SIEMENS

Local Area Network SINEC LI

Manual **Order No. 6ES5998-7LA21**

Contents	Order No.:
Instructions	GWA 4NEB 811 0545-02
Programming Instructions	GWA 4NEB 811 0546-02
Operating Instructions	GWA 4NEB 811 0520-02
Operating Instructions	GWA 4NEB 811 0521-02
Operating Instructions	GWA 4NEB 8110730-02

SIEMENS AKTIENGESELLSCHAFT

Instructions

Programming Instructions

Operating Instructions COM 530 on the PG 675

Operating Instructions COM 530 on the PG 615

Operating Instructions COM 530 with S5-DOS

SIEMENS

SIMATIC S5

SINEC L1 Local Area Network

Instructions Order No.: GWA4 NEB 811 0545-02d

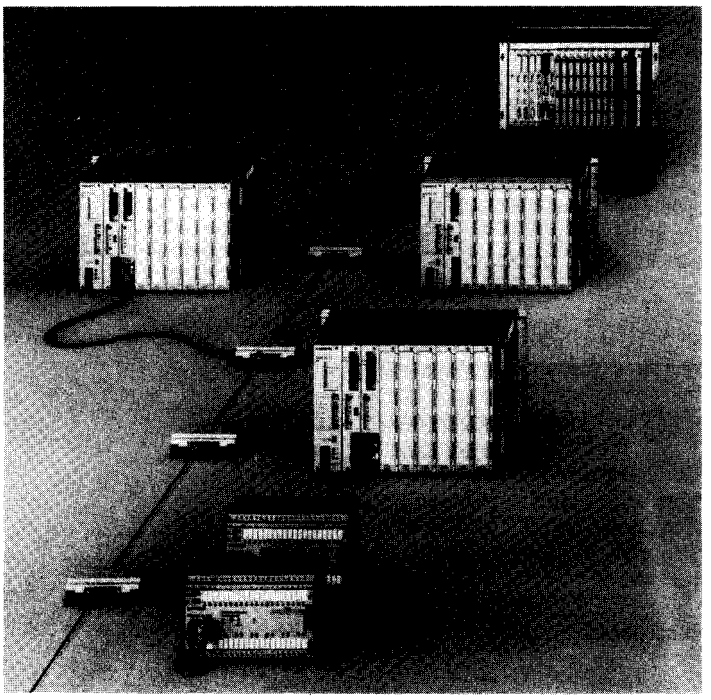


Fig. 1-1 SINEC L1 local area network

Contents

1. Description

1.1 Application

The interconnection of programmable controllers for the purpose of implementing distributed and hierarchical automation structures and the resultant enhancement of communications between the individual PCs is gaining in significance. Two SIMATIC S5 local area networks are available for this purpose:

- SINEC HI, for the high-performance range
- SINEC LI, for the low-performance range

These instructions apply exclusively to the components of the SINEC L1 network.

The SINEC L1 bus consists of three different components:

. CP 530 communications processor
. BT 777 bus terminal or transceiver (per node)
- Bus (4-core, shielded)

The CP 530 communications processor implements all the central coordination functions for the SINEC L1 bus, which operates on the master-slave principle. The CP 530, which controls the flow of information, is plugged into the master PC within the network. The slave PCs can be connected to SINEC L1 in two ways:

– In the case of the S5-1OOU, S5-1OI U and S5-115U programmable controllers, the connection is made via the existing programmer interface.
  N. B.: The 100 CPU cannot act as a slave with the S5-100U Pc.
– If the programmer interface is to be kept in the case of a slave S5-115U, a CP 530 can also be plugged into the slot to assume the slave interfacing function. The S5-135U and S5-150U PCs always communicate with the SINEC L1 bus via a CP 530 communications processor.

The maximum length of segments between two active BT 777 bus terminals is

– 1 km (0.6 miles) or 2,5 km (1.6 miles), depending on the **bus cable** used
– 2,5 km or 4 km, depending on the bus terminal used.

This gives a total bus length of 30 or 50 km (19 or31 miles). The maximum number of nodes is 31, including the master PC.

The CP 530 as master processor
: controls bus traffic
: interchanges data with the master PC (—)
: passes programming functions on to the nodes (-------)

The CP 530 as slave interface module
: exchanges data with its master PC (—)
: responds when referenced by the master
-for bus functions (.....)
–for programming functions (-------)

The BT 777 bus terminal constitutes the link between the individual nodes and the bus.

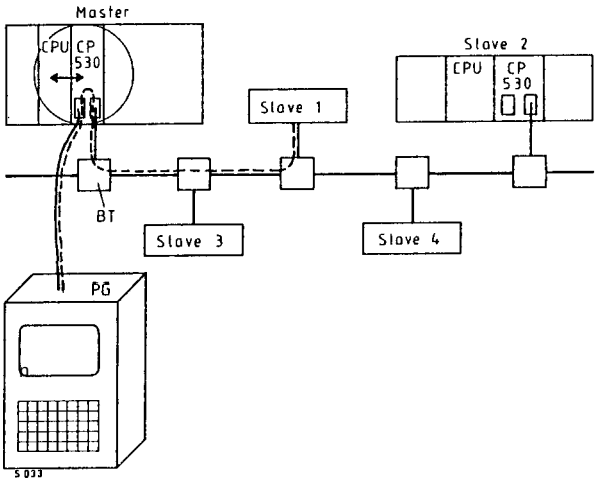


Fig. 1-2 Principle of operation as master processor

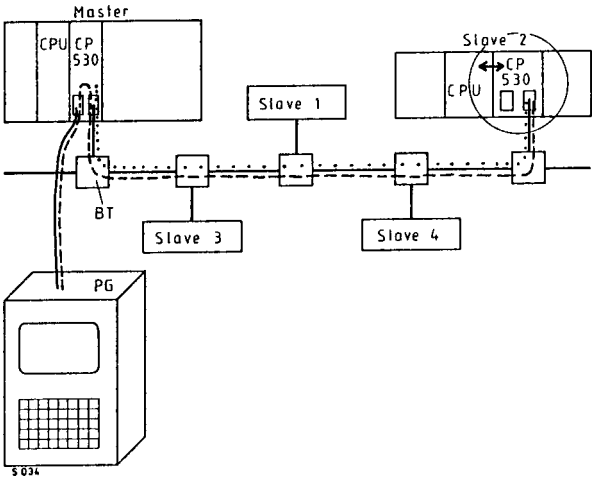


Bild 1-3 Principle of operation as slave interface module

1.2.1 Mechanical Design

There are two versions of the CP 530 communications processor:

- **Block-type module (43 mm or 1.7 in. wide)**
  (6ES5 **530-7 ...) for the** S5-115U **programmable controllers. The module is contained in a rugged plastics holder, which protects it from environmental influences and permits simple and vibration-free installation. A fan is not required.**

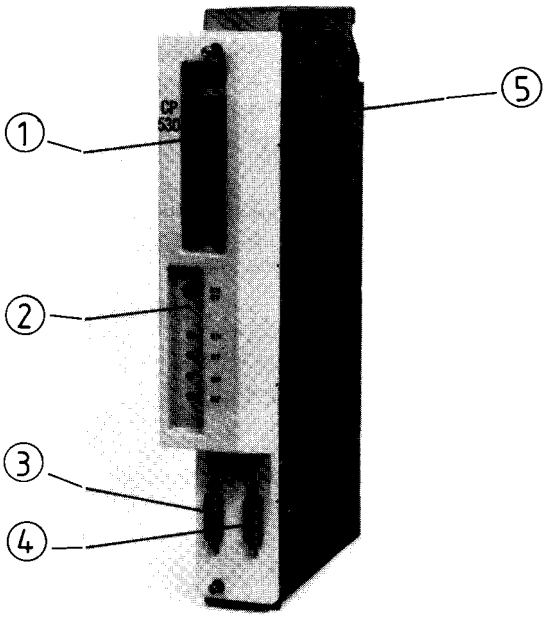


Fig. 1-4 CP 530 as a block-type module

. **Compact version (20 mm or 0.8 in. wide)**
  (6ES5 **530-3 ...) The module is a normal PCB and can be plugged into the** S5-135U **and** S5-150U **central controllers (also without fans); the module can also be plugged into the** S5-115U, **using an adapter casing.**

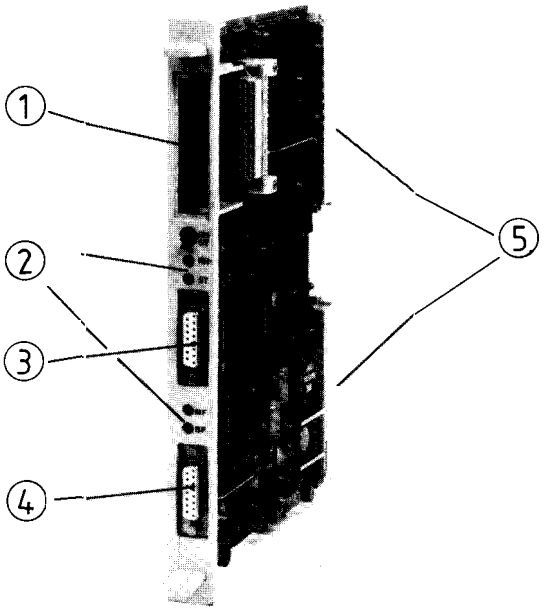


Fig. 1-5 CP 530, compact version

The frontplate has a receptacle for a memory submodule ① **(EPROM, EEPROM or RAM), two serial interface ports each with a 15-way Cannon socket connector for a programmer** (PG) ③, **a BT777 bus** terminal ④, **and a number of controls and** displays ②. **On the rear of the block-type module there is a** 48-**way connector of range R2 and on the compact module two such connectors** ⑤.

1.2 Construction of the Communications Processor

1.2.2 Controls and Displays

- Green LED: RN = RUN
  indicates that the CP 530 is servicing the interfaces and operating satisfactorily.
  The RUN-LED can only light up if the mode selector is in the "RUN" position.

- Red LED: ST= STOP
  indicates that the CP 530 is not servicing the bus interface. The red STOP LED can also light up in response to a "STOP" command from the programmer or "STOP" command from the CP if the mode selector is at "RUN"

- Red LED: MF = Module fault
  indicates a hardware fault of the CP 530.

- Red LED: BF = Bus fault
  indicates a fault in the course of bus traffic, e. g.
  – wrong parameters assigned
  – slave PC failure
  – bus wiring fault

- Switch: RN/ST = RUN-STOP
  This mode selector can put the CP 530 either from "RUN" to "STOP" (see above) or from "STOP" to "RUN" if no further stop condition applies.

When the CP is switched to STOP, this state becomes effective at the latest on completion of a current cycle through the polling list.

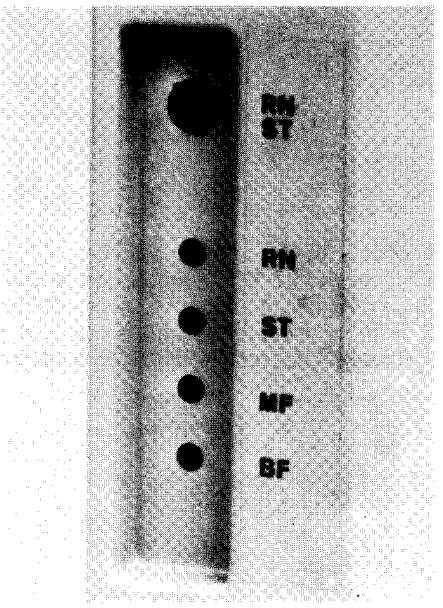


Fig. 1-6 Displays and controls on the CP 530 (block version shown)

1.2.3 Structure

The bus interface drives the SINEC LI bus in keeping with the parameters entered by the user in the user submodule. The CPU of the master PC can place data on the SINECL1 bus via the dual-port RAM and fetch data from the bus. The dual-port RAM is, as it were, a communications window between the relevant CPU and the CP 530. The operator can enter parameters via the programmer port and observe, control or test the bus.

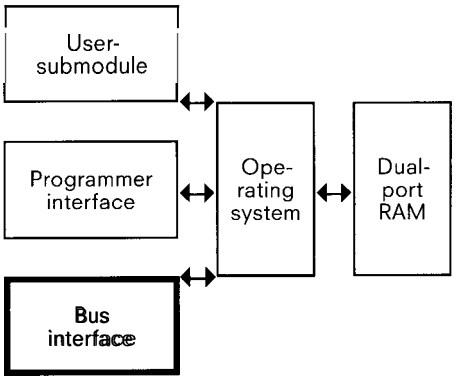


Fig. 1-7 Structure of the CP 530

1.2.4 Memory Submodules

The following three types of memory submodules are available for storing the user program in the CP 530:

EPROM submodule
EEPROM submodule
RAM submodule

The EPROM submodule can be programmed on the PG 615 or PG 675 programmer. A UV erasing facility is necessary for erasing the data.

The EEPROM is also a non-volatile memory which can be programmed and erased offline in a PG 615 or PG 675 programmer and online from 6/1986 in the CP 530.

The RAM submodule, on the other hand, is mainly **used for checking out the user program during system start-up. It should only be** used as a permanent program memory if the battery in the PC power supply is replaced at intervals of one to two years.

Caution:
- The RAM loses its data if the CP 530 is taken out of the central controller.
- In order to guarantee a safe initial start of the CP 530, the RAM or EEPROM submodule should first be erased.

Submodule	Order No.	Remarks
EPROM submodule		
8K bytes	6ES5375-OLA11	
16 K bytes	6ES5375-OLA21	
EEPROM submodule		
2Kbytes	6ES5 375 - 0LC II	Can only be programmed with programmers using S5-DOS or with COM 530/615
4Kbytes	6ES5 375 - 0LC21	
8K bytes	6ES5 375 - 0LC31	
16K bytes	6ES5 375 - 0LC41	
RAM submodul		
8K bytes	6ES5 375 - 0LD11	
16 K bytes	6ES5 375 - 0LD21	
32 K bytes	6ES5 375 - 0LD31	

The parameters of the communications processor require a maxi'mum user memory space of 500 words. However, 'all the above larger submodule types can be used.

1.2.5 Addressing in the Central Controller

The CP 530 can be plugged into a central controller along with other central processors or intelligent I/O modules. The CPU references the CP 530 via an address, which is set with the programmer. This is done by way of the "SYSID" function (system identification area). Addresses between 1 and 254 can be set.

1.2.6 Parameter Assignment

The CP 530 can be programmed with two lists (analogous to the programming of a SIMATIC S5 CPU):

- **a polling list = sequence for (max. 64 Byte) bus cycle**
- **a interrupt list = sequence for (max. 30 Byte) interrupt scanning**
- **There is also the SYSID area, which permits the defining of module functions by parameter assignment.**

If the CP 530 is to be used as a gateway in connection with programming functions, it must not have a polling list.

These lists are generated by the PG 675/PG 615 programmers with the support of the COM 530 software package and can be written into an EPROM, EEPROM or, if there is no memory submodule plugged in, direct into the internal RAM of the CP 530. The programmers are connected to the CP 530 by means of a cable with 15-way connector (as for the S5-I15U/S5-101 U).

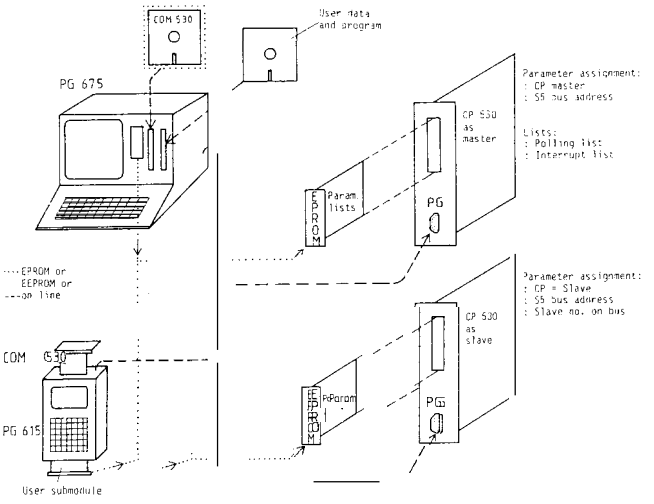


Fig. 1-9 Assigning parameters to the CP 530 as master or slave

1.3 Construction of the Bus Terminal

1.3.1 Mechanical Design

The bus terminal is accommodated in a rugged plastic casing and can be mounted in three different ways:

- Snapped onto a 35 mm sectional rail (EN 50 022)
- Snapped onto a 75 mm sectional rail, using mounting plate 3TX6 501-...
- Bolted to a flat surface using the fixing holes on the casing

A cable with a 15-way Cannon connector (1 or 2 m long) is brought out of the bus terminal to establish the connection to the SINEC LI node.
The four cores and shielding of the bus terminal are connected to five SIGUT screw-type terminals on either side of the casing ①.

An external 5V power supply (option) can be connected to another two terminals ②. The bus terminal is therefore capable of operating as a bus driver in a further section of the bus without having to be connected to the PC:
terminals C = + 5V (5.0 to 5.3 V; 0.3A)
D = OV reference potential (ground)

The two unlabeled screw terminals (bus terminal top left) must be bridged.

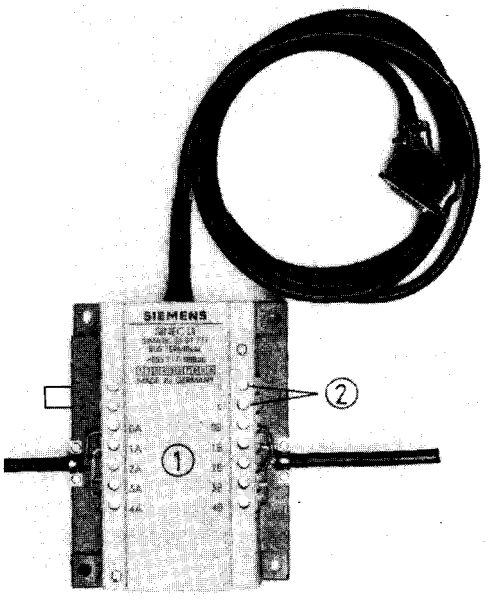


Fig. 1-10 BT 777 bus terminal

This power supply is not necessary for normal applications.

1.3.2 General

The BT 777 bus terminal is used for converting the 20 mA current loop signal from the serial interface of the module to a signal conforming to the EIA RS 485 standard on the bus.

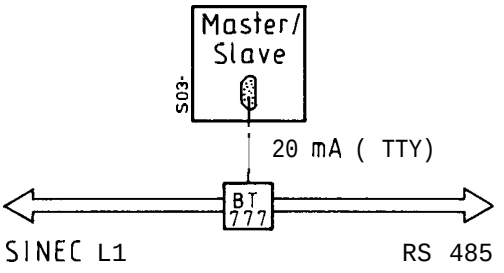


Fig. 1-11 Function of the BT 777 bus terminal

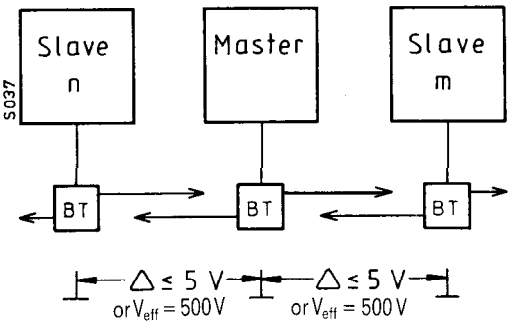


Fig. 1-12 Potential differences on the SINECL1 bus

The bus terminal of each node drives a bus segment. The potential difference between adjacent slave or master grounds should not exceed 5 V in the case of bus terminal 6ES5 777-0B.00.

With bus terminal 6ES5 777-1BC00, potential differences of up to V_{eff} = 500 V are permissible thanks to the optical isolation of the input.

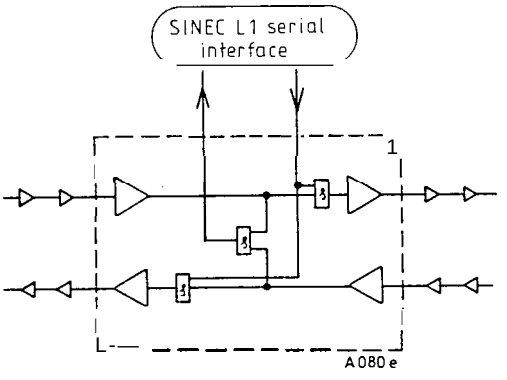


Fig. 1-13 Block diagram of the bus terminal

Fig. 1-13 shows a block diagram of the bus terminal. Each slave "hears" each signal on the bus with the exception of its own signal. Since signals can be sent and received from both ends, the master PC can be located at any point on the bus. A bus terminator at the extreme left or right-hand node is not required.

1.3.3 Connections

The BT 777 has terminals for bus cables with up to four cores. The core pairs for sending and receiving are connected up as shown opposite (see also cover of bus terminal):

The receiver core pair of the left-hand adjacent node are connected to terminals 1A/2A, while the sending core pair to the left-hand adjacent node leave at terminal 3A/4A.

The sending core pair to the right-hand adjacent node start at terminals 1 B/2B and the receiving core pairs from this node are connected to terminals 3B/4B.

> **Caution: When connecting the core pairs, make sure that they have the correct polarity.**
> **Make connections as shown in Fig. 1-14.**

Terminals OA and OB are for the cable screen. In this connection, please refer to Section 2.2. Installation guidelines.

The BT777 bus terminal contains a relay with four normally closed contacts. If the connector on a slave or master is withdrawn or if the power supply is disconnected, these contacts bridge the electronics of the bus terminal so that bus traffic for the other nodes remains unaffected (the bus terminal is a passive device).

The terminal can drive a bus of up to 2.5 km/4 km (1.5/2.5 miles) in length, i. e. should one of the nodes fail, the length of the cable between two nodes ① and@ still functioning must not be greater than 2.5 km/4 km (1.5/2.5 miles). If this length is exceeded, an extra power supply is necessary (see Section 1.3.4).

The connecting cable to the PC/CP can be extended to 25 m (165 ft). If the necessary cross-sectional area (1.3 mm' or 16 AWG) is not available for the supply cables, an external power supply must be provided (C= 5V, D = OV).
In this case, the following applies:

Required connections as above
(supply cables necessary, but smaller cross-section possible; monitoring to determine whether bus terminal is plugged in)

Required connection:
- Signal cores 6,7,9,2
- Power supply +5V: 3,14 Total cross-sectional area ≧ 1.3 mm² (16 AWG)
  M : 5,12 Total cross-sectional area ≧ 1.3 mm² (16 AWG)
- Shield : 1,8
- Open jumper Q 8 on the BT 777

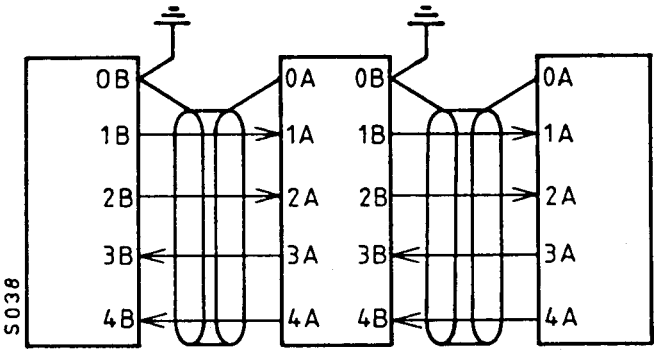


Fig. 1-14 Connecting the bus cable

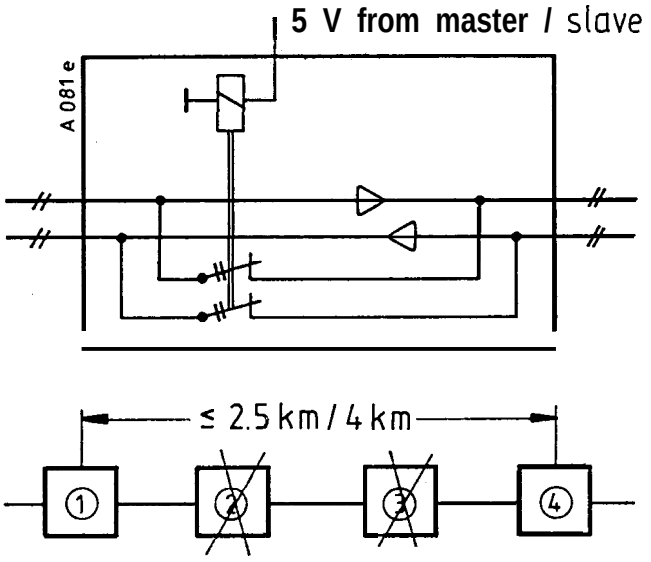


Fig. 1-15 Bridging a noda that has failed

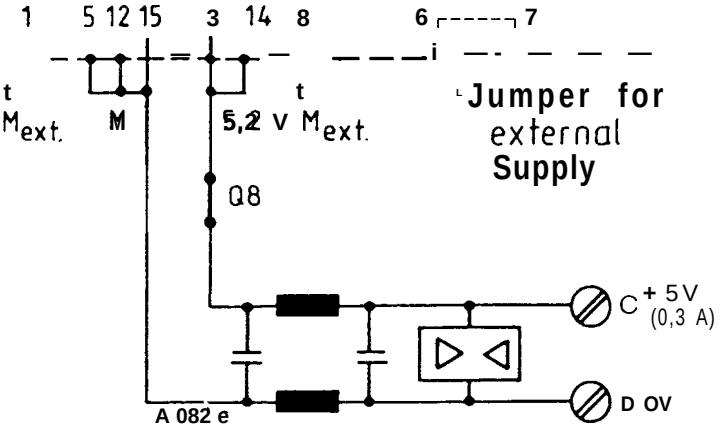
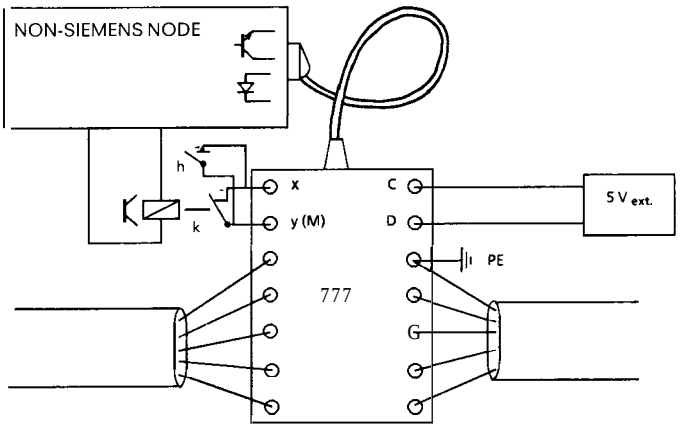


Fig. 1-16 Power supply

1.3 Construction of the Bus Terminal

1.3.4 Bus terminal for non-Siemens nodes
(No power on DSUB pins 3/5)

If the non-Siemens node is switched off or the cable from the bus terminal unplugged, this simulates a nonconducting interface transistor (20 mA current loop): a sustained line break is the result.



Remedy:

- Connect external 5Vpower supply to C.
  Connect M (OV reference) potential to D.
- Do not open jumper Q8 in the bus terminal.
- Monitor the voltage of the non-Siemens node with a relay; connect the NC contact via the two terminals at the top (labelled x and yin the case of bus terminal BT 777-1xxxx). Terminal y carries ground potential (like terminal D). The bus terminal cable can be unplugged if x and y are shorted. This is done by:
  * disconnecting the power supply of the non- Siemens node (k closed)
  * or operating a manual switch (h closed)

1.3.5 Bus driver

The bus terminal can also be used without programmable controller or communications processor as a bus driver or repeater. The external power can be connected in two ways:

Via 15-pin DSUB: Pin 3 = + 5.0 V (0.3A)
Pin 12= OV
lumper pins 6 and 7

Via terminals C/D: C = + 5 V, D = 0 V
lumper the two terminals (x,y) at the top left of the BT777.

1.3.6 Bus Terminal Selection

Bus terminal	6ES5777-QB.00	6ES5 777-1BC00
Maximum cable length as per Section 1.4		
– Segment	2.5 km (1.5 miles)	4 km (2.5 miles)
- Complete (BUS)	30 km (19 miles)	50 km (31 miles)
Isolation if powered		
a) via the PC/CP		
Input/Output	±5V	V_{eff} = 500V
input/Power supply	±5V	V_{eff} = 500V
Power supply/Output	–	–
b) externally		
Input/Output	±5V	V_{eff} = 500 v
input/Power supply PC/CP	V_{eff} = 500V[1)]	V_{eff} = 500 V[1)]
Power supply PC/C P/Output	V_{eff} = 500V[1)]	V_{eff} = 500 V[1)]
Impulse withstand voltage, terminals IAto 4B	–	5kV/50 µs (BEMA impulse)
Display for RXD/RXT(Typ test)	–	yes

1)Isolation by TTY-Network in PC/CP

1.4 Bus Cable

a) SIMATIC cable for distances of up to 1 km (0.6 miles)
Cable type: AZ1/514 G 5x0.14 mm^2 (64912014)
Design: 5 core dia. 0.42 mm/0.14mm^2 shielded
Outer diameter 5 mm
Order No.: 6ES5707-IAAO0

b) Indoor-type cable for 2.5/4 km (1.5/2.5 miles) (twisted pairs)
Cable type: 2YSTCY 2x2x0 .6411.5
Design: 2x 2 core, dia. 0.64 mm/0.32mm^2, (22 AWG) shielded
Outer diameter 7.9 mm
Order No.: 6ES5707-2AAO0

c) Buried cable for 2.5/4 km (1.5/2.5 miles) (twisted pairs)
Cable type: 2YSTC 2Y 2x2x0 .64/l.5
Design: 2x2 core, dia. 0.64 mm/0.32 mm', (22 AWG) shielded
Outer diameter 9.4 mm
Order No.: 6ES5707-3AAO0

d) Buried cable and lightning protection cable for 2.5/4 km (1.5/2.5 miles) (twisted pairs)
Cable type: 2YSTC2Y CC24 2x2x0 .6411.5
Design: 2x2 core, dia. 0.64 mm/O.32 mm^2, (22 AWG) shielded
Insulated 2xcopper braiding for lightning protection (16 mm^2)
Order No.: 6ES5707-4AAO0

Orders for all cable types to be placed with:
ZN-Werkstatt Fürth (FSZ Fürth);
specify lengths in meters

Use of customer-specific cables.

Other cables with similar ratings can also be used for SINEC L1. The following technical specifications can be used as an aid to selection:
– Loop impedance ≤ 110 Ω/km
– Capacitance per unit length ≤ 50 nF/km

Where cables of other manufacture are chosen, however, no functional guarantee can be given for the maximum segment length (cable distortion).

Note:
Increase core cross-sectional area to that required for SIGUT (screw-type) terminals (see 1.5.3).

1.5 Technical Specifications

1.5.1 Environmental Conditions

Degree of protection	: IP 00 (no protection against dust or water)
Permissible ambient temperature	: o to 55°C
Transport and storage temperature	: –40 to +85°C
Humidity rating (DIN 40040)	: ≤75% atmospheric humidity, annual mean for ≤35°C, no condensation
Mechanical stressing	: Installation in stationary equipment not absolutely free of vibrations
–Vibrations IEC 68-2-6	: 10...57 Hz 0.15 mm
	: 57...500 Hz 2g
–Shock IEC 68-2-27	30 g/18 ms, semisinusoidal

1.5.2 Technical Specifications of the CP 530

Mechanical Data

	Block-type modules	Compact version
PCB format	160x260 mm (6.2 in.x10.2 in.)	160x233.4 mm (6.2 in.x9.2 in.)
Frontplate width	43 mm (1.7 in.)	20,3 mm (0.8 in.)
Casing	203.2x302.6 mm (approx. 8 in.x11.9 in.)	—
Weight	–	–
Backplane connector (ES 902, range 2, 48-way)	1 x	2 x
Front connector (Cannon socket connector, 15-way)	2 x	2 x
RUN/STOP mode selector	1	1
LED displays RN (RUN) ST (STOP) MF (MODULE FAULT) BF (BUS FAULT)	4	4

Electrical Data

	Block-type module (6ES5 530-7LA11)	Compact version (6ES5 530-3LA11)
Power supply:		
+5V (backplane connector): Tolerance ±5%	Tolerance ±5%	
+5.2V (backplane connector): Tolerance ±5%	—	
+24V (backplane connector):		Tel. +25%/- 15%
+5.2V (front connector) : Tolerance ±5%	Tolerance ±5%	
Current rating:		
+5V (backplane connector):	1.0A (typ. 0.6A)	11A (typ. 0.6A)
+5.2V (backplane connector):	1.25A	–
+24V (backplane connector):		0.35A
+5.2V (front connector) :	1.25A	1.25A
Power losses :	6.5W	9.0W
Microprocessor :	SAB8031	SAB8031

Note
The S5-115U power supply modules can serve only one compact module (adapter casing):

Power supply module	
6ES5951-7LB12(3A)	-
6ES5 951-7LB21(7A)	1 x 530 compact
6ES5 951-7LD11 (15A)	1 x 530 compact

1.5.3 Technical Specifications of the BT777 Bus Terminal

Casing:	115 mm x 150 mm x 38 mm (4.5 in. x5.9 in. x 1.5 in.)
– for snapping onto	35 mm standard sectional rail
– with 3TX6-501 mounting plate on	75 mm standard sectional rail
– screw fixing	
Connections:	
– to the PCs	15-way Cannon connector with approx. 1 m (3.3 ft) or 2 m (6.6 ft) cable (expandable to 50 m)
– to the SINEC L1 bus	
incoming line	4 SIGUT terminals 1...2.5 mm²
outgoing line	4 SIGUT terminals (17 to 13 AWG)
– protective earth (PE) conductor	2 SIGUT terminals solid
Supply voltage	5V, tolerance ±5%
Current consumption	0.3 A (typ. 0.25 A)

1.5.4 SINEC L1 Bus Data

- Baud rate 9600 bits/s
- Bus cycle time, e. g. for 30 nodes:
  - 2 bytes of nettdata 0.75s
  - 64 bytes of nett data 5.2s
  - General: $Tu = (20 + L_m\ 1{,}8 + L_s\ 1{,}8 + t_Q)\ n$ [ins]

    Tu= Bus cycle time for n nodes with identical message length (m) master — slaves with identical message length (s) slaves — master
    L_m = Number of bytes, master
    L_s = Number of bytes, slave
    $t_{Q\text{-}20}$ ms, for cross communications only
    n = Number of nodes
- Maximum cable length with conductor cross sectional area 0.32 mm²

Bus terminal 6ES5777-OB.00
– segment 2.5 km (1.5 miles)
– complete length 30 km (19 miles)

Bus terminal 6ES5 777-1 BCOO
– Segment 4 km (2.5 miles)
– complete length 50 km (31 miles)

2.1 Packaging and Dimensions

Block-type CP 530:

The module is hooked into the mounting rack of the S5-115U PC and fixed with two screws.

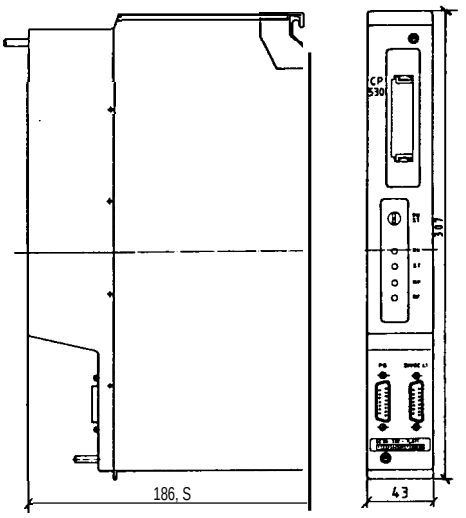


Fig. 2-1 Block-type CP 530

CP 530 compact:

The module is plugged into the central rack of the S5-135U and S5-150U programmable controllers and locked in position with the locking bar.

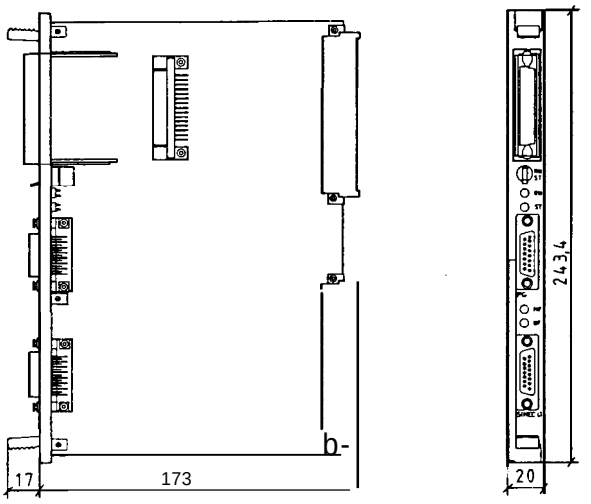


Fig. 2-2 Compact-type CP 530

The bus terminal can be attached in three different ways:

- Snapped onto a 35 mm standard sectional rail
- Snapped onto a 75 mm standard sectional rail
- Bolted, using the two fixing holes for screws 0 max. 5 mm.

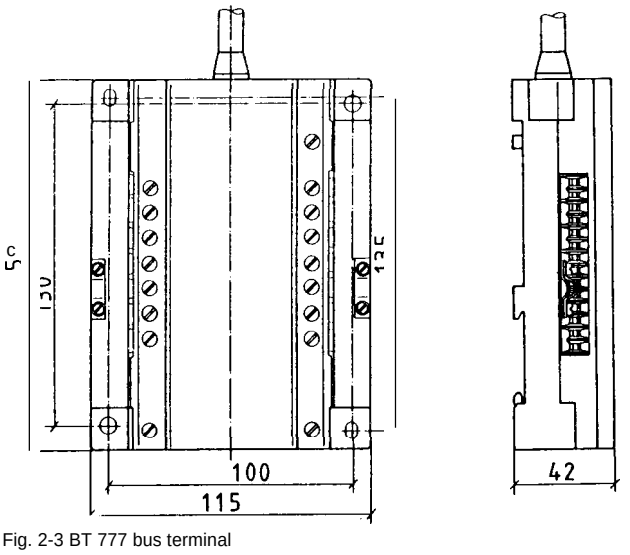


Fig. 2-3 BT 777 bus terminal

The BT 777 bus terminal with its terminals on both sides to the SINEC LI bus and the connecting cable to the serial interface on the relevant node should be kept separate from all input and output cables.
A clearance of at least 10 cm (4 in.) should be observed between the SINEC LI bus cable and adjacent power cables.

Immunity to noise can be enhanced if the screen is connected at both ends to the BT 777 bus terminal and the latter connected to PE (protective earth or ground). Make sure that the equalizing currents through the screen cable between the nodes do not become too high. If necessary, unload the screen by means of a separate earthing conductor of adequate cross section.

If the connecting cores have a cross-sectional area of less than 1 mm^2 (17 AWG), core end sleeves must be used (e.g. Etlinger & Co., München).

3. System Start-up and Troubleshooting

3.1 Self-Test

When the programmable controller is powered up, all the red LEDs on the CP 530 first light up and a module self-testis started. If this test proves successful, the red LEDs darken and the green RUN-LED indicates the RUN status.
Condition: Mode selector at “RUN”.

If the fault still permits programming functions to be carried out, the exact cause of the fault can be ascertained with the aid of the special “Modes” function, this enables the following responce: 10 hardware faults n (where n corresponds to the table opposite).

RUN	STOP	MF	BF	Cause	Remedy
*			*	Restart: CP waiting for SYNCH RON Operation: Break status on bus	SYNCHRON Bus connector?
	*		*	Restart: Error in SYSID;k polling list !nterrupt list; no list Operation: CP in STOP status Last slave cycle contained error(s)	New submodule connector Re-write list
	*			CP in STOP-status (mode selector) or STOP command from programmer or CPU	
		*		Hardware fault	Replace hardware
			*	Error(s) in network traffic	

The CP 530 must always be assigned the necessary parameters (SYSID function ≙ system identification). A CP 530 acting as a slave interface module only requires the SYSID parameter. If used as a bus master, the polling list and, where applicable, the interrupt list are also necessary.

The following parameters must be assigned in the SYSID area:

- CP 530 as master
  - Polling list required
  - Interrupt list required where applicable
  - No SINEC slave no.

  CP 530 as slave
  - No polling list permitted
  - No interrupt list
  - Slave no. on SINEC LI required

- Address in rack of the master PC
  - 1-254

  This is the internal master PC address under which the CP 530 is referenced from the CPU.

. Slave no. on the SINEC LI bus: this number is required when the corresponding slave PC is interfaced to the SINEC LI bus via CP 530. The number assigned must be between 1 and 30.

The polling list contains the sequence in which the slave PCs connected to the bus are polled from the master and is used at the same time for checking the number of nodes participating in the bus: if a referenced node does not reply, the red "Bus fault (BF)" LED on the frontplate of the master indicates a bus fault.

The user can change the sequence of the bus cycle as a function of process events by entering an interrupt list. In the event of an interrupt the normal polling is halted and the interrupt list is accessed. The slave PC triggering the interrupt is then located and serviced. The bus cycle is then resumed at the point of interruption. This list describes

a) the slave PCs from which interrupts are permitted

b) the sequence in which processing takes place when several slave PCs simultaneously trigger the group interrupt.

No interrupts are lost. If there are no interrupts, the bus master operates as prescribed by the polling list.

3.3 COM 530 Diagnostics

The CP 530 and all events on the bus can be observed and controlled by means of the "COM 530" program package.

The "CP status" programmer function enables the operator to access important dynamic statuses of the CP 530 and its error buffer (even in the STOP state).

The following statuses are displayed:

- CP in RUN status
- CP in STOP status
- Programmer has priority over CPU, i. e. the CPU can only read from the CP but no longer write into it
- Errors/no errors have occurred on the CP.
  Errors are indicated by the display of one or more error numbers.

  (For more details, please refer to:
  COM 530/675 Operating Instructions
  Order No.: GWA 4NEB 8110520-02,
  Section 3.10)

  The status of a maximum of two slave mailboxes can be selected from the programmer "TEST" function. In the case of the PG 615 programmer, only one mailbox can be selected. The next send/receive mailbox on the CP 530 to be sent to the relevant slave or the one last received by the slave is displayed on the programmer.

  In the case of cross communications, e. g. slave 2 sending data to slave 3, the send mailbox assigned to slave 2 is displayed, but the message received by slave 3 from slave 2 appears in the receive mailbox with the reference "Cross communications".

  The time required for this slave cycle is displayed with the contents of the receive mailbox.

  The operator can force a send mailbox for one slave cycle with the aid of the "STATUS" function: the STATUS display is frozen and all data can be modified. The send mailbox is brought into the bus cycle once by selecting a programmer softkey. By setting a "Programmer priority" identifier, the operator can disable the CPU-CP 530 bus interface so that the CPU of the master PC can no longer "get in his way" when he is forcing a send mailbox for one slave cycle.

  Selection of the "Bus test" function causes bus traffic to be automatically halted at the end of the polling list and the "Programmer priority" identification bit is set.

  The operator is now shown the send mailbox of the first slave in the polling list and given the opportunity to make any modifications. The send mailbox is sent to its destination via the CP 530 by selecting the "Send" softkey and the message returned from there is displayed as a receive mailbox from the destination slave.
  Interrupt messages can be incorporated in the bus test if required.

  The CP 530 keeps statistics on the
  - longest and
  - shortest bus cycle

  and displays these statistics with the last current time when requested by the "BUS CYCLE TIME" function.

  Bus cycle = once round the polling list.

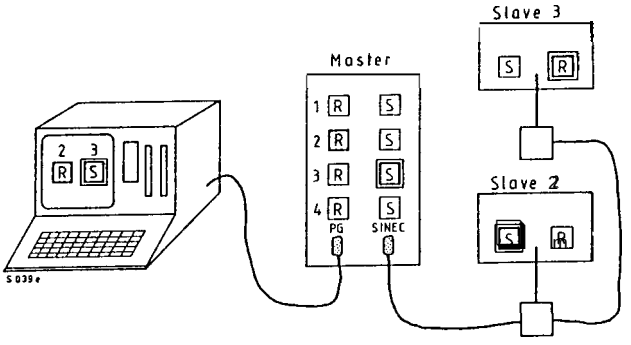


Fig. 3-1 Status of the receive mailbox 2 and send mailbox?)

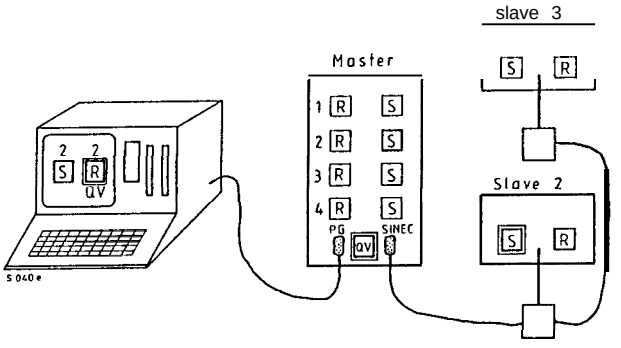


Fig. 3-2 Status of send mailbox 2 and receive mailbox 2 for cross communications from 2 — 3

Bus cycle time	
Current:	. ms
Minimum:	. ms
Maximum:	. ms

As an additional diagnostic aid, the PG 675 programmer can be looped in between the CP/CPU and bus terminal, thus enabling the operator to observe the bus at character level (ASCII) and record the traffic on the bus with the aid of the "FOX-PG" software package.

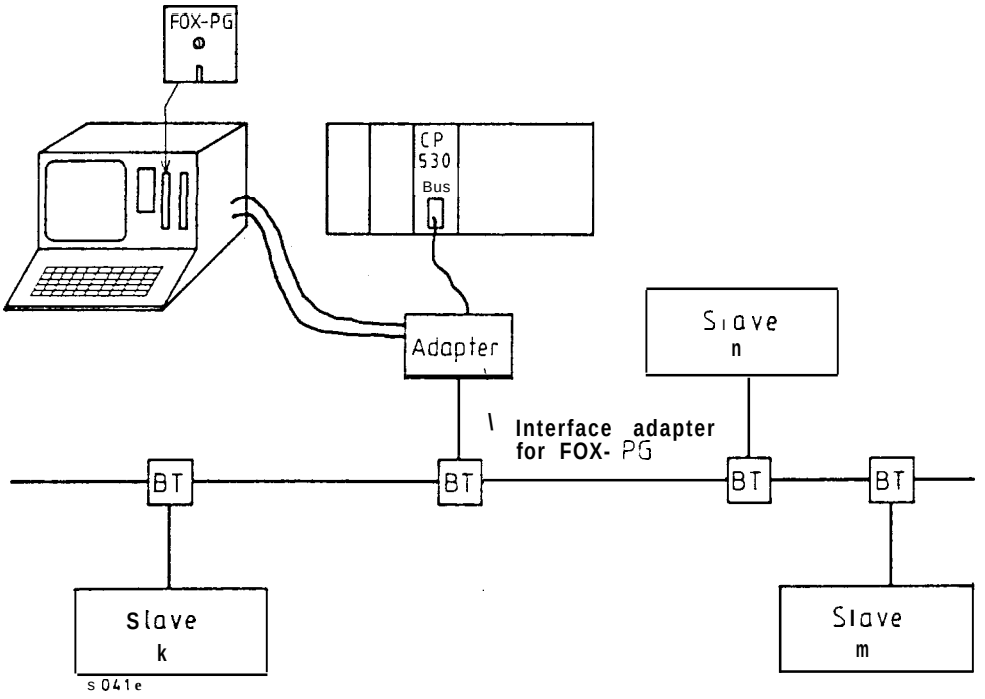


Fig. 3-3 Using the PG 675 programmer for observing the SINEC L1 network at message level

Ordering data:

Adapter	**6ES5321-OAA11**	**(ZN Werkstatt Fürth)**
Diskette	**6ES5875-OCA11**	**(GWK, Diskette for the PG 675 programmer)**

4. Appendix

4.1 Interface Pin Assignments

The electrical interfaces of the CP 530 are specified in more detail in the following subsections. These cover the following:

- Backplane connector (xl, x2)
- User submodule (x3)
- Serial port for PG (x4)
- Serial port for SINEC LI (x5)

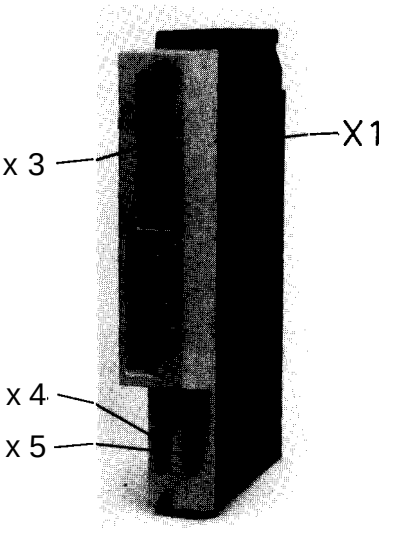


Fig. 4-1 Interface ports on the block-type module

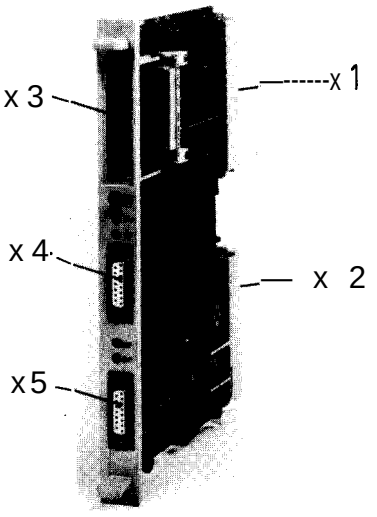


Fig. 4-2 Interface ports on the compact module

4.1.1 Backplane Connectors Xl and X2

The block-type and compact version modules differ in this case. The block-type module has only one 48-way backplane connector of range 2 and the compact version two such connectors. The different pin assignments can be seen in the following tables.

Compact version:

	d	b	z			d	b	z
2	+5.2V	M	+5V		2	(5.2V)	M	+5V
4	UBATT			xl	4	UBATT		
6	AD12	AD O	RESET		6	AD12	AD O	RESET
8	AD13	AD 1	MEMR		8	AD13	AD 1	MEMR
10	AD14	AD 2	MEMW		10	AD14	AD 2	MEMW
12	AD15	AD 3	RDY		12	AD15	AD 3	RDY
14	IRA	AD 4	DB O		14	IRA	AD 4	DB 0
16		AD 5	DB 1		16		AD 5	DB 1
18		AD 6	DB 2		18		AD 6	DB 2
20		AD 7	DB 3		20		AD 7	DB 3
22		AD 8	DB 4		22		AD 8	DB 4
24		AD 9	DB 5		24		AD 9	DB 5
26		AD IO	DB 6		26		AD IO	DB 6
28		AD I 1	DB 7		28		AD11	DB 7
30	+24 V		M24V		30			
32		M			32		M	

Block-type module:

	d	b	z		
2		M	+5V		
4					x2
6					
8					
10					
12					
14					
16					
18					
20					
22	TXD				
24					
26		RXD			
28					
30			M24V		
32		M	+24V		

4.1.2 User Submodule X3

This interface establishes the connection to the memory submodules listed in Section 1.2.4 (48-way connector to DIN 41612, Range C). A large number of test signals also use this interface.

	c	b	a	
1	AD12	M	+5V	x3
2	AD O	AD 1	AD 2	
3	AD 3	AD 4	AD 5	
4	AD 6	AD 7	AD 8	
5	AD 9	AD IO	AD I 1	
6	AD13	AD14	RD 1	
7	WR2	AD15	TEST1	
8	PSEN	RDYE	LIN1	
9	ALE	$\overline{\text{TEST3}}$	ADV	
10	DB O	DB 1	DB 2	
11	DB 3	DB 4	DB 5	
12	DB 6	DB 7	K 1	
13	CS 1	Cs 3	K 2	
14	$\overline{\text{CS 2}}$	UBATT	K 3	
15	$\overline{\text{TEST2}}$	PSW/BUSY	K 4	
16	5V	RD 1	K 5	

4.1.3 Serial Interfaces X4 and X5

There are two 15-way socket connectors on the frontplate:

- for programmers (PG 675 and PG 615)
- for the SINECL1 bus terminal

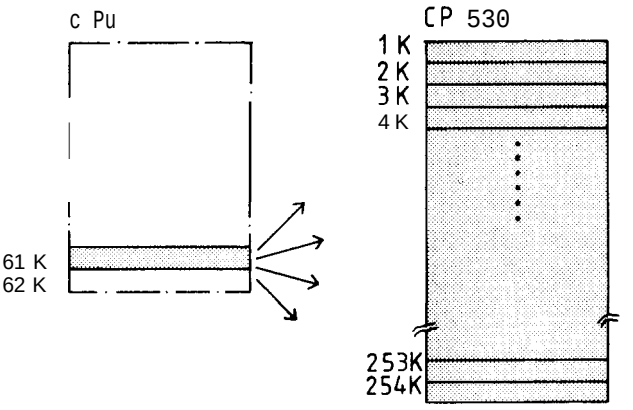
The interface can be operated either in active or passive mode (see Figs. 4-1 and 4-2).
The pin assignment of the two connectors is identical as far as the signal definition is concerned.

1	MEXT	(external 0 V reference or ground)
2	TTY IN –	(current output)
3	+ 5.2 V	
4	+ 24 V	
5	M	(internal 0 V reference)
6	TTY OUT +	(current input)
7	TTY OUT -	(current output)
8	MEXT	(external 0 V reference or ground)
9	TTY IN +	(current input)
10	OV reference for24 V	
11	20 mA current source of sender	
12	M	(internal 0 V reference or ground)
13	20 mA current source of receiver	
14	+ 5.2 V	
15	M	(internal 0 V reference or ground)

4.2 Adress Assignment on the S5 Bus
4.3 Constraints

The CP 530 has a different address area on the S5 bus, depending on the type of addressing:

Page addressing	CPU	CP 530
1 to 254	61 K to 62 K	1 K to 254 K



Page addressing

4.3 Constraints in Configurations with S5-101 U Slaves (up to and including the Z 05 operating system)

S5-101U 6ES5 101 -8U..3 Version 1

. The full data length of 2 x 64 bytes per slave cycle cannot be used. A maximum of 110 bytes may be exchanged between master and slave in one cycle.

E. g.: Master to slave: 64 bytes
Slave back to master: 46 bytes

or: Master to slave: 46 bytes
Slave back to master: 64 bytes

. EEPROM can not be used in the following PC.

S5-101U 6ES5 101 -8U..3 Versions 1 and 2

- Only a limited data lenth is possible for slave broadcasting; a maximum of 25 bytes may be transmitted per slave cycle.

Before broadcast: Master to slave / Slave back to master 25 bytes

Broadcast: Slave to all 25 bytes

After broadcast: Master to slave / Slave back to master 25 bytes

CP 530 communications processor (block-type module)	**6ES5530-7LAII**
CP 530 communications processor (compact version)	**6ES5530-3LAII**
Bus terminal (1 = 1 m or 3.3 ft) for 2.5 km (1.5 miles)	6ES5777-OBBO0
Bus terminal (1 = 2 m or 6.6 ft) for 2.5 km (1.5 miles)	6ES5777-OBCOO
Bus terminal (1 = 2 m or 6.6 ft) for4 km (2.5 miles)	6ES5 777-1 BCOO
Mounting plate for bus terminal	3TX6501
Fuse for bus terminal	TR5 F0,5A (Wickmann & Co.)

SINEC LI Manual

German	6ES5 998-7LA11
English	6ES5998-7LA21
French	6ES5998-7LA31
Spanish	6ES5998-7LA41
Italian	6ES5998-7LA51

COM 530/(CP/M-86)for PG675

German	6ES5 875-7UA11
English	6ES5 875-7UB11
French	6ES5 875-7UC11

COM530/modul for PG 615

German	6ES5 815-7UA11
English	6ES5 815-7UB11
French	6ES5 815-7UC11

COM530/S5-DOS for PG 635

German	6ES5 835-6SC11
English	6ES5835-6SC21
French	6ES5835-6SC31
Spanish	6ES5835-6SC41
Italian	6ES5835-6SC51

COM530/S5-DOS for PG 675/685/695

German	6ES5895-6SCII
English	6ES5895-6SC21
French	6ES5895-6SC31
Spanish	6ES5895-6SC41
Italian	6ES5895-6SC51

SINEC LI bus cables
Available from: ZN-Werkstatt Fürth **(FSZ** Fürth)
Cable lengths in meters (plaintext)

SIMATIC cable **5x 0.14 mm²** (26 AWG)	**6ES5707-IAAOO**
Indoortype cable **2x2x 0.32** mm^2 **(22** AWG)	**6ES5707-2AAO0**
Buried cable **2x2x 0.32** mm^2 **(22** AWG)	**6ES5707-3AAO0**
Lightning protection cable **2 x 2 x0.32** mm^2 **(22** AWG) + 16 **mm²**	**6ES5707-4AAO0**
Adapter	**6ES5321-OAAII**

SIEMENS

SIMATIC S5

SINEC L1 Local Area Network

Programming Instructions Order No.: GWA 4 NEB 811 0546-02d

Contents

1. Principle of Operation

1.1 Construction and Application

SINECL1 is a local area network that enables SIMATIC S5 programmable controllers of the U range to communicate with each other (Fig. 1 -1). It operates on the master-slave principle.

A single programmable controller

-the Master PC -

coordinates and monitors the entire data traffic in the SINEC LI network, as well as the routing through and monitoring of programming functions via the bus (Section 1.4). The other programmable controllers participating in the network are automatically

- the Slave PCs -.

The master PC must have a CP 530 communications processor, which assumes the master function and controls the entire flow of information.

There are two ways of sending data packets in the SINECL1 network:

- From any SINECL1 node to another;
- From any SINECL1 node simultaneously to all other nodes connected (broadcasting).

The data contained in the data packet may be the status of inputs, outputs (coils) and flags (internal relays) or the contents of data words.

Connection to the LAN bus:

- Master PC (S5-115U, S5-135U, S5-150U):
  via the CP 530 communications processor;
- S5-115U as slave:
  either via the programmer port of the PC or via the CP 530 communications processor;
- S5-135U and S5-150U as slave:
  via the CP 530 communications processor;
- S5-101 U as slave:
  via the programmer port of the PC.
- S5-100U (from 102 CPU onwards) as slave via the programmer port of the PC.

The BT777 bus terminal forms the connecting link between the programmable controller and the bus. This is the point at which the signal level is converted to that required for operation of the SINECL1 network.

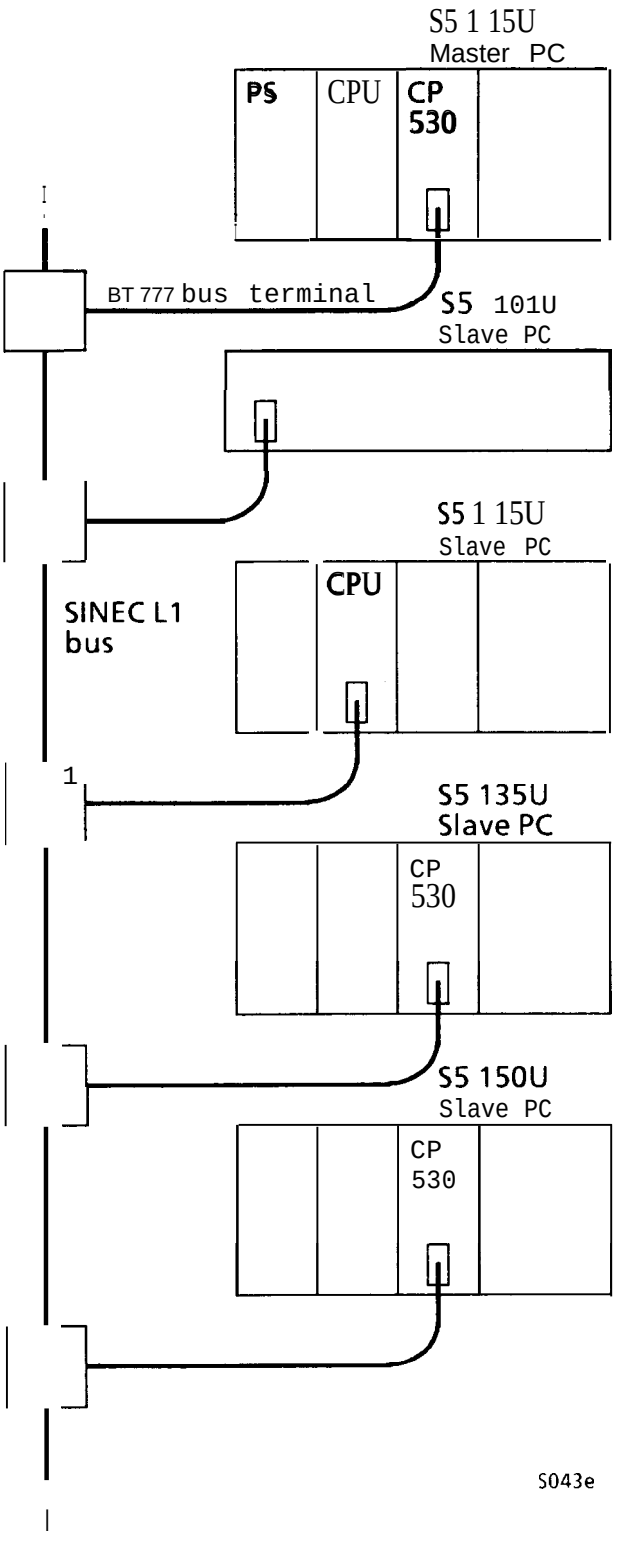


Fig. 1-1 A typical SINEC L1 configuration

The CP 530 of the master is not allocated an explicit node number, but the logical number "O" or the number "32" as destination address for sending from the slave to the master.

All slaves must have a slave number through which they can be referenced.
This number must lie between 1 and 30. The slave number cannot be assigned more than once in the network. Slave numbers allocated more than once or outside the permissible range of 1 to 30 result in network errors.

In the case of a slave with CP 530, the slave number is entered with the COM 530 program package via a programmer and stored on a memory submodule of the relevant CP 530 (cf. Section 2.2.1).

In the case of slaves without CP 530, the slave number is defined in a function block via the relevant CPU (for S5-101 U cf. Section 2.4.1 and for S5-115U Section 2.5.2.2).

The network can be compared to a post office that receives parcels and distributes them further to the various households. The CP 530 master is the post office, which is supplied with large numbers of parcels for the various households by a mail van (the master CPU), which, in turn collects parcels that have been delivered from the households to the post office.

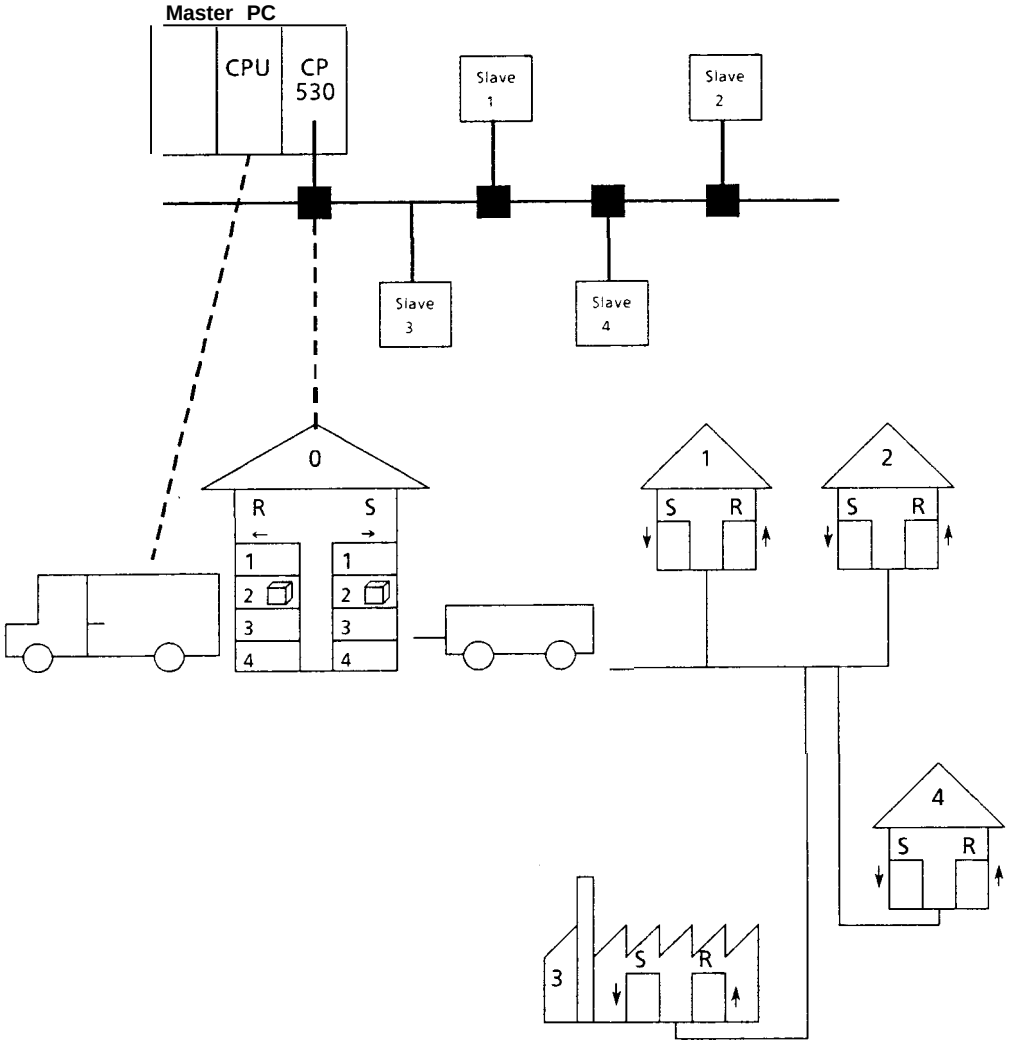


Fig. 1-2 Comparison between SINEC L1 and a post office model

1. Principle of Operation

1.1 Construction and Application

1.1.1 Establishing a Connection

In order to exchange data between a master and a slave in the SINECL1 network, a connection first has to be established in order to ensure that the data arrives at the correct partner.

The establishment of a SINEC L1 connection is always introduced by a “Break” (zero signal), which simultaneously clears an existing connection. After “Break”, the master sends the number of the desired slave PC and expects an acknowledgement with its own slave number as a reply from the referenced slave.

This connection is thus now established and has exclusive use of the bus. If the slave sends data back to the master, the latter also acknowledges.
A new connection can only be established again after a “Break” signal.

The driver of the mail van ensures that the parcels are delivered correctly by only delivering them if the recipient is at home.

1.1.2 Bus Protocol

The bus protocol controls the interchange of data in the network. In addition to the purely useful information or data, there is also control and error checking information on the bus which does not affect the user directly but which can be evaluated by him and, in certain cases, depending on system-specific conditions, must be evaluated. Further details are described where applicable in the following sections.
(See examples in the Appendix)

The STEP 5 user program communicates with SINECL1 through coordination bytes and, where data traffic takes place via the CP 530 as master or slave, also uses data handling blocks.

The send mailbox may contain:
O the “length” = quantity of data
O the “source”= sender
O up to 64 bytes of data.

The receive mailbox may contain:
O the “length”= quantity of data
O the “source”= sender
O up to 64 bytes of data.

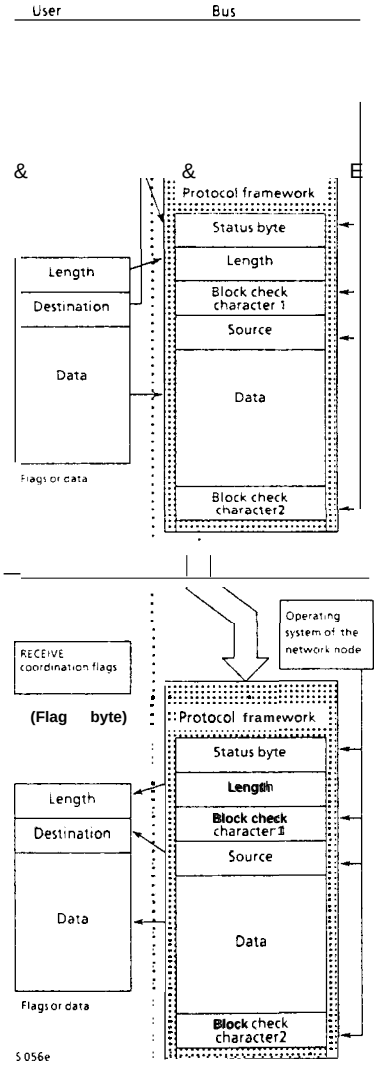


Fig. 1-3 Bus protocol – Data representation in the SINEC L1 system

In the post office analogy, the bus protocol corresponds to the packing and labelling of the parcel. The packing of the data and addressing of the parcel is the responsibility of the customer, i. e. the households, whereas the post office provides the box, the actual labels and the tying string.

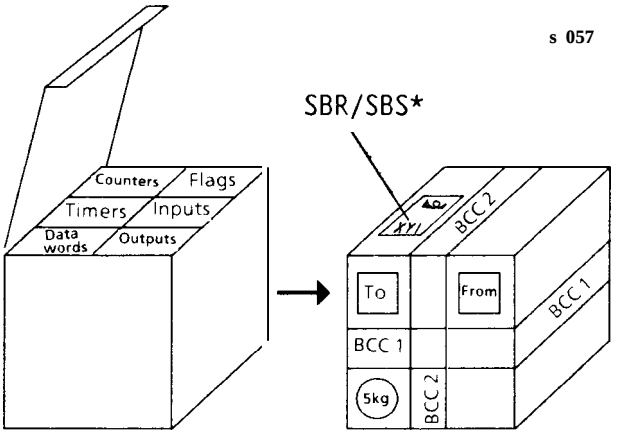


Fig. 1-4 Bus protocol in the post office model

1.1.3 Send and Receive Mailboxes

When assigning the initial parameters to a programmable controller that is to be connected to SINEC L1 via the programmer port, the user defines a receive mailbox and a send mailbox for data exchange via the bus, using the SYSID function. The starting addresses of these mailboxes are defined either by specifying a data block number and a data word number or a flag byte number (see Section 2). In addition to the net or useful data, both mailboxes contain a length specification of the destination in the case of the send mailbox or of the source in the case of the receive mailbox.

The length of the data should be specified in bytes; the destination/source is a slave number on the bus. When connecting a programmable controller to the SINEC LI network via the CP 530 communications processor, this is defined when initializing the data handling blocks (see Appendix).

All data that can be accessed with the STEP 5 operations available to the CPU, e. g. flags, data, inputs/outputs (byte or word), timers and counters etc., can be entered in the data field of the send mailbox. The data can also be fetched from the receive mailbox in byte or word mode and transferred to data/flag or input/output areas or, for instance, interpreted as times/counts.

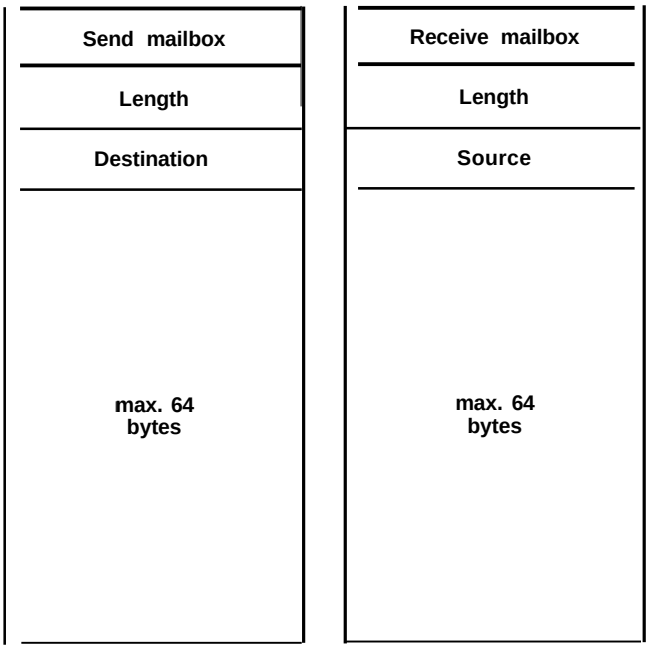


Fig. 1-5 Send snd receive mailboxes

1.1.4 Coordination with the User Program

Coordination of bus traffic with the user programs of the master and slave PCs differs depending on whether the PC participates in bus traffic direct via the programmer port or via a CP.
PCs that are connected direct use coordination bytes exclusively, while PCs connected via the CP 530 employ data handling blocks.

1.1 Construction and Application

1.1.4.1 Connecting the Slave PCs via the Programmer Port

The coordination bytes **provide** the user program with a powerful tool for interchanging data via the bus and participating in the bus control.
The coordination bytes are defined as a flag byte or left-hand data byte when initializing an S5-CPU participating direct in bus traffic: "Receive" coordination byte = CBR, "Send" coordination byte = CBS.

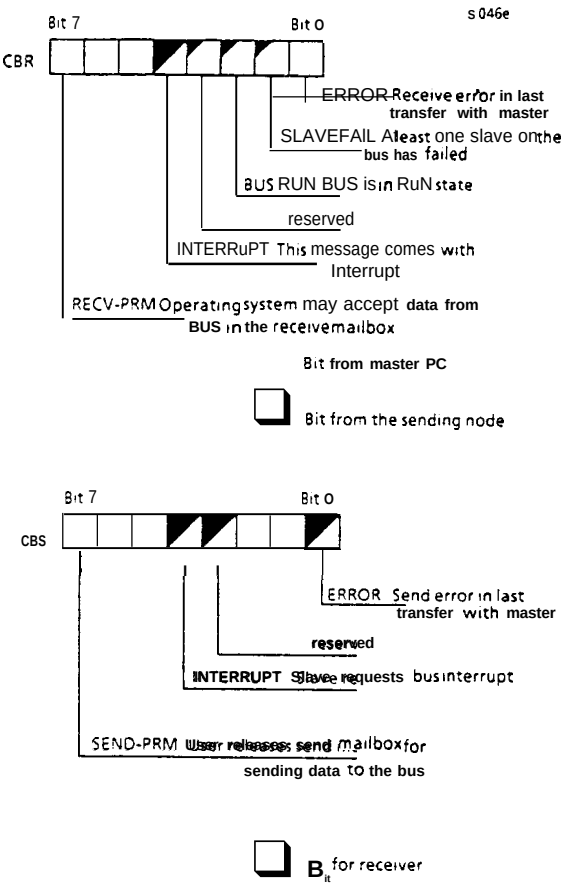


Fig. 1-6 Coordination bytes/PC as slave

Data Exchange

- After the send mailbox has been filled with the data to be sent, the "Permission to send" coordination bit (CBS, 7) is set via the user program and invokes the operating system to send the data. The operating system does not reset this bit until it has sent the data and received the appropriate acknowledgement, and new data can then be sent.

- The user program invokes the operating system to receive data from the bus and enter it in the receive mailbox by setting the "Permission to receive" coordination bit (CBR, 7). When the data has arrived, the operating system resets the "Permission to receive" bit and the user program can accept the data.

When sending and receiving data, the error bit "O" can be evaluated in order to decide whether, for instance, a transmitted message containing errors is to be repeated.

The sending of data via the CP 530 as master or slave is analogous: A data or flag area in the PC is defined as the send mailbox by initializing the SEND block and transferred to the CP 530. The data packet that has been received is transferred to the PC by the RECEIVE call and the packet specified in the initialization parameters written into the data or flag area. The "Condition code word" of the data handling blocks also has coordination bits; these are described in the Appendix.

Bus Control

The user programs of all slaves can trace and influence the sequences on the bus via coordination flags. When the "Permission to receive" bit (CBR, 7) is reset (i. e.: = data arrived), the operating system copies a number of bits from the bus status byte into the SBR; these bits can be evaluated by the user program:

o Interrupt (CBR, 4)
The data just received is express data (see Section 1.3)

O (CBR, 3) is reserved

o BUS RUN (CBR, 2)
SINEC LI is in the RUN mode and will also remain in this mode after data exchange

o SLAVE FAIL (CBR, 1)
At least one slave on the bus has failed.

In addition, the operating system of the slave sets the "Error" bit (CBR, O) to indicate whether the last data transfer took place without error.

The coordination information of the bits in CBR 1/2/3/4 is also transferred from the bus to the user program of the PC via the RECEIVE data handling block; bits 6 and 7 are sent in addition to the user program of a master PC (CP 530 = master): the sender is in the STOP or RUN mode.

When the "Permission to send" bit (SBS, 7) has been set, the user program can also send bit messages to the bus master or to the receiver:

o Interrupt (CBS, 4)
This data is to reach the receiver as express data (see Section 1.3)

O (CBS, 3) is reserved

O Error (CBS, O)
During the last data transmission, an error has occurred from the point of view of this slave

The coordination information of bit CBS 4 is also transmitted from the user program to the bus via the SEND data handling block; in addition, bits 6 and 7 are sent to the bus controller from the user program of a master PC (CP 530 as master): the addressee is to enter the STOP or RUN mode.
RUN and STOP are only possible with slave PCs connected direct via the PG (programmer) interface.

1.1.4.2 Connecting PCs via the CP 530

When using the CP 530, the transfer of data is achieved by the SEND and RECEIVE data handling blocks.
The user can program the status byte for sending (S6S) in the first byte of the send mailbox and the coordination information for receiving in the first byte of the receive mailbox.

The user specifies the number of bytes to be sent (from the send mailbox) with the parameter QLAE from the SEND data handling block.

The value of QLAE is obtained as follows:
QLAE = length + 2 bytes header
where length = the number of bytes of net data.

The user defines the number of bytes of data received by the parameter ZLAE in the RECEIVE data handling block.

The value of ZLAE is as follows:
ZLAE = length + 4 bytes header
where length = the number of bytes of net data.

'Caution: The header of the send request must always be reset (0000H). Exception: Direct slaves communicating via the CPU should be switched to STOP or RUN (cf. description of SBS).

Send mailbox

SBS	Not used	Header*
Data byte 1	Data byte 2	
Data byte 61	Data byte 62	Data
Data byte 63	Data byte 64	

Receive mailbox

SBR	Length	Header
Source	Not used	
Data byte 1	Data byte 2	Data
Data byte 61	Data byte 62	
Data byte 63	Data byte 64	

1.1 Construction and Application

Coordination information:

PC with CP 530 as master

The "Permission to receive and send" and "Receive error" flag bits are not used.

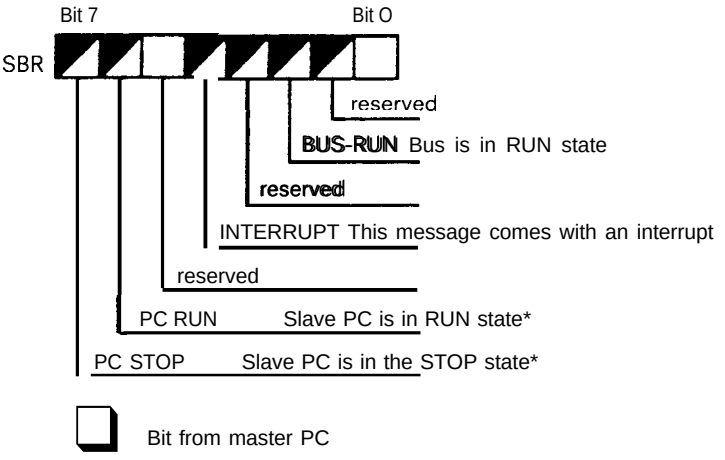
Their function is replaced by information in the data handling blocks and can be evaluated by the CONTROL block (see Appendix).

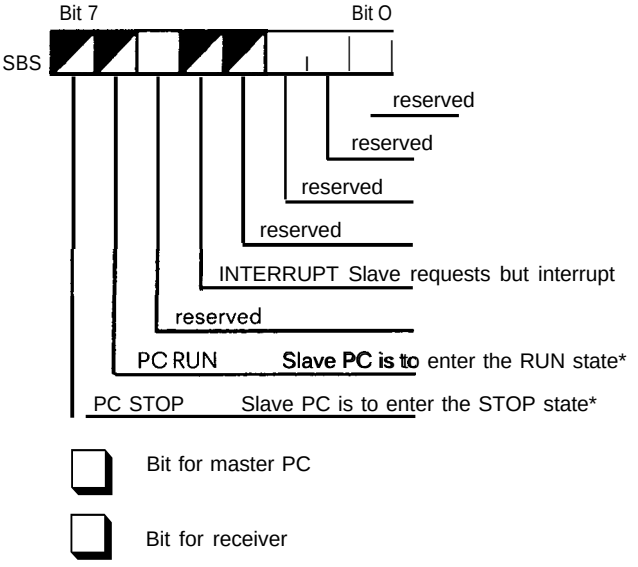
The master PC can use bit 7 of the SBS to put the addressed slave to the STOP mode and bit 6 for putting it into the RUN mode.

Bit 7 in the SBR, which the master PC receives from the slave, indicates whether the slave is in the STOP mode and bit 6 specifies whether it is in the RUN mode.

This RUN/STOP setting and acknowledgement is only possible in the case of slaves connected direct via the programmer interface.

Caution:
Even if only the SBS (status byte, send) is sent, at least two bytes must be transferred to the CP 530 by the send request (data handling blocks) (e.g. to switch directly connected CPUS to the STOP mode). These are the SBR (status byte, receive) and a meaningless slack byte (see table on p. 1 -6).





• Effective only with slave PCs connected direct via the programmer interface

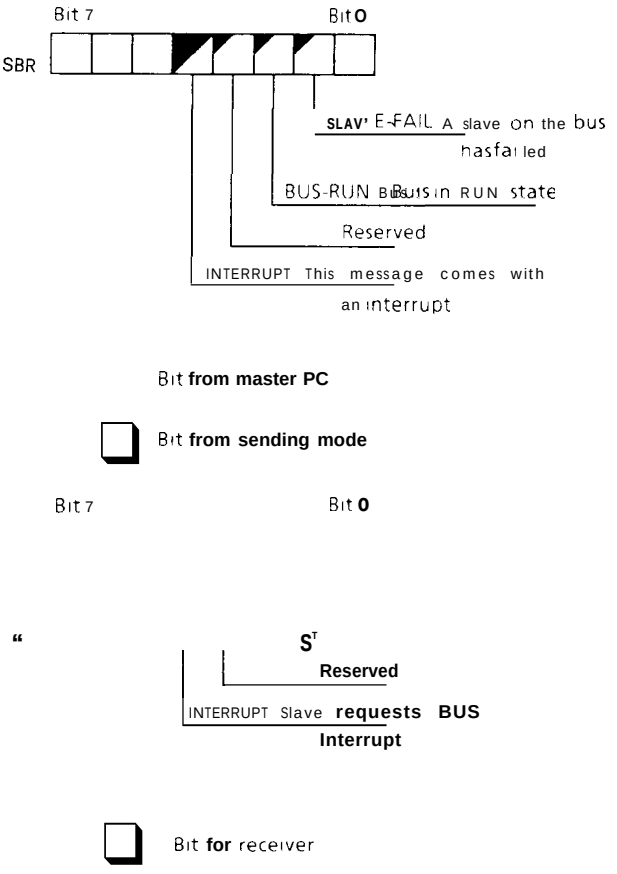
Fig. 1-7 Status bytea / CP 530 as master

Coordination information:

PC with CP 530 as slave

The "Permission to receive and send" and "Receive error" flag bits are not used.

Their function is replaced by information from the data handling blocks and may be evaluated by the CONTROL block (see Appendix).



Fig, 1-8 Status bytes/ CP 530 as slave

1.2 Normal Operation in the SINEC L1 Network

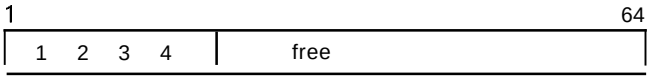
1.2.1 Polling List

The bus master (CP 530) requires a polling list for operating the SINEC LI network. This list contains the numbers of all slave PCs participating in the bus traffic and enables the master to detect and indicate failures (e. g.: via "BUS FAILURE" LED on the frontplate and network-wide by the setting of the "Slave failed" bit in the receive coordination byte (CBR, 1).

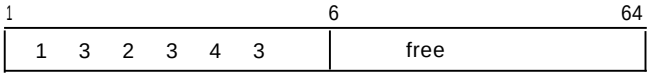
The user defines the bus cycle by specifying the number and sequence of the slaves in the list, i. e. the order in which the slaves are referenced by the master. The polling list contains a maximum of 64 slave numbers, which need not all be reserved; the bus cycle terminates with the last slave number entered and begins again from the beginning.

Priorities can be assigned by specifying a slave number more than once (slave 3 in the example).

Normal case without priorities:



Priority for slave 3



At the post office, there is a timetable telling the driver of the van the order in which the households are to be visited during a shift. – Normally, all are visited once; however, it may also happen, for example, that a small business concern (with the slave number "3") has to receive a better service.

1.2.2 Master ⟷ Slave Traffic

A slave cycle on the SINEC L1 network consists of the processing of one entry in the polling list; i. e. establishing a connection with this slave and exchanging status information and data. Since SINEC LI operates cyclically and completely asynchronously with the user program of the master and slave PCs, it may happen under certain circumstances that there is no data on the bus. On the other hand, data packets can also be sent in response to an input status (i. e. process-controlled). In this case, only the status bytes are sent. Data traffic takes place between the slave and master if the user program enters bus address "O" (direct) or "32" (request number for SEND block) as the data destination (bus master).

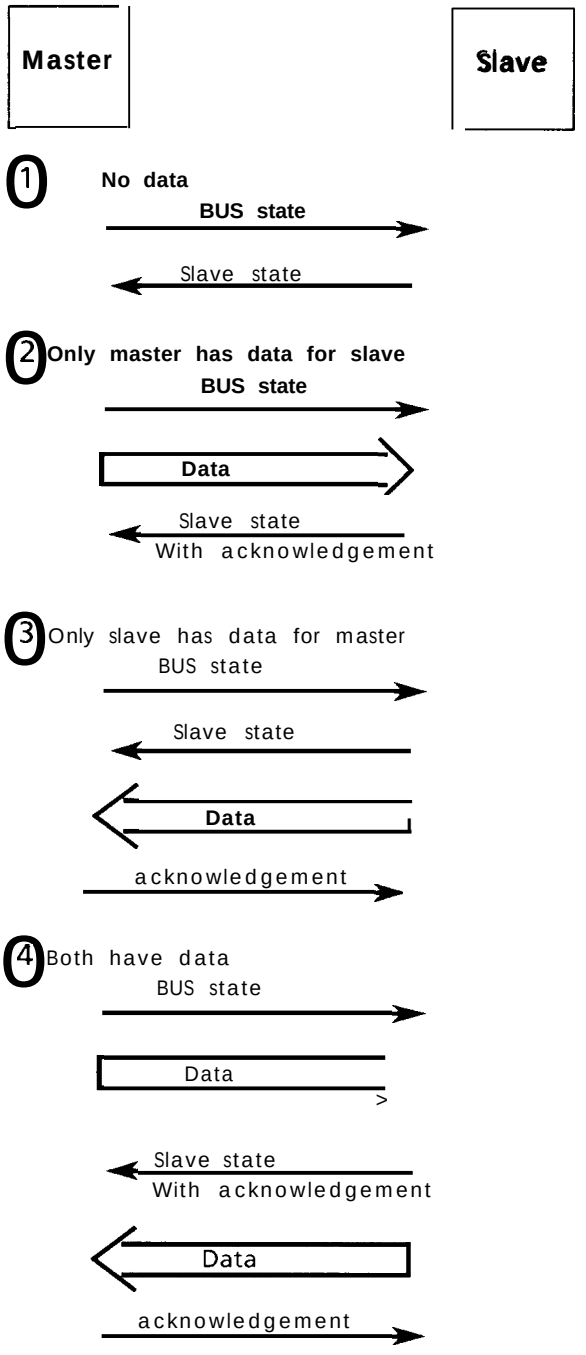


Fig. 1-9 Schematic of master — slave traffic

At the post office, these combinations have the following meaning:

① The mail van runs to and from the post office without parcels.
② The van brings a parcel to the recipient, but does not take a parcel back to the post office.
③ The van sets off for the receiver without parcels and brings a full parcel back to the post office.
④ The van delivers a parcel to the recipient and takes one back with it to the post office.

1.2.3 Slave ⟷ Slave Traffic

If the user program of the slave PC enters a destination number between 1 and 30 in the send mailbox, the bus automatically carries out slave-to-slave data traffic when the number in the polling list arrives at this slave in the course of the bus cycle. Slave-to-slave traffic is carried out without direct participation of the master.
The latter only performs monitoring functions.

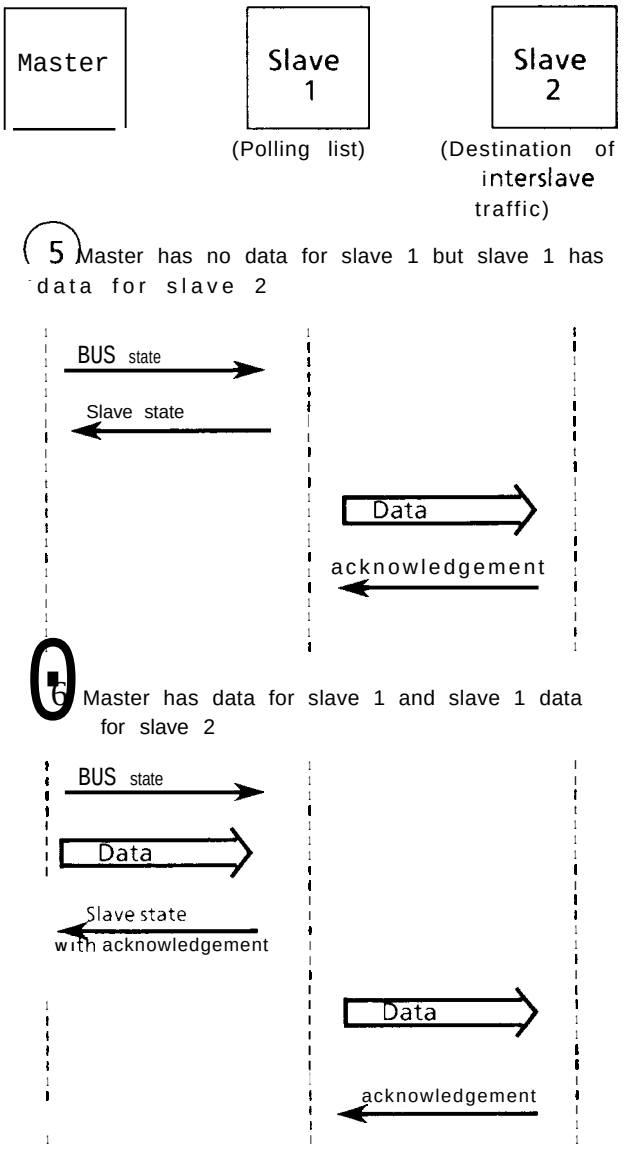


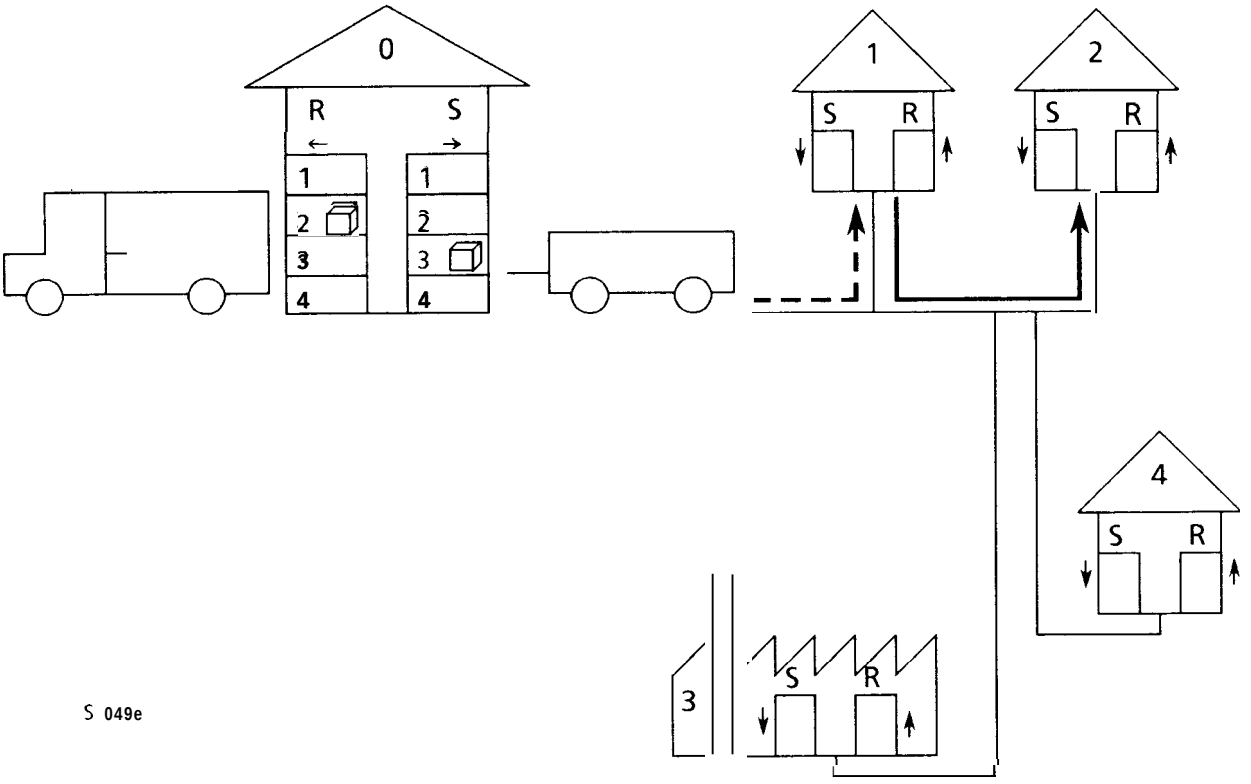
Fig. 1-10 Schematic of slave-slave traffic

Using the post office analogy:

The mail van heads for household 1, which is the next customer in the round.

⑤ Household 1 wishes to send a parcel direct to household 2. The van takes care of this immediately.

⑥ In this case, the van delivers a **parcel to household 1 before being requested,** as in ⑤, to deliver a **parcel directly** to household 2.



S 049e

Fig. 1-11 Slave-to-slave traffic in the post office model

1.2.4 Broadcasting

Each bus participant (master or slave PC) can send a message "to all" (broadcast) by using slave no. 31. In the case of the master, this message either consists only of the status byte (including STOP/RUN request to the PC) or of a status byte and data.

The slave can only broadcast if there is data to broadcast. Broadcasts are not acknowledged.

In the post office analogy, this is the equivalent of the post office or individual customers sending a circular to all other households including the post office.

1.3 Interrupting Normal Operation for Express Messages

1.3.1 Interrupt Mechanism in the User Program

During normal cyclic operation on the SINEC LI network, situations can arise where the user programs wish to send data immediately. This entails interrupting normal operation, i. e. an interrupt mechanism.
This is done with the “Interrupt” coordination bit (Section 1.1.4.4): In the event of an interrupt, the S5 program fills a send mailbox with data and sets interrupt bit CBS, 4 and the “Permission to send” bit. From this moment onwards, the operating system keeps trying to log this express message on with the bus master and thus send the message. As soon as the master recognizes this bid for the bus, it grants the slave requesting the interrupt an interrupt cycle and then resumes the interrupted bus cycle. The “Interrupt” information is sent along with the interrupt message and transmitted to the receiver as additional information in coordination byte CBR, 4. When an interrupt message reaches a CP 530 communications processor (master or slave), the CP 530 initiates a group interrupt on the internal PC interrupt line INTA and this interrupt can be handled in the interrupt organization block OB2 (S5-115/135 PCS),

1.3.2 Interrupt List

Should several interrupt requests be made simultaneously, the user specifies the order in which the interrupts are to be serviced by the bus master in an interrupt list on the CP 530. Only slaves whose numbers are in the interrupt list are allowed to place interrupt messages on the bus. The interrupt list contains up to 30 slave numbers and each slave may only be entered once.

Example:
Polling list: 1,2,3,4
Interrupt list:3, 1
Explanation: Slaves 1 to 4 are connected to the bus. If an interrupt request occurs, it can only be accepted if it comes from slave 1 or 3. Slave 3 has priority over slave 1.

Caution:
If data from the bus master and interrupt data of the slave total more than 100 bytes, the interrupt list should contain only 16 entries.

In the post office analogy, an alarm bell rings; the van driver calls the customers who are in his alarm list one after the other and goes to the customer wishing to send the express message.

1.3.3 Time Conditions in Connection with Interrupts

Interrupts are requested by the slave operating system during the addressing phase, i.e. interrupts are possible at each transition from the current slave to the next slave in the polling list, but not during ongoing slave data traffic.
(Fig. 1-12)

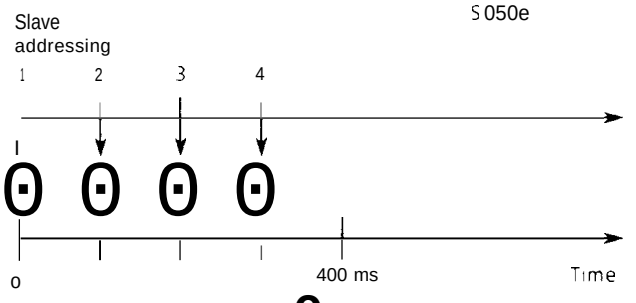


Fig. 1-12 Possible interrupt points: 0

Interrupt cycles are executed immediately they are recognized by the master and displace the bus cycle by the time required for the interrupt cycle.
(Fig. 1-13).

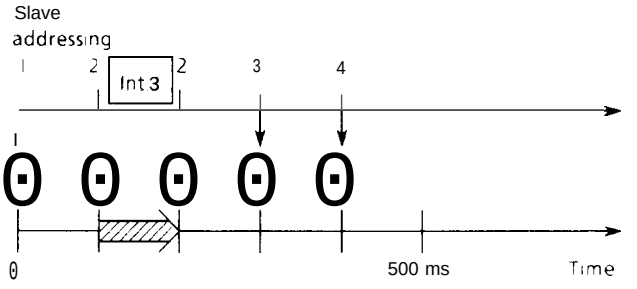


Fig. 1-13 Interrupt cycle with slave 3

If a number of interrupt requests are made simultaneously, an interrupt cycle is inserted (in the order specified by the interrupt list) and the bus cycle is extended accordingly. After dealing with the next slave in the polling list, the next interrupt cycle is inserted and the bus cycle once more extended accordingly (Fig. 1 -14) i.e. when several interrupts are requested simultaneously, the slaves in the polling list and interrupt list are processed alternately, keeping to the order in which they appear in the two lists.

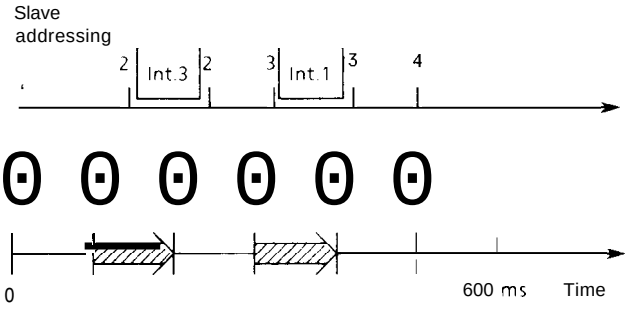


Fig. 1-14 Multiple interrupt requests at the same time

It is not only possible to exchange data with other nodes on the SINEC L1 LAN, but also to download programming functions in the other nodes.
Fig. 1 -15a gives a concise overview of the various configurations– ranging from the source programmer or PG/536 to the end node. These configurations can only be implemented with S5-DOS pro.grammers. The PG 615 programmer is capable of handling the PG - CP 530- END configuration only.

Conditions:

When implementing programmer functions, the CP 530 master on which they are to be executed must be in the STOP mode.

Addressing:

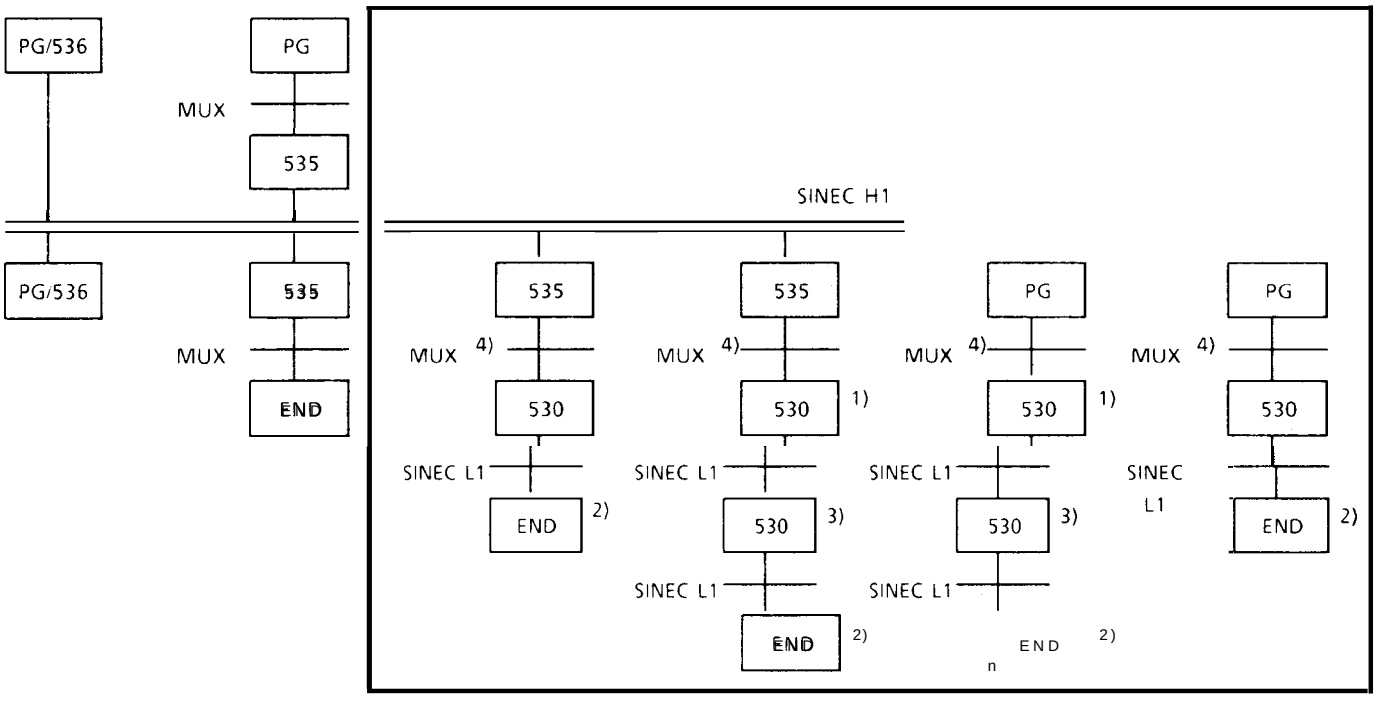
a) CP 530 master without polling list (gateway); only programmer functions can be downloaded.

b) CP 53o master with polling list (L1 bus master); In the RUN mode, normal L1 data traffic is carried out; In the STOP mode, programmer functions can be-implemented.

Constraints regarding the S5-135U and S5-150U:

Programmer functions are only possible in conjunction with an MUX; L1 data traffic is only possible with the CP 530 as slave.

1.4.1 Bus Selection Configurations



1) This CP 530 need not have a polling list (GATEWAY); the CP 530 is always in the STOP mode.
2) In the case of the S5-150 U and S5-135 U (single-processor mode), an MUX must be connected.
3) With this CP 530 master, the programmer bus address must be defined in line 7 of the SYSID.
4) MUX possible, but not mandatory.

Fig. I-15a Configurations for implementing programming functions on the CP 530

Caution: All nodes connected to the programmer bus must have programmer bus capability.

1.4 Programming Functions via the CP 530

CONFIGURATION No. 1 Programmer and Data Bus via One Bus Cable

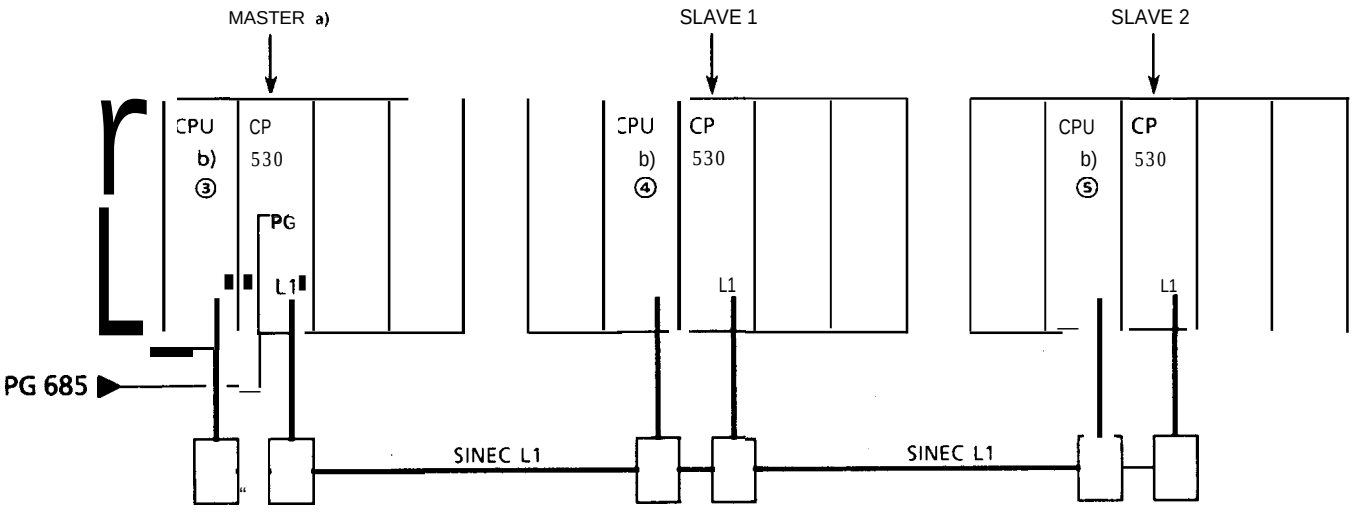
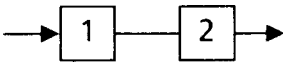


Fig. 1 -15b Programmer/data bus via one bus cable

a) Master programmed with polling list 

b) CPUs (e.g. 941/942) programmed as PROGR. BUS SLAVES, e. g. ③/④/⑤ (see Programming Example for programming of the CPUs as PROGR. BUS SLAVES).

Function: In normal LAN operation (L1 data bus), only slaves 1 and 2 are serviced by the MASTER. In programmer bus operation (master CP at STOP), programmer functions can be implemented on CPUs (SLAVES 3 to 5) after making the necessary bus selection in the menue.

CONFIGURATION No. 2 Programmer and Data Bus via Two Separate Bus Cables

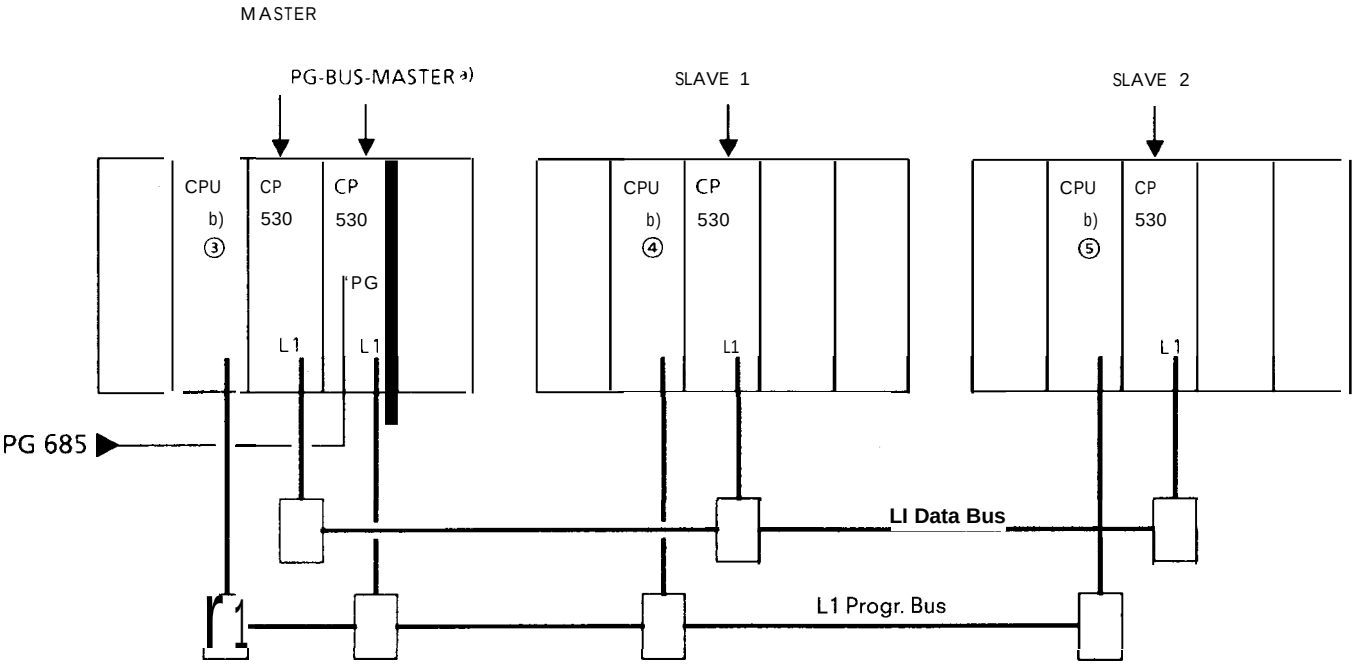


Fig. 1 -15c Programmer/Data bus via two separate bus cables

a) CP 530 as PROGR. BUS MASTER
The CP 530 is always in the STOP mode and does not require a polling list or SYNCH RON CALL if AUTO COLD RESTART "Y" is specified in the SYSID. In this case, the CP also does not require a page frame number (SSNR).

Caution: When programming the CP with the ON-LINE function (RAM/EEPROM), the CP must perform a defined start before carrying out any programmer functions. This is done either by putting the mode selector "- RUN - STOP" or automatically on each power recovery.

b) CPUs (e.g. 941/942) programmed as PROGR. BUS SLAVES, e.g.@)/@/@ (see Programming Example for programming the CPUs as PROGR. BUS SLAVES).

1.4 Programming Functions via the CP 530

CONFIGURATION No. 3 Programmer and Data Bus via Two Separate Cables with the 943 CPU

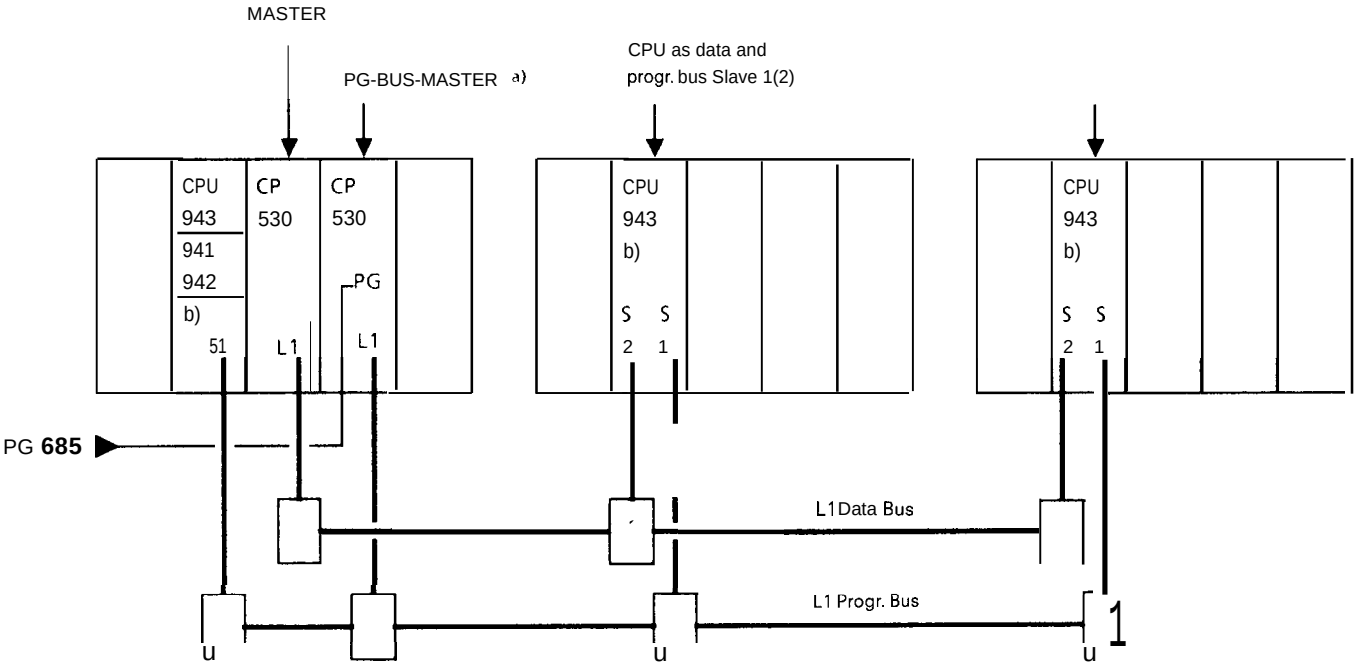


Fig. 1-15d Programmer/Data bus via two separate bus cables with the 943 CPU

a) CP 530 as PROGR. BUS MASTER
The CP 530 is always in the STOP mode, and does not require a polling list or SYNCHRON CALL if AUTO COLD RESTART "Y" is specified in the SYSID. In this case, the CP also does not require a page frame number (SSNR).

Caution: When programming the CP with the ON-LINE function (RAM/EEPROM), the CP must perform a defined start before carrying out any programmer functions. This is done either by putting the mode selector "- RUN - STOP" or automatically on each power recovery.

b) The 943 CPU is assigned a SINEC LI DATA SLAVE address and programmer bus address with the aid of the "L1-PG/DA" FB (see Programming Example).

Advantage: During normal data bus operation, programmer functions can be implemented on the CPU after making the necessary bus selection in the menue.

CONFIGURATION No. 4 Programmer and Data Bus with the S5-150U and S5-135U.

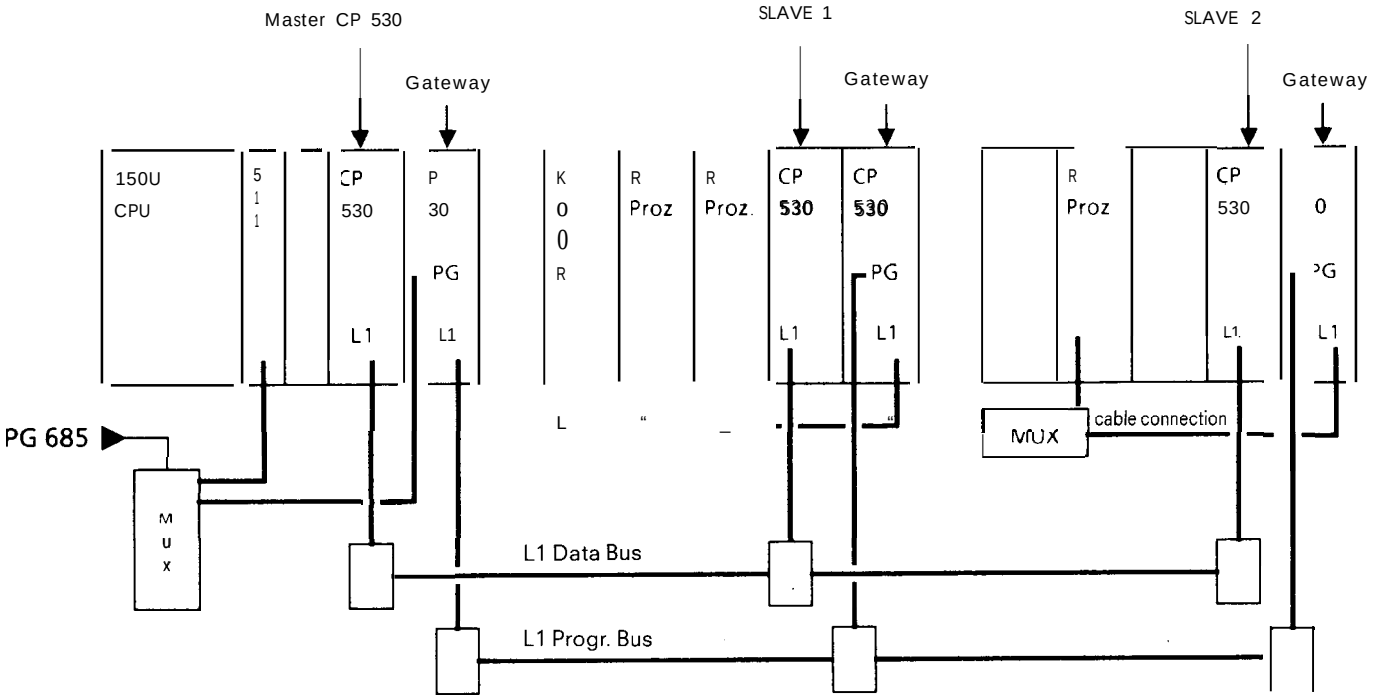


Fig. 1-15e Programmer/Data bus with S5-150 U and S5-135U.

The S5-150 U and S5-135 U programmable controllers can only be connected to the L1Progr. Bus via a CP 530 in conjunction with an MUX (single-processor mode).

Since the CP 530 used fulfills the functions of a gateway, make sure that the bus cable is plugged into the programmer connector of the CP 530.

1.4 Programming Functions via the CP 530

Example for Programming the CP 530 as a Gateway

	SYSID	CP 530
00	SUBMODULE IDENTIFIER:	
01	MODULE IDENTIFIER:	CP 530
02	FIRMWARE VERSION IDENTIFIER:	V2.0
03	PLANT DESIGNATION CODE:	EXAMPLE 530 GATEWAY
04	USER SOFTWARE GENERATION DATE:	xx.xx.xx
07	SLAVE NUMBER ON PROGR.BUS/SINEC L1 :	IMASTER
12	PAGE FRAME NUMBER (SSNR)	
13	AUTO, COLD RESTART:	Y (YES = Y, NO = N)
	BAUD RATE:	

Programming Example

Programming an S5-115U CPU to be connected to the SINECL1 local area network as a programmer bus node only

FBI

Segment 1
Name : PG-ADR
ID : PGAD I/Q/D/BIT/C: D KM/KH/KY/KS/KF/KT/KC/KG: K Y

```
:L RS 57                    Load SD 57

: LW      =PGAD             OR SD 57 and programmer
: O w                       bus address and write back
:TRS 57                     into SD 57
:BE

OB21      /OB22             Call FBs to specify progr. bus address in the initial start OBS

JU        FB1
NAME      :PG-ADR
PGAD      : KY 1,0          Progr. bus address= 1
```

Example for FB 255: Specifying Programmer bus and data slave addresses for S5-115 U CPUS on which data and programmer functions are to be possible (called in OBS 21/22).

```
FB255          SPRM-A
Segment 1
NAME : L1-PG/DA
ID     : PGDA    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : TCBR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID     : NCBR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : TCBS    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID     : NCBS    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : TSM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID     : NSM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : TRM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID     : NRM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
0044: LW =             PGDA LI PROGR. BUS/ LI
                       DATA BUS SLAVE NR.
    : T                FW200
UMW= FW
0040      : LW = PGDA  L1-PROGR.BUS/L1-DATA BUS-SLAVE-NO.
0042      :T   FW200
0044
0046
0048      : LW =TCBR   Type of coordination byte "R"
O04A      :T   FW202   (Receive)
O04C
O04E      : LW =NCBR   Address of CBR
0050      :T   FW203   DB or FB number/DW number
0052
0054      : LW =TCBS   Type of coordination byte "S"
0056      :T   FW205   (Send)
0058
O05A      : LW =NCBS   Address of CBS
O05C      :T   FW206   DB or FB number/DWnumber
O05E
0060      : LW =TSM    Type of send mailbox
0062      :T   FW208
0064      : LW =NSM    Address of send mailbox
0066      :T   FW209
0068      : LW =TRM    Type of receive mailbox
O06A      :T   FW211
O06C      LW = NRM     Address of receive mailbox
O06E      :T   FW212
0070
0072      :L   KHEED5  Transfer from F area into SD (system
0076      :L   KHEA7F  data area)
O07A      :TNB 14
O07C
O07E      :L   KHOOOO  Erase working flag words
0082      :T   FW200
0084      :T   FW202
0086      :T   FW204
0088      :T   FW206
O08A      :T   FW208
O08C      :T   FW210
O08E      :T   FW212
0090
0092      : BE
```

Meaning of the FB 255 parameters:

PGDA: Programmer bus address/data slave address
KY a, b
a) Programmer bus address
b) Data slave number

TCBR/TCBS: Type of coordination byte RECEIVE/SEND
KS ≙ possible identifier
FY ≙ flag byte
DW ≙ left-hand data byte

NCBR/NCBS: Number or address of coordination byte RECEIVE/ SEND
KYa, b
a) For type FY ≙ number of flag byte
For type DW ≙ number of data block
b) For type FY ≙ "O"
For type DW ≙ number of data word (left-hand data byte)

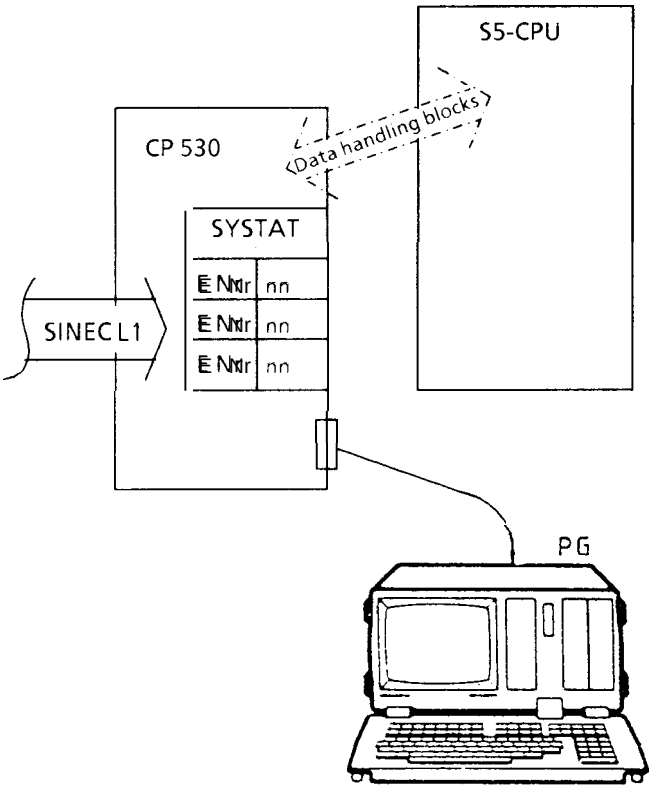
TSM/TRM: Type of SEND/RECEIVE mailbox
KS: FY ≙ flag byte and
DB ≙ data byte
are possible

NSM/NRM: Number of SEND/RECEIVE mailbox
KYa, b
a) Type FY ≙ number of flag byte at which the Send/Receive mailbox begins.
Type DB ≙ number of data block
b) Type FY ≙ "O"
Type DB ≙ number of data word at which the Send/Receive mailbox begins.

1.5 Error Messages CP 530— Master CPU

1.5.1 General

By using the RECEIVE 200 data handling block to read the SYSAT area, errors detected by the operating system of the CP 530 can be passed to the S5 user program.



A RECEIVE 221 maybe used at anytime to read the full SYSTAT area, or the RECEIVE 200 may be used to read only that error area which has a maximum of three error numbers. Each error message has two bytes – the error number and the supplementary error code; both are binary values.
If an error occurs, a call to RECEIVE 200 will cause the “Handshake meaningful” bit to be set.
After reacting to the new error, a call to RESET 200 may be used to cancel the error (or errors) in the SYSTAT area.

The RECEIVE 221 can be called at any time and the “Handshake meaningful” bit is always set.

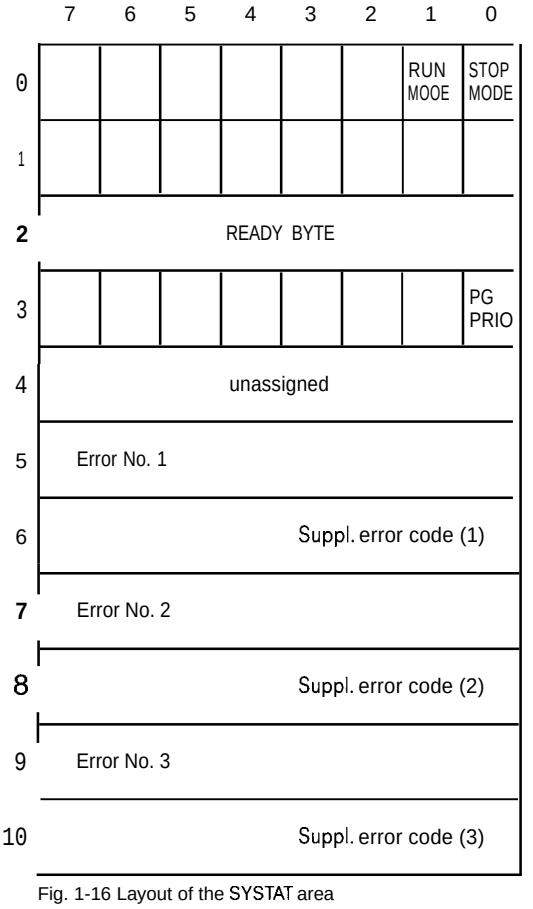


Fig. 1-16 Layout of the SYSTAT area

The error numbers may be broadly grouped in four classes, and most are accompanied by a supplementary error code to further define the problem (see the table "Error List").

a) Description of a group error number in more detail, e. g. the result of the self-test produces class 1 errors and these can only be eliminated by exchanging or repairing the module.
Example: 10,7 : Hardware error 7

b) Description relating to a request number containing an error.
Example: 54,177 : Request 177 not defined

c) Description@ slave with which the problem occurred.
Example: 60,3 : Slave 3 not in polling list.

1. Error Classes:

Division into error classes:
- Error types can be represented in compressed form.
- The class provides the user with information on how to eliminate the error.

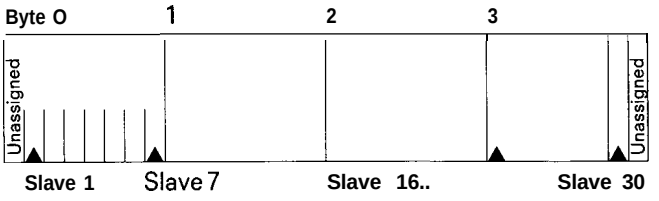
In addition to the errors described below, there are also a number of non-recoverable errors, which cannot be reported via SYSTAT since those parts of the system necessary to report the error, for instance, are faulted or missing.

Error Class

Class	Error No.	General Description	Operator reaction	Level of service
1	10-29	Possible hardware fault	Check HW Replace HW Repair HW	1. Operator 2. Service
2	30-49	Operational errors	Check Replace switch program submodule	Operator
3	50-69	Parameter assign-ment/programming error	Diagnostics by PG necessary S5-SW changes	Program-mer/ Syst. start-up engineer
4	70-90	Status messages	Record	Operator

RECEIVE 201 can be used to obtain a read-out ot all slaves that are recorded in the polling list but have not responded to the last call of the CP 530 master ("Slave failed").
This list is updated on every cycle as follows:
- "1" for slave has failed
- "O" for slave is OK

The data transmitted is four bytes long.
RECEIVE 201 is possible on the master CP only.



List: Failed slaves

1.5 Error Messages CP 530— CP

1.5.2 SYSTAT Error List

Class	Error No. (Decimal)	Suppl.	Description
I	10	xx	Error 10: Hardware Error No. XX
	11	XX	Error 11: Internal error message No. XX
II	30	○	Error 30: Waiting for SYNCH RON
	31	○	Error 31: Wrong CP module
	32	0	Error 32: PG priority active
	33	○	Error 33: PC is in Stop: noslavesend is possible
	34	○	Error 34: CPis not in the STOP mode.
	35	○	Error 35: CP RUN not possible: switch is set to STOP
III	50	○	Error 50: SYSID error
	51	○	Error 51: Polling list error
	52	0	Error 52: Interrupt list error
	53	XXX	Error 53: Request XXX not defined
	54	XXX	Error 54: Request identifier XXX unknown
	55	x x x	Error 55: Request No. XXX only for receive
	56	XXX	Error 56: Request No. XXX only for send
	57	x x x	Error 57: Request No. XXX only compatible with master
	58	x x x	Error 58: Request No. XXX only compatible with slave
	59	x x x	Error 59: Request too long
	60	x x x	Error 60: Slave XXX not in polling list
	61	x x x	Error 61: Request number XXX uses undefined list
	62	x x x	Error 62: Request number XXX SBS not allowed
Iv	70	○	Error 70: Bus error
	71	x x x	Error 71: Connection to slave faulted
	72	x x x	Error 72: The wrong slave (No. XXX) has replied
	73	XXX	Error 73: Slave No. XXX has failed

There are two activities involved when programming in the SINEC LI network:

a) Assigning the initial parameters to the CP 530 communications module(s) (SYSID)

b) Programming the messages (Send/receive/gate with the rest of the STEP 5 user program).

Programming is possible either with the PG 615 or PG 675 programmer. In the case of the PG 615, the activities under a) are carried out with a special COM 530/PG 615 operating system submodule and the activities under b) with the standard operating system of the PG 615.
With the PG 675, the activity under a) involves a "COM 530/ PG 675" diskette, while the activity under b) takes place with the standard LAD/CSF/STL package and the programs and parameters can be stored on a data diskette. When programming the CP 530 off-line with the PG 675 programmer, a program name must be entered; the parameter set for a CP is stored under this name on diskette and can be read back in when required.

Both programmers can be used off-line and on-line.

- Off-line: After they have been generated (e.g. on diskette), programs/ parameters are transferred to memory submodules, which are plugged into the receptacles on the corresponding modules. A memory space of 1 K bytes (max.) is required on the submodule for the CP 530.

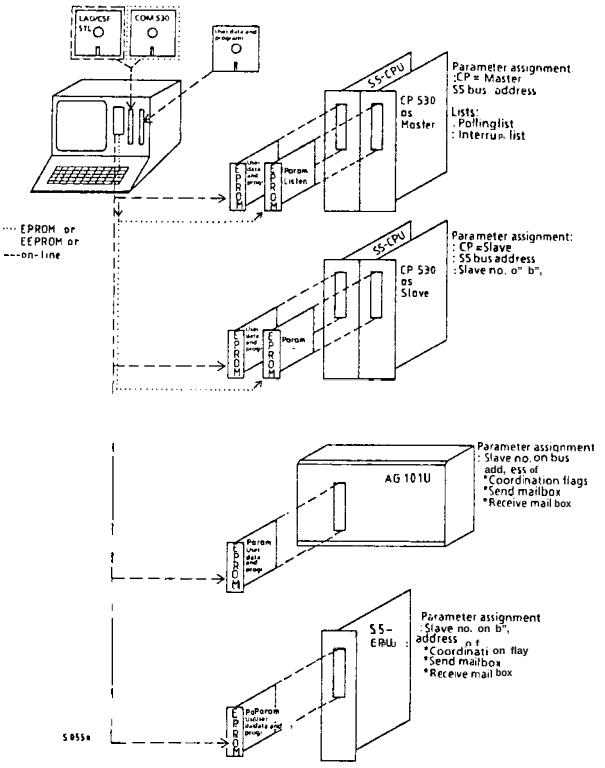


Fig. 2-1 Assigning the SINEC L1 parameters

- On-line: Once they have been generated, programs/parameters are transferred directly to the user memory of the modules.
  Condition: There must not be a user EPROM submodule plugged into the corresponding module.
  Diagnostic functions on the CP 530 are also possible in on-line mode. The operating mode is set in the basic menu of the programmer.
  A distinction must be made between four typical cases when programming the SINECL1 system (see Fig. 2-l).

1) Initialization of the CP 53o as master/slave

2) Programming the CPU of the master PC (S5-115U/S5-135U/S5-150U)

3) Programming the CPU of a slave PC with the CP 530 as slave (S5-115U/S5-135U/S5-150U)

4) Programming a PC as directly connected slave (S5-115U or S5-101U)

These four cases will be described below by way of example of a simple interconnected system.

2.2 Initializing the CP 530

2.2.1 SYSID (System Identification)

Initializing the CP 530 as master in the SINEC LI system.

The user defines the system characteristics by defining the SYSID parameters (system identification), proceeding as shown in the flowchart opposite (for COM 530).

Since the SYSID area has system-wide validity, not all the parameters are relevant for the CP 530.

An overview of the parameters is given in the table below.
ⓝ Mandatory parameter for SINEC L1
n Optional parameter

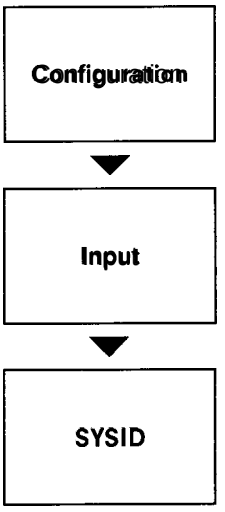


Fig. 2-2 Entry sequence on the programmer

Field	a) Definition	b) Examples	c) Formats	d) max. length
1	Module identifier	CP-530	ace. to. MLFB (ordering code)	8
2	Firmware version identifier	v-1 .2	free	8
3	Plant designation	Shop I PLNT4-AG7	free	19
4	Generation date	1—10/83	free	8
⑦	Slave No. on PG-/SINECL1	—/1 o –8/–4 –7/— —/—	P G PG/— Mini—/mm PG/Mini: PG/mm	5
⑫	Page No. (SSNR)	23	1-254	3
13	Automatic cold restart	Y	Y = automat. (a) cold restart	1
18	Baud rate on the SINECL1 bus*	9600 baud 300 baud		5

‘ not yet implemented

Table 2-1 SYSID for the CP 530

Meaning of the parameters (circled parameters are mandatory).

1) “CP 530” is entered as the module identifier; in this way, the user submodule can be easily identified as belonging to the SINEC L1 system.
When reading the SYSID in the “on-line” mode, the identifier is output from the firmware of the CP.

2) The user can enter the firmware status applying when his program was generated. When reading on-line, the version identifier is read out of the CP firmware.

3) Freely selectable (ASCII) characters

4) Freely selectable (ASCII) characters

⑦ O The CP 530 maybe a participant in a common programmer bus (PG bus); the number under which it can be referenced with programming functions via the programmer interface is between 1 and 30. “Blank” is entered if the programmer is connected directly to the CP 530.

0 The CP can be a participant or node in the SINECL1 network.
- The master must not be assigned a number ("Blank")
- A slave must be assigned a number between 1 and 30;

⑫ Page frame No.: The dual-port RAM of the CP 530 has a width of 1 K bytes; its location in the address area of the CPU is defined by entering the page frame number.
Area 1-255 e. g.:

Page frame No.	Base adress	Addr. range
1	I K	400 H - 7FF H
2	2K	800 H - BFF H
3	3K	COO H - FFF H

13 The cold restart characteristics define the behaviour of the CP 530 on recovery of the power supply:
O If "Y" is entered, the CP performs a cold restart and assumes bus operations (e. g. if there is no connection to the master Pc)
O If "N" is entered, the CP waits for a "SYNCHRON" command of the master PC before starting.

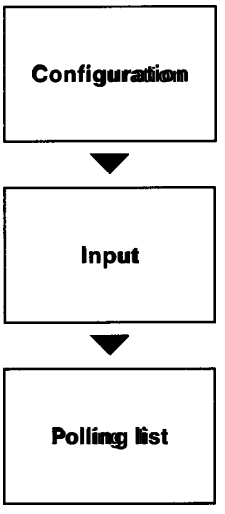
18 The baud rate for SINECL1 specifies the transmission speed on the bus; in this case, all slaves must be connected via the CP 530 communications processor and the same baud rate set (not yet effective; only 9600 baud possible at present).

Example:
Small' parts are manufactured in a number of sections in a factory shop. On 30 April 1985, initializing parameters are defined for the CP 530 bus master of plant section 1. The CP is plugged into the rack of an S5-115U and is addressed there with the Page No. 1. The CP is to start simultaneously with the CPU.

```
SYSID: 1 CP530
       2 .....,
       3 Small parts plant 1
       430.04.85

       12 : 1
       13 : N
```

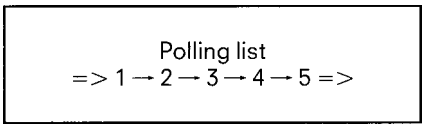
2.2.2 Polling List

The user can program the polling list by proceeding as shown in the flowchart opposite (for COM 530).
(see COM 530 Operating Instructions).

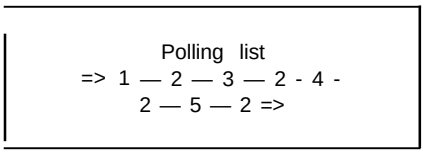


2.2 Initializing the CP 530

Five slaves on a bus are to be called in their natural sequence.

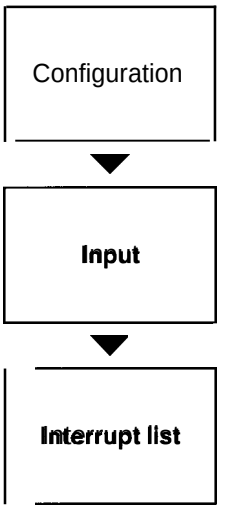


Slave 2 is to have a higher priority. This can be done by calling it more often than the other slaves in the bus cycle.

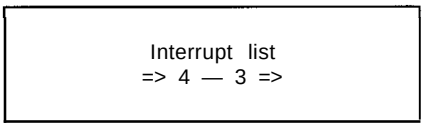


2.2.3 Interrupt List

The user can program the interrupt list by proceeding as shown in the flowchart opposite (for COM 530)
(see COM 530 Operating Instructions)



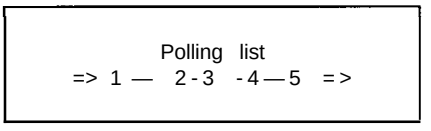
Of the five slaves in the polling list, slaves 3 and 4 are allowed to send interrupt messages.
If both wish to send at the same time, slave 4 is to have priority.

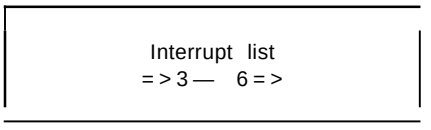


The interrupt list may contain slaves that are not in the polling list,

Advantage: These slaves can participate in bus traffic without loading the bus cycle if there is no need for cyclic data interchange.

Disadvantage: These slaves can fail without being noticed since they are not referenced cyclically by the master.





2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

2.3.1 General

This section describes data interchange between a CPU and a CP 530 plugged into the same rack as master or as bus slave.

The CPU can
- write
- read
- trace request states
- reset all requests
- initiate a cold restart

on the CP 530.

For these activities the CPU uses function blocks – referred to as data handling blocks – which are integrated in the operating system in the case of the CPU of the S5-115U. The function blocks are described in the Appendix.
In order to specify the request for the CP 530, the block is given request numbers defined according to Fig. 2-3. Only a subset of the requests is possible on the CP 530 acting as slave (Fig. 2-4).

CPU	Request number	Request identifier	Type of request
Writes	1	SEND	Slave 1
	30		Slave30
	31		to all slaves
	51		Interrupt to slave 1
	80		Interrupt to slave 30
	81		Interrupt to all slaves
	222		SYS ID CP
	42		Control byte, bus master
	43		Polling list
	44		Interrupt list
Reads	101	RECEIVE	Slave 1
	130		Slave 30
	100		Interrupt data
	200		Error list from SYSTAT
	201		List of failed slaves
	221		SYSTATCP
	223		SYS ID CP
	142		Control byte, bus master
	143		Polling list
	144		Interrupt list
Erases	200	RESET	Error list in SYSTAT
Erases	0	RESET	All requests on the CP (resetting of the CP)

Fig. 2-3 CPU as master

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

With requests 1 to 30 (51 -80) data is transferred to the CP 530 which the latter passes on to the slaves:

– Data from request no. 1 to slave 1
– Data from request no. 2 to slave 2
etc.

The SYSID, polling list and interrupt list can be transferred to the CP with requests 222,43 and 44.

Caution:
The CP must be in the "STOP" mode for these requests.

The bus can be started or stopped with request no. 42

The system identification of the CP can be written with request no. 222.

By analogy, data can be fetched from
slave no. 1 with request no. 101 and from
slave no. 2 with request no. 102
etc.

Arbitrary interrupt data can be copied to the CPU from any of the slaves with request no. 100.

The polling and interrupt lists can be transferred from the CP to the S5-CPU with requests nos. 143 and 144.

The system identification and CP status (SYSID and SYSTAT) can be transferred to the S5-CPU with requests nos. 223/221.

Before the first data requests (SEND/RECEIVE) are executed, the CP 530 must be synchronized with the CPU (see SYNCHRON).

The request RECEIVE 200 may be used to read the error list from the SYSTAT area, and request RESET 200 maybe used to reset the current errors.

In the master, the list of failed slaves can be read with request number 201.

CPU	Request number	Request identifier	Type of request
writes	1	SEND	Slave 1
	30		Slave 30
	31		to all nodes
	32		to master
	51		Interrupt to slave 1
	80		Interrupt to slave 30
	81		Interrupt to all nodes
	82		Interrupt to master
	222		SYSID CP
reads	101	RECEIVE	Slaves 1-30, master
	100		Interrupt data
	200		Error list in SYSTAT
	221		SYSTAT CP
	223		SYS ID CP
erases	200	RESET	Error list in SYSTAT
erases	0	RESET	All requests on the CP (resetting of the CP)

Fig. 2-4 CPU as slave

2.3.2 Programming Examples

2.3.2.1 Sending to a Slave

If input 14.0 changes its status from O to 1, a data word (net data) is to be sent from DB 11 (data = DW2) to slave 23.

```
FB2              SPRM-A

Segment 1
Name :S-Slave 23

000A                          Send to slave 23
000C
000E                          This block is scanned cyclically
0010      :
0012
0014      :C     DB10         Call DB1O
0016      :                   This block is for storing the
0018                          condition code words
001A
001C                          Input pulse edge evaluation
001E      :A     I4.0
0020      : AN   F2.1
0022      :=     F2.0         Edge flag
0024      :A     I4.0
0026      :=     F2.I
0028
002A
002C      :A     F2.0         Set flag for send initiation
002E      :S     F 1.1
0030
0032      :JU    FB247
0034 NAME : CONTROL
0036 SSNR :      KY0,I        Page frame number of the CP530
0038 A-NR :      KY0,23       Send to slave 23
O03AANZW  :      DW2
O03CPAFE  :      FY181
O03E
0040      :L     DW2          Relocate condition codeword
0042      :T     FWIO
0044
0046
0048      :
O04A
O04C
O04E
0050      :A     F 11.2       Evaluate positive-going edge
0052      : A N  F1.7         of the "Transmission completed
                              without error" message
0054      :=     F 1.6        "Transmission completed without
                              error" edge
0056      :A     F 1,2
0058      :=     F .7
O05A
O05C      : AN   F 82.0       No PAFE in last send request
O05E      :A     F .6         "Transmission completed without
                              error" flag
0060      :R     F .1         Reset send request
0062      :
0064      : A N  F11.1        No request pending
0066      :A     F 1.1        Send initiation flag
0068      :R     F 1.7        Auxiliary edge flag
O06A      : JC   FB244        Call function block FB244
O06CNAME  : SEND
O06E SSNR :      KY0,I        Page frame number of the CP530
0070 ANR  :      KY0,23       Send to slave 23
0072 ANZW :      DW4          DW4and DW5in DB10
0074 QTYP :      KSDB         Send mailbox is in DB area
0076 DBNR :      KY0,11       DB11
0078 QANF :      KF+1         From DW1
O07AQLAE  :      KF+2         2 words (net data = 1 word)
O07CPAFE  :      FY182        Flag byte 182
O07E
0080                          The first net data byte to be sent is
0082                          in DBI 1, DL2
0084
0086      : BE
```

2.3.2.2 Receiving from a Slave

```
FBI                    SPRM-A

Segment 1
Name : RECEIVE

000A        :                  Receive data from slave 23
000C                           in the master
000E        .                  This block is scanned cyclically
0010        :JU    FB247
0012 NAME : CONTROL
0014 SSNR :        KY0,1       Page frame number of the CP 530
0016 A-NR :        KY0,123     Receive from slave 23
0018 ANZW :        FW20
OOIAPAFE :         FY183       Flag byte 183
001C
OOIE        :
0020
0022
0024        : A    F21.0       Data available (bit Oof the ANZW)
0026
0028        :  JC  FB245       Function block FB245
O02ANAME : RECEIVE
O02CSSNR :         KY0, I      Page frame number of the CP530
O02E A-NR :        KY0,123     Receive data from slave 23
0030 ANZW :        FW24
0032 ZTYP :        KSDB        Receive mailbox in DB area
0034 DBNR :        KY0, I 1    DB 11
0036 ZANF :        KF+10       From DWIO
0038 ZLAE :        KF+3        3 words (net data from DW12)
O03APAFE :         FY184       Flag byte 184
O03C
O03E
0040        :                  The program section
0042        :                  "Evaluate data received"
0044                           should be scanned in the example
0046                           only if new data have been received
0048
O04A                           The first byte received (SBR)
O04C                           in the preamble of the receive data
O04E                           is used for this purpose in the
0050                           example.
0052                           This byte is written by the CP
0054                           every time data is received.
0056
0058        : C    DBI 1
O05A
O05C        : L    DL1O
O05E        : L    KBO
0060        : !=F
0062        : BEC              Program end, if no data have
0064                           been received
0066
0068                           ** Evaluate data received **
O06A
O06C                           The first net data byte received
O06E                           is in DB1 1, DL12 (left-hand byte)
0070
0072
0074                           Display (evaluate) data received
0076
0078        : L    DL12
O07A        : T    QB8
O07C
O07E
0080        : L    KBO
0082        : T    DL1O        Data have been evaluated
0084                           (Delete SBR)
0086
0088        : BE
```

2.3.2.3 Complete Example with the 150U as Master and the 135U/AG115U as Slave

Programming examples for the 115U PC as slave via the CP 530

```
0B22                SPRM-A    115U Pc

Segment 1
0000                          OB for power recovery
0002
0004        : A     F 255.7   Identifierfor FBI 11
0006        : R     F 255.7   Call from OB22
0008
000A
000C        :
000E        : JU    FB111     Call function block FBI 11
0010 NAME : START
0012
0014        : BE
```

```
FBI 11              SPRM-A    115U Pc

Segment 1
Name: Initial Start

000A        :
000C                          115U PC with the CP 530
000E
0010        : A     F 225.7   Identifier for entry
0012        : JC    = M001    lump if from OB21
0014
0016                  *
0018
001A        : L     KHIFFF    Time loop executed onlyon warm
001E M003 : L       KF+1      restart after power recovery
0022        : -F              (aprox. 5secswiththe941 CPU)
0024        : L     KF+0      Time required for CP hardware to
0028        : !=F             restart program
002A        : JC    =M002
002C        : TAK
002E        : JU    =M003
0030 M002 :
0032        :
0034 M001:
0036        :
0038        : JU    FB249     Call function block 249
O03ANAME : SYNCHRON
O03CSSNR :          KY0,1     Interface number 1
O03E BLGR :         KY0,5     Block size 256 bytes
0040 PAFE :         FY180     Initializing error byte
0042
0044        :
0046                  **
0048        : L     KHOIOO    Waiting time until CP
O04CMO05 : L        KF+1      completes restart Synchron
0050        : -F
0052        : L     KF+0
0056        : !=F
0058        : JC    =M004
O05A        : TAK
O05C        : JU    =M005
O05E M004 :
0060
0062        : BE
```

```
0B21                SPRM-A    115U Pc

Segment 1
0000                          OB for cold restart
0 0 0 2
0004        : A N   F 225.7   Identifier for FBI 11
0006        : S     F 225.7   Call from OB21
0008        :
000A        : JU    FB111     Call function block FBI 11
OOOCNAME : START
000E
0010        : BE
```

```
OB1                 SPRM-A    115U Pc
Segment 1

0000
0002        : A N   F 99.0    Send request in the clock pulse
0004        : L     KTIOO.O   example via flag 99.0
0008        : SE    T 1
000A        : A     T 1       Call function block FB111
000C        :  =    F 99.0
000E
0010
0012        : JU    FB32      Send tomaster(function block FB32)
0014 NAME : S-MASTER
0016 :
0018        : JU    FB101     Data receive (function block FB101)
001A NAME : RECEIVE
Oolc
001 E
0020                          Program section for changing
0022                          send data
0024        : A N   F 99.0
0026        : BEC
0028        : C     DB32
002A        : L     DL2
002C        : L     KBI
002E        : +F
0030        : T     DL2
0032        : BE
```

‘ LKH 7500 with the 942 CPU and the 943 CPU
“ LKH 0500 with the 942 CPU and the 943 CPU

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

```
FB32                SPRM-A    115U Pc

Segment 1
Name: S Master

000A                          Send to master
000C
000E        :A      F 99.0    Start send flag
0010        :S      F 1.1
0012
0014        :JU     FY247     Call function block FB247
0016 NAME : CONTROL
0018 SSNR :         KY0, I
OOIAA-NR  :         KY0,32    Send to master
001CANZW  :         FW10
001E PAFE :         FY181     Flag byte 181
0020
0022
0024
0026        :
0028
002A
002C        :A      F 11,2    Positive-going edge evaluation of
002E        :AN     F I . 7   the "Transmission completed with-
0030        :=      F 1.6     out error" message
0032        :A      F 11.2    "Transmission completed without
0034        :=      F1.7      error" edge
0036
0038        :AN     F 182.0   No PAFE in last send request
003A        :A      F 1.6     "Transmission completed without
                              error" flag
003C        :R      F 1.1     Reset send request
003E
0040        :AN     F 11.1    No request pending
0042        :A      F 1.1     Set flag for send initiation
0044        :R      F 1.7     Auxiliary edge flag
0046        :JC     FY244     Call function block FB244
0048 NAME : SEND
O04ASSNR  :         KY0,0
O04CA-NR  :         KY0,32    Send to master
O04E ANZW :         FW14
0050 QTYP :         KSDB      Send mailbox is in DB area
0052 DBNR :         KY0,32    DB32
0054 QANE :         KF+1      From DW1
0056 QLAE :         KF+2      2 words
0058 PAFE :         FY182     Flag byte 182
O05A
O05C                          The first net data byte to be
O05E                          sent is in DB32, DL2
0060
0062        :BE
```

```
FB1OI               SPRM-A    115U Pc

Segment 1
Name: Receive

000A                          Receive
000C
000E        :JU     FY247     Call function block FB247
0010 NAME : CONTROL
0012 SSNR :         KY0, I
0014 A-NR :         KYO,1O1
0016 ANZW :         FW20
0018 PAFE :         FY183     Flag byte 183
001A
OoIc
001E        :
0020
0022        :A      F21 .0    Data available
0024        :JC     FY245     Call function block FY245
0026 NAME : RECEIVE
0028 SSNR :         KY0, I
O02AA-NR  :         KYO,1O1   Receive (always No. 101 in case of
O02CANZW  :         FW24      slave)
002E ZTYP :         KSDB      Data receive mailbox in DB area
0030 DBNR :         KYO,1OI   DB 101
0032 ZANF :         KF+1      From DWI
0034 ZLAE           KF+3
0036 PAFE           FY184     Flag byte 184
0038
003A
003C                          The program section
003E                          "Evaluate data received"
0040                          is only to be processed
0042                          in the example when new data
0044                          have been received.
0046                          The first byte received (SBR)
0048                          in the preamble of the receive data
O04A                          is used for this purpose in the
O04C                          example,
O04E                          This byte is written by the CP
0050                          every time data is received.
0052
0054        :C      DB101
0056
0058        :L      DL1
O05A        :L      KBO
O05C        :!=F
O05E        :BEC              Program end if no data have been
0060                          received
0062
0064                          ** Evaluate data received **
0066
0068                          The first net data byte received is in
O06A                          DB101 DL3
O06C
O06E
0070                          Display (evaluate) data received
0072
0074        :L      DL3
0076        :T      AB8
0078
O07A
O07C        :L      KBO
O07E        :T      DLI       Data have been evaluated
0080                          (Delete SBR)
0082
0084        :BE
```

Programming example for the 135U PC (S processor) as slave via the CP 530

```
OB20                SPRM-B    135U PC

Segment 1
0000
0001        : A N   F 255.7   Identifier for FB111
0002        : S     F 255.7   Call from OB20
0003
0004
0005        : JU    FBI 11    Call function block FB111
0006 NAME : START
0007
0008        : BE


OB21                SPRM-B    135U PC

Segment 1
0000  :
0 0 0 1
0002        : A N   F 255.7   Identifier for FB111
0003        : S     F 255.7   Call from OB21
0004
0005
0006        : JU    FBI 11    Call function block FB111
0007 NAME : START
0008
0009                          Delete the condition codewords
000A                          of the send and receive requests
000B        : L     KB0
000C        : T     FW10
000D        : T     FW14
000E        : T     FW20
000F        : T     FW24
0010
0011        ' BE


OB22                SPRM-B    135U PC

Segment 1
0000
0001        : A     F 255.7   Identifier for FB111
0002        : R     F 255.7   Call from OB22
0003
0004
0005        : JU    FB111     Call function block FB111
0006 NAME : START
0007
0008                          Delete the condition codewords
0009                          of the send and receive requests
000A        : L     KB0
000B        : T     FW10
000C        : T     FW14
000D        : T     FW20
000E        : T     FW24
000F        :
0010        ' BE
```

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

```
FBI 11                SPRM-B    135U Pc

Segment 1
Name: Initial Start

0005        :
0006        :
0007        :
0008        : A     F 255.7     Entry identifier
0009        : JC    =M001       Jump if from OB20 or OB21
000A
000B        :L      KH3000      Time loop only executed after
OOODMO03 : L        KF+1        power recovery
000F        : -F                (approx. 5 secs)
0010        :L      KF+0        Time taken for CP hardware
0012        : !=F               to execute initial start program
0013        : JC    =M002
0014        : TAK
0015        : JU    =M003
0016 M002 :
0017        :
0018 M001:
0019
001A        : JU    FB125       Call function block FB125
001B NAME : SYNCHRON
001CSSDB :          DBIO        Queues DB (DB 10)
OOIDSSNR :          KY0,1       Interface number 1
001 E ANZW :        KYO,1O      Condition codeword FW 10
001F PAFE :         FO.5        Initializing error BIT F 0,5
0020
0021        : A     F 0.5
0022        :R      F99.7
0023"       : JC    =END        lump if PAFEerrorreported when
0024                            calling the FB Synchron
0025
0026        :                   Interlock flag
0027                            is reset
0028
0029
O 0 2 A .                       Start
O02B LOOP : JU FB126            Remove Synchron from queue
O02CNAME : ACTIVE
O02DSSDB :          DB1O
O02E        : A     F 10.6
O02F        : JC    = LOOP      lump till FB Synchron executed
0030
0031        :
0032        :
0033        :L      KH0400      Wait time till CP completes
0035 MO05 : L       KF+1        initial start
0037        : -F
0038        :L      KF+0
O03A        : !=F
O03B        : JC    =M004
O03C        : TAK
O03D        : JU    =M005
O03E MO04 :
O03F
0040        :                   FB Synchron executed without
0041        : AN F 10.4         error?
0042        : AN F1O.5
0043        : S     F99.7       User interlock flag
0044        :
0045
0046        :
0047 END : BE
```

```
OB1                   SPRM-B    135U Pc

Segment 1
0000
0001        :
0002
0003        : A     F 99.7
0004        : I C   FB126       Remove from queue (function block FB126)
0005 NAME : ACTIVE
0006 SSDB :         DB1O
0007
0008        :
0009
000A        : AN F 99.0         Send request initiation in clock
000B        :L      KT100.0     pulse example via flag 99.0
000D        : SE    T1
000E        : A     T 1
000F        : =     F 99.0
0010
0011        :
0012
0013        : JU    FB32        Send to master (function block FB32)
0014 NAME : S-MASTER
0015
0016        : A     F99.7
0017        : JC    FB126       Remove from queue (function block FB126)
0018 NAME : ACTIVE
0019 SSDB :         DB1O
001A
001B        :
Oo1c        : JU    FB101       Data receive (function block FB101)
001 DNAME : RECEIVE
001E        :
001 F
0020        : AN F 99.0         Change send data
0021        : BEC
0022        : C     DB32
0023        :L      DL2
0024        : L     KBI
0025        : +F
0026        :T      DL2
0027        : BE
```

```
FB32                SPRM-B   135U Pc

Segment 1
Name: S Master

0005                         Send to master
0006
0007        :A      F99.0    Set flag for send initiation
0008        :S      F1.1
0009
000A        :A      F99.7    Interlock if error in FB Synchron
000B        :JC     FB123    Function block FB123
OOOCNAME : CONTROL
OOODSSNR :          KY0,1
000E A-NR :         KY0,32   Send request to master
000F ANZW :         KY0,32   FW 10
0010 PAFE :         F0.1
0011
0012
0013        :A      F11.2    Positive-going edge evaluation of
0014        :AN     F1.7     the "Completed without error"
0015        :=      F1.6     message, "Completed without
0016        :A      F11.2    error"edge
0017        :=      F1.7
0018
0019        :AN     F10.4    No error
001A        :AN     F10.5    No error
001B        :AN     F0.2     No UELAerrorwhen sending
Oolc        :A      F1.6     "Completed without error" flag
001D        :R      F1.1     Reset send request
001E        :
001F
0020        :A      F99.7    Interlock if error in FB Synchron
0021        :AN     F14.6    No queue entry
0022        :AN     F11.1    No request pending
0023        :A      F1.1     Send request
0024        :R      F1.7     Reset auxiliary edge flag
0025        :JC     FB120    Function block FB120
0026 NAME : SEND
0027 SSNB :         DBIO
0028 A-NR :         KY0,32   Send to master
0029 ANZW :         KY0,14   FW 14
002A QTYP :         KSDB     Send mailbox in DB area
O02B DBNR :         KY0,32   DB32
002CQANF :          KF+1     From DW1
002DQLAE :          KF+2     2words
OOZE UELA :         F0.2
O02F
0030        :BE
```

```
FB101               SPRM-B   135U Pc

Segment 1
Name: Receive

0005                         Data receive
0006
0007        :A      F99.7    Interlock if error in FB Synchron
0008        :JC     FB123    Function block FB123
0009 NAME : CONTROL
OOOASSNR :          KY0,1
000B A-NR :         KY0,IO I Receive
OOOCANZW :          KY0,20   FW20
OOODPAFE :          F0.3
000E
000F        :
0010        :A      F99.7    Interlock if error in FB Synchron
0011        :AN     F24.6    No queue entry
0012        :A      F21.0    Data available
0013
0014        :JC     FB121    Function block FB121
0015 NAME : RECEIVE
0016 SSDB :         DBIO
0017 A-NR :         KY0,IO I Receive
0018 ANZW :         KY0,24   FW24
0019 ZTYP :         KCDM     Receive mailbox in DB area
OOIADBNR :          KY0,101  DB101
OOIBZANF :          KF+1     From DWI
OOICZLAE :          KF+3     3words
OOIDUELA :          F0.4
001E
001F
0020
0021                         The program section
0022                         "Evaluatedata received"
0023                         is only to be processed in the
0024                         example if new data have been
0025                         received.
0026                         The first byte received (SBR)
0027                         in the preamble of the
0028                         receive data is used for this
0029                         purpose in the example.
O02A                         This byte is written by the CP
O02B                         whenever data is received.
O02C
O02D
O02E        :C      DB101
O02F        :L      DLI
0030        :L      KBO
0031        :!=F
0032        :BEC             Program end if no new data have
0033                         been received
0034
0035                         "Evaluate data received"
0036
0037                         The first net data byte is in
0038                         DL30f DB101
0039        :L      DL3
O03A        :T      QBO
O03B
O03C
O03D        :L      KBO
O03E        :T      DLI      Data have been evaluated
O03F                         (Delete SBR)
0040
0041        :BE
```

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

Programming example for the 150U PC as master via the CP 530

```
OB20                SPRM-B   150U PC

Seament 1
0060                         OB for initial start
0001
0002          : AN  F 255.7  Identifier for FBIII
0003          : S   F 255.7  Call from OB20
0004
0005
0006          : JU  FB111    Function block FB111
0007          : START
0008          : BE
```

```
OB21                SPRM-B   150U Pc

Segment 1
0000                         OB for manual warm restart
0001
0002          : A N F 255.7  Identifier for FB
0003          : S   F 255.7  Call from OB21
0004
0005
0006          : JU  FB111    Function block FB111
0007 NAME : START
0008          : BE
```

```
OB22                SPRM-B   150U Pc

Segment 1
0000                         OBforwarm restart after power recovery
0001
0002          : A   F 255.7  FB identifier
0003          : R   F 255.7  Call from OB22
0004
0005
0006          : JU  FB" 11   Function block FB 11
0007 NAME : ANLAUF
0008
0 0 0 9
000A          : BE
```

```
OBI                 SPRM-B   150U Pc

Segment 1
0000                         Generate send criteria
0001                         (Example: send in clocked mode)
0002
0003          : AN  F 99.0
0004          : L   KT050.0
0006          : SR  T 1
0007          : A   T 1
0008          : S   F 99.0   Set flag for send initiation
0009          : =   F 99. I
000A          : A   F 99.0
000B          : L   KT050.0
000D          : SR  T2
000E          : A   T2
000F          : R   F 99.0
0010
0011                         FB Cycflag call for
0012                         distributing the data handling block
0013                         calls (time base)
0014                         (not required if calls are
0015                         distributed in the existing
0016                         user program)
0017          : JU  FB100    Function block FB100
0018 NAME : CYKFLAGS
0019
OOIA
OOIB          : A   F 112.2
Oolc          : I C F B 1 I  Send to slave 1
001 DNAME : S-SLAVE I
OOIE
OOIF          : A   F 112,4
0020          : JC  F B 1 2  Send to slave 2
0021 NAME : S-SLAVE2
0022
0023          : A   F 112,6
0024          : JC  F B 2 1  Receive from slave 1
0025 NAME : R-SLAVE I
0026
0027          : A   F 111.0
0028          : JC  F B 2 2  Receive from slave 2
0029 NAME : R-SLAVE2
O02A
O02B          :
O02C
O02D          : AN  F 99.1   Program section for changing
O02E          : BEC          send data
O02F          : C   DBI 1
0030          : L   DL2      For slave 1
0031          : L   KF+1
0033          : +F
0034          : T   DL2
0035          : C   DB12
0036          : T   DL2      For slave 2
0037          : BE
```

```
FBI 11              SPRM-B    150U  Pc
Segment 1
Name: Restart

0005                          Block for calling the
0006                          SynchronFB for the CP530
0007
0008
0009        :A      F 255.7   Evaluation of the entry identifier
000A        :JC     =M001     lump, if from 0820 and OB21
000B
000C
000D        :
000E
000F        :L      KB5       Time loop is only executed on warm
0010 M005 : L       KB1       restart after power recovery
0011        : -F              (approx. 5 secs)
0012        :T      FY255     Time elapsed until the CP
0013        :L      KHFFFF    hardware has executed the
0015 M003 : L       KB1       restart program
0016        : -F
0017        :L      KB0
0018        :!=F
0019        :JC     =M002
001A        :TAK
001B        :JU     =M003
001CM002 : L        FY255
001D        :L      KB0
001E        :!=F
001F        :JC     =M004
0020        :TAK
0021        :JU     =M005
0022 M004 :
0023        :
0024 M001:
0025
0026        :JU     FB185     Function block FB185
0027 NAME : SYNCHRON
0028 SSNR :         KY0,IO    Interface No. 10
0029 BLGR :         KY0,5     Block size 256 bytes
O02APAFE :          FY180     Initializing error (flag byte 180)
O02B
O02C
O02D
O02E
O02F        :L      KH2FFF    Time elapsed until CPfinished with
0031 M007 : L       KF+1      Synchron
0033        : -F
0034        :L      KHOOOO
0036        :!=F
0037
0038        :JC     =M006
0039        :TAK
O03A        :JU     =M007
O03B M006 :
O03C        :
O03D
O03E
O03F        : BE
```

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

```
FBI 1                SPRM-B    150U Pc

Segment 1
Name: S Slave 1

0005                           Send to slave 1
0006      :
0007      : A    F 99.0        Set flag for send initiation
0008      : S    F 1.1
0009      :
000A      : JU   FB184         Function block FB184
000B NAME : CONTROL
000C SSNR :      KY0, IO
000D A-NR :      KY0,1
000E ANZW :      FW10
000F PAFE :      FY181
0010
0011      :
0012
0013
0014
0 0 1 5
0016      : A    F 11.2        Positive-going edge evaluation of
0017      : AN   F 1 . 7       the "Completed without error"
0018      : =    F1.6          message, "Completed without
0019      : A    F 11.2        error" edge
001A      : =    F 1.7
001B
001C      : AN   F 182.0       No PAFE with the last send
001D      : A    F1.6          "Completed without error"
001E      : R    F 1.1         flag
001F                           Reset send request
0020      : AN   F 11.1        No request pending
0021      : A    F1.1          Send request pending
0022      : R    F 1.7         Reset auxiliary edge flag
0023      : JC   FB180         Function block FB180
0024 NAME : SEND
0025 SSNR :      KY0, IO
0026 A-NR :      KY0, I        Send to slave 1
0027 ANZW :      FW14
0028 QTYP :      KSDB          Send mailbox is in the DB area
0029 DBNR :      KY0, I 1      DB number 11
O02AQANF :       KF+1          From DW1
O02B QLAE :      KF+2          Send 2 words
O02CPAFE :       FY182         Flag byte 182
O02D
O02E                           The first net data byte to be
O02F                           sent is in DBI 1. DL 2
0030
0031
0032      : BE
```

```
FB12                 SPRM-B    150U Pc

Segment 1
Name: S Slave 2

0005                           Send to slave 2
0006
0007      : A    F 99.0        Set flag for send initiation
0008      : S    F 2. I
0009
000A      : JU   FB184         Function block FB184
000B NAME : CONTROL
000C SSNR :      KY0, IO
000D A-NR :      KY0,2
000E ANZW :      FW20
000F PAFE :      FY183         Flag byte 183
0010
0011
0012
0013      :
0014
0015
0016      : A    F21 .2        Positive-going evaluation of the
0017      : AN   F 2 . 7       "Completed without error" message
0018      : =    F2.6          "Completed without error" edge
0019      : A    F21 .2
001A      : =    F 2.7
001B
001C      : AN   F 184.0       No PAFE in last send request
001 D     : A    F2.6          "Completed without error" flag
001E      : R    F 2. I        Reset send request
001 F
0020      : AN   F 21.1        No request pending
0021      : A    F 2. I        Send request pending
0022      : R    F2.7          Reset auxiliary edge flag
0023      : JC   FB180         Function block FBI 80
0024 NAME : SEND
0025 SSNR :      KY0, IO
0026 A-NR :      KY0,2         Send to slave 2
0027 ANZW :      FW24
0028 QTYP :      KSDB          Send mailbox is in the DB area
0029 DBNR :      KY0,12        DB number 12
O02AQANF :       KF+1          From DWI
O02B QLAE :      KF+2          Send 2 words
O02CPAFE :       FY184         Flag byte 184
O02D
OOZE                           The first net data byte to be
O02F                           sent is in DB12, DL2
0030
0031      : BE
```

```
FB21          SPRM-B   150U Pc

Segment 1
Name: E Slave 1

0005                                Receive from slave 1
0006
0007    :JU   FB184                 Function block FB184
0008 NAME : CONTROL
0009 SSNR :   KY0, IO
000A A-NR :   KY0, IO I
000B ANZW :   FW30
000CPAFE  :   FY185                 Flag byte 185
000D
000E
000F    :
0010
0011    : A   F31 .0
0012    : JC  FB181                 Function block FB181
0013 NAME : RECEIVE
0014 SSNR :   KY0, IO
0015 A-NR :   KYO,1OI               Receive data from slave 1
0016 ANZW :   FW34
0017 ZTYP :   KSDB                  Receive mailbox in the DB area
0018 DBNR :   KY0,21                DB21
0019 ZANF :   KF+1                  From DWI
001A ZLAE     KF+3                  3 words
001 BPAFE     FY186                 Flag byte 186
001C
001 D
OOIE                                The program section
001 F                               "Evaluate data received"
0020                                is only to be processed in the
0021                                example if new data have been
0022                                received.
0023                                The first byte received (SBR)
0024                                in the preamble of the receive
0025                                data is used for this purpose
0026                                in the example.
0027                                This byte is described by the CP
0028                                whenever data is received.
0029
O02A    :C    DB21
O02B    :L    DB1
O02C    :L    KBO
O02D    :!=F
O02E    : BEC                       Program end if no data have
O02F                                been received
0030
0031                                **Evaluate data received **
0032
0033                                The first net data byte received
0034                                is in DB21, DL3
0035
0036
0037    :C    DB21
0038    :L    DL3                   Display data received
0039    : T   QB0
O03A
O03B
O03C    :L    KBO
O03D    :T    DLI                   Data have been evaluated
O03E                                (Delete SBR)
O03F    : BE
```

```
FB22          SPRM-B   150U Pc

Segment 1
Name: E Slave 2

0005                                Receive from slave 2
0006    :
0007    :JU   FB184                 Function block FB184
0008 NAME : CONTROL
0009 SSNR :   KY0,10
000A A-NR :   KY0,102
000B ANZW :   FW40
000CPAFE  :   FY187                 Flag byte 187
000D    :
000E    :
000F
0010
0011    : A   F41 .0                Data available
0012    : JC  FB181                 Function block FB181
0013 NAME : RECEIVE
0014 SSNR :   KY0,10
0015 A-NR :   KY0,102               Receive from slave 2
0016 ANZW :   FW44
0017 ZTYP :   KSDB                  Receive mailbox in DB area
0018 DBNR :   KY0,22                DB22
0019 ZANF :   KF+1                  From DW 1
001A ZLAE  :  KF+3                  3 words
OOIB PAFE :   FY188
0olc
OOID                                The program section
001 E                               "Evaluate data received"
001 F                               is only to be processed in the
0020                                example if new data have been
0021                                received.
0022                                The first byte received (SBR)
0023                                in the preamble of the receive data
0024                                is used for this purpose
0025                                in the example.
0026                                This byte is described by the CP
0027                                whenever data are received.
0028
0029    :C    DB22
O02A    : L   DLI
O02B    :L    KBO
O02C    :!=F
O02D    : BEC                       Program end if no data have
002E                                been received
O02F
0030                                **Evaluate data received **
0031
0032                                The first net data byte received is in
0033                                DB22, DL3
0034
0035    .
0036    : C   DB22
0037    :L    DL3                   Display data received
0038    :T    QB1
0039
O03A    :L    KBO
O03B    : T   DL1                   Data have been evaluated
O03C                                (Delete SBR)
O03D
O03E    : BE
```

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

```
FB100          SPRM-B     150U PC

Segment 1
Name: Cyc flags

0005    :                 Cycle flag shift register
0006    :
0007    : AN  F  112.1    1st cycle
0008    : AN  F 112.2     2nd cycle
0009    : AN  F 112.3     3rd cycle
000A    : AN  F 112.4     4th cycle
000B    : AN  F 112.5     5th cycle
000C    : AN  F  112.6    6th cycle
000D    : AN  F  112,7    7th cycle
000E    : AN  F 111.0     8th cycle
000F    : =   F 112.0
0010    :R    F 111.1
0011    :L    FWI 11
0012    : SLW 1
0013    :T    FW111
0014    : BE
```

2.3.2.4 Sending with Interrupt

```
FB3

Segment 1
Name: Interrupt

0014                                  This block is used for sending
0016                                  interrupt data to slave 1
0018
001A      : A     F 99.0              Send initiation edge
OoIc      : S     F 1.1               Set flag for send initiation
001E
0020      : JU    FB247               Function block FB247
0022 NAME : CONTROL
0024 SSNR :       KY0,I
0026 A-NR :       KY0,51
0028 ANZW :       FW10
O02APAFE  :       FY181               Flag byte 181
002C
002E      : A     F 11.2              Positive-going edge evaluation of
0030      : AN    F 1.7               the "Completed without error
0032      : =     F 1.6               message, "Completed without
0034      : A     F 11.2              error" edge
0036      : =     F 1.7
0038      :
003A      : AN  F 182.0               No PAFE with the last send
003C      : A     F 1.6               "Completed without error"
003E      : R     F 1.1               Reset send initiation
0040      :
0042      : AN F 1 1.1                "Request running"
0044      : AN    F 1.1               Start send
0046      : R     F 1.7               Reset auxiliary edge flag
0048
004A      :  JC FB244                 Function block FB244
O04CNAME : SEND
004E SSNR :       KY0,I
0050 A-NR :       KY0,51
0052 ANZW :       FW10
0054 QTYP :       KSDB
0056 DBNR :       KY0,100
0058 QANF :       KF+1
O05AQLAE  :       KF+4
O05CPAFE  :       FY182               Flag byte 182
O05E
0060      : BE
```

N. **B.:**
With the S5-150U, the RECEIVE 100 request must be in the cyclic program section since no OB2 entry is initiated. With the S5-1 15U/135U, the RECEIVE 100 call can be programmed in OB2, **as in the example. Please note that the RECEIVE 100 request can fail as a result of a PAFE error 91 and, should this be the case, OB2 is exited without the interrupt data being fetched.**

Remedy:
Repeat the request in OB2 or initiate an additional CONTROL/RECEIVE **100 in the cyclic user program.**

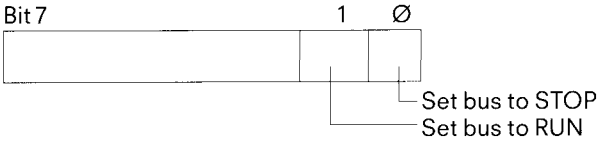
2.3.2.5 Receiving Interrupt data

```
OB2                 115U Pc
Segment 1
0000      :
0002                                  This block is used for receiving
0004                                  interrupt data, which have
0006                                  been received via the CP530
0008
000A
Oooc      :                           When an interrupt message is
000E                                  received, the CP 530 initiates
0010                                  a branch from the normal user
0012                                  program totheinterrupt OB2
0014      :                           (IR-A cable)
0016      : JU    FB 247              Function block FB247
0018 NAME : CONTROL
OOIASSNR  :       KY0,1               Caution with the 150U PC
001CA-NR  :       KYO,100             * * * * * * * * * * * * * * * * * *
001EANZW  :       FW30                Since the 150U PC has no interrupt
0020 PAFE :       FY 199              cable, there is also no branching
0022      :                           into the interrupt OB
0024                                  Remedy: Call Receive 100 cyclical-
0026                                  ly via Control
0028
002A      : A     F 31.0
002C      : JU    FB245               Function block FB245
002E NAME : RECEIVE
0030 SSNR :       KY0,1               Page frame No. of the CP530
0032 A-NR :       KYO,100             receive interrupt data
0034 ANZW :       FW30
0036 ZTYP :       KSDB
0038 DBNR :       KY0,110             DBI 10
O03AZANF  :       KF+1                From DW1
O03CZLAE  :       KF+5                5 words (net data 3 words)
003E PAFE :       FY200               Flag byte 200
0040      :
0042
0044      :
0046
0048      : BE
```

2.3 Initializing and Programming the S5-CPU with the CP 530 as Master or Slave

2.3.2.6 Send/Receive "Bus Master" Control Byte

If a slave connected to the bus fails, the bus is to be brought into the STOP state by way of the control byte (request 42). The control byte is not sent on to the slaves but is used only as control information for the master CP.



The bus will stop at the end of the current bus cycle, i. e. at the end of the polling list.

FB 101 -Bus-Stop SPRM-A

Segment 1 LAE = 43 SYM
Name: Bus-Stop Page 1

```
000A      :C     -Databox DB 11
000C
000E
0010
0012
0014      : AN F80.0      BUS STOP?
0018
001A      :JC    = JUMP   lump to program end
001C
001E      :
0022
0024      :
0026      :
0028      :L     KBI      Initialize control byte in
002A                      send mailbox =BUS "STOP"
002C      :T     DLIO
002E
0030      :C     -Cond. code DB 10 Call condition code DB
0032
0034      :JU    -SEND    FB 244 SEND call,
0036 NAME : SEND          Function block 244
0038 SSNR :       KY0.I   With request
O03AA-NR  :       KY0.42  Send control byte INFO "BUS
O03CANZW  :       DW20    STOP"
O03E QTYP :       KSDB
0040 DBNR :       KY0.I 1 The control byte is in
0042 QANF :       KF+10   data word IOof DB 11
0046 QLAE :       KF+1
0048 PAFE :       -PAFE-SEN FB 200
O04A      :
O04CIUMP  : BE
```

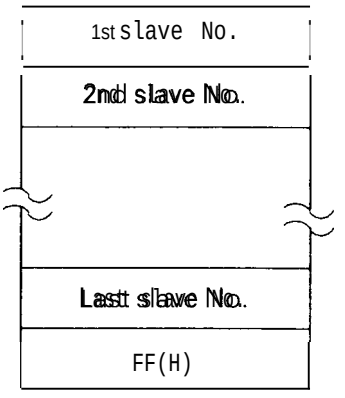
PAFE-SEN	= FB200	Parameter assignment error for send, flag byte 200
Databox	= DB 11	Send and receive mailbox for SINEC L1
Cond. code	= DB 10	Condition code double words for send and receive
Send	= FB244	Standard function block for CPs

The RUN mode can be reached again by setting bit 1 ("BUS RUN").
The bus cycle starts with the first slave in the polling list.

2.3.2.7 Sending and Receiving Lists

Like data, the polling and interrupt lists are transmitted with SEND/RECEIVE. After lists are written to the CP, the bus performs a RESTART, i. e. all send and receive mailboxes are erased, or requests (including those from the programmer) are cancelled and the bus cycle is restarted at the beginning of the polling list.
The interrupt and polling lists have the same format; the interrupt list has a maximum length of 30 bytes, while the polling list has a length of 64 bytes.

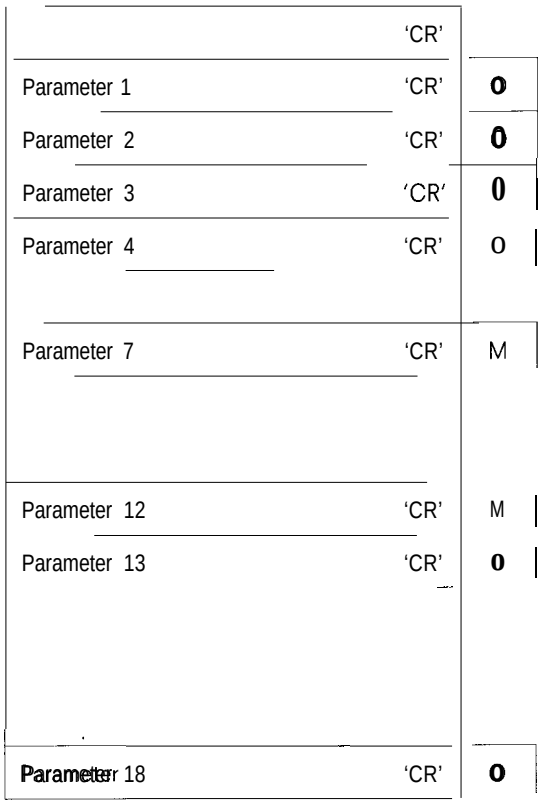
Note:
Make sure the CP is in the STOP mode before writing any lists.



```
AB5 FB5                    115U Pc

Segment 1
Name: Polling list

000A                       This example shows howa
000C                       polling list can be written,
000E                       using data handling blocks.
0010                       The CP 530 must first be
0012                       switched to "Stop" for this
0014                       purpose (e. g. using "Control
0016                       byte, bus master").
0018                       The "Slave numbers" data for
001A                       the polling list are stored
001C                       in DB20 from DW1 onwards.
001E                       The Send Block call should
0020                       be issued once only and is
0022                       monitored via a Control call
0024                       (call no. 34) in the cyclic
0026                       user program.
0028
002A

002C       :
002E
0030                                        "Enterwith RLO = "1" "
0032       :
0034       :JU   FB244      Function block FB244
0036 NAME : SEND
0038 SSNR :      KY0,I      Page frame no. of CP530
O03AA-NR :       KY0,43     Write polling list
003CANZW :       FW60
O03E QTYP :      KSDB
0040 DBNR :      KY0,20     Data in DB20
0042 QANF :      KF+1       from DW1 onwards
0044 QLAE :      KF+4       4 words
0046 PAFE :      FY201
0048       :
O04A
O04C
O04E       :
0050
0052       : BE

DB20       SPRM-A

0          KS= UMLI
1          KY= 001,002;
2          KY= 001,003;
3          KY= 001,004;
4          KY= 001,005
5
```

The SYSID list consists of various parameters, (see Section 2.2.1, p. 2-3). Each of these parameters is terminated with a carriage return 'CR', i. e. ODH. Parameters that are not assigned consist of only 'CR', and only ASCII characters maybe used.

Layout of the SYSID list



O = Optional
M = Mandatory

The SYSID DB **must** be generated in 'KH' format. The terminating parameter 'CR' cannot be represented in 'KS' format.

2.4 S5-101 U as Slave

2.4.1 Initializing the S5-101U

The SYSID function has not yet been implemented for the S5-101 U programmable controller. The slave number should be programmed by STEP 5 statements.
The slave number must be identified as a constant with the first two statements in the user program and, to distinguish it from applications without SINEC LI, identified by a SYSTEM flag, e. g.:

1st statement SF 63.0 – (unused SYSTEM flag)
2nd statement L KF 11 — slave number= 11

The coordination bytes for receiving (CBR) and sending (CBS) are permanently defined in flag bytes FY 61 for receive and FY 62 for send.

Send and receive buffer for the S5-101 U

S5-101U PC: DB 1

	LEFT-HAND BYTE	RIGHT-HAND BYTE
DW . .		
DW . .		

Receive mailbox

DW 40	Length of data packet	Source of data
DW 41	1st data item	2nd data i tern
DW 72	63rd data item	64th data item

Send ma i 1 box

DW 80	Length of data packet	Destination of data
DW 81	1st data item	2nd data item
DW 112	63rd data item	64th data item

The S5-101 U can receive and send a maximum of 64 data bytes in a block. A fixed area is reserved in data block 1 of the PC for the SINECL1 receive and send mailboxes.

2.4.2 Program Examples

2.4.2.1 Receiving

Example:
A slave receives data only if no other slave connected to the bus has failed.

```
      AF61.1        A slave has failed
      JC = MO02
      AF61.7        Permission to receive= 1 data invalid
      JC = M001                          = O data valid

      LDL41         Store data from flag byte 36 onwards
      TFY36
      LDR41
      TFY37
      LDL42
      TFY38
      LDR42
      TFY39

      AN F61.7
      SF61.7        Receive mailbox has been
                    evaluated and slave may
M001 : :            accept new data.
```

A slave receives four items of data and prepares itself for receiving new data

```
Data receive
      AF61.7        Permission to receive= 1: Received
      JC = M 001    data invalid
      LDL41         Store 1st item of data received in FY36
                    (flag byte)
      TFY36
      LDR41         Store 2nd item of data received in FY37
      TFY37
      LDL42         Store 3rd item of data received in FY38
      TFY38
      LDR42         Store 4th item of data received in FY39
      TFY39
      AN F61.7
      S F61.7       Receive mailbox has been evaluated
                    and slave may accept new data
M 001:
```

A slave receives data only if it has been sent from the master.

```
Data receive
      AF61.7        Permission to receive= 1: Data invalid
      JC = M 001                         = O: Data valid
      LDR40         Source = master?
      L KF +0
      > < F
      JC = M 002

      LDL41         1st item of data received in FY36
                    (flag byte)
      TFY36
      LDR41         2nd item of data received in FY37
      TFY37
      LDR42         3rd item of data received inFY38
      TFY38
      LDR42         4th item of data received in FY39
      TFY39

M 002: AN F61.7
      S F61.7       Receive mailbox evaluated and
                    slave can accept data
M 001:
```

2.4.2.2 Sending

A slave sends two items of data to the master.

```
      A F 62.7      If permission to send = 1, new
      JC = M 001    data may no longer be written
                    into the send mailbox.

      L KF+0        Destination = enter master in
      T DR 80                     send mailbox
      L KF+2        Length = enter2 bytes in
      T DL 80                     send mailbox

      L FW 50       Transfer 2 bytes to send mailbox
      T DW 81

      AN F 62.7
      S  F 62.7     Transfer contents of send mailbox
M 001 :             to bus
```

Interslave **traffic: A slave sends two items of data to slave 3**

```
      A F 62.7      If permission to send = 1, new
      JC = M 001    data may no longerbewritten
                    into the send mailbox.

      L KF=3        Destination = slave 3
      T DR 80
      L KF = 2      Length = 2 bytes
      T DL 80

      L FW 50       Transfer 2 bytes to the send mailbox
      T DW 81

      AN F 62.7
      S  F 62.7     Transfer contents of send mailbox to
M 001 :             bus
```

Broadcasting: A slave sends two items of data to all nodes participating in the bus

Caution:

This data is not acknowledged by any of the receiving nodes!

```
      S F 62.7      If permission to send = 1,
      JC = M 001    new data may no longer be written
                    into the send mailbox

      L KF+31
      T DR 80       Destination = "to all"
      L KF+2        Length = 2 bytes
      T DL 80

      L FW 50       Load 2 data bytes into send mailbox
      T DW 81

      AN F 62.7
      S  F 62.7

M 001 :
```

Interrupt: A slave sends an express message once onlv to the master **PC:**

```
      ONF 32.0      No interrupt (edge-triggered flag)
      O  F 62.4     Interrupt is not processed;
                    mailbox must not be changed
      A F 62.7
      JC = M 001

      L KF+0        Destination = master
      T DR 80
      L KF+2        Length = 2
      T DL 80

      L FW 50       Transfer 2 data bytes to
      T DW 81       send mailbox

      AN F 62.7
      S  F 62.7     Transfer contents of send mailbox
      S  F 62.4     to bus with interrupt request

M  0 0 1 :
```

A slave is only to send new data if the last transmission was free of error (CBR, **bit no. O = F 62.0); otherwise the send request is repeated.**

The user can define the number of repetitions by inserting a counting loop at the position marked@

```
      AF 62.7       If permission to send = 1,
      JC = M001     no new data maybe written into
                    the send mailbox
⊛     AF 62.0       If previous error:
      JC = M 002    Repeat message

      LK KF+0       Destination = master
      T DR 80
      L KF+2        Length = 2 bytes
      T DL 80

      L FW 50       Transfer 2 data bytes to
      T DW 81       send mailbox

      AN F 62.7
M O 0 2 : S F 62.7  Transfer contents of the
                    send mailbox to bus
M 001:
```

2.5 S5-115U as Slave

2.5.1 Parameter Assignment using SYSID

If the CPU of the S5-115 is connected direct to the SINEC LI network, it reacts in a similar way to the S5-101 U, the only differences being the coordination flags and the position of the send and receive mailboxes. The coordination flags are defined as a flag word by the user with the aid of the SYSID function. The left-hand byte contains the receive coordination flag bits and the right-hand byte the send coordination flag bits (parameter 11).
The send mailbox (parameter 12) is defined in a similar manner: it may be located in the data or flag area. In the case of data blocks, the user specifies the DB number and the data word number from which the mailbox starts in the data block.
In the case of flags, the number of the flag word at which the mailbox begins is specified.

The receive mailbox (parameter 13) is similar to the send mailbox; both mailboxes maybe located in different areas and in different DBs.

The table below contains an overview of the SYSID data possible in the S5-115 CPU. Since the SYSID function is not yet supported by programmers on CPUs, parameters cannot be assigned with SYSID as described below, but via OB22 using STEP 5 statements. This is described in Section 2.5.2.

General parameters

Field	Definition	Examples	Formats		Max. length
1	Module identifier	CP 530 IP nnn 931 B CPU	ace. to MLFB (ordering code)		8
2	Firmware version identifer	V– 1.2 Z03	free		8
3	Plant identification	Shop 1-M/C-4AG7	free		19
4	Generation date	2--10/83	free		8
7	Slave No. on programmer (PG) bus/ SINECL1	--/10 *) -8/-4 -7/_	PG SINECL1 PG/SINECL1:	PG/-- --/n PG/mm	5

Special parameters for the S5-115U

Field	Definition	Examples	Formats	Max. length
⑪	Address of coordination flag word	123	Flag area of the S5-115U	3
⑫	Address of send mailbox Area: Block No.: *1) Word address:	D-37-235 F-10	D = data block F = flag area	8
⑬	Address of receive mailbox Block No.: *1) Word address:	D-4-1 F-17	D = data block F = flag area	8

Explanation of special parameters 11 to 13

@Address of the coordination flag word
The number of a flag word or byte at which the coordination flags begin is specified here:
The first byte contains the coordination flags for RECEIVE and the next byte in memory the coordination flags for SEND. The coordination flags must be located in the non-retentive flag area (otherwise there is a danger of coordination errors).

@Address of the send mailbox:
This mailbox may be located in the data or flag area (“D” or “F”). If located in the “D” area, a valid “DB” number must be specified as well as the word address in this DB at which the send mailbox begins; if in the “F” area, a valid word address is necessary in the flag area.

⑬ Address of the receive mailbox
As for the send mailbox;
the send and receive mailboxes may be located in the same area or data block or independently of each other in different areas or different DBs.

2.5.2 Parameter Assignment from OB 22

The operating system of the 941 CPU supports operation of the SINEC LI master (CP 530) by means of data handling blocks which are available as integral function blocks from software release Z08 onwards.

2.5.2.1 Using the SINEC L1 LAN Bus

With each data transmission on the LI, control and status information is also sent. This information is made available to the STEP 5 user program via a coordination byte. The actual data in receive and send mailboxes which the user can access with load and transfer operations.

Slave number:

The slave number is used to assign the number by which it can be addressed on the L1 bus. It must lie in the range 1–30 inclusive.

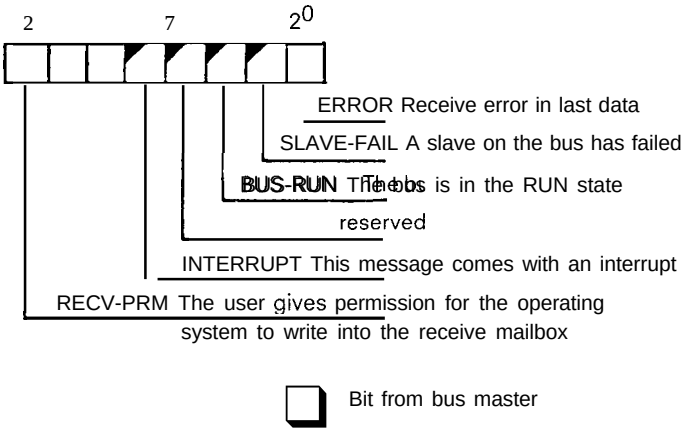
Coordination bytes Send/Receive (CBS/CBR)

These bytes represent the interface between the user program and the operating system. They are set as follows by the operating system on warm restart following power recovery.

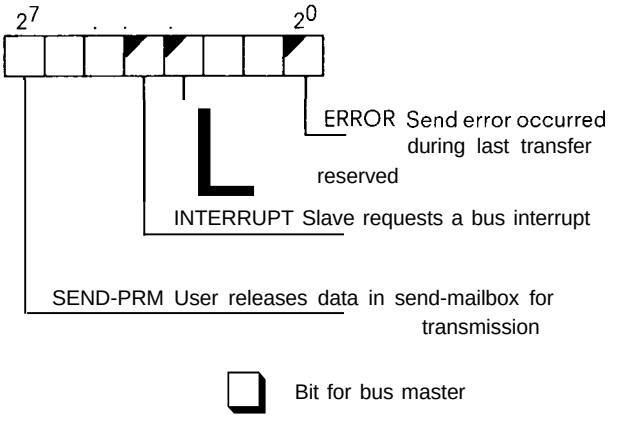
CBR = 80 H = ==> Permission to receive
CBS =00 H

2.5 S5-115U as Slave

Coordination Byte, RECEIVE **(CBR)**



Coordination Byte SEND (CBS)



The coordination bytes are managed by the operating system and may not be used for any other purpose.

The Send Mailbox:
This may be up to 66 bytes long. The user must place the data that is to be sent in the send mailbox starting at byte 3.
The layout is as follows:

Byte 1	Length of the net data (O. . . 64)	Byte 2	Destination No. (0... 30 or 31) 1)	Net data
Byte 3	1st data byte	Byte 4	2nd data byte	
Byte 5	3rd data byte	Byte 6	4th data byte	
. . . .				
Byte 63	61st data byte	Byte 64	62nd data byte	
Byte 65	63rd data byte	Byte 66	64th data byte	

1) Sl_{ave}. No. 0 ≙ Master
31 ≙ Broadcast

The Receive Mailbox:
The receive mailbox may be up to 66 bytes long.

The layout is as follows:

Byte 1	Length of net data (O. 64)	Byte2	Source No. (o. 30)'	Net data
Byte 3	1st data byte	Byte 4	2nd data byte	
Byte 5	3rd data byte	Byte 6	4th data byte	
Byte 63 Byte 65	61st data byte 63rd data byte	Byte 64 Byte 66	62nd data byte 64th data byte	

2) Slave No. O ≙ Master

2.5.2.2 Parameter Assignment for the SINECL1 Slave Firmware

When using the S5-115U CPU type 941/942/943, it is possible to program the location of the coordination bytes and of the send and receive mailboxes. The SINECL1 parameter block is in the system data area SD from word 57 onwards.

The SINECL1 parameter block:

SD 57	**PG bus address**	**Slave number**
SD 58	CBR **Location type**	CBR **DB or flag number**
SD 59	CBR DW number	**CBS** **Data identifier**
SD 60	**CBS** **DB or flag number**	**CBS** DW number
SD 61	**SM** **Data identifier**	**SM** **DB or flag number**
SD 62	**SM** DW number	**RM** **Data identifier**
SD 63	**RM** **DB or flag number**	**RM** **DW number**

CBS/CBR **(Coordination Bytes, Send/Receive)**

The locations of the coordination bytes are defined in each case by three data locations
They can be defined either as a flag byte or in the left-hand byte (DL) of the data word.

Coordination byte in flag area:

Byte 1	Data identifier	"F" (ASCII code)
Byte 2	Flag number	0...255
Byte 3	–	Irrelevant

Coordination byte in data block:

Byte 1	Data identifier	"D" (ASCII code)
Byte 2	DB	2...255
Byte 3	DW	0...255

The coordination bytes are in the left-hand byte (DL) of the data word.

SM/RM (Send/Receive Mailbox)
The beginning of the send and receive mailboxes is defined by three data bytes in each case. Parameter assignment is as described for the coordination bytes.

Mailbox overflow
If more data is received than will fit within the mailbox, the mailbox overflows and the data is lost. No error message results.
The end of the receive mailbox is determined as follows: for the flag area this is flag byte 255 and for the data area the last data word. In 256 word long data blocks it is data word 255.

2.5 S5-115U as Slave

Example for programming SINEC L1.
The parameters are set in (OB21) OB 22, using auxiliary FB 255 to enter the parameters. Any other unassigned FB could have been used instead of FB 255.

```
FB 255          SPRM-A
Segment 1
NAME  : L1-PG/DA
ID    : PGDA    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID    : TCBR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID    : NCBR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID    : TCBS    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID    : NCBS    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID    : TSM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID    : NSM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID    : TRM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KS
ID    : NRM     I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY

0040      : LW  =PGDA     LI -PG-BUS/LI-DATABUS-SLAVE-N O.
0042      :T    FW200
0044
0046
0048      : LW  =TCBR     Type of coordination byte “R”
O04A      :T    FW202     (Receive)
O04C
O04E      : LW  =NCBR     Address of CBR
0050      :T    FW203     DB or FB number/DW number
0052
0054      : LW  =TCBS     Type of coordination byte “S”
0056      :T    FW205     (Send)
0058
O05A      : LW  =NCBS     Address of CBS
O05C      :T    FW206     DB or FB number/DW number
O05E
0060      : LW  =TSM      Type of send mailbox
0062      :T    FW208
0064      : LW  =NSM      Address of send mailbox
0066      :T    FW209
0068      : LW  =TRM      Type of receive mailbox
O06A      :T    FW211
O06C      : LW  =NRM      Address of receive mailbox
O06E      :T    FW212
0070
0072      :L    KHEED5    Transfer from F area into SD (system
0076      :L    KHEA7F    data area)
O07A      :TNB  14
O07C
O07E      :L    KH0000    Erase working flag words
0082      :T    FW200
0084      :T    FW202
0086      :T    FW204
0088      :T    FW206
O08A      :T    FW208
O08C      :T    FW210
O08E      :T    FW212
0090
0092      : BE
```

Meaning of the FB255 parameters:

PGDA: Programmer bus address/ data slave address
KYa, b
a) Programmer bus address
b) data slave number

TCBR/TCBS: Type of coordination byte RECEIVE/SEND
KS ≙ possible identifier
FY ≙ flag byte
DW ≙ left-hand data byte

NCBR/NCBS Number or address of coordination byte RECEIVE/SEND
KY a,b
a) For type FY ≙ number of flag byte
For type DW ≙ number of data block
b) For type FY ≙ “O”
For type DW ≙ number of data word (left-hand data byte)

TSM/TRM: Type of SEND/RECEIVE mailbox
KS: FY ≙ flag byte and
DB ≙ data byte are possible

NSM/NRM: Number of SEND/RECEIVE mailbox
KYa, b
a) Type FY ≙ number of flagbyte at which the Send/Receive mailbox begins.
Type DB ≙ **number of data block**
b) Type FY ≙ “O”
Type DB ≙ number of data word at which the Send/Receive mailbox begins.

2.5.3 Programming Examples

```
Send to test date integrity
OB 21          SPRM-A                    LNG = 13 ABS
                                          Page 1

Segment 1
0000      :
0002      :                 115U CPU as slave
0004
0006      :JU   FB113       Function block FB113
0008 NAME : SINECL1
000A
000C
000E      : BE
```

```
OB 22          SPRM-A                    LNG = 14 ABS
                                          Page 1

Segment 1
0000
0002
0004      :                 115U CPU as slave
0006
0008      :JU   FB113       Function block FB113
000A NAME : SINECL1
000C      :
000E
0010      : BE
```

```
FB 113         SPRM-A

Segment 1
Name: SINEC-L1

000A                        Call function block FB255 for
000C      :                 direct connection of an S5-115U
000E      :                 941/942 CPU as slave to the
0010      :                 SINECL1 network bus
0012      :JU   FB255       Function block FB255
0014 NAME : L1-PG/DA
0016 PGDA :     KY0,1       Slave No.: 1
0018 TCBR :     KSFY        Type of CBR is "FB" (flag byte)
001A NCBR :     KY61,0      CBR is flag byte 61
001CTCBS :      KSFY        Type of CBR is "FB" (flag byte)
001 E NCBS :    KY62,0      CBR is flag byte62
0020 TSM  :     KSDB        Type of send mailbox "DB"
0022 NSM  :     KY1OO,8O    Send mailbox in DB100 from DW80up
0024 TRM  :     KSDB        Type of receive mailbox in "DB"
0026 NRM  :     KY100,40    Receive mailbox in DBIOOfrom DW40 up
0028
002A      : BE
```

Caution:

A CPU acting as slave returns data to the Sender. The returning of data is monitored by evaluating Send Error Bit CBS bit O and, if necessary, repeated.

```
OB 1           SPRM-B                    LNG = 34ABS
                                          Page 1

Segment 1
0000                        115U CPU as slave on
0002      :                 SINEC LI network bus
0004
0006
0008      :JU   FB 1        Function block FBI
000A NAME : SINECL1
000C
000E      : BE
```

```
FBI            SPRM-A

Segment 1
NAME :SINECL1

000A      : A   DB 100      Data-DB
000C      :
000E      : O   F 1,0       Disable receive flag
0010      : O   F 61,7      Receive CBRdata flag
0012      :JC   =JUM1       lump if no data received
0014      :
0016      : AN  F 1,0       Disable setting of receive flag
0018      : S   F 1,0
001A
Oolc      :L    DW40        Relocate length/source to
001E      :T    DW80        length/destination
0020      :L    DW41        Relocate data
0022      : T   DW81
0024      :
0026      : AN  F 61,7      Message from user progr. ==> BESY
0028      : S   F 61,7      Receive mailbox free
002A      :
002CJUM1  :
002E      : AN  F 1,0       End if no data received
0030      : BEB
0032
0034      : O   F 1,1       Disable send flag
0036      : O   F62,7       Send CBS flag
0038      :JC   =JUM2       lump if no receive mailbox free
003A
003C      : AN  F 1,1       Set flag for "Send disable"
003E      : S   F 1,1
0040      :
0042      : AN  F62,7       Send message from user
0044      : S   F62,7       progr. ==> BESY
0046      :R    F 1,2       Reset aux. edge flag
0048 JUM2 :
004A      :                 Edge evaluation for Send
004C                        completed
004E      : AN  F 62,7      CBS Send flag
0050      : AN  F 1,2       Aux. edge flag
0052      :  =  F 1,3       "Send completed" edge flag
0054      : A   F 1,3
0056      : S   F 1,2
0058      :
O05A      :A    F 1,3       "Send completed" edge
O05C      : AN  F62,0       "Send error" CBS flag
O05E      :R    F 1,0       Receive enable
0060      : R   F 1,1       Send enable
0062      :A    FI03        "Send ready" edge
0064      :A    F62,0       "Send error" CBS flag
0066      :R    F 1,1       Send enable for repeat
0068
O06A      : BE
```

2.6 Example of Small Parts Plant

Parameter Assignment with the CPU 102

Declarations:

- Coordination byte, receive (CBR) → flag byte FY 100
- Coordination byte, send (CBS) flag byte FY 101
- Send mailbox data block DB2 from DW0
- **Receive mailbox** (RM) **data block** DB3 **from** DW0
- Flag bytes FY64 to 77 are **used as buffers.**

FB10 SPRM-A

NAME: PARA 102

```
L   KF   1        - Load slave No.
T   FY   65         andstorein flag byte 65
L   KH   4D00     - Load "Flag" data identifier
T   FW   66         and store in flag byte 66
L   KY   100.0    - Load flag byte lOOand
T   FW   67         store in flag byte 67
L   KH   4D00     - Load "Flag" identifier and
T   FW   69         store in flag byte 69
L   KY   101.0    - Load flag byte 101 andstore
T   FW   70         in flag byte 70
L   KH   4400     - Load "Data word" identifier
T   FW   72         and store in flag byte 72
L   KY   2,0      - Load DB No. "2" and DB No. "3"
T   FW   73         in flag bytes 73 abd 74
L   KH   4400     - Load "data word" identifier and
T   FW   75         store in flag byte 75
L   KY   3.0      - Store DB No. "3" and DW No. "0"
T   FW   76         in flag bytes 76 and 77
                  - Transferflagarea FY 64 to
                    77tothesystem data area
L   KH   EE4D     - Load upper source address
L   KH   EA7F     - Load upper destination address
T   NB   14       - Transferdata frameof 14 Bytes
                  Reset all buffers.
L   KH   0000     - Load hexadecimal number "0000"
T   FW   64       - Set all bits of FY 64 to 77 to "O"
T   FW   66
T   FW   68
T   FW   70
T   FW   72
T   FW   74
T   FW   76
                  CBR default setting:
                  Data can be received from the bus,
L   KH   0080     - Load binary number lOOOOOOO
T   FY   100      - Set bit 7 to "I" and bits 6...0 to "O"
                  CBS default setting:
                  Program has access to the Send mailbox

L   KH   0000     - Load binary numberOOOOOOOO
T   FY   101      - Set bits 7... 0 to "0"
BE                Block end
```

Programming Example with the CPU 102

The control program has to handle the following tasks:

. The Send and Receive mailboxes must be enabled, and the data in these mailboxes processed.

- The coordination bytes must be managed (e. g. send job, error evalution)

Example:

Data traffic with the Master as Slave 1

Declarations:

- Slave 1 receives 3 bytes from Master O.
- The information is stored in the PIQ (QB0, QBI, QB2).
- Slave 1 sends 3 bytes (IB0, IB1, IB2) to the Master.
- Parameter assignments is implemented in FBI, as shown in Fig. 11.6

Programming the individual blocks:

```
    OB22            SPRM-A
    SEGMENT 1
    JU   F1         OB22 executes once following power-up,
                    It calls FBI, which assigns the slave parameters,
    BE
    OB1             SPRM-A
    SEGMENT I
    JU   FB2        OB1 executes cyclically, It calls FB2, which services
                    the Send and Receive mailboxes,
    BE
    FB2             SPRM-A
    SEGMENT 1
    NAME: PROG 102
    c    DB3        Receive mailbox (DB3)
    A    F100.7     Check whether access to Receive mailbox is allowed.
                    CBR/bit 7 = O: access allowed
                    CBR/bit 7 = 1: access prohibited
    JC   = MO01     Skip Receive box evaluation if access prohibited
    L    DRO        Check whether number of source (Master 0)
    L    KF + O     is in byte 2 of the Receive mailbox
    ><F
    JC   = MO02     Skipevaluationof Receive mailbox
                    if source No, = O
    L    DLI
    T    ABO
    L    DRI        Transfer Receive mailbox to
    T    QB1        the PIQ
    L    DL2
    T    QB2
M2: AN   F100.1     Set CBR/bit 7to "1", i.e. permit PC access
                    Program access isnot permitted again
    s    F100.7     until the PC has reset this bit.
MI: A    F1OI.7     Check whether access is permitted to Send
                    mailbox,
                    CBS/bit 7 = O: access permitted
                    CBS/bit 7 =1: access prohibited
    JC   = MO03     Skip evaluation of Send mailbox
                    if access prohibited
    c    DB2        Call Send mailbox (DB2)
    L    KF + 3     Specify length of data packet
    T    DLO        in byte 1 of Send mailbox
    L    KF + O     Load destination No, 0 (Master)
    T    DRO        in byte 2 of Send mailbox
    L    IB3
    T    DL1        Load input bytes 3, 4and 5 in
    L    IB4        Send mailbox
    T    DRI
    L    IB5
    T    DL2
    AN   F101.7     Set CBS/bit 7, i.e. PC may access
    s    F1OI.7     Send mailbox
M3: NOP  O
    :BE
```

A plant for manufacturing small parts from bar stock consists of three substations (S5-101 U slaves)

Station 1 supplies station 2 with raw material from the bar magazine.
Station 2 machines the raw material and passes the finished part on to station 3.
Station 3 checks whether the manufacturing tolerances have been adhered to.

A master station supplies the slaves with data and also receives data from the slaves.

Monitoring station

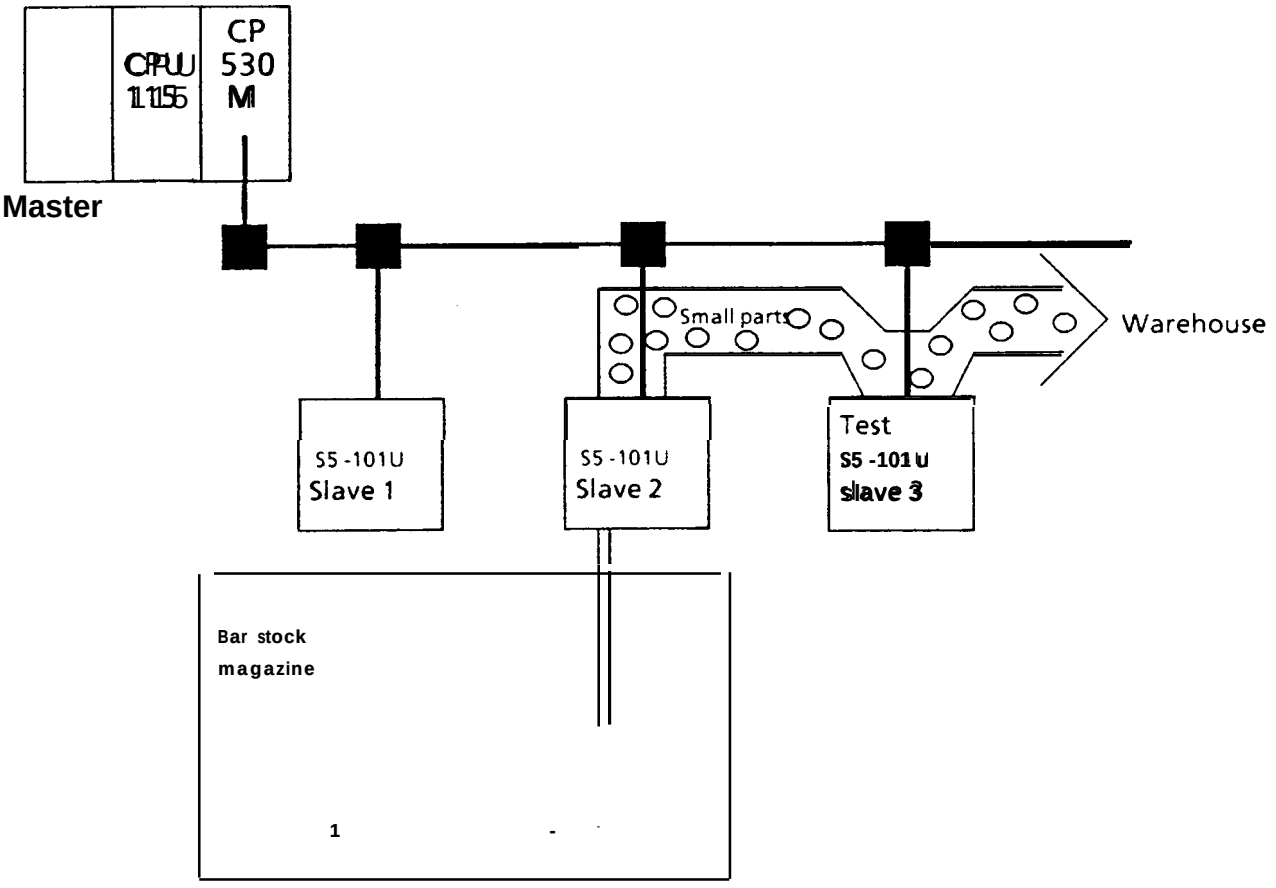


Fig. 2-6 Layout of a small parts manufacturing plant

Examples of STEP 5 programs:

Example 1: Data exchange between the master and slave 3

The master tells slave 3 which dimensions of the finished part are to be tested and which manufacturing tolerances can be accepted. The three-byte long block of data is stored by the slave in flag bytes FY 36 to FY 38.
For statistical purposes, slave 3 informs the master which manufacturing tolerances have occurred. The two-byte long data word is stored in FW 50. **Data is to be transferred to the master only once at the end of each test operation. For this purpose, flag F 32.0 is set by the control program at the end of the test operation.**

Name: EXAMPLE 1

```
0006      : S    F 63.0
0008      :L     KF+3
000A      : A    I0.0        Control program for slave 3
000C             :
000E
0010
```

2.7 Example of Small Parts Plant

```
0012        : =     Q 1.0
0014
0016        :
0018        :                       RECEIVE DATA
OOIA
OoIc        : A     F 61.7          Permission to receive= 1: data received
001 E       : JC = M 001            not valid; slave is receiving data
0020        :
0022                                Evaluate receive mailbox:
0024        : L     DR 40           Source
0026        : L     KF+0            Accept only data from O (master)
002A        : ><F
002C        : JC = FO04
002E
0030        : L     DL 41           Store Ist item of data received in FY 36
0032        : T     FY 36           (flag byte FY 36)
0034        : L     DR 41           Store 2nd item of data received in FY 37
0036        : T     FY 37           (flag byte FY 37)
0038        : L     DL 42           Store 3rd item of data received in FY 38
003A        : T     FY 38           (flag byte FY 38)
003C        :
003E M 004  : A N   F 61.7          Permission to receive= 1: data received
0040        : S     F 61.7          have been evaluated and slave may accept
0042                                new data
0044
0046        :
0048                                SEND DATA
004A        :
004C M 001  : A     F 62.7          Permission to send= 1: slave sends
004E        : I C = M 002           but send mailbox may not be changed
0050
0052                                Edit send mailbox:
0054        :                       Test operation not completed
0056        : L     KF+0            Length: O
005A        : T     DL 80
005C        : L     KF+0            Destination slave: O (master)
0060        : T     DR80
0062        : A N   F 32.0          F 32.0= 1: Test operation completed
0064        : JC = M 003
0066                                Test operation completed
0068        : L     KF+2            Length: 2
O06C        : T     DL 80
O06E
0070        : L     FW50            FW50: Manufacturing tolerances
0072        : T     DW81            (1st and2nd items of data sent)
0074        : A     F 32.0          Reset "Test operation completed"
0076        : R     F 32.0          identifier
0078        :
O 0 7 A  M 003 : AN F 62.7          Permission to send= 1: send mailbox
O07C        : S     F 62.7          ready to send
O07E M 002  : NOP O
0080
0082        : BE
```

Example 2: a) Data exchange between the master and slave 1
b) Data exchange between slave 1 and slave 2 (interslave traffic)

a) The master informs slave 1 of the magazine compartment in which it will find the raw material for the next processing batch. Slave 1 stores this information in flag byte FY 35.
Slave 1 tells the master how many bars of raw material have been removed from the particular magazine compartment. For this purpose, the current status of counter CO is transferred.

b) Slave 1 accepts the data byte sent by slave 2 and maps bit 1 to flag F 34.0 of its flag area (F 34.0 = 1: supply new bar of raw material).

Name: Example 1

```
0006        : S     F63.0
0008        :L      KF+1
000A        :A      I 0.0       Control program for slave 1
000C
000E        :         :
0010                  .
0012        : =     Q 0.0
0014
0016        :
0018                            RECEIVE DATA
001A        .
001C        : A     F61.7       Permission to receive= 1: data received
001E        : JC = M 001        invalid; slave receiving data
0020
0022        :L      DR 40       Source
0024        :L      KF+0        Source: master?
0028        :><F
002A        : JC = M 002
002C                            Evaluate receive mailbox from master
002E
0030        :L      DL 41       Store number of magazine compartment in FY 35 (flag byte FY 35)
0032        :T      FY 35       (lst item of data received)
0034        :
0036        : JU = M 003        Permission to receive= 1: receive mailbox has
0038        :                   been evaluated; slave may accept new data
003A M 002 : L      DR 40       Source
003C        : L     KF+2        Source: slave 2?
0040        :><F
0042        : JC = M 003
0044                            Evaluate receive mailbox of slave 2
0046
0048        :L      DL 41       Map bit Oto F 34.0
004A        :L      FY 34       (1st item of data received: OOOOOOOX)
004C        : Ow
004E        :T      FB34        (flag byte FY 34)
0050         :
0052 M 003 : AN     F61.7       Permission to receive= 1: receive mailbox
0054        : S     F61.7       has been evaluated; slave may accept new data
0056
0058        :
O05A
O05C                            SEND DATA
O05E        :
0060 M 001: A       F62.7       Permission to send= 1: slave sending data;
0062        : JC = M 004        send mailbox may not be changed
0064
0066
0068                            Edit send mailbox
O06A        :L      KF+2        Length: 2 (counter contents take up 2 bytes)
O06E        : T     DL 80
0070        : L     KF+0        Destination slave: O (master)
0074        : T     DR 80
0076
0078        :L    C     O       Number of bars removed
O07A        :T      DW 81       (1st and 2nd items of data sent)
O07C         :
O07E M 002 : AN     F62.7       Permission to send= 1: send mailbox
0080        : S     F62.7
0 0 8 2
0084 M 004 : NOP    O
0086
0088        : BE
```

2.7 Example of Small Parts Plant

Example 3: a) Data exchange between the master and slave 2
b) Data exchange between slave 2 and slave 1 (interslave traffic)

a) The master sends slave 2 a three-byte long identifier telling it in coded form how the raw material is to be machined. Slave 2 stores this information in flag bytes FY 38 to FY 40.
Slave 2 tells the master which machining step the control system is at that moment. This information is contained in flag byte FY 10.

b) The control program of slave 2 sets flag F 34.0 when slave 1 is to supply a new bar of raw material. Slave 2 informs slave 1 of the status of this flag. Slave 1 then transfers the status of this flag. Slave 1 then transfers the status of the flag to F 34.0 of its flag area.

NAME: EXAMPLE 1

```
0006        :S    F63.0
0008        :L    KF+2
000A        :A    I0.0        Control program for slave 2
000C              :
000E
0010              .
0012        :=    Q1.0
0014
0016
0018        :                 RECEIVE DATA
OOIA
Oolc        :A    F61.7       Permission to receive= 1: data received
001E        :JC=  M001        invalid; slave receiving data
0020
0022                          Evaluate receive mailbox:
0024        :L    DR40        Source
0026        :L    KF+0        Accept only data from source O (master)
002A        : ><F
002C        : JC=M004
002E                          Data for parts machining
0030        :L    DL41        Store 1st item of data received in flag byte FY38
0032        :T    FY38
0034        :L    DR41        Store 2nd item of data received in flag byte FY39
0036        :T    FY39
0038        :L    DL42        Store 3rd item of data received in flag byte FY40
003A        :T    FY40
003C        :
003E M004 : AN    F61.7       Permission to receive= 1: receive mailbox
0040        :S    F61.7       has been evaluated; slave may accept new data
0042
0044
0046        :
0048                          SEND DATA
004A        :
004C M001 : A     F62.7       Permission to send= 1: slave is sending data;
004E        :JC=  M002        send mailbox may not be changed
0050
0052        :A    F60.0       F60.0 = 1: supply send mailbox of slave 1
0054        :A    F34.0       with new raw material
0056        :JC=  M003
0058
O05A        :                 Edit send mailbox for master
O05C
O05E        :L    KF+1        Length: 1
0062        :T    DL80
0064        :L    KF+0        Destination slave: O (master)
0068        :T    DR80
O06A        :L    FY 10       FY 10 (flag byte 10): active machining step
O06C        :T    DL81        (1st item of data sent)
O06E
0070        :AN   F62.7       Permission to send= 1: send mailbox ready
0072        :S    F62.7       to send
```

```
0074          : S     F60.0     F60.0 = 1: send mailbox slave 1
0076          : JU =  M002
0078
007A                            Edit send mailbox for slave 1
007C
007E M003 :   L       KF+1      Length: 1
0082          :T      DL80
0084          :T      DR80      Destination slave: 1
0086
0088          :L      FY34      Transfer F 34.0 only
008A          :L      KF+1
008E          : AW              Mask out F34.1 to F34.7
0090          :T      DL81      (1st item of data sent)
0092
0094          :  AN   F62.7     Permission to send = 1: send mailbox
0096          : s     F62.7     ready to send
0098          : R     F60.0     F60.0 = O: send mailbox of master
009A          :
009C M002 :   NOP   O
009E          : BE
```

3. Appendix

3.1 Matrix of the Data Handling Block Numbers in the Various PCs

The "Data handling blocks" program package contains the following function blocks (FBs):

For S5-	115U[1]	135U with S processor[2]	R processor	150U
SEND	FB 244	FB 120	FB 120	FB 180
RECEIVE	FB 245	FB 121	FB 121	FB 181
FETCH	FB246	FB 122	FB 122	FB 182
CONTROL	FB 247	FB 123	FB 123	FB 183
RESET	FB 248	FB 124	FB 124	FB 184
SYNCH RON	FB 249	FB 125	FB 125	FB 185
ACTIVE	—	FB 126		—
UP ACTIV		FB 127	—	—
SEND-A		—	FB 126	
REC-A	—	—	FB 127	

1) In the case of the S5-115U, these function blocks are contained in the operating system of the CPU.

2) The data handling blocks of the S processor differ from the others with respect to queue processing (see ACTIVE and UP ACTIV function blocks).

FY 244 **"SEND"**

FY244 SPRM-A LNG = 35 SYM
Page 1

Segment 1

NAME	: SEND		
ID	: SSNR	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG : KY
ID	: A-NR	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG : KY
ID	: ANZW	I/Q/D/B/T/C : I	BI/BY/W/D: W
ID	: QTYP	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG: KS
ID	: DBNR	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG: KY
ID	: QANF	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG: KF
ID	: QLAE	I/Q/D/B/T/C : D	KM/KH/KY/KS/KF/KT/KC/KG: KF
ID	: PAFE	I/Q/D/B/T/C : Q	BI/BY/W/D: BY

O03A : BE

Description:

The SEND block is used for initiating a "Send data" request to the CP 530.

Description of parameters:

SSNR: Number of logic interface

A-NR: Interface request to be initiated (send message)

ANZW: Address of the condition code double word (address ANZW and ANZW + 1) in which processing of the initiated request is flagged to the user.

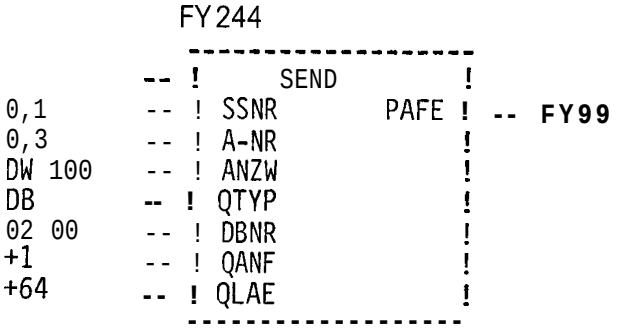
QTYP: Type of data source from which the data are to be transferred to the CP.

DBNR: Number of data block for QTYP XX, SS, DB and DX

QANF: "Relative starting address" of the data source

QLAE: Number of source data

PAFE: Error condition codes in response to parameter assignment errors



3.2 Standard FBs for CPU ⟷ CP 530 Traffic

FY 245 "RECEIVE"

```
FY245          -RECEIVE SPRM-A                    LNG = 35 SYM
                                                        Page 1
Segment 1
NAME      : RECEIVE
ID        : SSNR     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KY
ID        : A-NR     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KY
ID        : ANZW     I/Q/D/B/T/C : I  BI/BY/W/D : W
ID        : ZTYP     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KS
ID        : DBNR     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KY
ID        : ZANF     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KF
ID        : ZLAE     I/Q/D/B/T/C : D  KM/KH/KY/KS/KF/KT/KC/KG : KF
ID        : PAFE     I/Q/D/BIT/C : Q  BI/BY/W/D : BY

O03A      :BE
```

Description:

The RECEIVE block is used for examining whether there is a request on the CP 530.

Description of parameters:

SSNR: Number of the logic interface

A-NR: Number of the receive message to be fetched on the CP.

ANZW: Address of condition code double word in which processing of the request is flagged to the user.

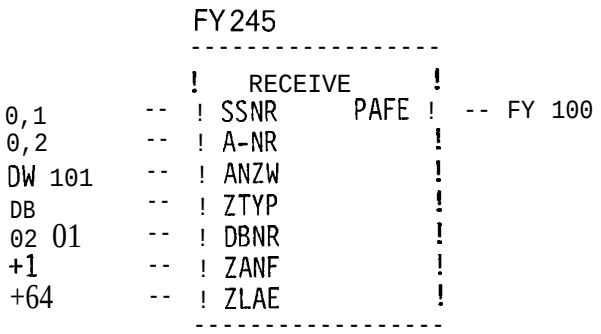
ZTYP: Type of data destination in which the data of the CP are to be entered

DBNR: Number of the data block for ZTYP XX, DB and DX

ZANF: Relative starting address of the data block

ZLAE: Number of data items to be accepted (length of net data + 4)

PAFE: Condition codes in response to parameter assignment errors



FY 247 "CONTROL"

```
FY247          -CONTROL SPRM-A                LGN = 23 SYM
                                                Page 1
Segment 1
NAME : CONTROL
ID     : SSNR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : A-NR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : ANZW    I/Q/D/B/T/C : I BI/BY/W/D : W
ID     : PAFE    I/Q/D/B/T/C : QBI/BY/W/D : BY

0022       : BE
```

Description:

The CONTROL block performs the following functions:

– Updating of the condition code word if the associated request was initiated in an interrupt branch (SEND)

– Scanning of the CP to determine whether a definite request of an interface module is active

– Scanning of the CP to determine which request is being processed

No acknowledgements take place between the CONTROL block and the interface module; the control block only transfers the condition codes from the "Request status" to the initialized condition code word. The block does not depend on the RLO (result of the logic operation) and must be linked into the cyclic part of the STEP 5 program section by the user.

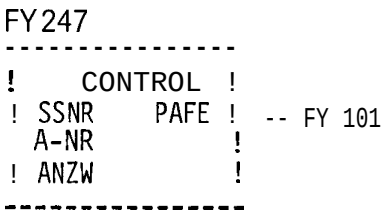
Description of parameters:

SSNR: **Interface number**

A-NR: **CP 530 request to be monitored to tell whether a message is present or not**

ANZW: **Condition code word in which the result of the scan is flagged to the user**

PAFE: **Condition code byte for parameter assignment errors**



FY 248 "RESET"

```
FY248          -RESET SPRM-A                  LGN = 20 SYM
                                                Page 1
Segment 1
NAME : RESET
ID     : SSNR    I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : A-NR    I/Q/D/BIT/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID     : PAFE    I/Q/D/B/T/C : Q BI/BY/W/D : BY

001 C      : BE
```

Description:

The "RESET" function block knows the following operating mode in conjunction with the CP 530:
– RESET all; resets all requests of this logical interface to which parameters have been assigned

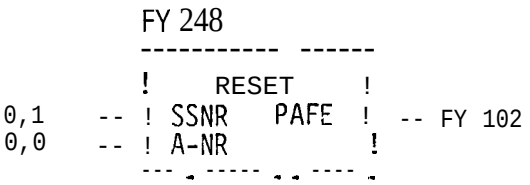
3.2 Standard FBs for S5 I15U ⟷ CP 530 Traffic

Description of parameters:

SSNR: Logical interface number

A-NR: Request number

PAFE: Error condition codes in response to parameter assignment errors



FY 249"SYNCHRON"

FY249 -SYNCH RON SPRM-A LGN = 20 SYM
Page 1

Segment 1
NAME : SYNCHRON
ID : SSNR I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG : KY
ID : BLGR I/Q/D/B/T/C : D KM/KH/KY/KS/KF/KT/KC/KG: KY
ID : PAFE I/Q/D/B/T/C : Q BI/BY/W/D : BY

Oolc : BE

Description:

The block synchronizes the PC with the CP 530 on restart of the PC. At the same time, the transfer area of the interface is erased and initialized and the packet size between the CP 530 and the PC defined.

Description of parameters:

SSNR: Logical interface number

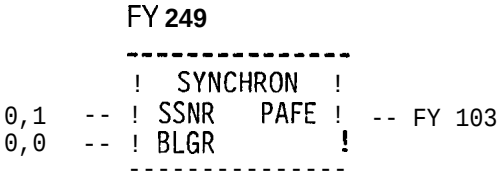
BLGR: Packet size

PAFE: Parameter assignment error

Block sizes:

O ≙ Standard size
1 ≙ 16 bytes
2 ≙ 32 bytes
3 ≙ 64 bytes
4 ≙ 128 bytes
5 ≙ 256 bytes
6 ≙ 512 bytes

Note:
The packet size in the case of the CP 530 from version 2.0 onward is preset at 5 ≙ 256 bytes and cannot be changed.



a) Bit O Handshake Meaningful

Set/ By the data handling blocks in keeping with the condition code in the request status byte.
Reset: Handshake bit (= 1) is only meaningful with the RECEIVE block (as information on whether a message is present or not),

Evaluation: By user for RECEIVE enquiry (enquiry as to whether message is present or not).

b) Bit 1 Request Executing

Set: By the data handling blocks if request issued to CP

Reset: By the data handling blocks if a request is processed by the CP (e. g. acknowledgement received).

Evaluation: By the data handling blocks
A new request is only issued if the "old" request has been processed
By the user
in order to find out whether it is meaningful to trigger a new request or not.

c) Bit 2 Request Completed Without Errors

Set: By the data handling blocks if the corresponding request has been completed without errors.

Reset: By the data handling blocks if the request has been retriggered.

Evaluation: By the user in order to check whether the request has been completed without error,

d) Bit 3 Request Completed With Error

Set: By the handling blocks if the corresponding request has been completed with error, The cause of the error is then in the high byte of the condition code word in coded form.

Reset: By the data handling blocks if the request is retriggered.

Evaluation: By the user in order to check whether the request has been completed with error or not. If the "Request completed with error" identifier bit is set, the cause of the error can be read from the high byte of the condition code word.

e) Bit 4 Data Acceptance/Data Transfer Executing

Set: By the SEND and RECEIVE data handling blocks if transfer/acceptance has been started for a request, e. g. if data is being exchanged in response to the broadcasting function (DMA substitute), but the request has been initiated with SEND direct.

Reset: By the SEND and RECEIVE data handling block if the data exchange for a request is terminated (last partial block of data transmitted).

Evaluation: By the user
During data transmission from the CP to the PC, the user is not allowed to make any further changes to the packet of a request. In the case of packets shorter than the blocking limit, this is uncritical since data exchange can be implemented in one block pass. However, larger quantities of data can only be transmitted in packet form with blocking distributed over a number of PC cycles. In order to ensure data integrity, the user must therefore first check whether the data packet has just been transmitted before he changes the data of a request.

f) Bit 5 Data Transfer Completed

Set: By the SEND data handling block when data transfer for a request has taken place.

Reset: By the SEND data handling block when the transfer of data for a new request (retriggering) has been started.
By the user if evaluation has already taken place (pulse edge generation).

Evaluation: By the user
This bit can tell the user whether the packet for a request has already been transmitted to the CP or when a new packet can be readied for a current request.

g) Bit 6 Data Acceptance Completed

Set: By the RECEIVE data handling block if the acceptance of data for a request has been completed.

Reset: By the RECEIVE data handling block if transfer of the data to the PC has started for a new request (retriggering).
By the user if evaluation has been completed (pulse edge generation).

Evaluation: By the user
This bit tells the user whether the record of a request has already been transferred to the PC or when a new record for the current request has been transferred to the PC.

h) Bit 7 Disable/Enable Data Block

By the user in order to prevent writing into an area by the RECEIVE block or reading out of an area by the SEND block (in case of the lst data packet).

Reset: By the user in order to release the associated data area.

Evaluation: By the SEND and RECEIVE data handling blocks.
If bit 7 is set, the blocks do not execute any data transfers but report this "error" to the CP. How this disabled data block is treated depends on the type of CP.

i) Bits 8 to 11 Error Byte

If the interface module issues an error identifier for a request, the data handling blocks enter this identifier in the high byte of the condition code word.
Error messages:
6 : Parameter assignment error
7 : Slave not in the polling list
8 : CP overload
9 : Operating mode error
A : Submodule failed or incorrect type
B : STOP/RUN switch set to STOP
C: Dialogue with handling block interrupted
D : Error in data traffic
E : List not found
F : Data has been overwritten

3.4 Length Word

The condition code word is followed immediately, i. e. in the next memory address location, by the length word, in which the number of request data items exchanged between the PC and CP are stored.

Writing: By SEND and RECEIVE during data interchange.
The length word is calculated from
the packet management + "reserved" pointer

Reset: By overwriting or with each new SEND, RECEIVE or FETCH request.

Evaluation: By the user
- If the "Request completed without error" or "Data transfer/acceptance completed" bit is set, the length word contains the current source and/or destination length.
- If the "Request completed with error" bit is set, the length word contains the number of data items transferred up to the point of error.

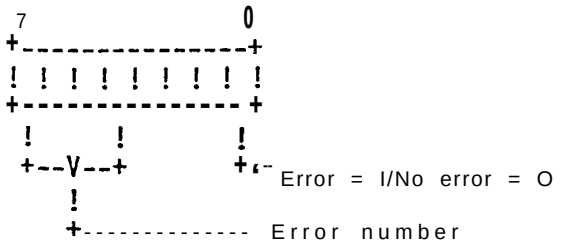
3.5 PAFE: Condition Code for Parameter Assignment Errror

> PAFE: Condition code for parameter assignment error

The "BYTE" specified here (output, input, flag) is set if the block detects a parameter assignment error, e. g. interface (module) does not exist or illegal parameter asignment of QTYP/ZTYP; QANF/ZANF; QLAE/ZLAE.

Type of parameter: Output, byte
Permiss. range : IB0.IB 127
QBO QB 127
FYO. FY255

Format PAFE byte



0 No error
1 Wrong ORG (operating system) format
2 Area non-existent (no DB)
3 Area too small (DB etc.)
4 Time-out
5 Wrong condition code word
6 No source/destination parameter for SEND RECEIVE ALL
7 Interface does not exist
8 Interface not ready
9 Interface overloaded
B Interface not acknowledged or enabled
C Interface (CP) does not acknowledge or gives negative acknowledgement
D Parameter assignment error high byte (e. g. blocking)

Order No.: GWA4NEB Bll 0546-02d
Printed in the Federal Republic of Germany
AG 6870.5 Fl 66 En

SIEMENS

SIMATIC S5 / SINEC LI

CP 530 Communications Processor CP 530
COM 530 on the PG 675 Programmer

Operating Instructions — Order No.: GWA 4NEB 811 0520-02b

Contents

1 Introduction

The SINECL1 Local Area Network permits communication between up to 31 SIMATIC S5 programmable controllers of the U range in master/slave mode.
The COM 530 software package for the PG 675 programmer is used to assian parameters to the SINEC L1 CP530 communications processor, 'to select the LAN operating mode, to control data traffic and to document and archive the bus parameters on mini diskettes.

All operator inputs are made via interact ve screen forms (entry fields) and function keys (or softkeys).
The significance of the function keys can be seen in the menu displayed in the bottom threelines of the screen.

In addition, the familiar functions of the keys on the PG 675 programmer, such as acknowledgement / r@ /, Abort, / ❑ /, etc. have been retained.

2 Definitions

The following are described:

- o Layout of the interactive screen forms
- o Meaning of the entry and output fields in the interactive forms
- o Meaning and effect of the softkeys and function keys.

Output fields in the interactive forms are marked xxxxx. In these fields COM 530 shows current statuses or data entered with the previous operations.

Entry fields in the interactive forms are marked ##### and appear on the screen of the PG675 in inverse video. Entries can be made in these fields by means of the alphanumeric keyboard and, in some cases, with the function keys (HELP function).

Error messages of COM 530 always appear in the last line on the screen before the menu.

Bus parameters: All parameters necessary to operate the LAN and generated by the user with the aid of the COM 530 software (polling list, interrupt list, SYSID identifiers, see SINEC L1 Operating and Programming Instructions).

The bus parameters are generated when programming (see Section 3.3) and stored by means of function keys.
Where they are stored depends on the COM 530 mode set, i.e. ONLINE or OFFLINE.

In ONLINE mode, the user data are stored directly in the CP 530 communications processor, and, in OFFLINE mode, on, for example, diskette in drive FD1 or in an EPROM or EEPROM submodule.

When changing over to programming bus parameters these are read from the communications processor or minidiskette - that is, if user data are already stored there and displayed on the respective forms where they can be modified and re-stored.

Program name: A user-selectable name for identifying all user data assigned to a CP 530 interface and stored on diskette.

The significance of the cursor control keys and function keys

a) Cursor control keys

In most of the interactive forms used, the significance of the control key functions is as follows:

↑ ↓ : The cursor is positioned to the first entry field in the line above (below).

==> <== : The cursor is positioned to the previous (next) entry field.

-> <-- : The cursor is positioned to the previous (next) character within an entry field.
If the left (or right) field limit is exceeded, the cursor moves to the previous (next) field.

b) Function keys

The COM 530 software package generally permits parallel input both by means of softkeys and function keys. The operator therefore does not have to relearn programmer operations.

In most of the interactive forms used the significance of the keys is as follows:

↵ : Jump to next entry field

⌦ Deletion of a character at the cursor position

⎀ : Insertion of a character at the cursor position

⎋ : This key always has the same. meaning as function key F8 EXIT (parallel key).
This results in the next-highest level in the form hierarchy being entered without the input data in the current screen form being evaluated.

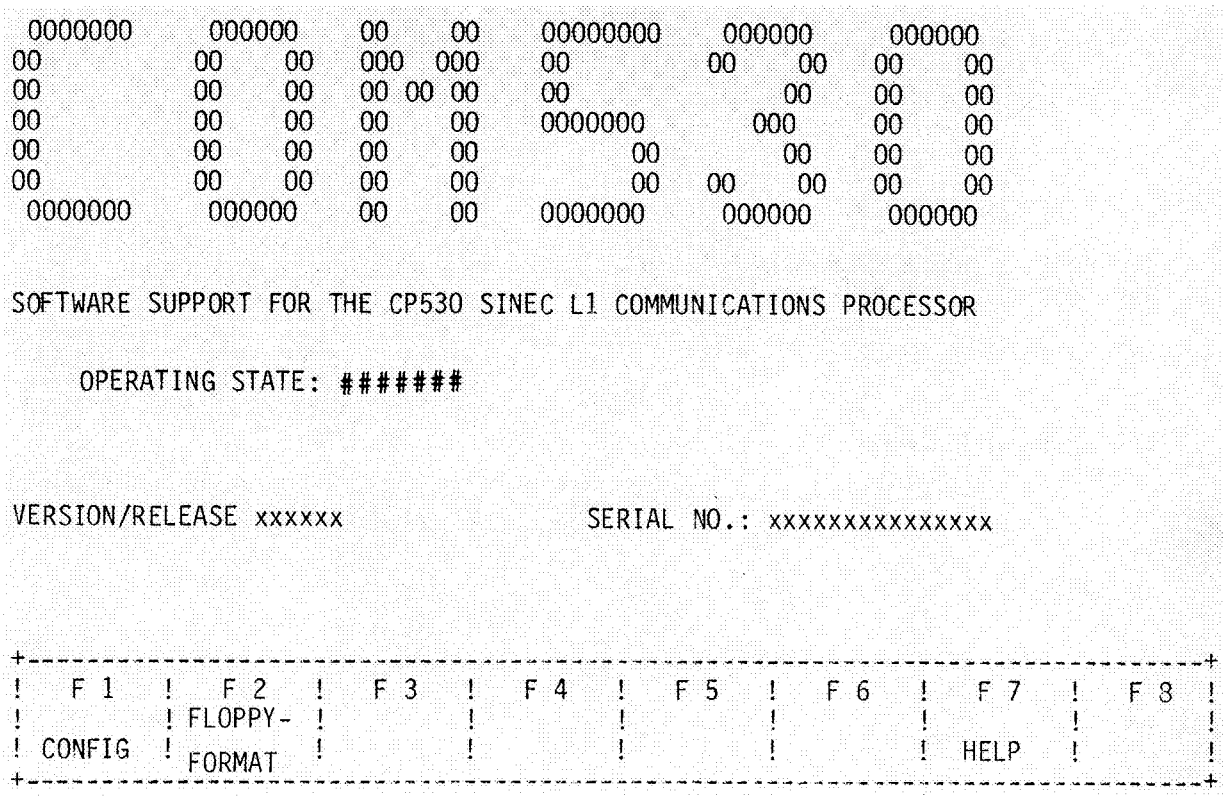
F6 : This key means "Store" if softkey F6 STORE has been defined in the form. These are also parallel keys and have the same function

HELP : The HELP key has the same function as function key F7 - HELP if it has been defined in the form.

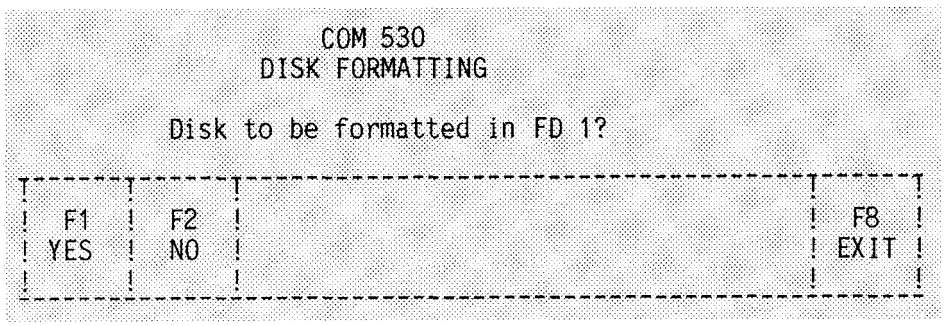
3 Operator input and operation of the COM 530

3.1 COM 530 basic form

After the COM 530 has been cal led (diskette in drive FDØ, and after the PG 675 programmer has been switched on), the following form appears:



3.1.1 **Disk formatting form**



Assignment of the function keys:

F1 YES — The disk in FD1 is formatted
N.B. any existing files on it are lost.

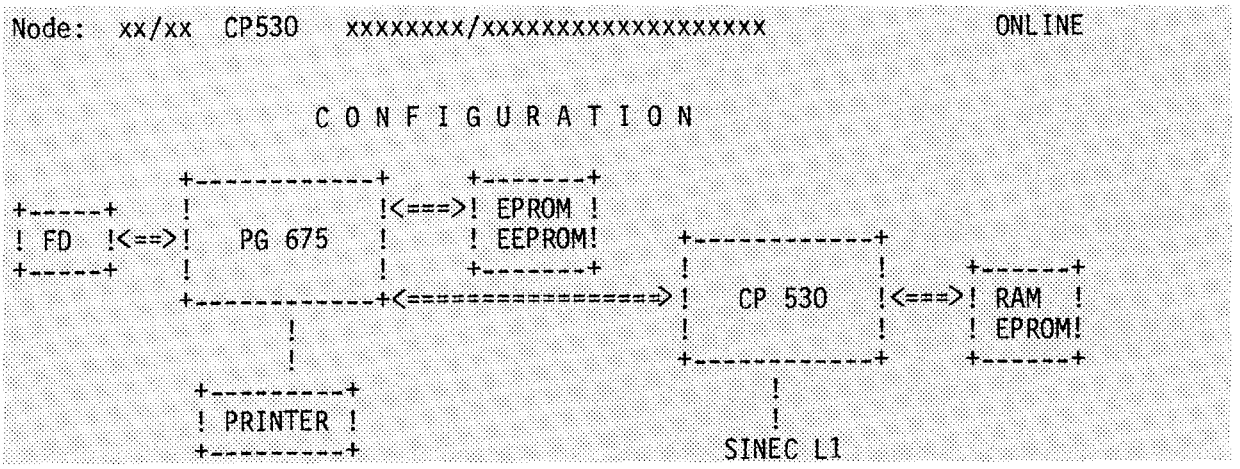
F2 NO — Return to main menue

F8 EXIT — Return to main menue

3.2 CONFIGURATION form

Either the CONFIGURATION (ONLINE) or CONFIGURATION (OFFLINE) form appears, depending on which mode is active.

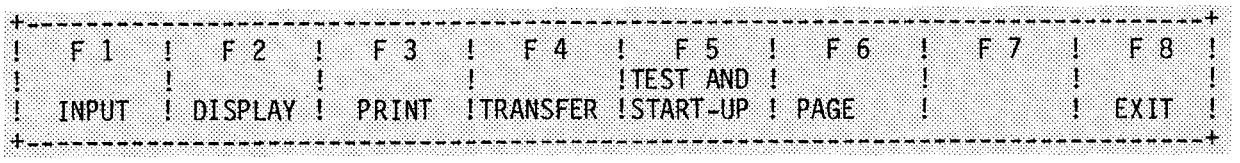
(1) ONLINE programming:



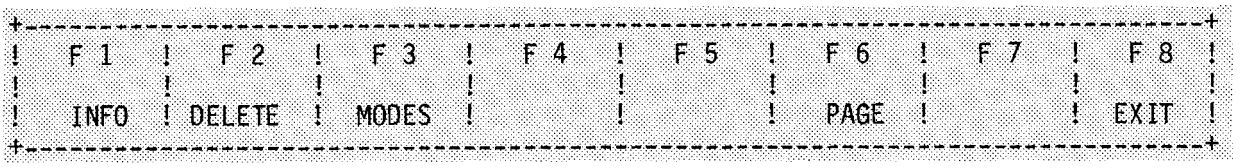
CONFIGURATION (ONLINE) form

The menu for the CONFIGURATION (ONLINE) form is in two parts:

Menu 1a):



Menu lb):



The meaning of the output fields in the CONFIGURATION (ONLINE) form:
The 'Nodes" field in the header displays a two-part node number, which is read out of the SYSID area of the CP 530.
For the master the second part is /00; for a slave it is a number greater than o (1-30).

The next two fields show the version and symbolic designation of the CP 530 (only if the PG675 is connected directly or indirectly to the CP 530, i.e. in ONLINE mode).

Assignment of the function keys:

Menu 1a):

F 1 : Entry of user data

The following are possible:
- Programming the system parameters (SYSID),
- Generation of a polling list,
- Generation of an interrupt(priority)list.

Selects the ENTRY form.

F 2 : Display of user data

If available, SYSID identifiers, the polling list and the interrupt list can be displayed.
Selects the DISPLAY form.

F 3 : Listing user data

The user data generated can be output for documentation purposes on a printer connected to the PG 675.
Selects the PRINT form.

F 4 : Transfer of SYSID identifiers, polling and interrupt lists

This function permits the transfer of bus parameters from the CP 530 onto mini-diskette. EPROMS and EEPROMS can also be programmed with the bus parameters.
Selects the TRANSFER form.

F 5 : Selects the tests of data traffic on the SINEC L1 (TEST form)

Note: A user diskette must be in drive FD 1, even in ONLINE mode.

F 6 : Selects menu 1b

F 8: Initiates return to the COM 530 basic form

Menu 1b:

F 1 : Selects the "Information mode

Information is given on whether SYSID identifiers and/or polling list and/or interrupt list are in the CP 530 or EPROM/EEPROM submodule.

In addition, information is given on the programs stored on the user diskette (program names) and their contents (INFO form)

F 2 : Selects the "Delete" mode

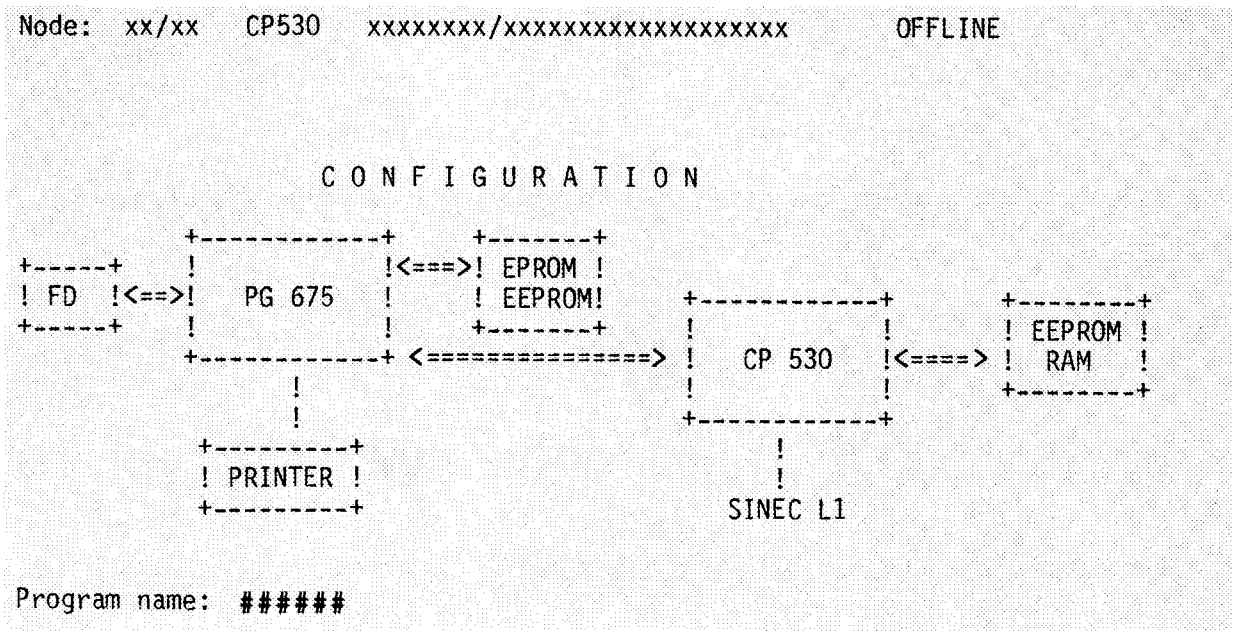
Deletes SYSID identifiers, polling or interrupt lists (DELETE form)

F 3 : Selects mode setting (MODE form)

F 6 : Selects menu 1a

F 8 : Initiates return to the COM 530 basic form

(2) OFFLINE programming:



CONFIGURATION (OFFLINE) form

The menu for the CONFIGURATION (OFFLINE) form is also in two parts:

Menu 2a):

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
INPUT	DISPLAY	PRINT	TRANSFER		PAGE	HELP	EXIT

Menu 2b):

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
INFO	DELETE				PAGE	HELP	EXIT

For the meaning of the entry and output fields:

PROGRAM NAME:

In OFFLINE mode, the user must enter a name here which he can use to identify new or already existing bus parameters (lists and SYSID). This specifies the file name under which the user data is read from, or written onto, the diskette.

The HELP function can be used to enter all program names on the diskette.

The meaning of the output fields: see CONFIGURATION (ONLINE) form.
The "Version no." and "Symbolic designation" fields of the CP 530 are blank in OFFLINE mode.

Assignment of the function keys:

Menu 2a):

F 1, F 2, F 3: see Menu 1a.

F 4 : Transfer of SYSID identifiers, polling and interrupt lists

This function permits bus parameters to be transferred from mini-diskette to EPROMs or EEPROMS.
Selects the TRANSFER form.

F 6 : Selects menu 2b

F 7 : HELP function

With the aid of the HELP function, the names of all programs (i.e. bus parameters) on the user diskette can be entered in the "PROGRAM NAME" field.

F 8 : Initiates return to the COM 530 basic form

Menu 2b):

F1: Selects the "Information" mode

Information is given on whether SYSID identifiers and/or polling list and/or interrupt list are available in the EPROM. In addition, information is given on the programs stored on the user diskette (program names) and their contents. (INFO form)

F 2 : Selects the "Delete" mode

Deletion of SYSID identifiers, polling or interrupt list (DELETE form)

F 6 : Selects menu 2a

F 7 : HELP function

The HELP function can be used to enter the names of programs (i.e. bus parameters) on the user diskette in the "PROGRAM NAME" field.

F 8: Initiates return to the COM 530 basic form

Possible error messages:

Error 2a: "WARNING N. EPROM driver on the system diskette. No EPROM calls!!!"

The attempt to program data direct into an EPROM/EEPROM or to read data out of an' EPROM/EEPROM causes the program to crash ===> The "INTERRUPT TRAP HALT" message appears.

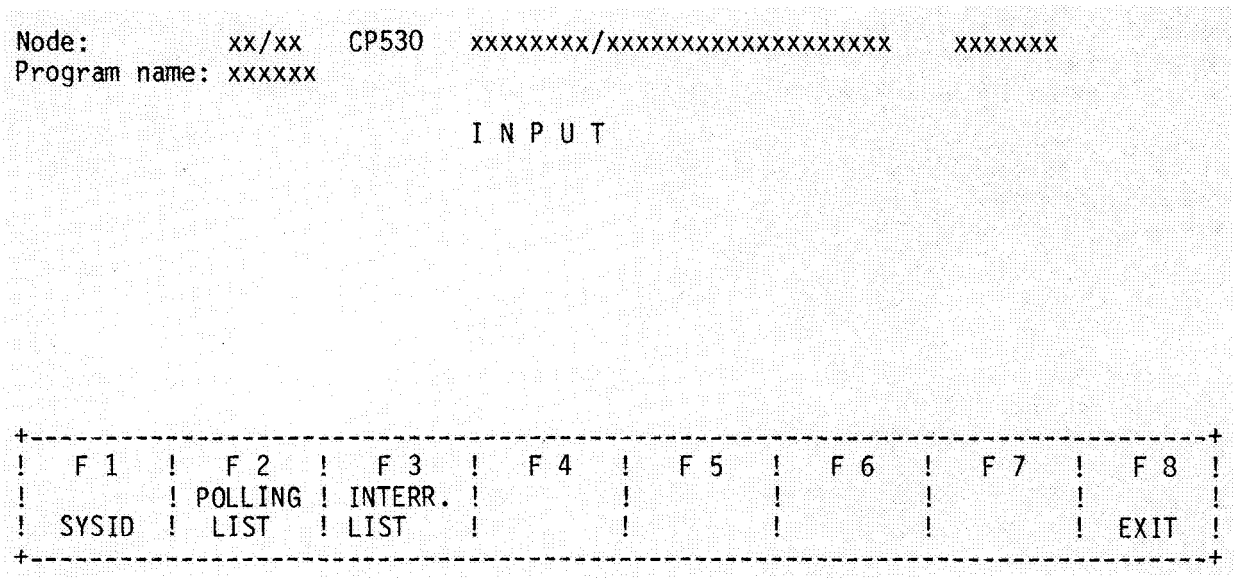
Remedy: Load the original COM 530 system diskette into drive FDØ and restart program.

Error 46: "Incomplete entry"

A freely selectable program name must be specified in OFFLINE mode.

3.3 Entering (programming) user data

INPUT form



The following applies to the header in this and all subsequent forms:

The node numbers and identifiers are taken from the SYSID area of the CP 530 communictions processor in ONLINE mode. If OFFLINE mode has been selected, this data is taken from the SYSID identifiers stored on the user diskette. If the SYSID identifiers stored on the user diskette under the program names specified in OFFLINE programming are not (yet) available, the fields in the header remain vacant.

The "Program name" field is only displayed when programming OFFLINE. It then contains the name specified in the CONFIGURATION (ONLINE) form.

Assignment of the function keys: Softkeys F2 and F3 only appear if the CP 530 is master in the SINEC L1 network.

F l : Selects programming of the system parameters (SYSID-ON form)

F 2 : Creation of the polling list (POLL-INP form)
(Only possible when programming a master)

F 3 : Creation of the interrupt (priority) list (INTERRUPT-INP form)
(Only possible when programing a master)

F 8 : Initiates return to the CONFIGURATION form.

3.3.1 Programming the system parameters (SYSID-INP form)

The SYSID data area is a memory area in the CP 530 used for identification purposes.

Purpose of the SYSID:

- To uniquely describe the role of an intelligent module (e.g. CP 530) in an programmable controller system.
  To transfer parameters to an intelligent module, causing certain reactions
  To give information on the firmware and software status of the module.

```
Nodes:  xx/xx  CP530     xxxxxxxx/xxxxxxxxxxxxxxxxxx     xxxxxxx
Program name:  xxxxxx

                          S Y S I D     C P 5 3 0


        Submodule identifier:                 xxxx ###-#####
        Module identifier:                    xxxxxxxx
        Firmware version identifier:          xxxxxxxx
        Plant designation:                    ####################
        User software generation date:        ########
        Slave No. on PG/SINEC L1:             ## x ##
        Ident No.:                            ###
        Automat. restart:                     #    (Y=Yes, N=No)
        Master/Slave identifier:                   (M=Master, S=Slave)
        Transmission speed:                   ######
+-------------------------------------------------------------------------+
!  F 1  !  F 2  !  F 3  !  F 4  !  F 5  !  F 6  !  F 7  !  F 8  !
!       !       !       !       !       ! STORE !       !       !
!       !       !       !       !       ! ON xx ! HELP  ! EXIT  !
+-------------------------------------------------------------------------+
```

Description of the SYSID INP form:

(1) ONLINE parameter assignment:

When the CP 530 is started, the operating system transfers the "submodule identifier", "Version identifier", and "Firmware version identifier" parameters to the SYSID area reserved in the RAM. Then it transfers the other parameters to the SYSID area from the memory submodule connected (EPROM/EEPROM).

In ONLINE programming, the SYSID identifiers of the CP 530 appear in this form and can be modified by overwriting them.

(2) OFFLINE programming:

The form also appears completed on the screen in OFFLINE programming if a SYSID file exists on the floppy disk under the program name specified (see CONFIGURATION form).
Otherwise the entry fields are vacant or defaulted with possible alternatives.

The contents of the SYSID (except for the first three parameters) can be generated OFFLINE on the programmer and transferred to an EPROM/EEPROM. This means that the EPROMs/EEPROMs can be identified at any time and can program the CP 530 on start-up.

Output field F6 is defaulted with FD (floppy disk) (OFFLINE programming) or with CP(CP530) in ONLINE mode.

Meaning of the entry fields in the SYSID-INP form:

PLANT DESIGNATION:

Name freely selectable with up to 19 alphanumeric characters
e.g.: "SHED1 PLA4 PC7°

USER SOFTWARE GENERATION DATE:

The generation date of the user software can be specified in a freely selectable format (e.g.: "21.01.85")

SLAVE-NO. AT PG/SINEC L1:

Specification of the node number of the CP 530 on the PG or SINECL1 bus.
Examples: "-- /10" ..... only node number for SINECL1
"-8/-4" ..... two-tier node number
"-7/--" ..... only node number for PG bus

("-" stands for a blank; the "/" is provided by the programmer)

Range: Slave no. on PG bus: 1 . . . 30
Slave no. on SINECL1: 1 . . . 30

AUTOMAT. COLD RESTART:

The alternatives YES (Y) and NO (N) can be entered using the HELP function.

Default: "Y"

MASTER/SLAVE IDENTIFIER:

The CP 530 can be programed as master or slave. This is shown either by "M" (Master) or
"S" (Slave).
HELP functions are possible.

Default: "M"

TRANSMISSION SPEED:

This permits selection of transmission speed on the bus.
Possible entries: 9,600 baud 4,800 baud 2,400 baud
1,200 baud 600 baud 300 baud

Default: "9,600 baud"

Assignment of the function keys:

F 6 In ONLINE mode: Storing the SYSID identifiers in the CP 530
In OFFLINE mode: Storing the SYSID identifiers on diskette under the program name specified.
After storage - return to the INPUT form.

F7: HELP functions by paging possible alternatives (only in the "Master/Slave identifier", "Addressing mode" and "Automatic cold restart" fields.) Depressing the HELP function key causes a description to be displayed in the other input fields. This can be exited with F 8 (Exit), returning to this form.

F8: Return to the INPUT form without storing the SYSID identifiers.

3.3.2 Generation of a polling list (POLL-INP)

For minimal operation of the SINEC LI, only the polling list is required. This list contains 64 locations for entering slave numbers; the sequence thus laid down defines the order in which the slaves are to be addressed in LAN operation. The simplest case is the natural sequence; if all 64 places are used and certain slaves are specified several times, a certain priority can be achieved.

Simple case:

```
+---+---+---+---+---- ----+---+
! 1 ! 2 ! 3 ! 4 ! . . . . . ! n !
+---+---+---+---+---- ----+---+
```

Assigning priority to slave 3:

```
+---+---+---+---+---+---+---+---+---+---- ----+---+---+
! 1 ! 2 ! 3 ! 4 ! 5 ! 6 ! 7 ! 3 !... . . .! 3 ! n !
+---+---+---+---+---+---+---+---+---+---- ----+---+---+
```

POLL-INP form:

When entering a polling list for the first time, the following form appears:

```
Node: xx/xx    CP 530  xxxxxxxx/xxxxxxxxxxxxxxxxxxx     xxxxxxx
Program name: xxxxxx

                          P O L L I N G   L I S T

  ==> ##  ==>

+-------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8   !
!         !         !         !         !         ! STORE   !         !         !
!  DELETE ! INSERT  !         !         !         ! ON xx   !         !  EXIT   !
+-------------------------------------------------------------------------+
```

The entry field between the arrows symbolizes the first location in the polling list. The number of the first node in the list can be entered here. The first double arrow signifies the beginning of the list, the final double arrow the end of the list.

The F 1, F 2 and F 6 function keys have no significance at this time as the list does not contain any nodes yet.

After a node number (e.g. 10) has been input in the entry field, the form changes as follows:

```
Node:       xx/xx   CP530     xxxxxxxx/xxxxxxxxxxxxxxxxxx       xxxxxxx
Program name: xxxxxx

                              P O L L I N G   L I S T


  => 10 -->  ##  ==>
```

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
DELETE	INSERT				STORE ON xx		EXIT

It can be seen that, after a node has been entered in the list, a new entry field is offered for entering a further node and thus building up the list.
The slave numbers (1. to 30) are keyed into the entry fields in the sequence in which they are to be addressed. The same slave can appear several times, thus receiving higher priority than other slaves.

The polling list can have up to 64 entries.

After a certain number of slave numbers have been entered (e.g. 20) the form looks like this:

```
Node:       xx/xx    CP 530   xxxxxxxx/xxxxxxxxxxxxxxxxxx     xxxxxxx
Program name: xxxxxx

                              P O L L I N G   L I S T


    ==>  10 -->  11 -->  12 -->  1 -->  2 -->  3 -->  10 -->  4 -->
    -->   5 -->  10 -->   6 -->  7 -->  8 --> 10 -->   9 -->  2 -->
    -->  13 -->  14 -->  10 -->  2 --> ## ==>
```

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
DELETE	INSERT				STORE ON xx		EXIT

The slave numbers can be changed once the respective field has been selected using the cursor control keys.

The function keys now have the following assignments:

F1: Operation of this key deletes the slave entry in the field on which the cursor is presently positioned (current entry field). All slaves following this entry field are then automatically moved one place backwards. If the last field is deleted, the cursor automatically jumps to the first field.

F2: Operation of this key releases the current field, i.e. all slave numbers following this are moved one position to the right.
The key has no effect if the cursor is in the last (free) entry field or if 64 entries have already been made.

F 6: Storing the polling list in the CP 530 (ONLINE mode) or on mini-diskette (OFFLINE mode) and return to the INPUT form.

F 8: Return to the INPUT form without storing the polling list.

Possible error messages:

Error 01: "Illegal input"!

At least one slave number must be entered in the list
Possible slave numbers: 1 to 30.

Error 26: "Blank fields illegal"!

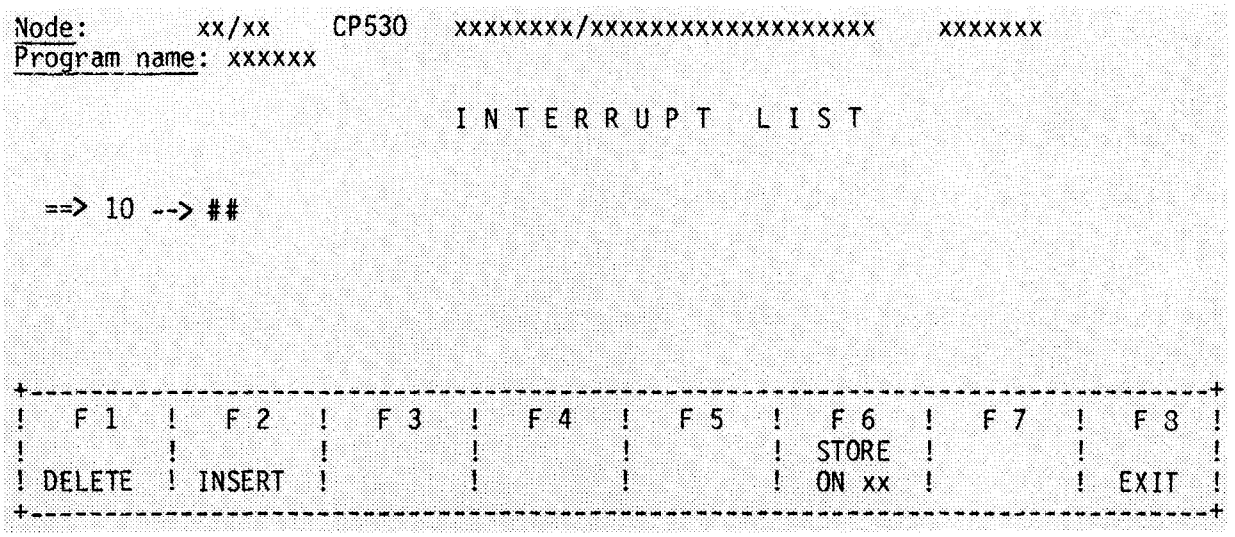
The polling list must not have blank fields. The cursor is in the first blank field that has been found.
Remedy: - Delete the field or
enter slave number in the respective field.

3.3.3 Generation of an interrupt list (INTERRUPT-INP form)

The interrupt list is only required if slaves can interrupt bus traffic with an interrupt request. The bus master then searches for the cause of interrupt in the sequence specified in the interrupt list.

Generation and display of the interrupt list is the same as for the polling list.

The following form therefore appears when an interrupt list is generated for the first time:



Further operations and functions of the softkeys can be seen from the description of polling list generation (Section 3.3.2).

The numbers of the slaves (1 to 30) must be keyed into the entry fields in their order of priority in the event of a bus interrupt. The slave entered in the first field has the highest priority, the second the second-highest priority, etc.

NOTE: Each slave number may only be specified once!

Possible error messages:

Error 27: "No double slave numbers in interrupt list!"

Each slave may only appear once in an interrupt list. The cursor is in the field in which the slave number appears for the second time.
Remedy: - Delete the respective field or
- Enter another slave number in the respective field.

See the POLLING-INP form for other error messages.

3.4 Output (display of the user data)

DISPLAY form

```
Node:  xx/xx   CP530   xxxxxxxx/xxxxxxxxxxxxxxxxxx     xxxxxxx

                          D I S P L A Y

               Source:     ##########

Specify only for diskette:  ######
   Program name:

+-------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8  !
!         ! POLLING !INTERRUPT!         !         !         !         !        !
!  SYSID  ! LIST    !LIST     !         !         !         !  HELP   !  EXIT  !
+-------------------------------------------------------------------------+
```

Significance of the entry fields in the DISPLAY form:

SOURCE :

The source can be specified here without changing the mode. An EPROM (EEPROM) can also be specified as source!

Possibilities:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP 530, EPROM!

These alternatives can be entered with the HELP function.

Default: ONLINE mode .... CP 530
OFFLINE mode ... FD

PROGRAM NAME:

If a floppy disk (FD) is specified as source, a program name must. be entered under which the user data has been stored on the diskette.
The HELP function can be used to review all program names stored on the user diskette.

Assignment of the function keys: (Softkeys F 2 and F 3 only appear if the CP530 is master!)

F1: Causes display of the system parameters (SYSID-DISP form)

F 2: Displays polling list (POLL-DISP form)
(Only for display of master data)

F 3: Display of interrupt (priority) list (INTERRUPT-DISP form)
(Only for display of master data)

F 7: HELP function for entering source and program name
(if source = floppy disk)

F 8: Return to CONFIGURATION FORM

Possible error messages:

Error 01: "Illegal input!"

Source must be stated. The following possibilities exist:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP530, EPROM

Error 31: "No polling list!"
Error 32: "No interrupt list!"
Error 33: "No SYSID identifier!"

One of these three error messages appears if the respective data type does not exist in the CP 530, the EPROM or in the programs specified on the user diskette.

Error 46: "Incomplete input"

A program name must be specified if the diskette (FD) is the source.

3.4.1 SYSID-DISP form

Node: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxx xxxxxxx
Program name: xxxxxx

S Y S I D C P 5 3 0

Submodule identifier:	xxxxxxxxxxx
Module identifier:	xxxxxxxx
Firmware version identifier:	xxxxxxxx
Plant designation:	xxxxxxxxxxxxxxxxxx
User software generation date:	xxxxxxxx
Slave No. on PG/SINEC L1:	xx x xx
Ident-No.:	xx
Automatic cold restart:	x (Y = Yes, N = No)
Master/Slave identifier:	x (M = Master, S = Slave)
Transmission speed:	xxxxxx

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
INPUT							EXIT

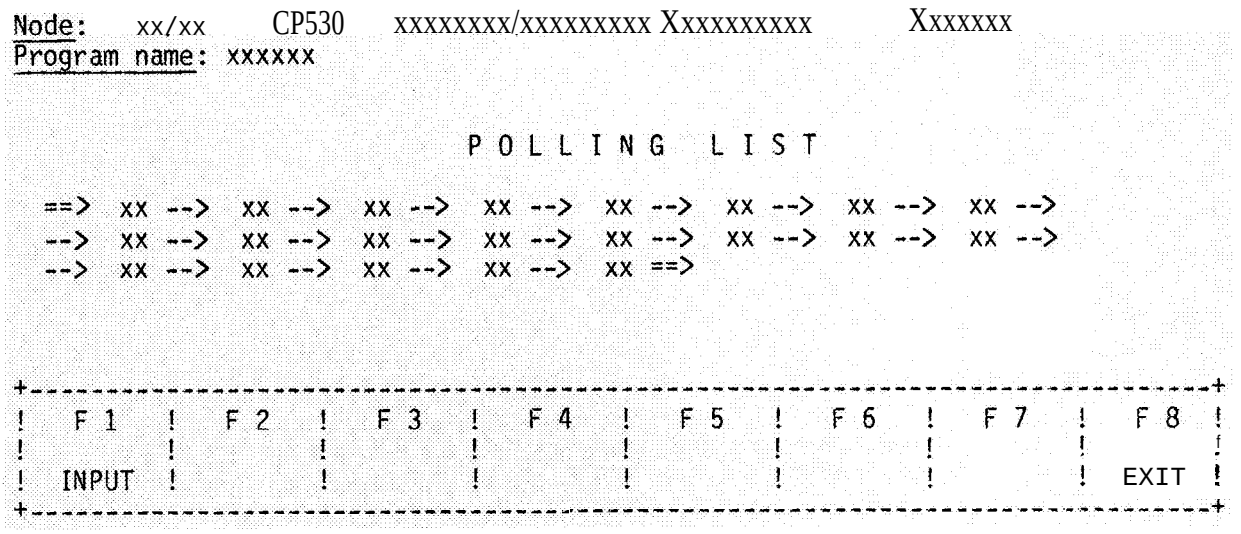
The significance of the data in the output fields can be seen from the SYSID-DISP form.
In output field 5 (ONLINE or OFFLINE) EPROM is specified as source if an EPROM is used.

Assignment of the function keys: (Softkey F 1 remains unassigned if display is from an EPROM!)

F1: This key initiates display of the form for SYSID identifier input (see Section 3.3.1)

F8: Causes return to the DISPLAY form

3.4.2 POLL-DISP form



In output field 5 (ONLINE or OFFLINE) "EPROM" appears as source if an EPROM is used. The polling list stored is displayed as generated (see Section 3.3.2) with the following differences:

- There are no entry fields
  There is no vacant field at the end of the list

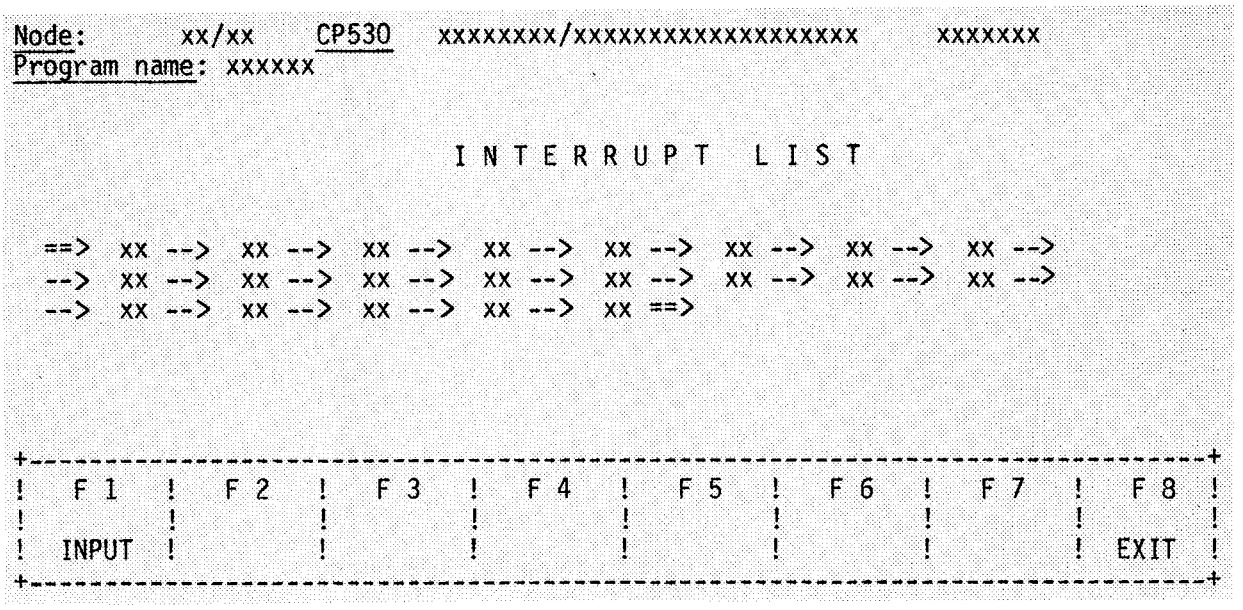
The slave numbers appear in the output fields as generated.

Assignment of the function keys: (Softkey F1 does not appear if display is from an EPROM!)

F1: This key permits entries to be made in the polling list (see Section 3.3.2) for modification, etc.

F8: Causes return to the DISPLAY form

3.4.3 INTERRUPT-DISP form



In output field 5 (ONLINE or OFFLINE) "EPROM" appears as source if an EPROM is used.

The interrupt list stored is displayed as for input mode (see Section 3.3.3), with the following differences:

- There are no entry fields:
- There is no vacant field at the end of the list.

The slave numbers appear in the output field as entered.

Assignment of the function keys: (Softkey F 1 does not appear if display is from an EPROM!

F1: This key permits entries to be made in the interrupt list (see Section 3.3.3) for modifications etc.

F 8: Causes return to the DISPLAY form

3.5 PRINT form

```
Node:   xx/xx   CP530      xxxxxxxx/xxxxxxxxxxxxxxxxxxx        xxxxxxx

                                P R I N T

                             Source: ##########


Specifiy only for diskette:
                Program name:  ######


+-----------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8   !
! PRINTER ! TOTAL   !         !         !         !         !         !         !
!PARAMTERS! PRINTOUT!         !         !         !         ! HELP    ! EXIT    !
+-----------------------------------------------------------------------+
```

Significance of the entry fields in the PRINT form:

SOURCE :

Here it is possible to specify the source without changing the mode. An EPROM can also be given as source!

Possibilities:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP530, EPROM

These alternatives can be entered using the HELP function.

Default: ONLINE mode CP530
OFFLINE mode . . . . . FD

PROGRAM NAME:

If a floppy disk (FD) is specified as source, a program name must be entered here under which the bus parameters have been stored on diskette. The HELP function can be used to review all program names from the user diskette.

Assignment of the function keys:

F 1: Causes transfer to the PRINTPAR form for assignment of parameters to the PG printer output.

F2: Printing all user data from the CP 530 (in ONLINE mode) or the bus parameters stored on the diskette under the program name specified (in OFFLINE mode). The printed pages have header and footnote lines (user texts in accordance with F 1) and are numbered. When finished, the program returns to the CONFIGURATION form.

F7: HELP function for entering the source.

F 8: Return to the CONFIGURATION form.

Possible error messages:

Error 01: "Illegal input!"

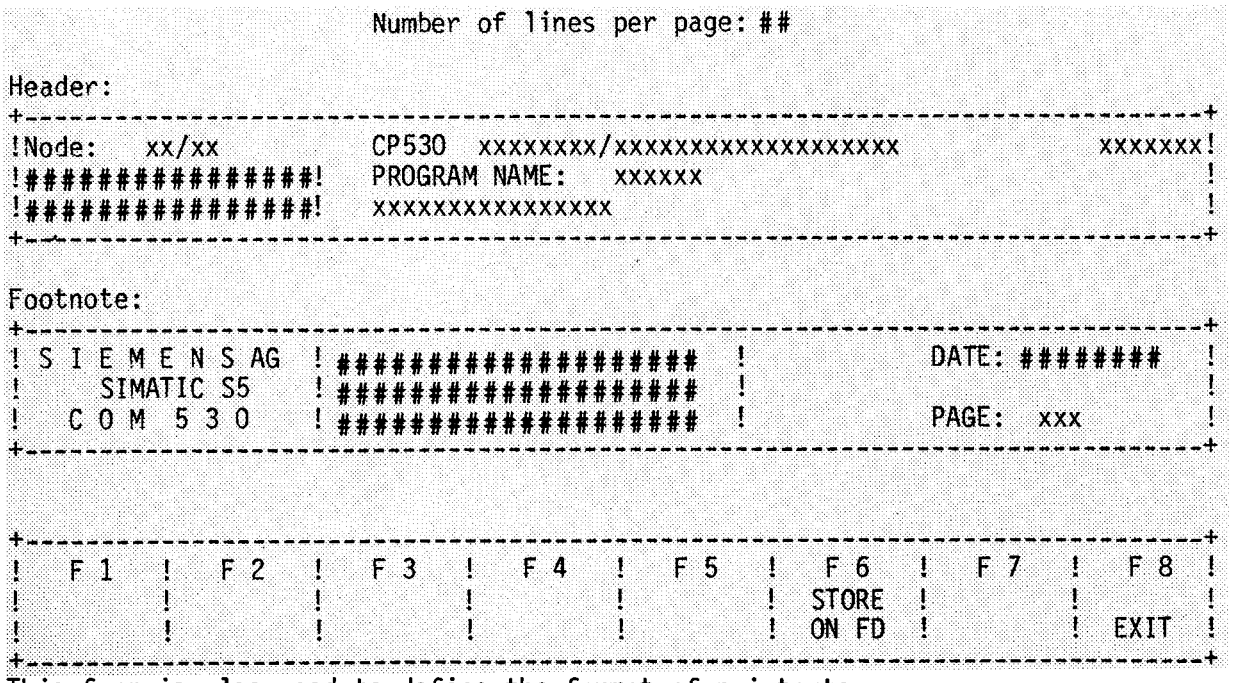
Source must be specified. The following are possible:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP 530, EPROM

Error 46: "Incomplete input"

If the diskette (FD) is stated as source, a program name must be specified.

3.5.1 PRINTPAR form



This form is also used to define the format of printouts.

Significance of the entry fields in the PRINTPAR form:

NUMBER OF LINES PER PAGE: Specification of number of lines per page for the PG printer connected.
Range: 40 ... 65

DATE: The actual date can be entered here in any format.

The other entry fields can be filled with user text as required.

In this case, the output fields only indicate that dynamic texts appear in the printout here.
('SYSID', 'Polling list', page number etc.)

Assignment of the function keys:

F 6: Operation of this function key causes the parameters generated for printout to be stored on the user diskette. They are not assigned to any program name.
The next time parameters are assigned to the printer, this form already has these parameters as defaults.

F 8: Return to the PRINT form

Possible error messages:

Error 01: "Illegal input!"

Permissible number of lines 40 . . . 65

3.6 TRANSFER form

```
Node:        xx/xx    CP530    xxxxxxxx/xxxxxxxxxxxxxxxxxx        xxxxxxx

                              T R A N S F E R

                  Transfer from ##########   to ##########

                  SYSID:
                  Polling list:
                  Interrupt list:

Specify only for diskette:
                  Program name:   ######


+-----------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8  !
!         !         !         !         !         !         !         !        !
!TRANSFER !         !         !         !         !         !  HELP   !  EXIT  !
+-----------------------------------------------------------------------------+
```

Significance of entry fields in the TRANSFER form:

TRANSFER FROM/TO:

In these two fields, the source and destination of the transfer are to be specified. (This is also possible using the HELP key).

Possibilities:

Mode	Source	Destination
OFFLINE	FD, EPROM	FD, EPROM
ONLINE	FD, CP530, EPROM	FD, CP530, EPROM

Source and destination must not be identical.

Default: Source: 'DISKETTE' ('FD')
Target: 'EPROM'

SYSID, POLLING LIST, INTERRUPT LIST:

In these three fields, the entries "Y" for yes or "N" for no can be selected.

If the CP 530 is configured as a slave, only "N" is possible for both lists (as a CP 530 slave has neither a polling nor an interrupt list).
If the EPROM is specified as destination, all user data are automatically transferred.
Default: "Y"

PROGRAM NAME:

If the floppy disk is specified as either dest " nation or source, a program name must be entered under which the user data is stored on the diskette or are to be stored on it.
The HELP key can be used to review al' program names on the user diskette.

Assignment of the function keys:

F l : Initiation of transfer.
In the message line the messages "Act ve!, "Completed" or error messages appear.

F7: HELP function (paging through the alternatives) in the first five fields.
In addition, the names of all programs on the user diskette can be reviewed in the "PROGRAM NAME" field.

F 8: Return to the CONFIGURATION form.

Possible error messages:

Error 01: "Illegal input!"

Source and destination must be specified. The following possibilities are available.

Mode	Source	Target
OFFLINE	FD, EPROM	FD, EPROM
ONLINE	FD, CP 530, EPROM	FD, CP 530, EPROM

Error 21: Source = Destination?

In "Transfer" mode, source and destination must not be identical.
Possibilities:

Destination / Source	DISKETTE	CP530	EPROM
FLOPPY	-	+	+
CP530	+	-	+
EPROM	+	+	-

\+ permissible illegal

Error 36: "No polling list!" or
"No interrupt list!" or
"No SYSID identifier!"
One of these three error messages appears if the respective data type does not exist in the CP 530 or EPROM or on the user diskette under the program name specified.

Error 46: "Incomplete input"

If the source or destination = DISKETTE; a program name must be specified.

3.7 Test and startup

3.7.1 TEST form

(Refer also to instructions GWA 4NEB 811 . . . for the L1 network, Section 3.3)

```
Node:   xx/xx        CP530   xxxxxxxx/xxxxxxxxxxxxxxxxxx      ONLINE

                          T E S T   A N D   S T A R T U P

                 Is data diskette in drive FD1?

+-------------------------------------------------------------------------+
!  F 1  !  F 2  !  F 3  !  F 4  !  F 5  !  F 6  !  F 7  !  F 8  !
!       !       !       !       !DISKETTE !       !       !       !
!       !       !       !       !IN DRIVE 1!      !       ! EXIT  !
+-------------------------------------------------------------------------+
```

Assignment of the function keys:

F5: The message line is deleted and the following menu appears on the screen:

```
+-------------------------------------------------------------------------+
!  F 1  !  F 2  !  F 3  !  F 4  !  F 5  !  F 6  !  F 7  !  F 8  !
! STAT/  !       !BUS CYCLE!     !       !       !       !       !
! FORCE  !BUS TEST !TIME    !     !       !       !       ! EXIT  !
+-------------------------------------------------------------------------+
```

Assignment of the function keys: Softkeys F 2 and F 3 appear if the CP530 is Master!

F1: Selects display of contents of mailboxes (STAT/FORCE form)

F2: Selects bus test. The bus is stopped and further processing can then take place step by step in accordance with the polling list. (BUS TEST form)

F3: Selects display of the bus cycle time. (CYCLE TIME form)

F 8: Return to the CONFIGURATION form

Possible error messages:

Error 35: 'Illegal return message from PC!"

Error 36: "PC-Usart error!"

Error 37: No message from CP!"

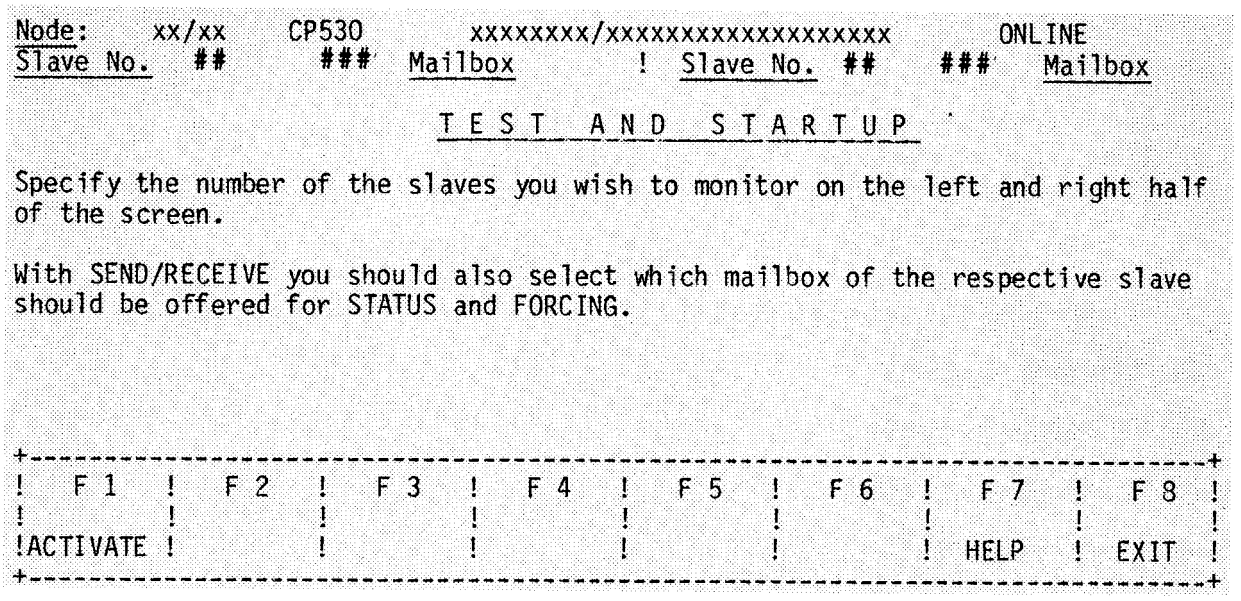
Error 38: "USART error PG side!"

Error 39: "USART error PC side!"

Error 3A: "Interface not ready!"

Error 3B: "Abort by CP!"

3.7.2 STAT/FORCE form



Significance of the entry fields in the STAT/FORCE form:

SLAVE No:

Specifies the numbers of the slaves to be tested.

Range: 1 ... 30

MAILBOX:

Specifies which of the mailboxes (send or receive mailbox) is to be updated in the left or right half of the screen.
SEND can be specified for the send mailbox or REC for the receive mailbox of the slave.

Default: . . . . SEND in both fields

Assignment of the function keys:

F 1: The input data are transferred and the CP test function activated. The mailboxes specified are read (selects STATUS form).

F7: SEND or REC can be entered in entry fields 2 and 4.

F8: Return to the TEST form.

Possible error messages:

Error 01: "Illegal input!"

If a number outside the upper or "ower limits is d' played in fields 1 and 3 or if there are blanks in these fields.
If something other than SEND, REC or blank appears in fields 2 and 4.

Error 35: "Illegal return messages from CP!"

Error 36: "CP Usart error!"

Error 37: "CP does not report!"

Error 38: "USART error PG side!"

Error 39: "USART error PC side!"

Error 3A: "Interface not ready!"

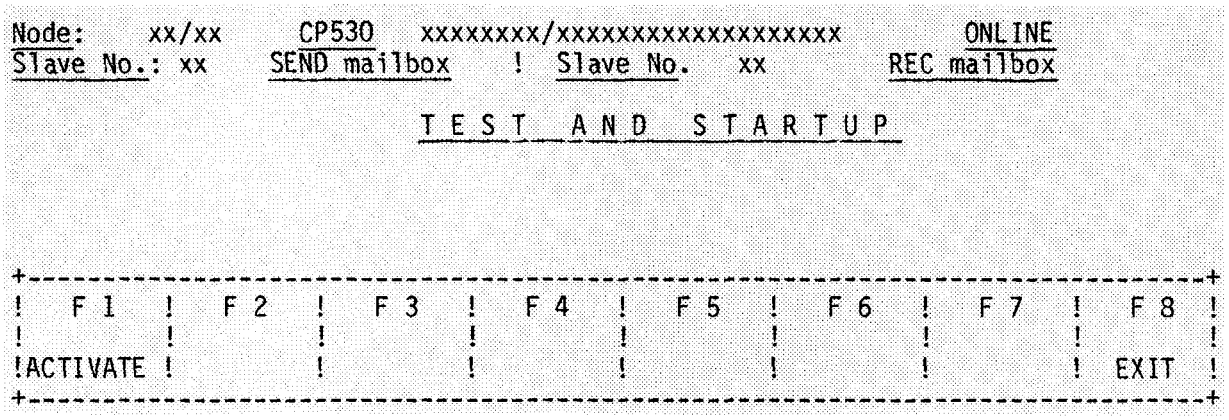
Error 3B: 'Abort by CP!"

Error 58: "Slave xx failed!"

Error 59: "Slave xx not on bus!"

In these last two error messages the number of the respective slaves is entered in the output fields.

If the CP 530 is configured as a slave, only its send and receive mailbox can be monitored and the STAT/FORCE form appears as follows:



Significance of the output fields:

SLAVE No.:

The extension of the node number from the header is written into these output fields.

Range: 1 ... 30

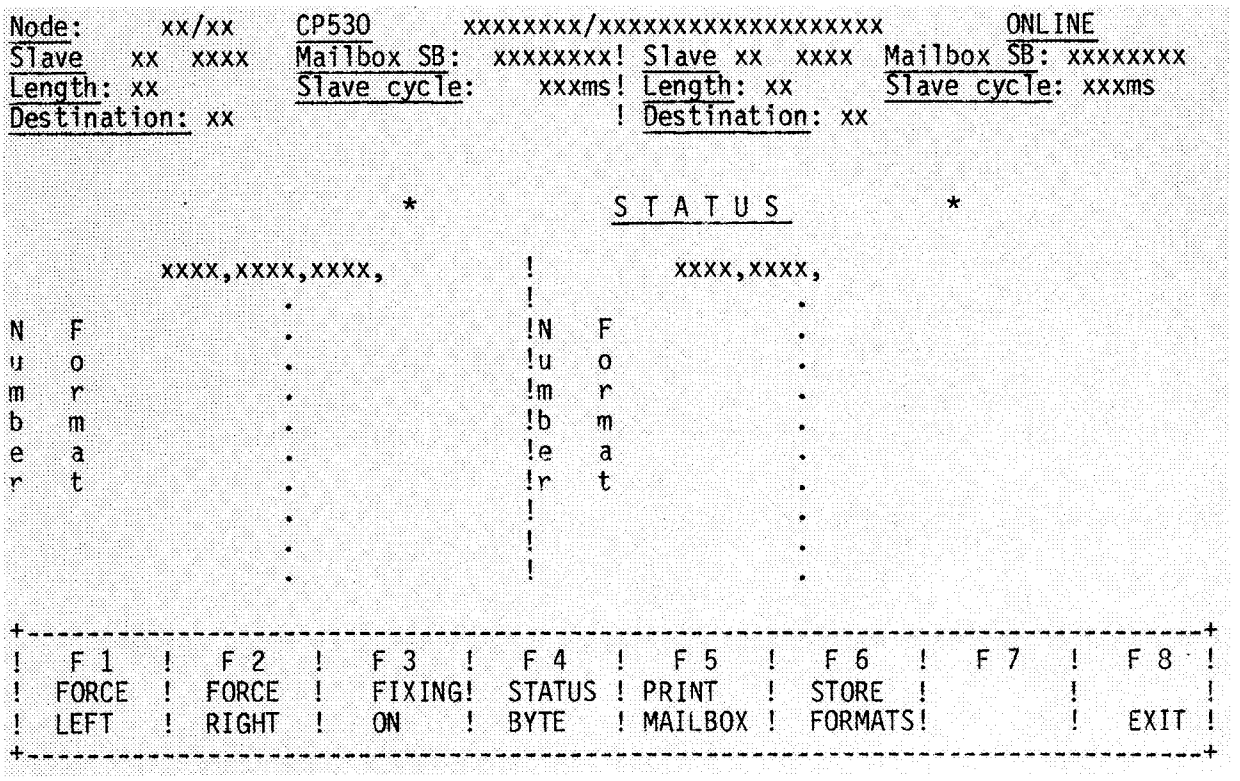
Assignment of the function keys:

same assignment as for the master

Possible error messages:

as for the master

3.7.3 STATUS form



The following possibilities are available for menu labell ing:

Mode 1 eft	Mode right	Possible softkeys
SEND	SEND	F 1 - F 6, F 8
SEND	REC	F 1, F3 - F 6, F 8
REC	SEND	F 2 - F 6, F 8
REC	REC	F 3 - F 6, F 8

If the CP 530 is configured as a slave, the "FORCE LEFT" and "FORCE RIGHT" functions cannot be implemented.

Significance of the output fields:

SLAVE: In the case of a receive mailbox, the number of the slave selected is entered. In the case of a send mailbox, 00 is entered as identifier for the master.

MAILBOX: The data selected are taken from the previous form.

SB: The status byte of the respective slave is displayed in binary code. It can be examined in more detail with function key 4 (magnifier function).

LENGTH: The length of the mailbox in words

SLAVE CYCLE: The reaction time of a slave until its mailbox is received (in milliseconds).

DESTINATION: In the case of a receive mailbox, 00 is entered for the master in this field; in a send mailbox this number is identical with that of the slave selected.

*: The appearance or disappearance of an asterisk in this field whenever a mailbox arrives indicates data traffic between the PG 675 and the CP 530.

The blinking frequency of the asterisk thus permits a rough estimate of the bus cycle time.

The part of the form between header and menu shows the contents of the mailboxes in the usual S 5 form divided into left and right mailboxes. The contents are constantly updated.

The contents of the longer mailboxes can be paged up and down with the "Page" function keys.

Function keys F 1 to F 3 can be used to modify display formats or enter new contents in the send mailboxes.

Assignment of the function keys:

F 1: Forcing the left mailbox. Both mailboxes are fixed, i .e. the display is frozen (not updated). The contents of the left mailbox can be modified.
Selects the FORCE form.

F 2: Forcing the right mailbox. Both mailboxes are fixed, i.e. the display is frozen (not updated). The contents of the right mailbox can be modified.
Selects the FORCE form.

F 3: Both mailboxes are fixed. The cursor is in the first format field of the left mailbox. Now the formats for the display can be modified (for both mailboxes). The menu labelling for F 3 is changed to "FIXING OFF". In addition, softkey F 4 enables the cursor to be positioned in the desired mailbox (see menu, next page).

Actuating F 3 once more causes a return to updating of the mailbox contents in the new formats. The form appears as shown in the diagram above.

F 4: This function enables the slave status byte, which appears as a binary pattern at the top right of each mailbox display, to be more closely examined. However, the status byte cannot be modified.
Causes the STATUS BYTE form to appear. (Section 3.7.5).

F 5: Listing of the contents of both mailboxes on the printer connected to the programmer. The display is frozen during printing.

F 6: The formats for the respective mailbox are stored on the user diskette. The display is frozen during the storage procedure. For each slave only one list of formats can be stored for the send mailbox and for the receive mailbox.

Note: There must be a user diskette in drive FD 1 although online mode is active.

F 8: Return to the STAT/FORCE form

Menu after actuation of the "FIXING ON" key:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
FORCE LEFT	FORCE RIGHT	FIXING OFF	FORMAT RIGHT	PRINT MAILBOX	STORE FORMATS		EXIT

F4 is 1 abell ed "right" or "left", depending on whether the cursor is in the left or right mai 1 box.

Possible error messages:

Error 13: "Unknown data identifier!"

The formats can be modified in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted"
Possible in conversions from all formats to the formats I(T, KC, I(S or KG.

Error 17: "Beginning - no further!"

If the mailbox is paged upwards but the first line has already been displayed.

Error 18: "End - no further!"

If the mailbox is paged downwards and the last line has already been displayed.

Error 2C: "Left mailbox empty!"

If the interface module brings a left mailbox with length zero, this message is displayed.

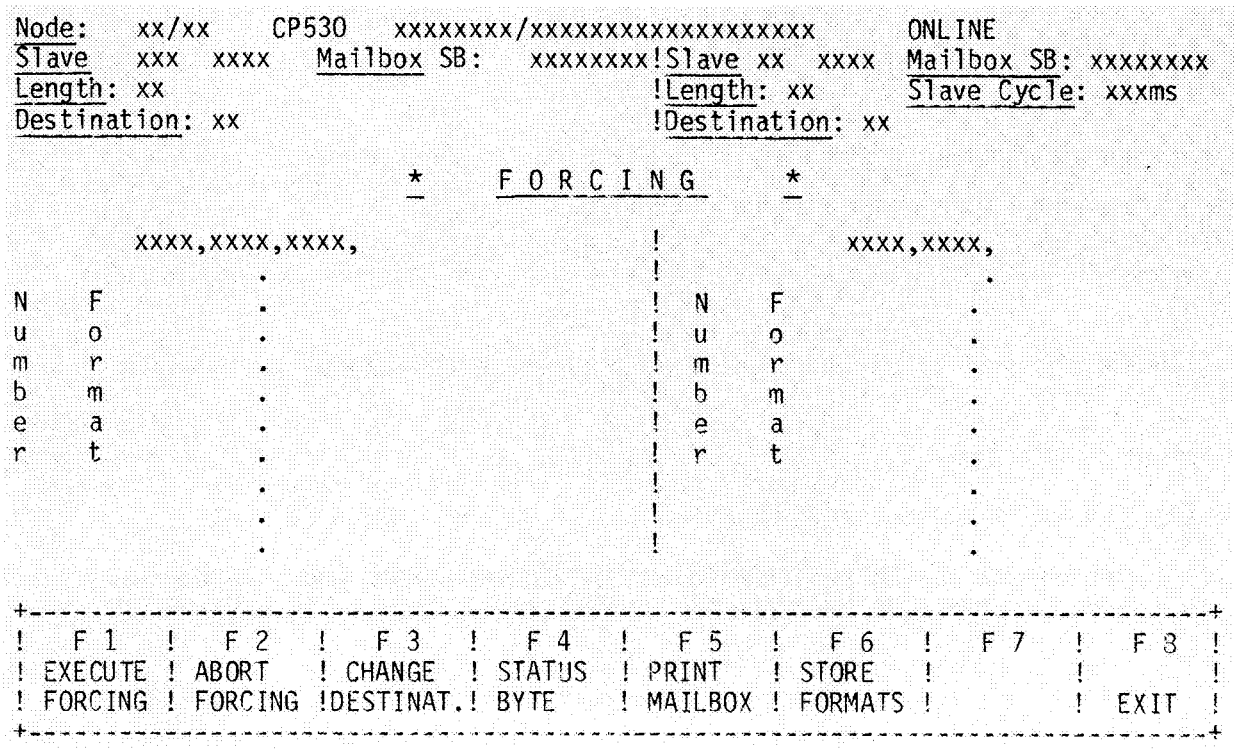
Error 2D: "Right mailbox empty!"

If the interface module brings a right mailbox with the length zero, this error message is displayed.

Message 01: "Active"

If the F6 function key, the Enter key or the F5 function key or hardcopy key are actuated, this message appears.

3.7.4 FORCE form



Significance of the output fields:

See STATUS form, Section 3.7.3

Assignment of the function keys:

F 1: Forcing is executed, i.e. the modified send mail 150 x is sent to the destination node and the bus cycle continues. A return is made to the STATUS form and any newly entered formats taken over.

F 2: Forcing is not executed. The bus cycle continues. Return to the STATUS form. New formats are not taken over.

F 3: The cursor jumps to the "Destination: xx" field which now becomes an entry field and permits a new destination slave to be entered.
Range: 1 . . . 30

F 4: This function permits the status byte of the slave which appears as a binary pattern at the top right of each mailbox display and can be more closely examined and modified.
Causes the STATUS BYTE form to appear.

F 5: Printout of the contents of both mailboxes on the PG printer.

F 6: The formats for the respective mail box are stored on the user diskette.

F 8: Return to the STAT/FORCE form without forcing being executed.

Possible error messages:

Error 0B: "Inhibited key!"

Error 12: "INSERT/ERASE" not possible here!"

The cursor is in the last line and an attempt has been made to erase or insert in the mailbox.

Error 13: "Unknown data identifier!"

The formats can be changed in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted!"

Possible on conversion from all formats to the KT, KC, KS or KG formats.

Error 17: "Beginning - No further!"

If the first line in the mailbox is displayed and an attempt is made to page upwards.

Error 19: "Do not press function key!"

If the cursor is in the first field, i.e. where the editor is expecting an initial word address, it is not permitted to actuate a function key.

Error 1A: "Repetition factor too high!"

A repetition factor can be specified when editing a mailbox. If the total length of the mailbox, taking the repetition factors into consideration, is more than 32 words, the editor outputs this error message.

Error 2C: "Left mailbox empty!"

If the CP 530 supplies a left mailbox with the length zero, this message is displayed.

Error 2D: "Right mailbox empty!"

If the interface module supplies a right mailbox with the length zero, this error message is displayed.

Error 35: "Illegal return message from PC!"

Error 36: "PC - Usart error!"

Error 37: "CP does not report!"

Error 38: ""USART error, PG side!"

Error 39: "USART error, PPC side!"

Error 3A: "Interface not ready!"

Error 3B: "Abort by CP!"

Error 4A: "Slave already receiving data!"

Error 58: "Slave xx failed!"

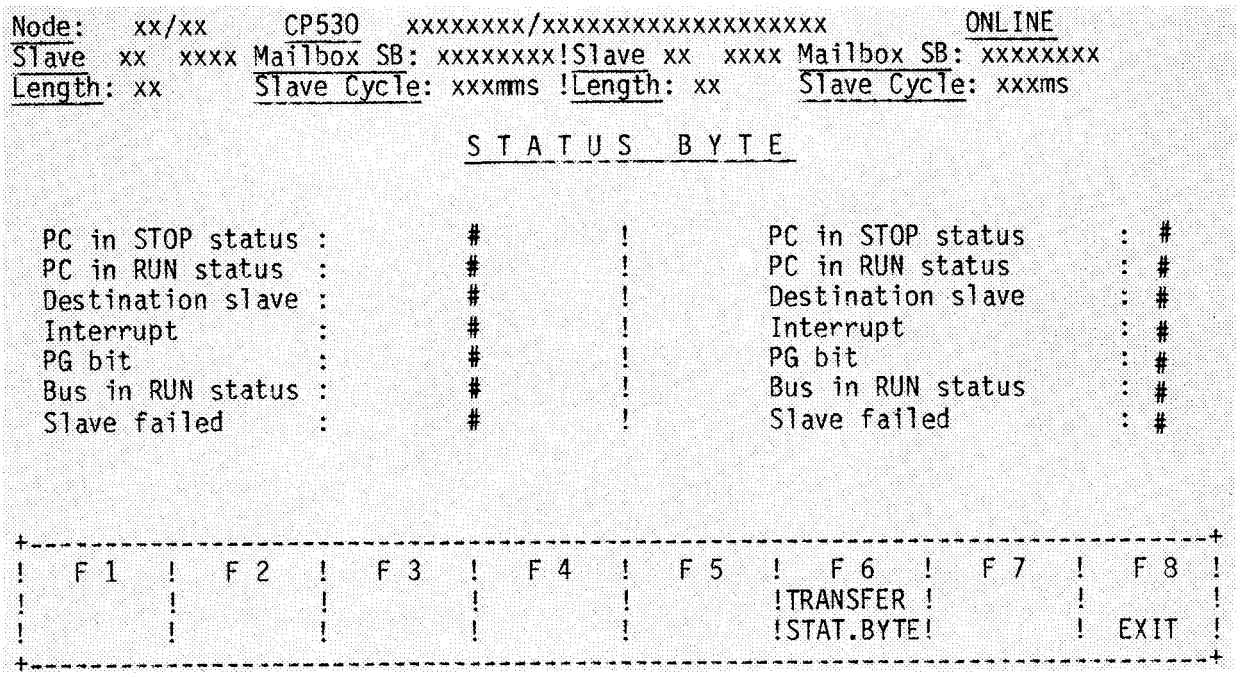
Error 59: "Slave xx not on bus!"

In these last two error messages the number of the respective slave is entered in the output fields.

Message 01: "Active!"

If the F6 function key, the transmit key or the F5 function key or hard-copy key are actuated, this message appears.

3.7.5 STATUS BYTE form



Significance of the output fields:

See STATUS form, Section 3.7.3.

Significance of the entry fields:

The status byte is displayed for both slaves. By changing the bit statuses (in the entry fields), the status bytes are transmitted to the slaves. This, however, is only possible when changing from the FORCE form and actuating the F6 key.

Assignment of the function keys:

F6: Transmitting the status bytes as displayed on the form to the slaves. Return to the FORCE form.

F8: Return to the FORCE form without changing the status bytes.

Possible error messages:

Error 01: "Illegal input!"

If a digit other than 0 or 1 or a blank appears in the entry fields. If the first two bits have the same status, i.e. if both are set or reset, this message also appears.

Example of STATUS FORCING:

```
Node: 23/00   CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx      ONLINE
Slave No. 5   REC Mailbox    !  Slave No. 16    SEND Mailbox

                          T E S T   A N D   S T A R T U P


        Specify the number of the slaves you wish to observe on the left and right
        halves of the screen.

        Specify with SEND/REC which mailbox of the respective slave is to be offered
        for STATUS and FORCE.

+-------------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8  !
!         !         !         !         !         !         !         !        !
!ACTIVATE !         !         !         !         !         !  HELP   !  EXIT  !
+-------------------------------------------------------------------------------+
```

Actuating F1 causes the STATUS form to appear. The interface module then supplies, for example, the right mailbox first, displays it on the screen and then displays the left mailbox.
Function key 3 is depressed while the interface module accesses the data of the slave.

The following display appears on the screen:

```
Node: 23/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx        ONLINE
Slave  5  REC Mailbox SB:  01001100!Slave 00 SEND Mailbox SB:10001110
Length: 24 Slave Cycle: 159ms! Length:  5   Slave Cycle: 234ms
Destination: 00                !  Destination: 16

                          S T A T U S

  0:     KH= ABCD               !   0: I(T= 735.2
  1:     KM= 0100110001110000 !   1: KY= 234,189
  2:     KH= 12CF               !   2: KS= ";
  3:  5  KH= 3E6A               !   3:  KM= 1011100010101001
  8:     KG= +1423148-05        !   4:  KF=  -21555
  9:11 KS= §Q                   !   5:
 20:     KM= 1110000111010110 !
 21:     KC= 391                !
 22:     KM= 0101101001011010 !
 23:     KY= 24, 1              !
 24:                            !
                                !

+-------------------------------------------------------------------------------+
!  F 1  !  F 2   !   F3   !   F4   !   F5   !   F6   !   F 7  !   F   8 !
!       ! FORCE  ! FIXING ! FORMAT ! PRINT  ! STORE  !        !       !
!       !  RIGHT ! OFF    ! LEFT   ! MAILBOX ! FORMATS !        !  EXIT !
+-------------------------------------------------------------------------------+
```

The cursor is now positioned in the left mailbox. The formats can now be changed in the mailboxes. The following formats are permissible: KH, KF, KS, KM, KT, KC, KG, and KY.

A "?" appears for characters which cannot be converted to the format specified.

The next function selected is "FORCE RIGHT". The following appears on the screen:

```
Node: 23/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx          ONLINE
Slave 5 REC Mailbox SB: 01001100!Slave 00 SEND Mailbox SB:10001110
Length: 24 Slave Cycle: 159ms! Length: 5   Slave Cycle: 234ms
Destination: 00                !  Destination: 16
```

F O R C I N G *

0:	KH= ABCD	0:	KT= 735.2
1:	KM= 0100110001110000	1:	KY= 2'34,189
2:	KH= 12CF	2:	KS= ";
3:	KH= 3E6A	3:	KM= 1011100010101001
4:	KH= 3E6A	4:	KF= 21555
5:	KH= 3E6A	5:	
6:	KH= 3E6A		
7:	KH= 3E6A		
8:	KG= +1423148-05		
9:	KS= §Q		
10:	KS= §Q		
11:	KS= §Q		

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
EXECUTE FORCING	ABORT FORCING	CHANGE TARGET	STATUS BYTE	PRINT MAILBOX	STORE FORMATS		EXIT

The contents of the right mailbox can now be edited. Using the F4 key, the status byte can be examined more closely and edited. Each bit has a certain function and can be set or reset.

For a description of the function keys, see Section 3.7.4.

3.7.6 BUS TEST function

The BUS TEST function of the COM 530 enables minibus traffic to be executed step by step, whereby the data exchanged between CP 530 and the nodes can be observed in the form of mailboxes and, in certain cases, modified. When the bus test is called, the SINEC 1 cycle is stopped at the end of the polling list.
Operator procedure depends on whether or not interrupts triggered by slaves occur during bus processing.

1. Procedure without interrupts (master-slave traffic)

Each stepin the bus test consists of two parts:

A) In the first part, the operator can examine and also modify the send mailbox of the CP 530 to the current slave (the slave currently being processed). This First part is completed by actuating the F1 "SEND MAILBOX" softkey.

The following reaction is displayed on the programmer screen:

- Acknowledgement by the slave of the send mailbox of the CP 530

- The receive mailbox received by the slave and its acknowledgement.

- The cycle time for the entire traffic
  Send-receive .... in master-slave traffic
  Send-receive-send ..... in slave-slave traffic

- Softkey F1 now has the inscription "NEXT STEP", enabling the second part.

B) The second part, which is also initiated by softkey F1 (now "NEXT STEP"), enables the minibus to poll the next slave in the polling list.

The following reaction is displayed on the screen of theprogrammer:

- The send mailbox of the CP 530 to the next slave in the polling list.

  Softkey F1 has the label "SEND MAILBOX" once more, i.e. the first part is active again (see 1. A.).

2. If an interrupt occurs

In this case, after the first part has been executed, an operator input must now be made:

A) The following reactions appear on the screen:

- The slave causing the interrupt is acknowledged

  The send mailbox of the CP530 to the interrupting slave appears

  The send mailbox from the interrupting slave to another node appears

  The "INTERRUPT CYCLE" message appears

  Softkey F1 ("NEXT STEP") permits the execution of the next step.

B) After this intermediate step, the previous procedure is followed as under 1.B. as long as no further interrupt occurs.
If a further interrupt occurs, the screen reaction is as in 2.A.

In the BUS TEST forms, send mailboxes of the CP 530 appear on the left of the screen. Only these mailboxes can be modified. On the right half of the screen, the mailbox which is sent back to the CP 530 by the respective slave (direct traffic) appears
or
the mailbox which is sent from the current slave to another node (cross traffic). appears.

As the SINEC-L1 traffic is executed step by step in the BUS TEST, the occurrence of frequent interrupts could prevent normal processing of the polling list. For this reason, a soft key function permits interrupts to be enabled or disabled.

The forms for the two parts of the operation for master-slave and slave-slave traffic follow:

BUS TEST form 1 - 3

This form appears in the first step:

```
Node:    xx/xx        CP530   xxxxxxxx/xxxxxxxxxxxxxxxxxx    ONLINE
SEND CP 530 to slave xx SB: xxxxxxxx!
Destination: xx                Length: xx!

                               B U S   T E S T

                  xxxx,xxxx,xxxx          !
                                          !
N   F                                     !
u   o                                     !
m   r                                     !
b   m                                     !
e   a                                     !
r   t                                     !
                                          !
                                          !
                                          !
                                          !

+-------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8   !
! SEND    !DISABLE  ! CHANGE  ! STATUS  ! PRINT   ! STORE   !         !         !
! MAILBOX !INTERRUPT!DESTINAT.! BYTE    ! MAILBOX ! FORMATS !         ! EXIT    !
+-------------------------------------------------------------------------+
```

The left half of the screen shows the send mailbox of the CP 530 which is to be sent to the current slave.
Modifications can be made with the usual S5 operator input procedures.

Assignments of the function keys:

F1: Sending the send mailbox to the current slave. Causes the BUS TEST 2 form to appear for the second step.

F2: The inscription of the key is alternately "DISABLE INTERRUPT" or "ENABLE INTERRUPT". This key makes it possible to service the slaves in accordance with the polling list even if an interrupt occurs.

F 3: This key makes it possible to come to the "DESTINATION" field in the form header and change the destination for the send mailbox.

F4: Function as in FORCING (Section 3.7.4)

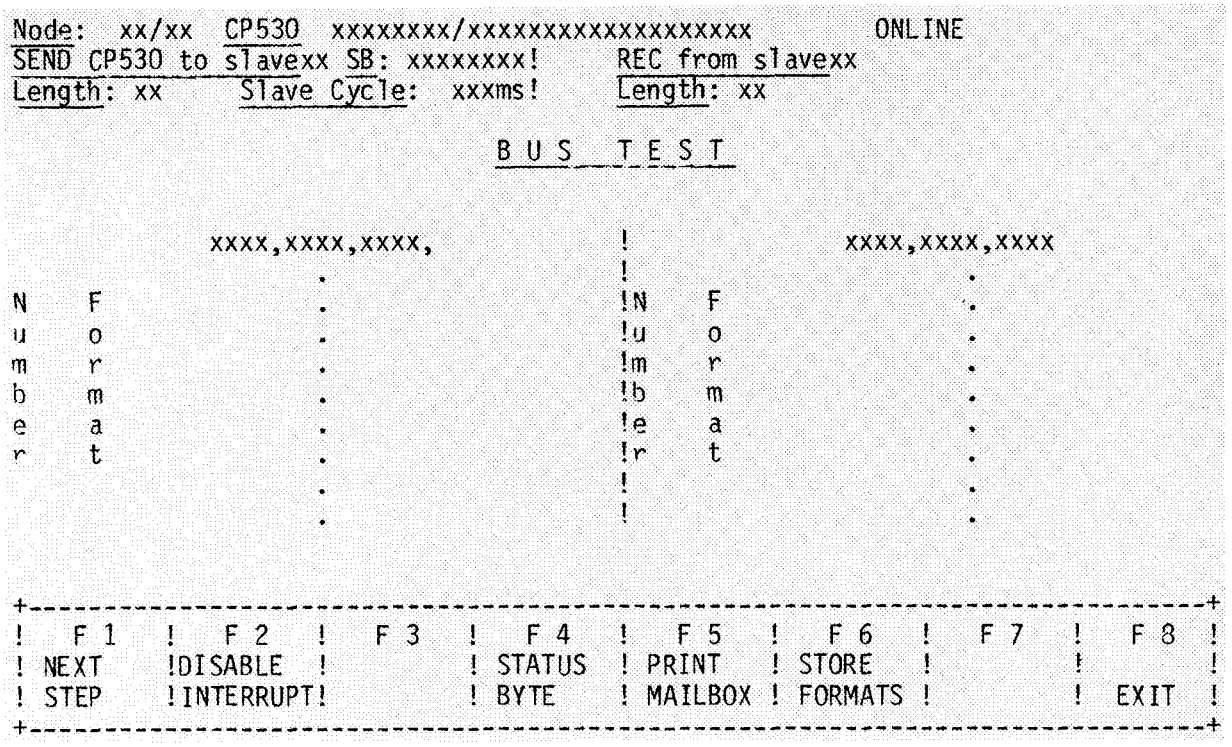
F5: Function as in FORCING (Section 3.7.4)

F6: Function as in FORCING (Section 3.7.4)

F8: Return to the TEST form. The SINEC L1 cycle continues running.

BUS TEST 2 form

This form is displayed in the second step:



The send mailbox from the previous form remains displayed. It can no longer be edited, however. The receive mailbox for the current slave appears on the right of the screen. The formats can be modified here.

Assignment of the function keys:

Fl: Polling the next slave. Return to the BUS TEST 1 form.

For all other function key assignments, see BUS TEST 1 form.

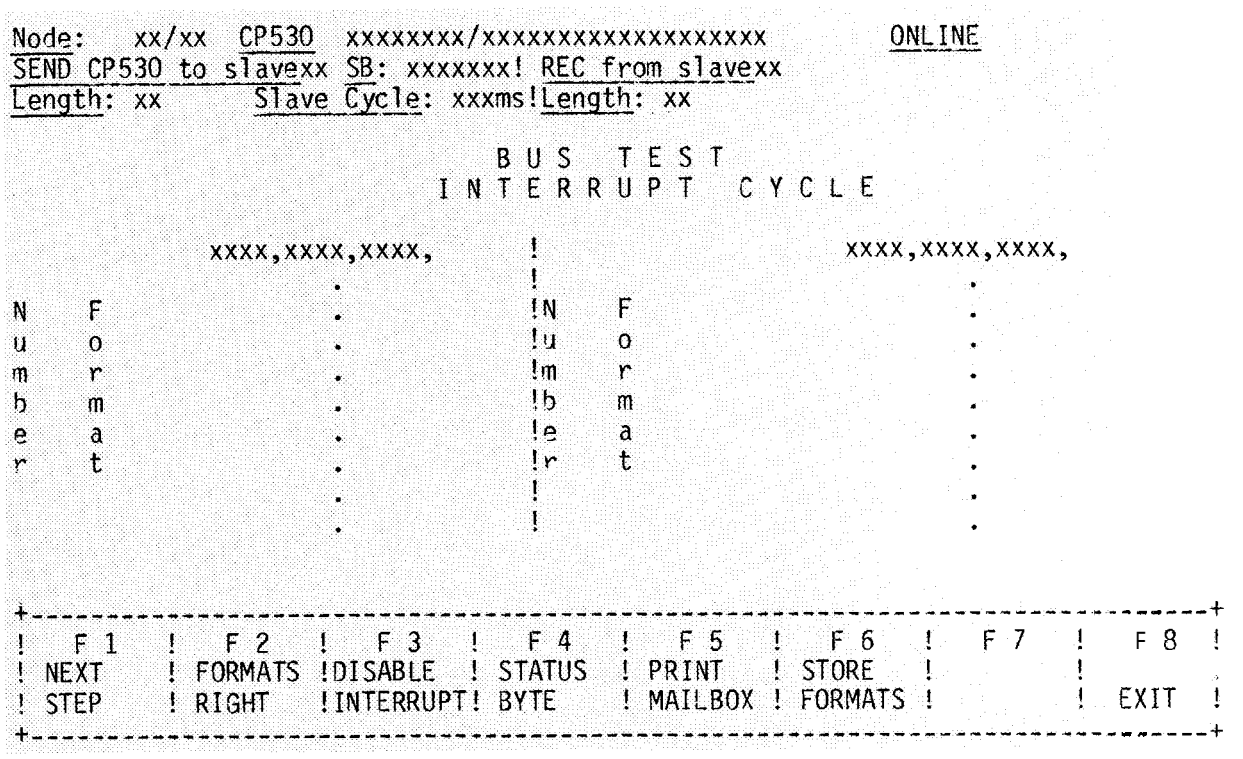
On the right of the screen, the mailbox from the slave to the master appears or the mailbox of the slave to another slave appears (slave-slave traffic).

In the case of cross traffic, the following form header appears:

Node: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxx ONLINE
SEND CP530 to slave xx SB: xxxxxxxx! CROSS TRAFFIC Slave xx to slave xx
Length: xx Slave Cycle: xxxms! Length: xx

BUS TEST 3 form

This form appears (in the second step) if an interrupt occurs:



The left half of the screen displays the send mailbox from the CP530 to the slave sending the interrupt.
On the right, the send mailbox from the current slave to the master or another slave appears. Editing is not possible in either mailbox, but the formats can be changed. Whether or not slave-slave traffic is taking place is indicated in the header.

Assignment of the function keys:

F1: Bus traffic continues. If no further interrupt occurs, the BUS TEST 2 form appears once more. The original send mailbox for the slave whose turn it should have been is displayed on the left and on the right the respective receive mailbox appears.
If, however, a further interrupt occurs, the BUS TEST 3 form reappears with the mailboxes to and/or from the slave sending the interrupt.

F2: This key must be actuated to position the cursor for changing the formats in the required mailbox. The inscription is alternately FORMATS RIGHT and FORMATS LEFT.

For further function key functions, see the BUS TEST 1 form.

Possible error messages:

Error 0B: "Inhibited key!"

Error 12: "INSERT/ERASE not possible here!"

The cursor is in the last line and an attempt has been made to insert or erase in the mailbox.

Error 13: "Unknown data identifier!"

The formats can be changed in the mailboxes. This error occcurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted"

In connection with conversions from all formats to the formats KT, KC, KS or KG.

Error 17: "Beginning - no further!"

If the mailbox is paged upwards but the first line is already displayed.

Error 18: "End - no further!"

If the mailbox is paged downwards and the last line is already displayed.

Error 19: "Do not press function key!"

If the cursor is in the first field, i.e. where the editor is expecting an initial word address, it is not permitted to press a function key.

Error 1A: "Repetition factor too high!"

A repetition factor can be specified when editing a mailbox. If the total length of the mailbox, taking the repetition factor into account, is greater than 32 words, the editor outputs this error message.

Example of-BUS TEST

The bus test function has been selected. The mailbox of the master to the slave, e.g. slave 17, is displayed.

```
Node: 8/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx          ONLINE
SEND CP530 to slave 5 SB: 10111110!
Destination: 5          !Length: 21

                                B U S T E S T

        0:  KH= ABCD                !
        1:  KM= 1011001110001111    !
        2:  KH= 2345                !
        3:  KH= 2345                !
        4:  KH= 2345                !
        5:  KH= 2345                !
        6:  KH= 2345                !
        7:  KH= 9876                !
        8:  KG= +1423148-05         !
        9:  KF= +32767              !
       10:  KY=  17,30              !
       11:  KC= 789                 !


+-------------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8   !
! SEND    !DISABLE  ! CHANGE  ! STATUS  ! PRINT   ! STORE   !         !         !
! MAILBOX !INTERRUPT!DESTINAT.! BYTE    ! MAILBOX ! FORMATS !         !  EXIT   !
+-------------------------------------------------------------------------------+
```

The mailbox can be edited. If function key F1 is actuated, the mailbox is sent to the slave and the receive mailbox is read.

Three different cases can now occur:

a) Slave sends mailbox to master

b) Slave sends to another slave

c) Interrupt

If case a) occurs, the form has the following appearance:

```
Node:  8/00  CP530  xxxxxxxxx/xxxxxxxxxxxxxxxxxx        ONLINE
SEND CP530 to slave 5 SB: 10111110      !               REC from slave 5
Length: 21            Slave Cycle: 320ms!               Length: 3

                                B U S   T E S T


        0:  KH= ABCD                          !   0:  KT= 735.2
        1:  KM= 1011001110001111              !   1:  KY= 234, 189
        2:  KH= 2345                          !   2:  KS= ",
        3:  KH= 2345                          !   3:
        4:  KH= 2345                          !
        5:  KH= 2345                          !
        6:  KH= 2345                          !
        7:  KH= 9876                          !
        8:  KG= +1423148-05                   !
        9:  KF= +32767                        !
       10:  KY= 17, 30                        !
       11:  KC= 789                           !

+-------------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8  !
! NEXT    !DISABLE  !         ! STATUS  ! PRINT   ! STORE   !         !        !
! STEP    !INTERRUPT!         ! BYTE    ! MAILBOX ! FORMATS !         ! EXIT   !
+-------------------------------------------------------------------------------+
```

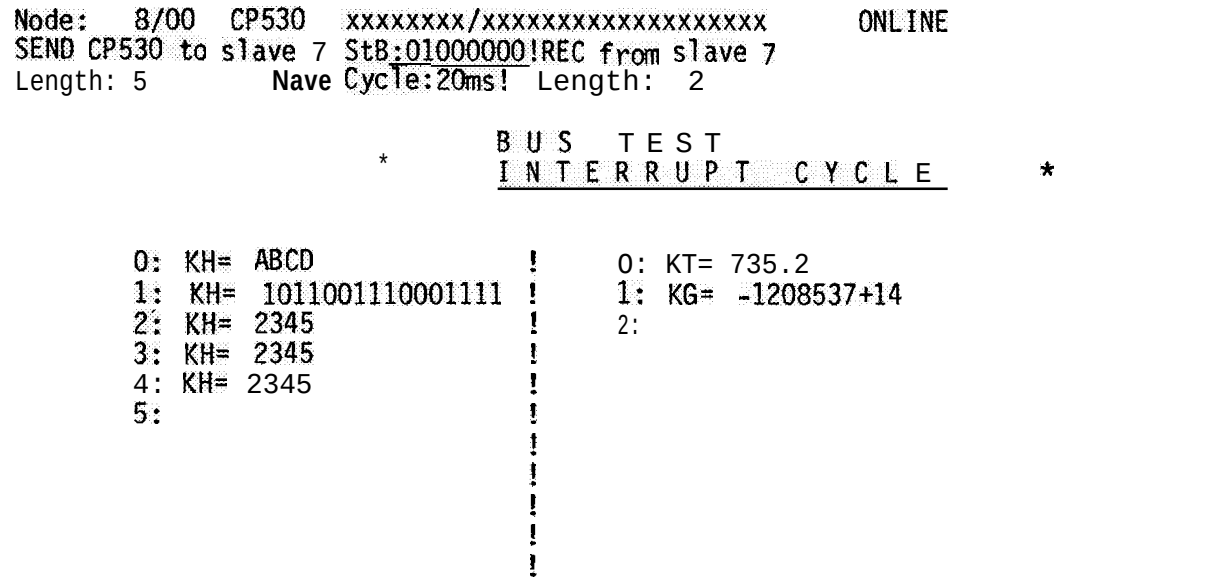
If case b) occurs, the form appears as follows:

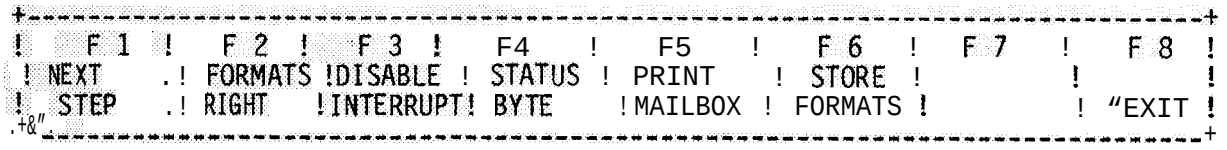
```
Node: 8/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx          ONLINE
SEND CP530 to slave 5 SB: 10111110       !CROSS TRAFFIC   Slave 5 to slave 9
Length: 21       Slave Cycle: 320 ms     !Length: 1

                              B U S   T E S T


        0:  KH  ABCD                      !   0:  KT 735.2
        1:  KM= 1011001110001111          !   1:
        2:  KH= 2345                      !
        3:  KH= 2345                      !
        4:  KH= 2345                      !
        5:  KH= 2345                      !
        6:  KH= 2345                      !
        7:  KH= 9876                      !
        8:  KG= +1423148-05               !
        9:  KF= +32767                    !
       10:  KY= 17, 30                    !
       11:  KC= 789                       !

+-------------------------------------------------------------------------------+
!   F 1   !   F 2   !   F 3   !   F 4   !   F 5   !   F 6   !   F 7   !   F 8  !
! NEXT    !DISABLE  !         ! STATUS  ! PRINT   ! STORE   !         !        !
! STEP    !INTERRUPT!         ! BYTE    ! MAILBOX ! FORMATS !         !  EXIT  !
+-------------------------------------------------------------------------------+
```

In both cases, function key F 1 “NEXT STEP” can be used to display the send mailbox Of the master to the next slave in the polling list. However, only the formats can be changed here.

Case c) Interrupt

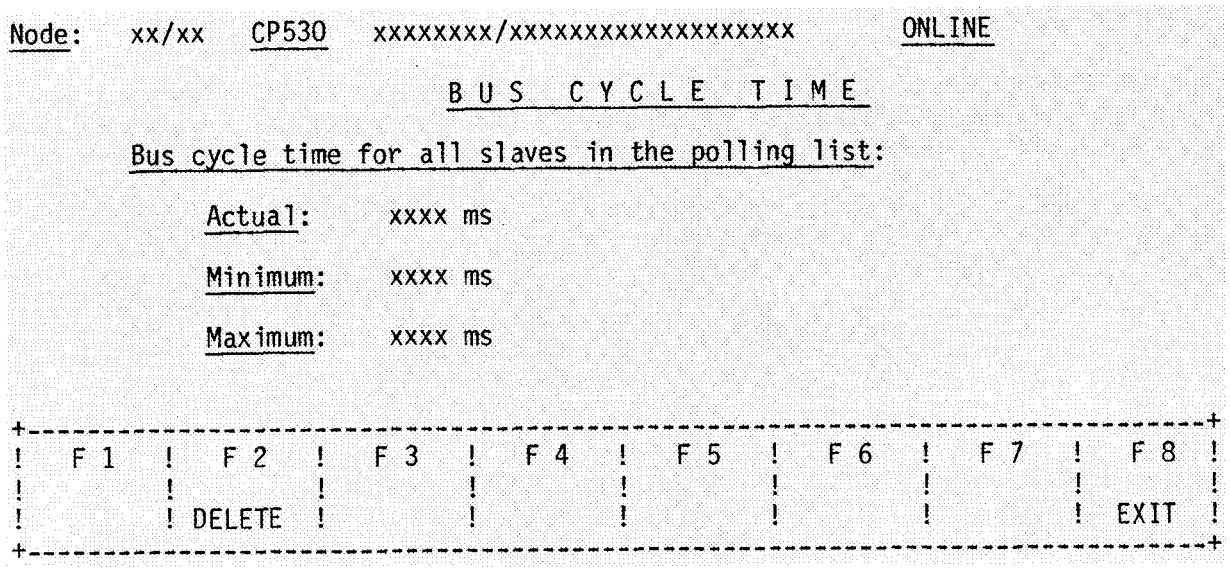
An interrupt has occurred, i.e. the send mailbox of the master to the slave sending the interrupt and the mailbox of the slave sending the interrupt to the master or to another slave (slave-slave traffic) are displayed.





Now the left or right formats can be changed.
Actuation of the F 1 key causes a return to case a) or b) if no further interrupt occurs.

3.7.7 CYCLE TIME form



Significance of the output fields:

ACTUAL : In this field the current bus cycle time is displayed. It is updated approximately every half second.

MINIMUM: A non-return pointer shows the lowest value.

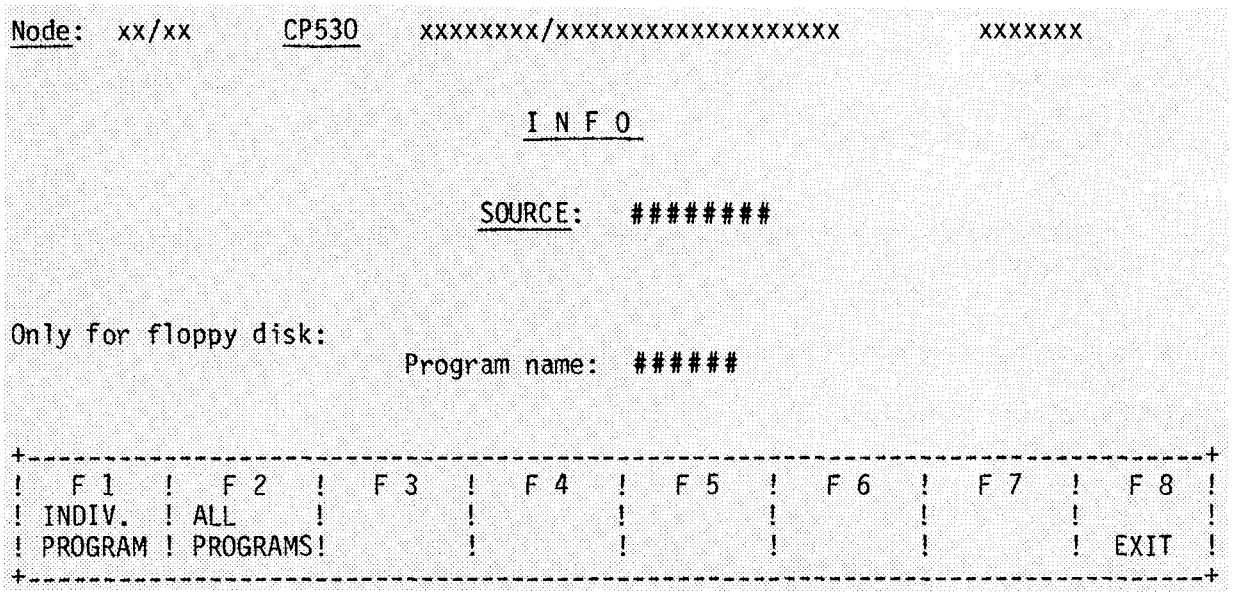
MAXIMUM: A non-return pointer shows the highest value.

Assignment of the function keys:

F2: The interface module receives the command to delete the bus cycle times.

F 8: Return to the TEST form.

3.8 INFO form



Significance of the entry fields in the INFO form:

SOURCE:

Here it is possible to specify the source without changing the mode. An EPROM can also be given as source!

Possibilities:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP530, EPROM

These alternatives can be entered using the HELP function.

Default-: ONLINE mode . . . . . CP 530
OFFLINE mode . . . . FD

PROGRAM NAME:

If a diskette (FD) is specified as source, the program name under which the user data are stored on the diskette must be entered here.
All program names on the user diskette can be entered with the aid of the HELP function.

Assignment of the function keys:

F1: Information is given on whether a SYSID identifier and/or polling list and/or interrupt list is available either in the CP 530 or in the EPROM or on the user diskette under the program name specified.
Selects the INFO 1 form.

F2: All program names on the user diskette are listed. Selects the INFO 2 form to appear.
(Only possible for 'FD' source!)

F7: HELP function for entering the source and available program names.

F8: Return to the CONFIGURATION form.

Possible error messages:

Error 01: "Illegal input!"

Source must be specified. The following are possible:

Mode	Source
OFFLINE	FD, EPROM
ONLINE	FD, CP 530, EPROM

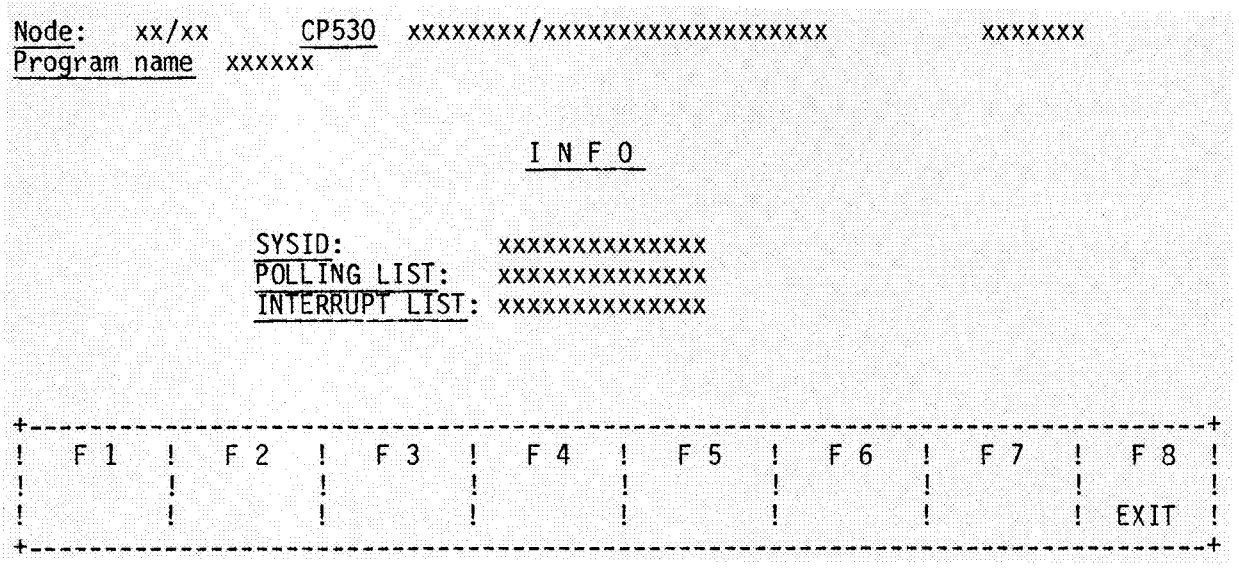
Error 29: "No program name for EPROM and CP 530!"

If the EPROM or CP 530 is designated as source, no information can be received on "all programs" (i.e. all program names on the user diskette) as only one user program can be stored in the EPROM or CP 530.

Error 46: "Incomplete input"

If a diskette (FD) is given as source, a program name must be specified.

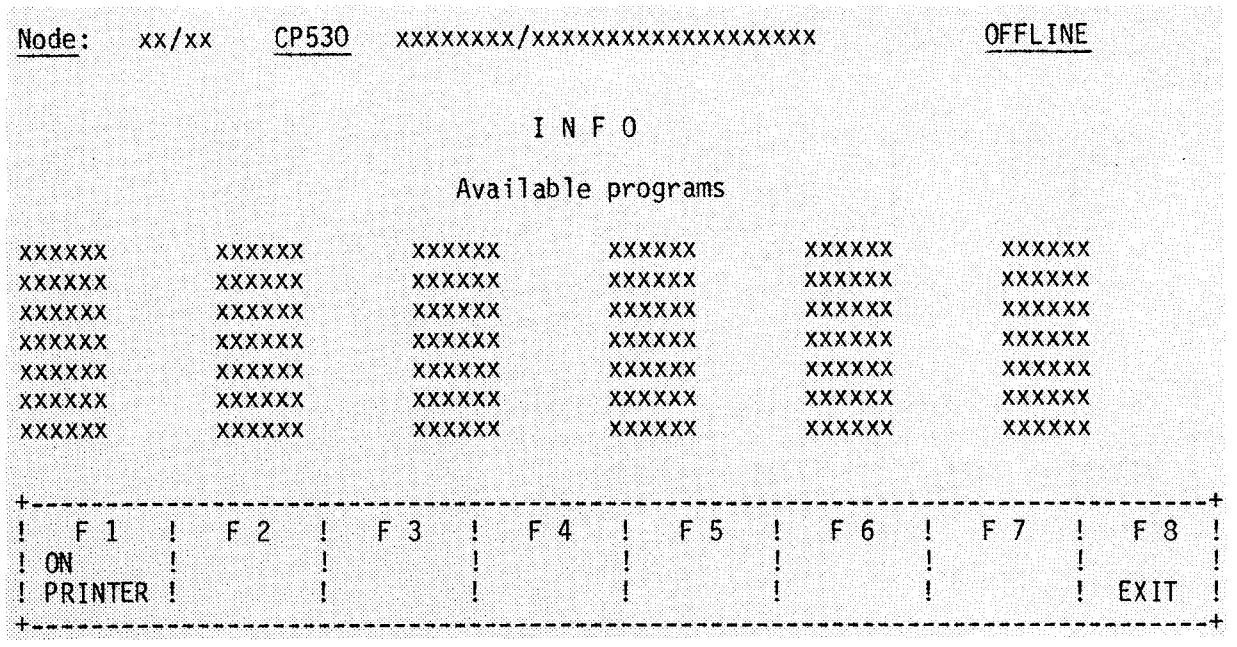
INFO 1 form



AVAILABLE or NOT AVAILABLE appear in the output fields.

F 8: Return to the CONFIGURATION form.

INFO 2 form



All available CP530 programs on the diskette are listed.

Assignment of the function keys:

F l: Listing on the printer and return to the INFO form

F 8: Return to the INFO form

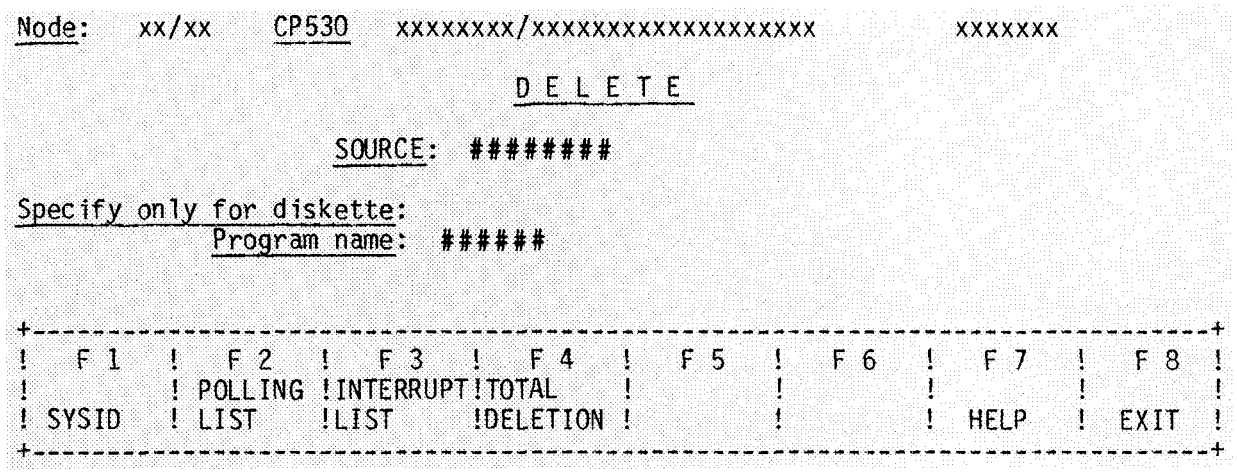
3.9 DELETE form

In "Delete" mode, user data stored in the CP 530 or on diskette under a certain program name can be deleted.

The following possibilities are available:

Individual deletion (i.e.: Deletion of SYSID, polling or interrupt lists)

- Total deletion (i.e.: Deletion of SYSID, polling and interrupt lists)



Significance of the entry fields in the DELETE form:

SOURCE:

It is possible to specify the source here without changing the mode.
Possibilities:

Mode	Source
OFFLINE	FD
ONLINE	FD, CP 530

The alternatives can be entered using the HELP function.

Default: ONLINE mode . . . . . CP 530
OFFLINE mode . . . . . FD

PROGRAM NAME:

If diskette (FD) is specified as source, a program name under which the bus parameters are stored on the diskette must be entered here.

All program names on the user diskette can be entered with the aid of the HELP function.

Assignment of the function keys:

F 1: Deletion of the SYSID identifier in the source specified.
Deletion is indicated with the "Active" and "SYSID deleted" displays.

F2: Deletion of the polling list in the source specified:
Deletion is indicated by the "Active" and "Polling list deleted" messages.

F3: Deletion of the interrupt list in the source specified:
Deletion is indicated by the "Active" and "Interrupt list deleted" messages.

F4: Delete all user data:
The SYSID, polling list and interrupt list in the source specified (i.e. in the CP 530 or on diskette under the name stated) are deleted.
In the error message line the message 'Delete all ?' appears. The user is prompted to acknowledge.

Menu for acknowledgement:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
YES	NO						EXIT

The deletion is indicated by the "Active" and "SYSID deleted" or "Polling list deleted" or "Interrupt list deleted" messages.

F7: HELP function for entering the source and the program names on the floppy diskette.

F 8: Return to the CONFIGURATION form.

Possible error messages:

Error 01: "Illegal input!"

Source must be specified. The following are possible:

Mode	Source
OFFLINE	FD,
ONLINE	FD, CP 530

Error 31: "Polling list does not exist!"
Error 32: "Interrupt list does not exist!"
Error 33: "SYSID identifier does not exist!"

One of these three error messages appears if the respective data type is not available on the CP 530 or on the user diskette under the program name specified.

Error 46: "Incomplete input"

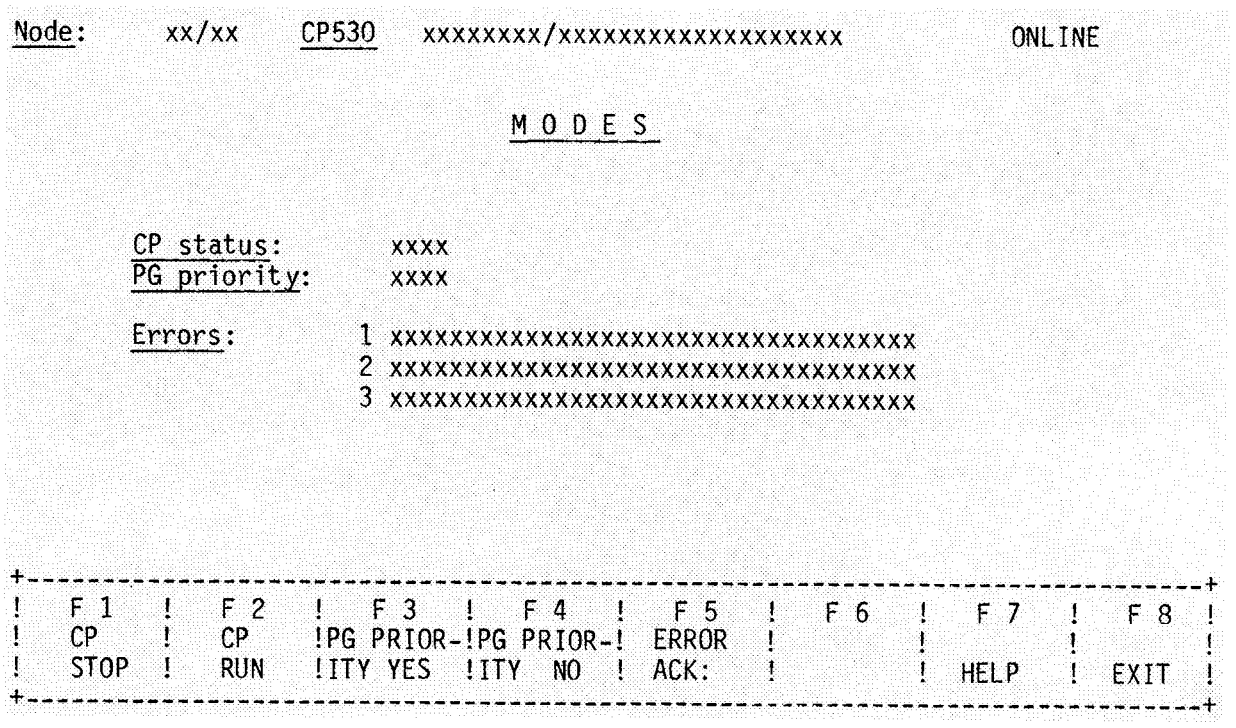
If diskette (FD) is given as source, a program name must be specified.

3.10 Setting the operating mode (MODES form)

3.10.1 Operating Modes

The foil owing operating modes can be set:

- Starting the CP 530 (CP RUN),
- Stopping the CP 530 (CP-STOP),
- PG priority (ON and OFF),
- PG priority ON means that the programmer connected directly or indirectly to the PG (programmer) interface has absolute priority.



Significance of the output fields in the MODES form:

CP STATUS:

Here the current status of the CP530 is displayed: The possibilities are 'STOP' or 'RUN'.
The 'STOP' or 'RUN' states can be forced with function keys F 1 or F 2.

PG PRIORITY:

The PG priority field also shows the current CP status (possibilities: YES and NO).
These settings can be changed with the F 3 and F 5 function keys.

ERRORS:

The error messages collected in the CP 530 are displayed in the ERROR fields and updated approximately every half second.

After function keys F 1 to F 4 have been actuated, the "CAUTION: DANGER(IUS STATE - Operate keys again" message is output and only the relevant function keys appear in the menu.

Assignment of the function keys:

F 1: The CP 530 is brought to the STOP state at the end of the current cycle through the polling list. As a dangerous system status can occur, the user is prompted to acknowledge with the following menu:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
CP STOP	CP RUN						EXIT

F 1: After positive acknowledgement, the CP 530 is brought to STOP: This is indicated as follows:
The "EXECUTED" message is output and
in the CP STATUS field the word "STOP" appears.

F 2: The operating mode is not changed as the CP 530 is already in the RUN state. The "CP IN RUN STATE" message appears.
Only
a) Positive acknowledgement with F 1 or
b) Negative acknowledgement with F 8 are meaningful.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

F 2: The CP 530 is brought to the RUN state. As a dangerous system status can occur, the operator is promted to acknowledge with the following menu:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
CP STOP	CP RUN						EXIT

F 1: The operating mode is not changed as the CP 530 is already in the STOP state. The "CP in STOP STATE" message appears.

Only
a) positive acknowledgement with F 2 or
b) negative acknowledgement with F8 are meaningful.

F 2: After this positive acknowledgement, the CP 530 is set to RUN. This is indicated as follows:
- The "Executed" message is displayed and
in the CP status field the word RUN appears.

F 8: A negative acknowledgement means that the operating mode is not executed. Return to main menu.

F 3: The CP 530 is brought to the "PG heck YES'' mode. As a dangerous system status can occur, the operator is prompted to acknowledge with the following menu:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
		!PG PRIOR-!ITY YES	!PG PRIOR-!ITY NO				EXIT

F 3: After this positive acknowledgement the CP530 is brought to the "PG check YES" state.
This is indicated as follows:
- The "Executed" message is displayed
- "YES" appears in the PG PRIORITY field

F 4: The mode is not changed as the CP 530 is already in the "PG check NO" mode. The "PG PRIORITY OFF" message appears.
Only
a) Positive acknowledgement with F 3 or
b) Negative acknowledgement with F 8 are meaningful.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

F 4: The CP 530 is brought to the "PG check NO" mode. As a dangerous system status can occur, the operator is prompted to acknowledge with the following menu:

F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8
		!PG PRIOR-!ITY YES	!PG PRIOR-!ITY NO				EXIT

F3: The operating mode is not changed as the CP 530 is already in the "PG check YES" mode. The "PG PRIORITY ON" message appears.
Only
a) Positive acknowledgement with F 4
b) Negative acknowledgement with F 8 are meaningful.

F 4: After this positive acknowledgement, the CP530 is brought to the "PG check NO" status.
This is indicated as follows:
- The "Executed" message is displayed
- "NO" appears in the PG priority field.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

F7: The HELP function causes a description of the operating modes which can be set with F 1 to F 4 to be displayed on the screen. This description can be exited with function key F 8 (EXIT).

F 8: Return to the CONFIGURATION form.

3.10.2 Error Messages from the CP 530

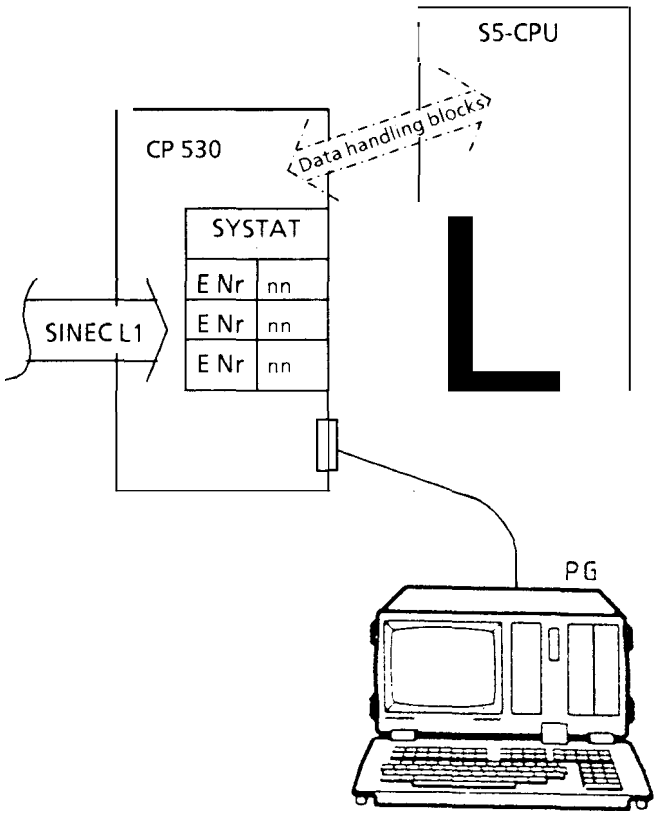
In the field for these messages in the mode form, messages will appear when problems occur with data transfer from the CP 530 to the network or the S5 CPU.

The error number may be broadly classified in 4 classes, and most errors come with an extension number to further define the problem (see the table "Error List"). The extension may be used as follows.

a) Description of a group error number in more detail, e.g. the result of the self test produces class 1 errors and these can only be cured by exchanging the module. Example: 10,7 is hardware error 7.

b) Description relating to a job number containing an error
e.g. 54,177: job 177 not defined

c) Description of a slave with which the problem occurred
60,3: slave 3 not in poll list.



Error classes

Introduction to error classes:

more efficient error messages result
- the user can define a general reaction
to a failure type.

There are also other errors but these are not shown in this way since the failure relates to those parts of the system necessary to transmit the error number.

Error class

Class	Error No.	General Description	Operator reaction	Level of service
1	10-29	Possible hardware fault	HW.-Test / exchange	1. Operator 2. Service
2	30-49	Operational errors	Switches, Sub Module, Sequence: Check / correct	Operator
3	50-69	Parameter / program error	Diagnosis by PG necessary S5-SW changes	Programmer / Commissioning Engineer
4	70-90	Status messages	Record	Operator

Error list SYSTAT

Class	Error No.	Error Ext.	Description
1	10	XX	Error 10: Hardware error No. XX
	11	XX	Error 11: Internal error message No. XX
2	30	0	Error 30 : Waiting for SYNCHRON
	31	o	Error 31: Wrong CP module
	32	0	Error 32: PG Function operative
	33	0	Error 33: PC is in stop: no slave send is possible
	34	0	Error 34: CP is not in the STOP mode
	35	o	Error 35: The CP may not RUN: switch is set to STOP
3	50	0	Error 50: Erronious (absent) SYSID
	51	0	Error 51: Erronious (absent) Poll list
	52	o	Error 52: Erronious (absent) Alarm list
	53	XXX	Error 53: Job XXX not defined
	54	XXX	Error 54: Job description XXX not recognized
	55	XXX	Error 55: Job No. XXX only for reveive
	56	XXX	Error 56: Job No. XXX only for send
	57	XXX	Error 57: Job No. XXX only compatible with the master
	58	XXX	Error 58: Job No. XXX only compatible with the slave
	59	XXX	Error 59: Job too long
	60	XXX	Error 60: Slave XXX not in polling list
	61	XXX	Error 61: The job number XXX uses an undefined list
	62	XXX	Error 62: Job number XXX SBR not allowed
4	70	o	Err& 70: Bus error
	71	XXX	Error 71: Connection to slave has an interference
	72	XXX	Error 72: The wrong slave (No.XXX) has replied
	73	XXX	Error 73: Slave No. XXX has failed

4. APPENDIX

4.1 COM 530 Error List with hints on how to proceed

The following message texts appear in the message line; the numbers are internal references only and should not appear.

Error 01: Illegal input

The COM 530 executes validity checks for the data entered: The cursor blinks in the field in which an illegal entry has been made.

In the field only certain limit values are-permissible
- In the field only certain alternatives are permissible
- For further details, see Operating Manual.

Error 02: Illegal key!

The key which has just been actuated is illegal in the field in which the cursor is blinking. The following are possible:
- Only digits and letters
- Only letters
- Only digits
- Only special characters
- Only digits and '+' and '-'
- Only digits and '+' and '-' and '='
- Only hexadecimal characters
- Only digits between 0 and 3
- Only digits 0 and 1

are permissible.

Error OB: Inhibited key!

Error OD: Programmer memory overflow!

Action: The programmer must be switched off with the power switch and switched on again and the COM 53"0 must be reloaded. If this error occurs frequently, the manufacturer must be contacted.

Error OE: System file not in drive 0!

A system file required for correct execution of COM 530 is not in drive 0. Action: Insert. system diskette in drive 0.

Error OF: Program load error!

A system file required for correct execution of COM 530 cannot be loaded. Action: Use original system diskette..

Error 10: Starting address Length ofmailbox!

Error 11: Mailbox empty!

Error 12: INSERT/ERASE not possible here!

The cursor is in the last line and an attempt has been made to erase or insert data in the mailbox.

Error 13: Unknown data identifier!

The formats can be changed in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 15: Input missing!

Error 16: Data cannot be interpreted!

Possible in conversions from all formats to formats KT, KC, KS or KG.

Error 17: Beginning - No further!

If the mailbox is paged upwards and the first line is already displayed.

Error 18: End - No further

If the mailbox is paged downwards and the last line is already displayed,

Error 19: Do not press function key!

If the cursor is in the first field, i.e. where the editor is expecting an initial word address, no function key may be pressed.

Error 1A: Repetition factor too high!

When a mailbox is edited, a repetition factor can be specified. If the entire length of the mailbox taking the repetition factor into account, is greater than 32 words, the editor outputs this error message.

Error 21: Source = Destination?

In "Transmit" mode, source and destination devices must not be identical.

Possibilities:

Destination / Source	DISKETTE	CP 530	EPROM
DISKETTE	-	+	+
CP530	+	-	+
EPROM	+	+	

\+ permissible
illegal

DISKETTE User diskette in drive 1
CP 530 SINEC L1 communications processor
EPROM Plug-in EPROM/EEPROM submodule

Error 23: Transmission error

This error message indicates an error in all transmission types:

TRANSMIT PG675 ==> CP 530
- No transmission cable plugged in
- Transmission cable incorrectly plugged in
- Wrong transmission cable
- An EPROM submodule is inserted in the CP 526, all accesses for writing or deletion are possible
- CP 530 not in operation
- Transmission time exceeded

PG675 ==> FD
Please contact manufacturer
- Hardware fault
- COM 530 system error

Error 24: Read SYSID identifier

Error 25: SYSID, polling and interrupt lists non-existent!

In "Print all" mode, no data can be listed as no user data are stored in
the CP 530 memory (ONLINE mode) or
- on the diskette under the program name specified (OFFLINE mode)

Error 26: No blank fields permitted!

Blank fields are not permitted in polling lists or interrupt lists.
The cursor is in the first vacant field found.
Action: - Delete the respective field or
- Enter a slave number in the respective field.

Error 27: No double slave number in interrupt list!

Each slave may only appear once in an interrupt list.

Error 28: No EPROM/ EEPROM submodulepluggedin!
An EPROM/EEPROM submodule must be plugged into the receptacle for it on the programmer.
See SINEC L1 Operating Instructions.

Error 29: No program names for EPROM and CP 530!
"Info" mode:
Where EPROM or CP 530 is specified as source, no information can be given for "All programs" (all program names on diskette), as only one user program can be stored in the CP 530 or EPROM.
Source = CP 530 or EPROM:
Only "Individual program" mode is possible in which information is given on whether the SYSID identifier and/or polling list and/or interrupt list exist or not.

Error 2A: WARNING: No EPROM driver on the system diskette. No EPROM calls!!!
The COM 530 draws the user's attention to the fact that there is no EPROM driver on the system diskette being used.
WARNING: An attempt to program data into an EPROM or to read data from an EPROM causes a program crash. The "INTERRUPT TRAP HALT" message appears in the form.
Action: Put originalCOM530 system into drive 0 and start program once more.

Error 2C: Left mailbox empty!
If the interface supplies a mailbox (left) with length 0, this error message appears.

Error 2D: Right mailbox empty!
If the interface supplies a mailbox(right)withlength 0, this error message appears.

Error 31: Polling list does not exist!

Error 32: Interrupt list does not exist!

Error 33: SYSID identifier does not exist!
These three error messages appear in
- output
- Delete
- Transmit

modes if the respective data type does not exist in the CP 530 or in the EPROM or on the user diskette under the program name specified.

Error 35: Illegal return message from PC!

Error 36: PC-Usart error!

Error 37: CP does not report!

Error 38: Usart error on programmer side!

Error 3A: Interface not ready!

Error 3B: Abort by CP!

Error 3C: List non-existent!

Error 3D: Diskette directory full!
The diskette directory of the user diskette is full. A new file cannot be initialized.
Action: Use new diskette
Delete files not required

Error 3E: User diskette full!
The space available on the diskette being used is insufficient for storir the file.
Action: Use another diskette
Delete files not required

Error 41: EPROM not erased!
The EPROM must be completely erased before it is programmmed.

Error 42: VPP error!

The supply voltage level (for the EPROM) is not within the permissible range. This suggests a hardware fault on the EPROM or at the PG 675 interface.

Action: Service or repair

Error 43: Comparison error!

After bus parameters have been programed into an EPROM from diskette or the CP 530, the contents of the EPROM are compared with the data on the diskette or CP 530. If the data are not identical, this error message is output.

Action: Erase EPROM and reprogram

Error 45: Address out of range!

Error 46: Incomplete input!

In the
- output
- transmit
- print
  info
- delete

modes, a program name must be given for the source diskette (the cursor is in the PROGRAM NAME field).

Error 47: Slave failed!

Error 48: Wrong mode!

Error 49: Slave not in polling list!

Error 4A: Slave already receiving data!

Error 54: EPROM/EEPROM submodule type illegal!

The wrong EPROM/EEPROM submodule is being used:
See SINEC L1 Operating Instructions

Error 56: Mode change CP in STOP

Error 57: Faulty slave cycle

Error 58: Slave xx failed!

Error 59: Slave xx not on bus!

In both of these error messages, the number of the respective slave is entered in the output fields.

Error 60: Only hexadecimal characters permitted!

Error 61: -32.768 fixed-point +32.767!

Error 62: FORMAT: aaa,bbb;aaa,bbb... only Up to 255!

Error 63: ASCII characters only!

Error 64: FORMAT: aaa, b ; b . . . 0 to 3!

Error 65: Wrong counter word!

Error 66: Only keys 0 and 1 permissible!

Error 67: FLOATING-POINT SYNTAX: +1234567-89

Error 70: Incorrect CP module!

Error 71: Error when programming the EEPROM on the CP.

Error 72: It is not possible to set the CP to Run since the Stop/Run switch is at Stop.

Error 73: CP 530 is a slave.

Error 75: Slave failure

Error 77: PG data has been overwritten.

Error 78: Slave not in polling list.

COM 530 Error list

The following error messages do not appear in the error message line as usual but in the body of the form. These are all fatal error messages which means you are now at the Operating System Level.

x stands for 'A' (drive O) or 'B' (drive 1)

BDOS error on x:

If this error message appears, a write access was not made on the diskette The diskette is write-protected or initialized for read-only access. In the second case, the diskette can be initialized for read/write access by reloading COM.

BDOS ERR ON x: had sector

This error message suggests a hardware fault or a wrongly or badly formatted diskette. Reformat the diskette and subsequently reload COM 530.

By actuating the return key " <--'" the error can be ignored.

ERROR: DISK WRITE: "x:XXXXXXXX.$$$"

There is no more room on the diskette in drive 1.
Action: Delete files not required
Use another diskette

ERROR: NO DIRECTORY SPACE - Xxxxxxxx'$$$

The diskette directory is full.
Action: Delete files not required
Use another diskette

ERROR: USER ABORTED

Transfer has been aborted by actuating a key.

SECTOR NOT FOUND:

A sector of the diskette cannot be accessed. This means that data might be lost.
By actuating the return key " <--" the error can be ignored.

4.2 References

/1/ S5DOS Description
for PG675 (Herr Trapp/E814)

SIEMENS

SIMATIC S5 SINEC L1 Local Area Network

CP 530 Communications Processor
COM 530 on the PG 615 Programmer

Instructions Order-No.: GWA 4NEB 8110521 -02c

Contents

1. Introduction

The SINEC L1 Local Area Network permits communications between up to 31 SIMATIC S5 programmable controllers of the U range in master/slave mode.

The COM 530 software package described here for the PG 615 programmer is used for the following: to assign parameters to the SINEC L1 communications processor, for SINEC L1 mode selection, for controlling data traffic (including diagnostics) and for the documentation and archiving of bus (network) parameters on EPROM/EEPROM submodules and on the printer.

Procedures for programming and assigning parameters are described in the "Programming instructions" for SINEC L1. All operator inputs are interactively via screen forms (entry fields) and function keys. The meaning of the function keys is shown either on the last display line or in the texts assigned to the numbers.

The following are described:

- Format of the interactive screen forms
- Meaning of the entry and output fields in the forms
- Meaning and effect of the softkeys and function keys

Output fields in the interactive form are marked xxxxx.
In these fields COM 530 shows current statuses and data entered in previous operations.

Entry fields in the interactive form are marked #####
Entries can be made in these fields via the hexadecimal keyboard or the funktion keys.

COM 530 error messages are shown in the second last display line.

Bus parameters: All parameters necessary to operate the LAN are generated by the user with the aid of the COM 530 software (SYSID identifiers, polling list, interrupt list, see SINEC LI Operating and Programming Instructions).

Meaning of the cursor control and function keys

⇓⇑ : The cursor is positioned to the first entry field in the line above (below). The next (previous) part of the interactive screen form appears in the last (first) entry field. If the form contains further parts, this is shown by arrows pointing up and down on the form. If the form has no entry fields, it is only used for consultation.

⇐⇒ : The cursor is positioned to the next (previous) character within an entry field so that a character may be changed without changing the whole field.

DEL Deletion of a character at the cursor position.

INS Insertion of a character at the cursor position.

BRK This key allows progression movement up to the next form in the hierarchy without evaluating the data in the current form.

ENTER This key means „ Store".

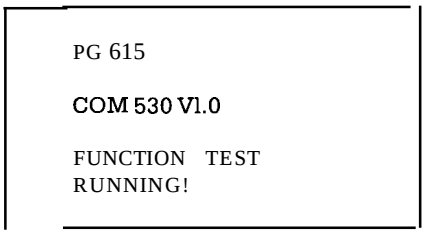
0...9
without Shift key : Entry of numbers O to 9 or selection of functions

0...5
with Shift key : Entry of hexadecimal numbers A to F.

3. Operator Input and Operation of the COM 530 on the PG 615 Programmer

3.1 Power up

After the power is switched on, the following display appears:



This indicates that internal tests are taking place. After a few seconds, the Defaults form appears and the cursor blinks at the ON-LINE position.

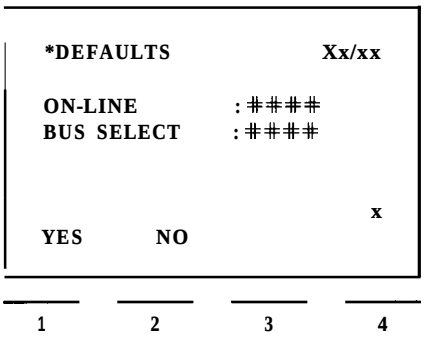
In ON-LINE mode, the output fields in the top line show a two-digit mode number which has been entered in the Bus Select form. The master has the number xx/0, and a number greater than O applies for a slave.

YES is the default for the first entry field and NO for the remaining entry fields.

The function keys and their meaning:

1 : YES is entered at the cursor position

2 : No is entered at the cursor position

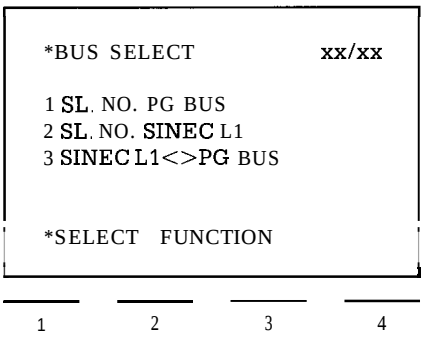


ENTER : Change to INITIAL STATE form

If YES is entered in the entry field of the BUS SELECT line, it is possible to page further in the BUS SELECT menu. This is indicated by an arrow pointing downwards in the last column of the second-last line. The BUS SELECTform appears if the ⇓ key is pressed while the cursor is in the bottom-most entry field.

3.3 BUS SELECT Form

With this function a mode can be selected. To do this it is necessary to enter the slave number on the programmer bus/or the slave number on the SINEC LI LAN.



N. **B.:** The Bus Select process is integral only from the V2.0 software version onwards.

The function keys and their meaning:

1 : Change to the SL.NO.PG-BUS form

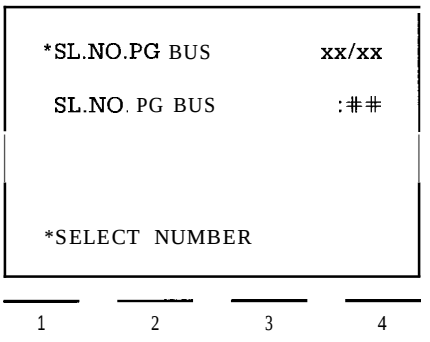
2 : Change to the SL.NO.SINEC L1 form

3 : Change to the SINEC LI /PG Bus form

BRK : Return to the DEFAULTS form

3.3.1 SL.NO.PG-BUS Form

The number of the slave with which the PG 615 is to communicate via the programmer bus (PG-Bus) can be entered with the numeric keys 0-9.



The function keys and their meaning:

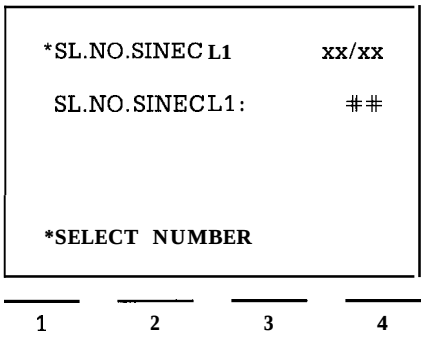
ENTER : Transfer the slave number entered and return to I NIT STATE form

BRK : Return to BUS SELECT form

3.3.2 SL.NO.SINEC LI Form

If it is desired to select a CP 530 which is connected to the SINEC LI network, and if the programmer is plugged directly into the relevant Master CP, then the slave number of the CP 530 is entered in this form with the numeric keypad.

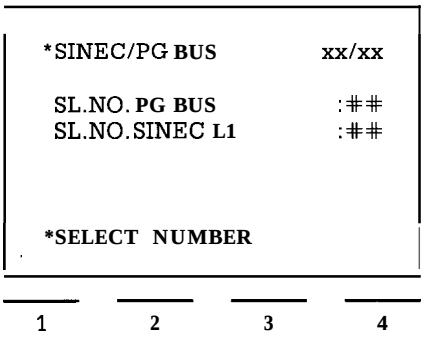
The meaning of the function keys is the same as in Subsection 3.3.1.



3.3.3 SINEC L1/PG-BUS

If the programmer is at a random position on the programmer bus and it is to communicate with a CP 530 connected to SINECL1 as a slave, then the number of the relevant Master CP through which the slave CP is to be addressed must be entered in this form, followed by the number of the slave CP itself.

The meaning of the function keys is the same as in Subsection 3.3.1.



3.4 I NIT STATE Form

Pressing the ENTER key after power-up brings the user to the I NIT STATE form.

The form consists of two parts which can be accessed with the ⇓ ⇑ cursor keys.

The function keys and their meaning:

ENTER : The Enter key effects a return to the start of INIT STATE form.

1 : Change to INPUT form.

2 : Change to OUTPUT form

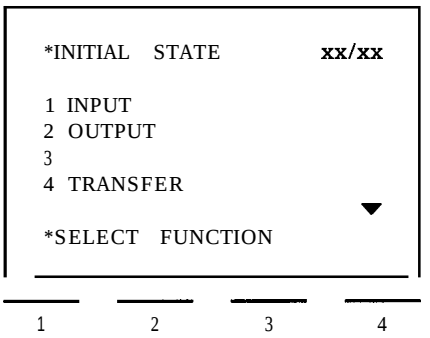
4 : Change to TRANSFER form.
With this function the various transfer modes can be selected

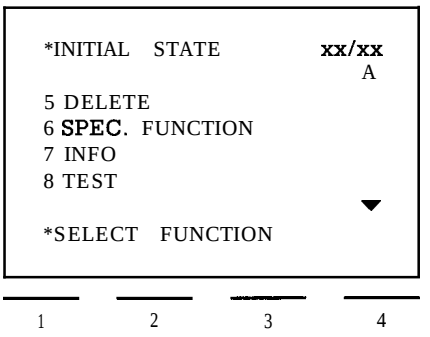
5 : Change to the DELETE form
With this function bus parameter can be deleted in the programmer, the CP 530 or also in an EEPROM submodule.

6 : Change to the SPECIAL FUNCTIONS form (Defaults, Operating modes)

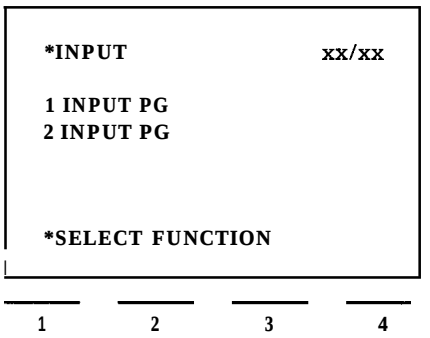
7 : **Information about the bus parameters in the programmer, the CP 530 or the submodule.**

8 : Change to testing the data traffic on SINEC LI LAN (TEST form)





INPUT form:



Keylock **switch in position 1: Input not possible**

Keylock **switch in position II:**
By pressing key I in the initial state, the user reaches the INPUT form.

The node numbers are taken from the Defaults form, and are used for addressing on the LAN in connection with "ON-LINE" functions.
The contents of the SYSID **data (Slave No. = O — "Master", Slave No. = 1 . . . 30— "Slave") controls the following forms for INPUT/OUTPUT:**
MASTER : Polling list and interrupt list maybe selected.
SLAVE : Error message of polling list or interrupt list is selected.

In the OFF-LINE mode, these identifiers originate from the SYSID **stored in the programmer. If these are not available or if "MASTER" is entered in the** SYSID, **a polling list or an interrupt list can be selected.**
If the polling list and interrupt list have been assigned parameters, the programmer may only accept a SYSID **for "Master".**

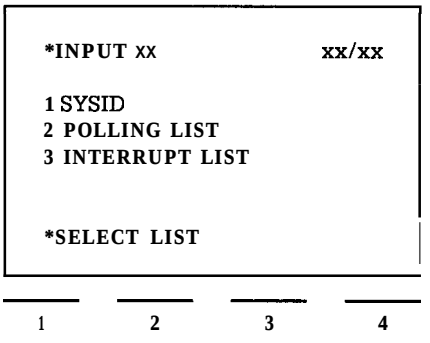
The function keys and their meaning:

1 **: Input to the programmer**

2 **: Input to the CP 530 (only possible in ON-LINE mode)**

BRK **: Return to "INIT STATE" form**

The following form appears on pressing keys 1 or 2:



PG or CP **appears in the output field in the top line, depending on whether 1 or 2 was pressed on the "INPUT" form.**

Function keys and their meaning:

1 **: Change to input of** SYSID **identifiers**

2 **: Generate polling list**

3 **: Generate interrupt list**

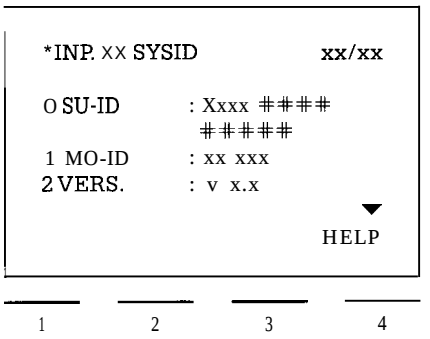
BRK **: Return to INPUT form**

If there is already a SYSID, **a polling list or an interrupt list in the** CP **530 or in the programmer, this will be indicated on input and can be changed.**

3.5 Input of User Data

3.5.1 SYSID INPUT Form

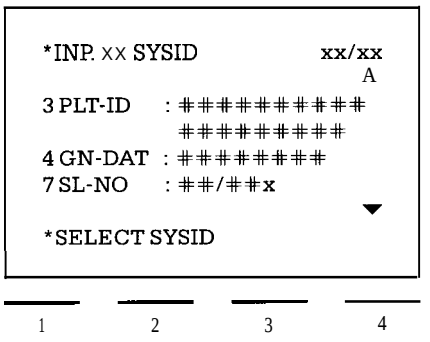
The SYSID data area is a memory area in the CP 530 used for general identification purposes.



Purpose of the SYSID:

- to give a clear description of the classification of an intelligent I/0 module (e. g. CP 530) in a programmable controller.
- to transfer parameters to an intelligent I/0 module which will lead to definite performance characteristics.
- to provide information about the firmware and software releases of the module.

SYSID identifiers appear in these forms when parameters are assigned ON-LINE. They can be overwritten.
The form sections are also filled in during OFF-LINE programming should SYSID identifiers have already been generated by entering them in the programmer, or transferred to programmer. Otherwise, the entry fields are empty.



Explanation of SYSID mnemonics:

SUB-ID : For submodule identification (e. g. 6ES5 375-0LA11)

MD-ID : Module identifier (e.g. CP 530)

VERS. : Software release

PLT-ID : Plant identifier
Free choice of name of up to 19 alphanumerical characters (only hexadecimal numbers can be entered on the PG 615 programmer).

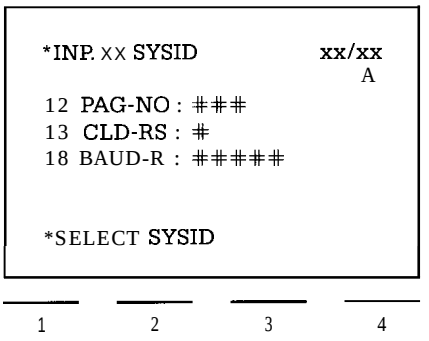
GN-DAT : Date of generation of bus parameters.
The format is optional (e.g. 1.5.85).

SL-NR : Slave number on the programmer (PG) bus (1-30) and slave number on SINEC L1.

PAG-NR : 3-digit number (l-254)

RSTRT : Cold restart

BAUD-R : Entry and changing of baud rate (only, 9,600 baud presently possible).



The form consists of three parts, which can be paged through with the cursor control keys.

Function keys and their meaning:

If no texts are displayed over digits 1-4 in the bottom line of the display, keys 1-F have their normal meaning. The numbers 1 -F can be written into the entry fields.
In some of the entry fields, the text assigned to keys 1-4 appears above the digits in the display. If one of the keys is pressed, the text assigned to it is written into the entry field.

BRK : Return to the “INPUT-XX” form.

ENTER : Store edited text in the programmer (PG) or CP 530. Return to the ‘INPUT’ form.

3.5.2 POLLING LIST INPUT Form

For minimum operation of SINEC LI only the SYSID and polling list are necessary. This list contains 64 locations for the entry of slave numbers; the order of these numbers establishes the order in which the slaves are to be addressed during network operation. In the simplest case, the natural order is maintained; however, by using all 64 locations and naming certain slaves several times, a certain order of priority can be achieved.

Simple case:

1	2	3	4	...	n

Giving priority of slave 3

1	2	3	4	5	3	6	7	...	...	3	n

The following form appears for entering a new polling list:

```
*INP. xx PO-LI                xx/xx

01:##



*SELECT POLL. LIST
```

1 2 3 4

The cursor is in the first entry field, in which a two-digit slave number can be entered. When both digits have been entered, the cursor jumps to the next field, which is then available for the next entry (max. 64). If only one digit is entered, the next field is reached by pressing the ⇒ key. Each position in the edited polling list can be reached via the cursor control keys and changes can then be made.

After several slave numbers have been entered, the form looks like this:

```
*INP. PG PO-LI                xx/xx
                                A
07:18    1   2   19   3   4
13:20   21   3    4   5  12
19:23   24  10   11  12  12
25:25   26  27   28  29  30
                              ▼
*SELECT POLL. LIST
```

1 2 3 4

Function keys and their meaning:

DEL : This key deletes the contents of the field in which the cursor is located. The subsequent slave numbers are then moved one location left.

INS : This key moves all slave numbers one location on. A new slave number can then be entered in the vacant location.

ENTER : This key causes the polling list to be stored in the communications processor or the programmer and then effects a return to the "INPUT" form.

BRK : Return to the "INPUT" form without storage of data (the data are simply lost).

3.5.3 INTERRUPT LIST INPUT Form

An interrupt list is generated in the same way as the polling list. The following form appears for entering a new interrupt list:

```
*INP. xx PO-LI                xx/xx

01:##



*SELECT INTR. LIST
```

1 2 3 4

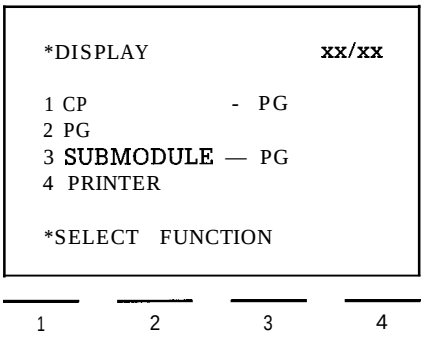
Further inputs and the meaning of the function keys are the same as when generating the polling list, (subsection 3.4.2).

In the input fields the slave numbers (1 –30) should be entered according to their priority in case of a bus interrupt.

N. B.: Each slave number may only be entered once

3.6 Display of User Data

If the key 2 is pressed while the "I NIT STATE" is being displayed, access is gained to the "OUTPUT" form.



Function keys and their meaning:

1 : Output of user data from the CP 530 and transfer into the programmer. If user data have already been generated in the programmer, the question "Overwrite?" must be acknowledged by pressing the ENTER key, if these data are no longer needed. Should these data still be needed, the user can abort with the BRK key and the data can then be saved (e.g. on a memory submodule).

2 : Output of user data from the programmer.

3 : Output of user data from the EPROM/EEPROM submodule. If there are already data in the programmer which are to be overwritten, the question "Overwrite?" must be acknowledged with the ENTER key.

4 : Output of user data on the printer connected to the programmer.

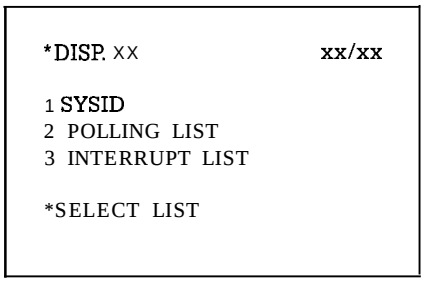
BRK : Return to "I NIT STATE" form.

3.6.1 Output of User Data on the Display

Pressing function keys 1 to 3 in the "DISPLAY" form enables access to the following form:

Form DISPLAY-XX:

CP 530 is master



The letters CP, PG or SM appear in the first output field of the top line, depending on which of the first three functions keys (1 -3) was pressed in the "DISPLAY" form.

The function keys and their meaning:

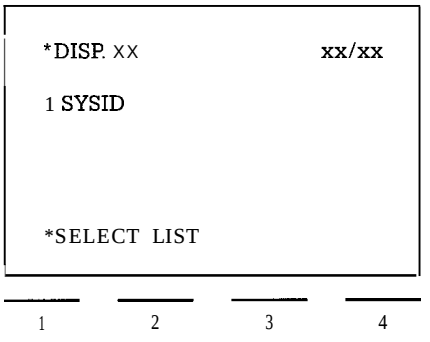
1 : Change to display of system parameters ("SYSID DISPLAY" form)

2 : Display of polling list (D ISP-POLI form) in the case of master data only.

3 : Display of interrupt (D ISP-INLI form) in the case of master data only.

BRK : Return to the "DISPLAY" form.

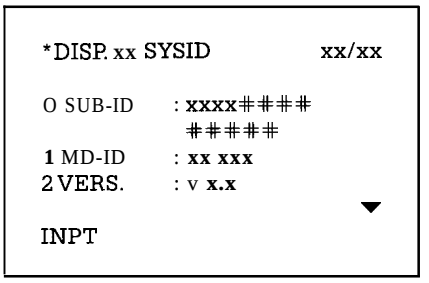
CP 530 is slave

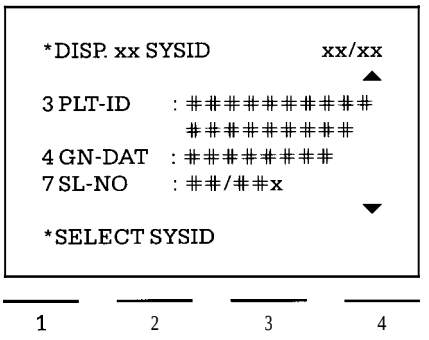


3.6.1.1 SYSID DISPLAY Form

The SYSID DISPLAY form consists of three parts which can be paged through using the cursor control keys.

Further details: 3.5.1 SYSID INPUT form.



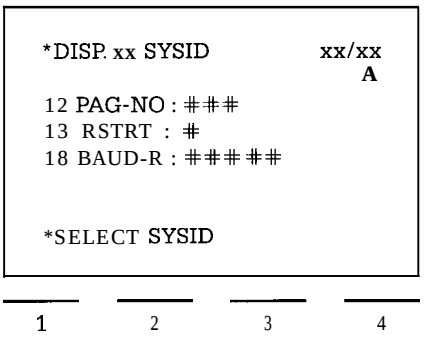


The function keys and their meaning:

1 : With this key, control can be passed direct to the input of SYSID identifiers, if the keylock switch is in position II (see Subs. 3.4.1).

BRK : Return to DISPLAY xx form

The text above digit 1 INP in the display does not appear when output from a memory submodule. It is therefore not possible to change to input.



3.6.1.2 POLLING LIST DISPLAY Form

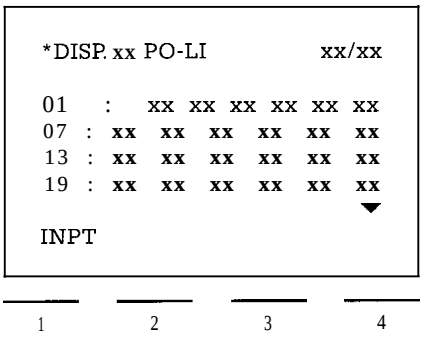
The following form appears on pressing key 2 in the DISPLAY xx form:

The polling list is displayed as generated

The function keys and their meaning:

1 : This key branches direct to the input of the interrupt list, if the keylock switch is in position II (Subs. 3.4.2).

BRK : Return to DISPLAY xx form.

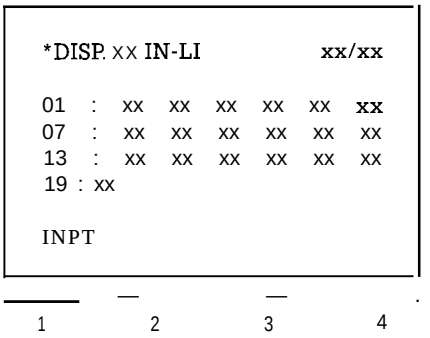


The text above digit 1 INP in the display does not appear if data is being output from a memory submodule. It is therefore not possible to branch the input.

3.6 Display of User Data

3.6.1.3 INTERRUPT LIST DISPLAY Form

On pressing key 3 in the **DISPLAY xx form,** the **following form** appears:



The interrupt list is displayed as generated.

The function keys and their meaning:

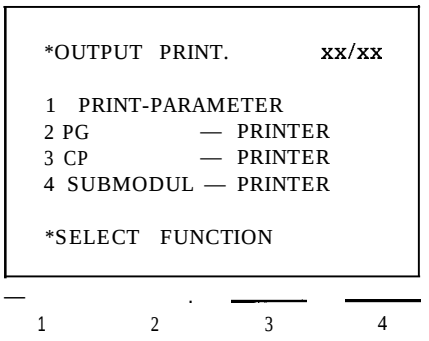
1 : This key branches direct to the input of the interrupt list, if the keylock switch is in position II (Subs. 3.4.2).

BRK : Return to DISPLAY xx form

The text above digit 1 INP in the display does not appear if data is being output from a memory submodule. It is therefore not possible to change to input.

3.6.2 Output of User Data on the Printer

With this function, all user-generated data can be output via the printer from the CP, the PG or the EPROM/EEPROM submodule. By pressing key 4 in the DISPLAY form, the following form appears:



The function keys and their meaning:

1 : Change to setting of print paramters (PRINT-PAR form)

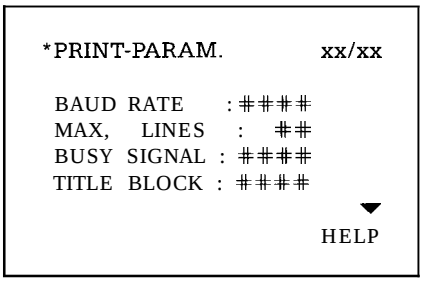
2 : Change to printing of user data generated in the programmer (PRINT START form)

3 : Change to printing of user data generated in the CP 530 (PRINT START form)

4 : Change to printing of user data from the EPROM/EEPROM submodule (PRINT START form)

BRK : Return to the OUTPUT form

3.6.2.1 PRINT PARAMETER Form



Explanation of print parameters:

BAUD RATE : Transmission speed between the PG 615 and the printer. If the cursor is in the BAUD RATE entry field, the BAUD RATE can be set between 300 and 9600 baud, using key 4 (HELP), The default on programmer power-up is 600 baud.

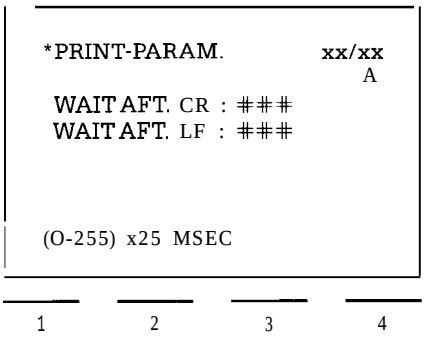
MAX. LINES : Maximum number of lines per page. If the cursor is positioned at the entry field, this number can be set to 72, using key 1, and to 66, using key 2. The default is 72 lines.

BUSY SIGNAL : If the printer has a busy line, “YES” must be set. “YES” and “NO” can be set using numeric keys 1 and 2, respectively. The default is “NO”.

TITLE BLOCK : Frame for the data to be printed. YES is set using key 1 and NO with key 2.

WAITAFT. CR. : Waiting time after “CR”. After the control character “CR”, there is a waiting period so that the print head can return to the beginning of the line. This period is a multiple of 25 ms and can be entered with the keys O to 9. The default is 20.

WAITAFT. LF : Waitina time after “LF” (as for waiting time after “-CR”).



The PG 615 transmits CR and LF as control characters for the printer. Before starting to print, the paper must be set to the top of the form.

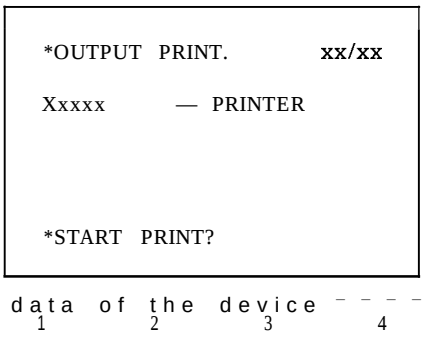
The function keys and their meaning:

BRK : Return to the “OUTPUT PRINT” form

ENTER : Transfer of edited values and return to “OUTPUT PRINT” form.

3.6.2.2 PRINT START Form

The following form appears on pressing keys 2 and 4 in the OUTPUT PRINT form:



The text PG, CP or SM appears in the entry field of the third line, depending on which key was pressed in the OUTPUT PRINT form.

The function keys and their meaning:

ENTER : The printout is started. All user data of the device specified are printed out, followed by return to the DISPLAY FORM.

BRK : Return to the OUTPUT PRINT form.

3.7 Transfer of User Data

When key 4 is pressed in the I NIT STATE form, the TRANSFER form appears. It consists of two parts, which can be paged using the cursor control keys ⇓⇑.

The function keys and their meaning:

1 : Transfer of user data from the CP 530 to the programmer

2 : Transfer of user data from the programmer to the CP 530

3 : Transfer of user data from the programmer to the EPROM/ EEPROM submodule

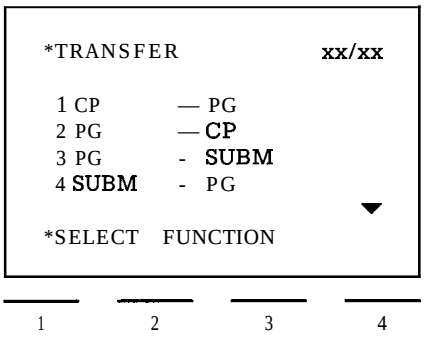
4 : Transfer of user data from the EPROM/EEPROM sub-modul

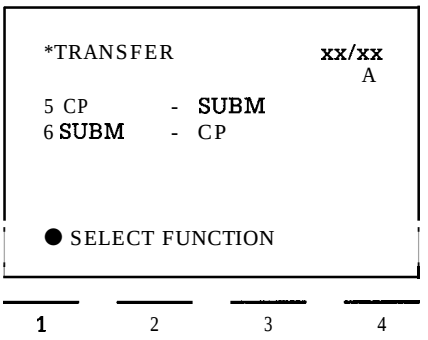
5 : Transfer of user data from the CP 530 to the EPROM/ EEPROM submodule

6 : Transfer of user data from the EPROM/EEPROM sub-module to the CP 530

Keys 1, 2, 4 and 6 effect a change to the TRANSFER xx form. Keys 3 and 5 effect a change to the SUBMODULE SELECT form after power-up or if the submodule has changed.

BRK : Return to INIT STATE form.





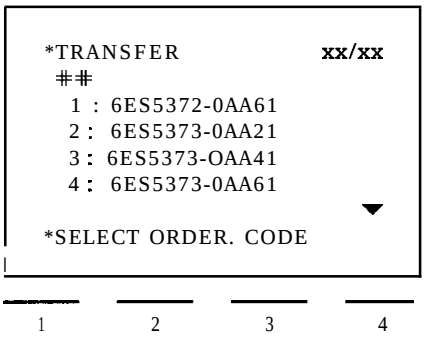
SUBMODULE SELECT form

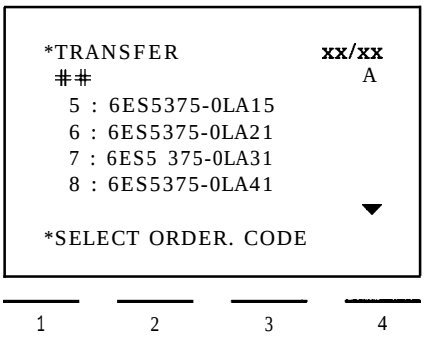
The ordering codes of permissible EPROM/EEPROM submodules can be examined in the following forms. The digit assigned to the MLFB number of the connected module must be entered in the input field. After pressing the transition key in the assigned module type is then set until the module is changed.

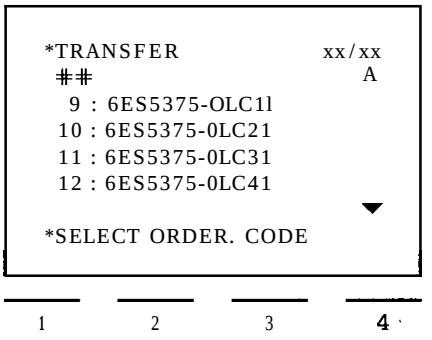
The function keys and their meaning:

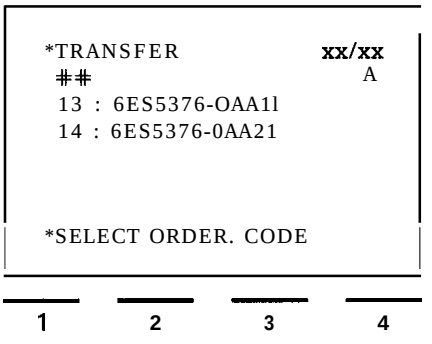
ENTER : Enter ordering code assigned to the digit and change to the TRANSFER xx form.

BRK : Return to the I NIT STATE form.









3.7 Transfer of User Data

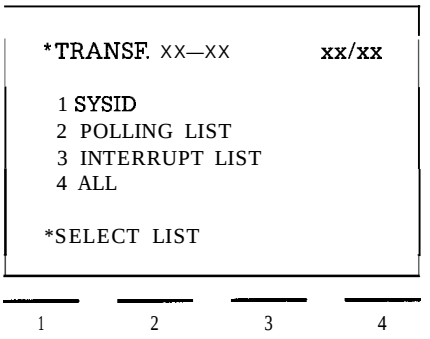
3.7.1 TRANSFER xx form

The function keys and their meaning:

1: Initiates transfer of SYSID identifiers
2: Initiates transfer of the polling list
3: Initiates transfer of the interrupt list
4: Initiates transfer of all source user data

Keys 1 to 4 effect transition to the TRANSFER ACKN form

BRK : Return to the TRANSFER form



In this form only those lists are offered which are in the source.

3.7.2 TRANSFER ACKN Form

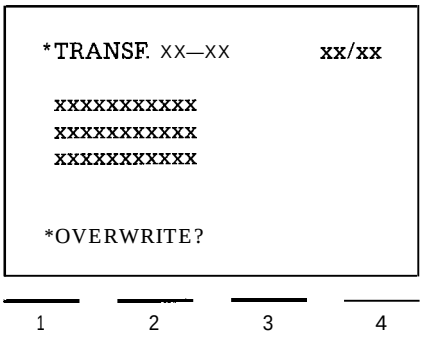
This form only appears when transfer is to be made to the programmer and lists already in the programmer are to be overwritten.

The designations of the user data to be transferred (SYSID, POLLING LIST, INTERRUPT LIST), are entered in the output fields of lines 3 to 5.

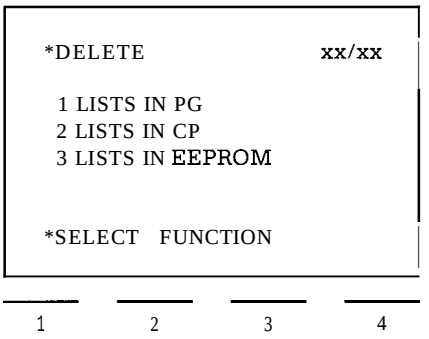
The function keys and their meaning:

ENTER : The transfer process is positively acknowledged and executed. Transfer to the module can take anything up to a minute, depending on memory type. On completion, return to TRANSFER form.

BRK : Return to TRANSFER xx form



With this function, user data generated in the CP 530, the programmer or in the EEPROM submodule can be deleted. It is possible to delete the polling list, the interrupt list and the SYSID individually.
After pressing key 5 in the I NIT STATE form, the DELETE form appears:



The function keys and their meaning:

1 : Deletion of user data in the programmer.
2 : Deletion of user data in the CP 530.
3 : Deletion of user data in the EEPROM submodule

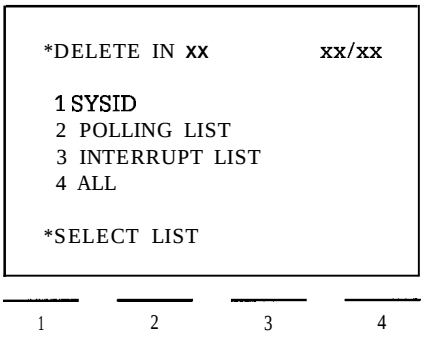
Keys 1 to 3 effect change to the DELETE xx form

BRK : Return to the I NIT STATE form

3.8.1 DELETE xx Form

This form always only displays the lists stored in the respective devices (e. g. CP 530).

In the first output field of the top line the text PG, CP or SM appears, depending on which key was pressed in the DELETE form.



The function keys and their meaning:

1 : Deletion of the SYSID
2 : Deletion of the polling list
3 : Deletion of the interrupt list
4 : Deletion of all user data

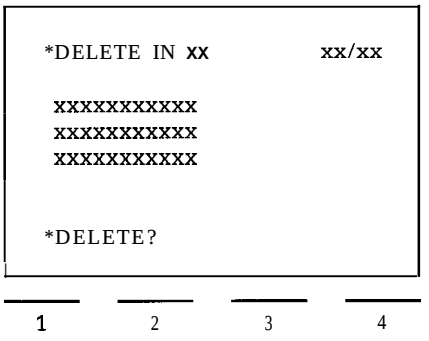
N. B.: On deletion of all user data, only the first 2Kbytes of the EEPROM submodule are deleted even if the submodule has a memory capacity of more than 2Kbytes. The erasing process can last up to a minute.

Keys 1 to 3 effect a change to the DELETE ACKN form.

BRK : Return to the DELETE form.

3.8.2 DELETE ACKN Form

In the first output field of the top line the text PG, CPor SM appears, depending on which key was pressed in the previous form. The designations of the user data to be deleted appear in the output fields of lines 3 to 5.



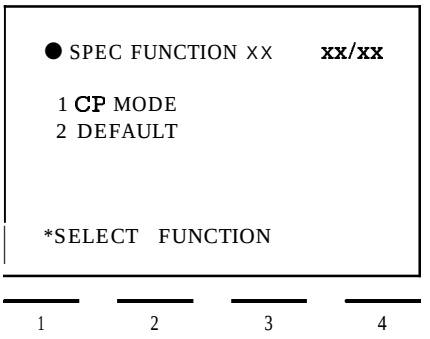
The function keys and their meaning:

ENTER : Deletion process is activated, followed by return to the DELETE form

BRK : Return to the DELETE xx form

3.9 SPECIAL FUNCTIONS Form

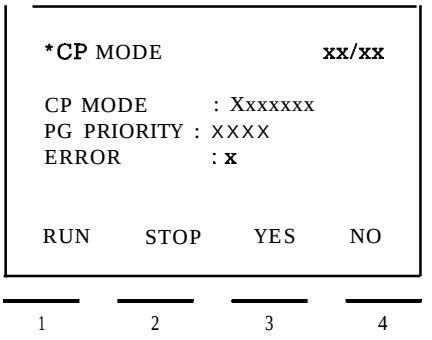
When key 6 is pressed in the I NIT STATE form, the following form appears:



The function keys and their meaning:

1 : Change to the CP MODE form (only possible ON-LINE)
2 : Change to the DEFAULTS form (see Subs. 3.2)
BRK : Return to I NIT STATE form

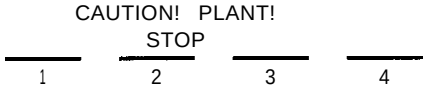
3.9.1 CP MODE Form



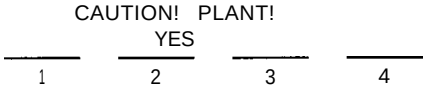
The information in this form is continually updated by the CP in a function similar to "Status".

The current status of the CP 530 **(RUN, STOP) is displayed** in the "CP Status" field. The "PG Priority" field likewise displays the current status of the CP 530. (YES, NO). Function keys 1 and 2 refer to the CP status, and function keys 3 and 4 to the PG programmed priority.
If the CP is in the RUN status, for example, and the PG priority is YES, only function keys 2 and 4 can be used in the menu. If one of these function keys is pressed, (e. g. STOP), the message CAUTION! PLANT! appears, and in the menu line key 2 only. By pressing function key 2 again, the CP is brought to Stop status and only function keys 1 an 4 appear in the menu line. This process is executed for-all function keys. The texts RUN, STOP, YES and NO do not appear if the keylock switch is set to " I " .

After function key 2 is pressed, the menu appears as follows:



After function key 3 is pressed, the menu appears as follows:

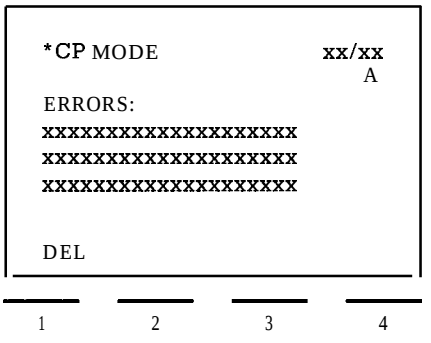


The message "CAUTION! PLANT!" appears. The relevant function key must then be pressed again.

The function keys and their meaning:

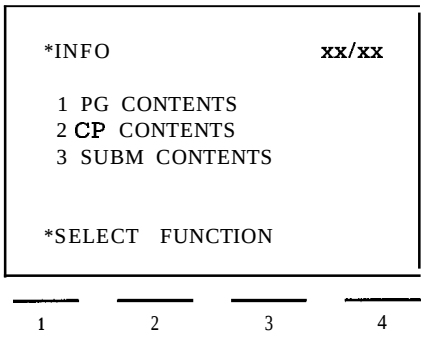
1 : The CP 530 is set to the Stop status. STOP appears in the "CP status" field.
2 : The CP 530 is set to the Run status. RUN appears in the "CP status" field.
3,4 : With these keys the CP 530 can be set to "PG PRIORITY YES" mode or "PG PRIORITY NO" mode.
BRK : Return to INIT STATE form

In the "Error" field the number of errors accumulated in the CP 530 is displayed. The plaintexts of the errors can be paged using the cursor control keys ⇓⇑ The following form type appears:



The plaintext of the error appears in the output fields, (see Subs. 4.2) The displayed error can be acknowledged with function key 1.

By pressing key 7 in the I NIT STATE form the following form appears:



The function keys and their meaning:

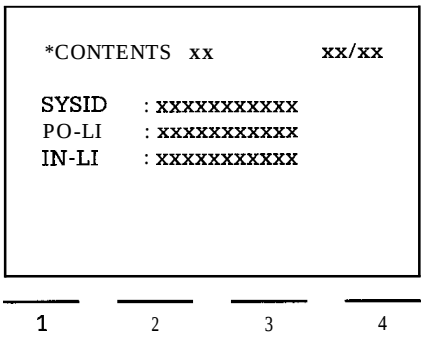
1 : Information about user data in the programmer
2 : Information about user data in the CP 530
3 : Information about user data in the EPROM/EEPROM submodule

Function keys 1 to 3 effect a change to the CONTENTS form.

BRK : Return to the INIT STATE form

3.10.1 CONTENTS Form

PG, CP or SM appears in the first output field of the top line, depending on which information function was selected in the INFO form. The output fields in lines 3 to 5 display which user data are available or not.



The function keys and their meaning:

BRK : Return to the INFO form.

3.11 Test and Start-up

3.11.1 TEST Form

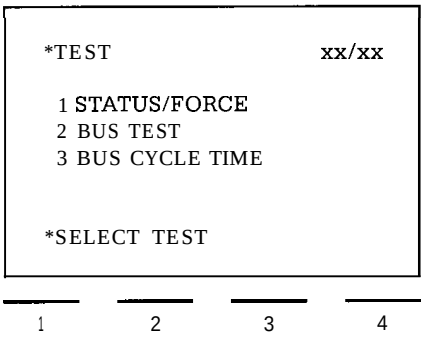
The function keys and their meaning:

1 : Change to the STATUS-FORCE form (display of mailbox contents)

2 : Change to the BUS TEST form. Data traffic on the SINEC LI LAN is stopped and can be continued step by step in keeping with the polling list.

3 : Change to the BUS CYCLE TIME form

BRK : Return to the I NIT STATE form



3.11.2 STATUS-FORCE Form

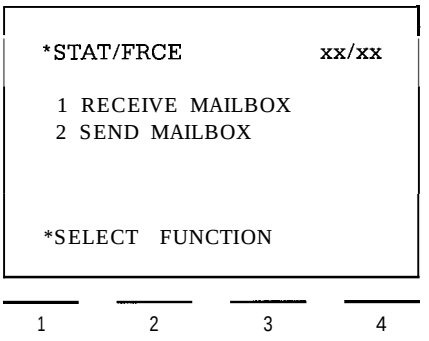
The function keys and their meaning:

1 : Display of the status of the receive mailbox

2 : Display of the status of the send mailbox

BRK : Return to the TEST form

The function keys also effect a change to the STAT/FRCE-1 form.



3.11.2.1 STAT/FRCE 1 Form

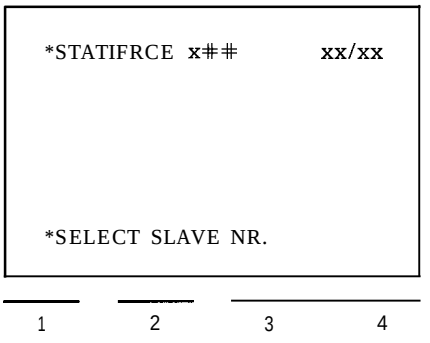
R or S appears in the first output field of the top line, depending on whether a receive mailbox or a send mailbox was selected in the STAT/FRCE form. The cursor is positioned to the entry field in the top line. A slave number (1 to 30) must be entered.

If the programmer is plugged direct into the CP 530 and the CP 530 is set as a slave, this form is skipped and the slave number is entered automatically.

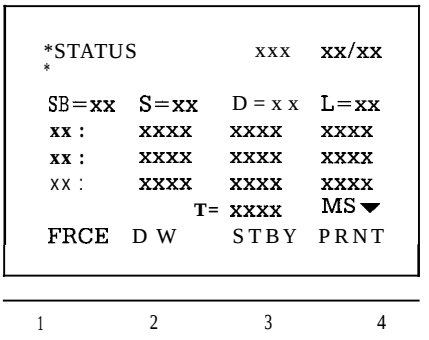
The function keys and their meaning:

ENTER : Enter the slave number and change to the STATUS form.

BRK : Return to the STAT/FRCE form.



3.11.2.2 STATUS Form



The data in the output fields of the top line are accepted by the STAT/FRCE form.

In addition, the following are displayed:

SB : Slave status byte in hexadecimal form. For the contents of the status byte see Subs. 3.11.2.4
S : Source of displayed data (O = CP 530)
D : Destination of displayed data (O = CP 530)
L : Length of the mailbox in bytes
T : Time in milliseconds between referencing a slave and receiving its mailbox contents.

If the status is running an asterisk blinks in the second display line. If the asterisk is no longer blinking, there is no longer any data traffic between the CP 530 and the programmer. The part of the form between the top line and the menu displays the mailbox contents in hexadecimal form. Words are separted by blanks. The contents are updated continuously. The cursor control keys ⇓⇑ can be used to page through longer mailboxes.

The function keys and their meaning:

1 Forcing the mailbox. The mailbox contents are fixed, i. e. the display is no longer updated (the asterisk in the display line no longer blinks). The contents of the mailbox can be changed.
Change to the FORCE form

N. B.: The text FRCE above digit 1 on the display only appears in the send mailbox when the keylock switch is at "II".

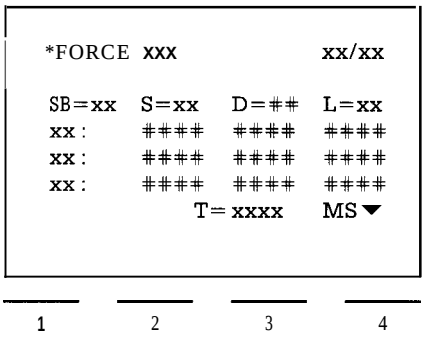
2 The cursor jumps to the data word number in the first line of the mailbox display. The menu line is deleted and a data word number can be entered, If the ENTER key is pressed, the data word number is accepted and the mailbox display rolls to the corresponding data word. With the BRK key, the previous status can be displayed again.

3 : With this function, the slave status byte, which is displayed hexadecimally in every mailbox, can be more closely examined.
Change to the STATUS BYTE form.

4 Printout of mailbox contents on the printer. The display is frozen during printing.

BRK : Return to the STAT/FRCE form

3.11.2.3 FORCE Form



The cursor jumps to the left-hand data word in the first line of the mailbox. Any data can then be selected using the cursor control keys and changes can thus be made in the mailbox contents.

The function keys and their meaning:

DEL : The data word at the cursor position is deleted. Subsequent data word are moved one place to the left and the length of the mailbox is reduced by one.

INS : A data word is inserted at the cursor position. Subsequent data words are renumbered one number higher and the length of the mailbox is increased by one.

ENTER : The first time this key is pressed, "CAUTION! PLANT!" appears. The second time it is pressed, forcing is executed, i. e. the modified send mailbox is sent to the destination node. Data traffic between the CP 530 and the programmer continues in the STATUS function.

BRK : Return to the FORCE form without executing forcing.

3.11.2.4 STATUS BYTE Form

Send mailbox

```
*STATUS          xxx   xx/xx
*
STOPPC                 :####
RUN PC                 :####
SEND ENABLE SL         :####
INTERRUPT              :####
                              ▼
FRCE                   PRNT
```

1 2 3 4

Receive mailbox

```
*STATUS          xxx   xx/xx
*
PC STOPPED             :####
PC RUNNING             :####
CROSS COMM.            :####
INTERRUPT              :####
                              ▼
                       PRNT
```

1 2 3 4

```
*STATUS XXX              xx/xx
*                          A
PG ROSTS BUS STOP
BUS RUNNING            :####
ONE SL FAILED          :####
DP WITH ERROR  :  xxxx
FRCE                   PRNT
```

1 2 3 4

```
'STATUS          xxx   xx/xx
*                          A
PG ROSTS BUS STOP
BUS RUNNING            :####
ONE SL FAILED          :####
DP WITH ERROR          :xxxx
                       PRNT
```

1 2 3 4

The display of the status byte is updated continuously. A change in the slave status is therefore immediately displayed.

The function keys and there meaning:

1 : Forcing of the status byte. The status byte display is frozen. Control is passed to the FORCE STATUS BYTE form. FORCE appears only in the send mailbox of a master.

4 : Print out of mailbox contents and status byte on the printer.

3.11.2.5 FORCE STATUS BYTE Form

The status byte can now be changed for the specified slave. By changing the bit statuses (in the entry fields) and entering with the ENTER key, the status byte is accepted by the send mailbox and the FORCE form is selected.

The function keys and their meaning:

1 : YES is entered in the entry field at the cursor position

2 : NO is entered in the entry field at the cursor position

BRK Return to the STATUS BYTE form

ENTER : Enter the status byte in the send mailbox and branch to FORCE of the send mailbox.

Note: If contradictory statuses are entered in the status byte, e. g.
STOP PC : YES and
RUN PC : YES
then the message: ILLEGAL INPUT appears. The status byte can be accepted only when the contradictory inputs have been corrected.

Send mailbox

```
*FORCE           xxx   xx/xx
*
STOP PC                :xxxx
RUN PC                 :xxxx
SEND ENABLE SL         :xxxx
INTERRUPT              :xxxx
                              ▼
YES     NO             PRNT
```

1 2 3 4

```
*CONTROL         xxx   xx/xx
*                          A
PG Bit                 :xxxx
BUS RUNNING            :xxxx
ONE SL FAILED          :xxxx
DP WITH ERROR          :xxxx

YES     NO             PRNT
```

1 2 3 4

3.11.3 Bus Test

The BUS TEST function of the CP 530 enables SINEC LI to be tested step by step, while the data exchanged between the CP 530 and the nodes can be examined at the same time in the form of "mailboxes" and can also be changed. On changing to the BUS TEST, the bus is stopped before the start of a new bus cycle (the CP 530 goes from RUN to STOP).

Sequence without interrupt (direct traffic):

In the first **substep** the operator can examine and also change the send mailbox of the CP 530 at the first slave of the polling list. This substep is terminated by pressing the ENTER key and causes the mailbox contents to be sent.

As a result of this, the following is displayed:

The receive or cross communications mailbox of the referenced slave. The mailbox can be paged through with the cursor control keys ⇓⇑ but it cannot be changed.

In the second **substep,** the next step is initiated with key 1 (CONT). The SINEC LI LAN is released until the next slave is polled (the green LED on the CP 530 flashes momentarily).

As a result of this, the following is displayed:

The send mailbox of the CP 530 for the next slave in the polling list. The user is back at the start of the first substep for the second slave in the polling list. When the polling list has been processed, the cycle starts again. If the bus test is interrupted, the CP 530 remains in the Stop status and can be switched back into RUN with the programmer or the mode selector.

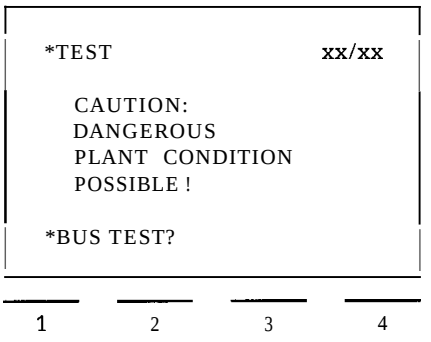
Sequence with interrupt:

In this case, an operator input step is inserted after the first substep:
The send mailbox of the CP 530 at the slave responsible for the interrupt is displayed. The mailbox can be paged through with the cursor control keys, but the contents cannot be changed. Function key 1 calls up the display of the receive or cross communications mailbox of the slave responsible for the interrupt. Function key ENTER leads to the second substep in the processing of the polling list slave, If there is another interrupt, the relevant node must wait until at least one normal step has been executed in the polling list.

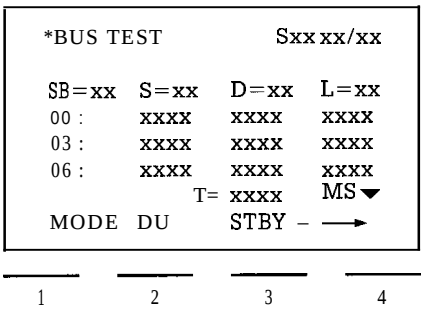
In the BUS TEST forms only those send mailboxes of the CP 530 which are at the current slave in the polling list can be changed. Since data traffic on the SINECL1 LAN takes place step by step during the bus test, a build up of interrupts can hinder normal processing of the polling list (this can be avoided by disabling the interrupt from the programmer).

When key 2 is pressed, the following message appears:

The question "BUSTEST" can be positively acknowledged with the ENTER key and the PG 615 then takes the CP 530 into the BUS TEST. Data traffic on the SINEC LI LAN is stopped at the beginning of the polling list. Change to the BUS TEST SEND MAI LBOX form.
The TEST form appears again when the BRK key is pressed.



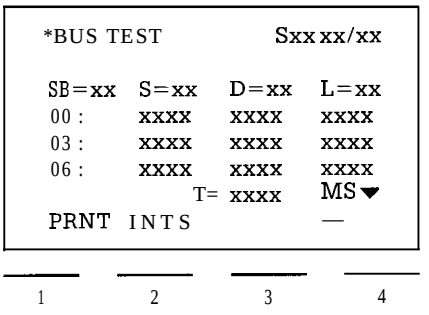
3.11.3.1 BUS TEST SEND MAILBOX Form



The send mailbox of the CP 530 at the current slave on the polling list is displayed. Changes can be entered in this mailbox.

The function keys and their meaning:

Characters above key 4 ⟶
1 : Modify the mailbox: Change to the MODIFY form
2 : Function as in STATUS (see Subs. 3.11.2.2)
3 : Function as in STATUS (see Subs. 3,11.2.2)
4 : Switch menu line

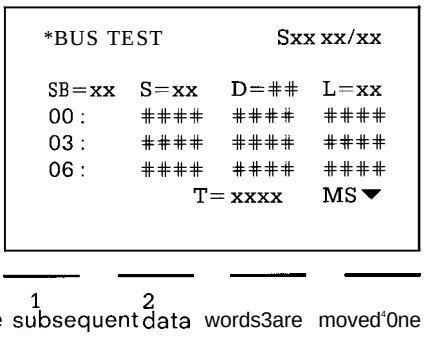


Characters above key 4 —
1 : Function as in STATUS key 4 (see Subs. 3.11.2.2)
2 : Disable interrupt/enable interrupt
4 : Switch menu line

ENTER : Send mailbox and change to BUS TEST MAILBOX form or BUS TEST INTERRUPT CYCLE if an interrupt occurs.

BRK : Return to TEST form. The SINEC LI LAN remains in the Stop status.

3.11.3.2 MODIFY MAILBOX Form



The cursor jumps to the left-hand data word in the first line of the mailbox display. The mailbox can be paged through with the cursor control keys; the data words can be modified.

The function keys and their meaning:

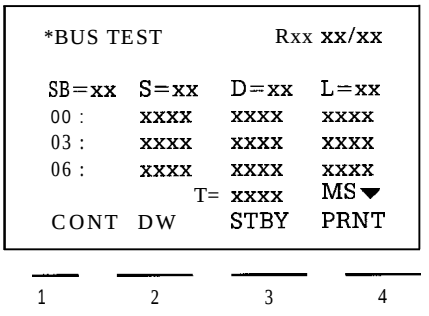
DEL : The data word at the cursor position is removed. The subsequent data words are moved one number left and the mailbox is shortened by one.

INS : Space for a data word is inserted at the cursor position. The subsequent data words are moved one number right and the mailbox is lengthened by one.

ENTER : Send the mailbox and change to the second segment (BUS TEST RECEIVE MAILBOX form)

BRK : Return to the BUS TEST SEND MAILBOX form (the modified data are lost).

3.11.3.3 BUS TEST RECEIVE MAILBOX Form

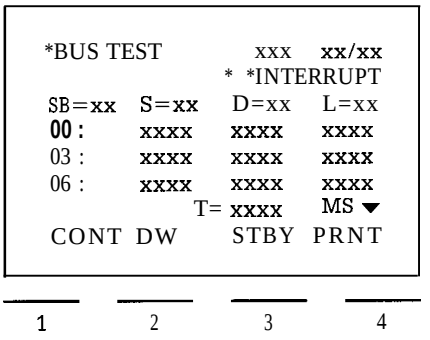


The receive or cross communications mailbox of the current slave is displayed. In the case of a cross communications mailbox, the message "CROSS COMMUNICATIONS" appears in the second line. The receive mailbox can be paged through with the cursor control keys ⇓⇑, but no changes can be made.

The function keys and their meaning:

1 : Change to the send mailbox display of the next slave
2 : see BUS TEST SEND MAILBOX form
3 : see BUS TEST SEND MAILBOX form
4 : see BUS TEST SEND MAILBOX form
ENTER : Polling of the next slave. Return to the BUS TEST MAILBOX form

3.11.3.4 BUS TEST INTERRUPT CYCLE Form



This form appears after the first substep on the recurrence of an interrupt.

The function keys and their meaning:

1 : Change to the receive mailbox display or the cross communications mailbox display of the slave responsible for the interrupt.

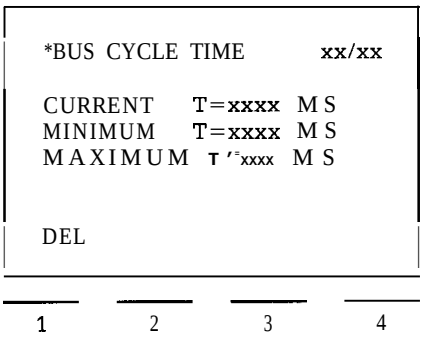
2 : see BUS TEST SEND MAILBOX form

3 : see BUS TEST SEND MAILBOX form

4 : see BUS TEST SEND MAILBOX form

ENTER : Bus traffic is continued. If no further interrupt occur, the BUS TEST RECEIVE MAILBOX form appears in which the receive mailbox or cross communications mailbox of the current slave on the polling list is displayed.

3.11.4 Bus Cycle Time



The current cycle time is displayed (CURRENT). The values MINIMUM and MAXIMUM indicate the extreme values of cycle time like non-return pointers.

The function keys and their meaning:

1 : The minimum and maximum values are set to zero. Capture of extreme values begins again.

BRK : Return to the TEST form

4.1 Errors displayed in the seventh Display Line

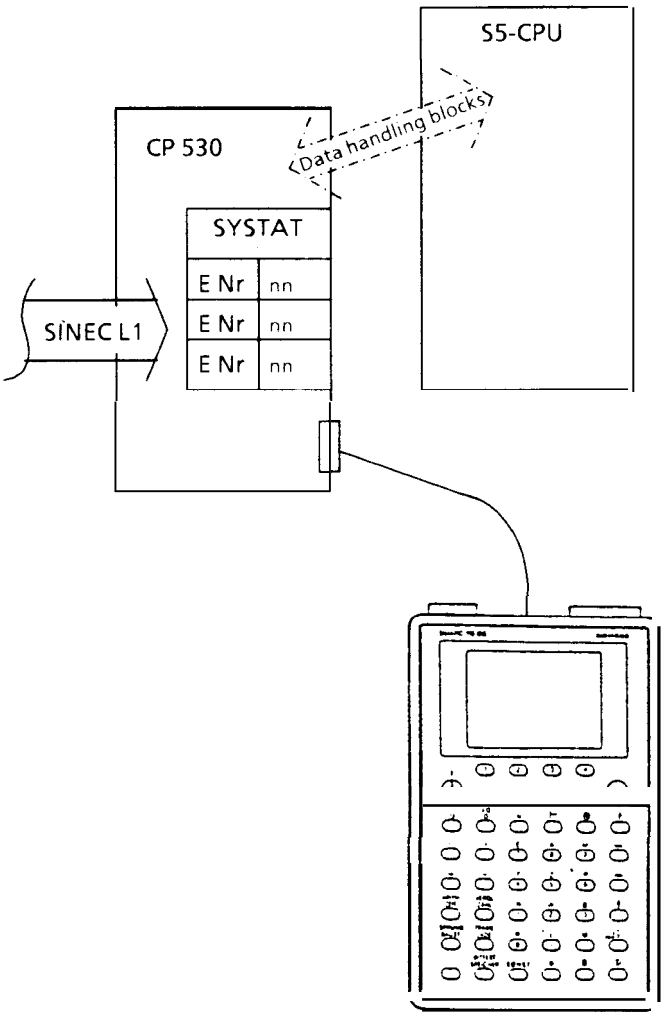
Error	Meaning and remedy
CPABORT	The programmer has recognized a BREAK. Check that the programmer is correctly connected to the CP 530. Repeat function, if necessary.
AREA NOT EMPTY	The memory area into which the list in the submodule is to be programmed is not empty. Delete area.
MODE SELECTOR	CP should be set to RUN even though mode selector is at STOP. Put switch to RUN.
NO REPLY FROM CP	The CP 530 has not send an expected character within a certain time. Switch power supply off and on again.
MAI LBOX FULL	Send mailbox full. Do not enteranv more data.
WRONG MODE	Function not possible in this mode. Change mode.
WRONG SUBMODULE	The submodule plugged in does not match ordering code entered. Enter correct ordering code.
PROGRAMMING ERROR	An error has occurred in programming. Delete module and repeat function.
WRONG FIRMWARE	The COM 530 submodule is plugged into the wrong unit
WRONG CP SUBMODULE	There is an EPROM submodule plugged into the CP which cannot be programmed by the CP. Connect an EEPROM module.
ERROR IN EXTERNAL RAM	The RAM chio in the oroarammer is defective. The programmer must be repaired.
ERROR INTERNAL RAM	The internal RAM of the processors defective. The processor must be replaced.
ERROR PROG.EEPROM	An error has occurred when programming the EEPROMsubmodule. Replace EEPROMsubmodule.
NO SUBMODULE PLUGGED IN	There is no user submodule plugged into the programmer. Plug submodule in.
NO LIST AVAILABLE	There is no list available in the programmer, the CP 530 or the submodule.
LIST FULL	The polling list already contains 64 slave numbers ortheinterrupt list already 30.
LIST NOT AVAILABLE	The desired list is not in the CP 530, the programmer or the user submodule. If necessary, enter list.
LISTALREADYAVAILABLE	The list is already available in the user submodule. Delete list if necessary.
LIST NO DELETED	List in user submodule not deleted. Delete list if necessary.
ON-LINE ONLY PERMISSIBLE	The desired function is only permissible ON-LINE. If necessary, set programmer to ON-LINE status.
ONLY IN POSITION II	Keylock switch in positionI, and all input functions are therefore disabled. Set switch to positionII.
ONLY PERMISSIBLE IN CASE OF MASTER	This function is only allowed if the CP 530 has been initialized as a master. If necessary, initialize the CP as a master.
EEPROM ONLY PERMISSIBLE	There is an EPROM submodule plugged in and its data is supposed to be erased. If necessary, plug an EEPROMsubmodule in since an EPROM can only be erased with UV light.
INTERFACE NOT READY	The CP 530 is in a mode which does not allow this function, or there is an EPROM submodule plugged into the CP 530 and an attempt is being made to transfer a list to the CP 530. Change mode or remove EPROM.
SLAVE FAILURE	Slave does not report to CP 530. Check slave.
SLAVE NUMBER ASSIGNED MORE THAN ONCE	A slave number appears more than once in the interrupt list. Delete the relevant slave number until it occurs once in the interrupt list.
SLAVE NOT IN POLLING LIST	The send or receive mailbox of the slave cannot be observed in Status mode because the slave is not in the polling list. If necessary, enter slave in polling list.
SLAVE ALREADY FORCED	The slave is already being forced by the CPU, and therefore cannot be forced by the programmer. Otherwise, programmer priority must be set.
INVALID LIST	There is an invalid list in the user submodule. Delete list and re-enter it.
ILLEGAL INPUT	This input exceeds its permitted parameter limit. Observe the parameter limits.
ILLEGAL KEY	This key is not permissible in this function. Use permitted keys only.
RECEIVE ERROR, PROGRAMMER	Programmer has received character it cannot interpret. Repeat function.
RECEIVE ERROR, COMMUNICATIONS PROCESSOR	The CP 530 has received a character it cannot interpret. Repeat function.
PRINTER NOT READY	Printer not connected or not switched on.
DELETE LISTS	A master is to be re-initialized as a slave by the SYSID and the CP already contains the polling list and/or interrupt list.

4.2 Errors displayed in CP MODE

Errors in connection with data traffic on SINEC LI and between the CP and the programmable controller are entered in the "Error" field and displayed in the "Mode" form.

The error numbers make it possible to classify the errors roughly and the appended digits provide more details on the error, depending on the error number ("Error classes" table).

Detailing a group error. In Class 1, these details are the results of a self-test and should be reported to the works in the event of a replacement or repair, e. g. Error 10: "Hardware error 7".



Error classes:

Breakdown into error classes:
- **Compressed representation of error types possible**
- **The error class provides the user with information on how to eliminate the error.**

Besides the errors described below, there are also irreparable errors that cannot be reported via SYSTAT, e. g. because the equipment for their transmission is faulted or does not exist.

Error classes

Class	Error No.	Description	Operator response	Person involved
1	10–29	Hardware configuration of module suspect	Check hardware, and replace or repair it	1. Operator 2. Service engineer
2	30–49	Operator error	Mode selector Check program submodule	Operator
3	50–69	Initializing/programming error	Diagnostics with programmer necessary S5 software modification	Programmer
4	70–90	Status messages	Note error	Operator

4.2 Errors displayed in CP MODE

Error class	Error No.	Error supplement	
	decimal	representation	
I	10	xx	ERROR 10: HARDWARE ERROR XX
	11	xx	ERROR 11: INTERNAL ERROR MESSAGE NO. XX
II	30	0	ERROR 30: WAIT FOR SYNCH RON
	31	0	ERROR 31: WRONG CP SUBMODULE
	32	0	ERROR 32: PROGRAMMER ACTIVE PRIORITY
	33	0	ERROR 33: CP IN STOP STATUS; SLAVE CANNOT SEND
	34	0	ERROR 34: CP NOT IN STOP STATUS
	35	0	ERROR 35: CP RUN NOT POSSIBLE
III	50	0	ERROR 50: SYSID ERROR
	51	0	ERROR 51: POLLING LIST ERROR
	52	0	ERROR 52: INTERRUPT LIST ERROR
	53	0	ERROR 53: UNSPECIFIED TASK
	54	0	ERROR 54: UNKNOWN TASK
	55	0	ERROR 55: UNSPECIFIED SEND
	56	0	ERROR 56: UNSPECIFIED RECEIVE
	57	0	ERROR 57: A-NO.: MASTER ONLY
	58	0	ERROR 58: A-NO.: SLAVE ONLY
	59	0	ERROR 59: A-NO.: JOB TOO LONG
	60	0	ERROR 60: SLAVE NOT IN POLLING LIST
	61	0	ERROR 61: A-NO.: THE JOB NUMBER USES AN UNDEFINED LIST
	62	0	ERROR 62: A-NO.: JOB NUMBER SBR NOT ALLOWED
IV	70	0	ERROR 70: BUS ERROR
	71	0	ERROR 71: CONNECTION TO SLAVE HAS AN INTERFERENCE
	72	0	ERROR 72: WRONG SLAVE REPORTING
	73	0	ERROR 73: SLAVE EXTRACTED

Order No.: GWA 4NEB 8110521 -02c
Printed in the Federal Republic of Germany
AG 4871.5 Fl 30 En

SIEMENS

SIMATIC S5 / SINEC L1

CP 530 Communications Processor COM 530 with S5-DOS

(PG *635,* **PG** *675,* **PG** *685,* **PG** *695)*

Operating Instructions — Order No.: GWA 4NEB 811 0730-02a

Contents

1. Introduction

The SINEC L1 Local Area Network permits communication between up to **31** SIMATIC **S5 programmable controllers of the U range in master/slave mode.**
The COM **530 software package for all programmers on which** S5-DOS **can execute** (PG **635, PG 675, PG 685, PG 695)** is used to assign parameters to the SINEC LI CP 530 communications processor (generation of SYSID as well as polling list and interrupt list), to select the LAN operating mode, to control data traffic and to document and archive the bus parameters on mini diskettes.

All operator inputs are made via interactive screen forms (entry fields) and function keys (or softkeys).
The significance of the function keys can be seen in the menu displayed in the bottom three lines of the screen

In addition, the familiar functions of the keys on the PG 675 programmer, such as acknowledgement (⏎), Abort (✱), etc. have been retained.

See also the S5-DOS Operating Instructions in your programmer manual.

The S5-DOS version of the COM 530 is available in two diskette sizes.

3½ in. for the PG 635
5¼in. for the PG 675, PG 685 and PG 695

Starting COM 530:

PG 635 **and PG** 675
1. Insert the PCP/M 86 diskette in drive A and switch the programmer on.
2. Insert the ZEFU diskette (PG 675: ZEFU 3/3).
3. Remove the PCP/M 86 diskette and insert the COM 530 diskette.
4. Start S5-DOS (by typing in S5).
5. The programmer displays a menu listing all the programs available on diskette. Select COM 530
6. Insert a formatted diskette in one of the drives.

PG 685 and PG 695
1. Copy the COM 530 diskette onto the hard disk.
   Remove the diskette.
2. Start S5-DOS (by typing in S5)
3. The programmer displays a menu listing all the programms available on diskette. Select COM 530.
4. Insert a formatted diskette in one of the drives. The data may also be stored on the hard disk (drive C).

The following are described:

- Layout of the interactive screen forms
- Meaning of the entry and output fields in the interactive forms
- Meaning and effect of the softkeys and function keys.

Output fields in the interactive forms are marked xxxxx. In these fields COM 530 shows current statuses or data entered with the previous operations.

Entry fields in the interactive forms are marked ##### and. appear on the programmer screen in inverse video. Entries can be made in these fields by means of the alphanumeric keyboard and, in some cases, with the function keys (HELP function).

Error messages of COM 530 always appear in the last line on the screen before the menu

Bus parameter: All parameters necessary to operate the LAN and generated by the user with the aid of the COM 530 software (polling list, interrupt list, SYSID identifiers, see SINEC LI Operating and Programming Instructions).

The bus parameters are generated when programming (see Section 3.3) and stored by means of function keys.
Where they are stored depends on the COM 530 mode set, i.e. ONLINE or OFFLINE.

In ONLINE mode, the user data are stored directly in the CP 530 communications processor. In OFFLINE mode, this is done on a minidiskette, on a hard disk or on an EPROM/EEPROM submodule.

When changing over to programming bus parameters these are read from the CP 530 communications processor or minidiskette or hard disk – that is, if user data are already stored there – and displayed on the respective forms where they can be modified and re-stored.

Program name:A user-selectable name for identifying all user data assigned to a CP 530 interface and stored on diskette

The Significance of the Cursor Control Keys and Function Keys

a) Cursor Control Keys

In most of the interactive forms used, the significance of the control key functions is as follows:

↑ ↓ The Cursor is positioned to the first entry field in the line above (below).

= = > < = = The cursor is positioned to the previous (next) entry field.

--> <-- The cursor is positioned to the previous (next) character within an entry field.
If the left (or right) field limit is exceed, the cursor moves to the previous (next) field.

b) Function Keys

The COM 530 software package generally permits parallel input both by means of softkeys and functions keys. The operator therefore does not have to relearn programmer operations.

In most of the interactive forms used the significance of the keys is as follows:

J Jump to next entry field

Deletion of a character at the cursor position

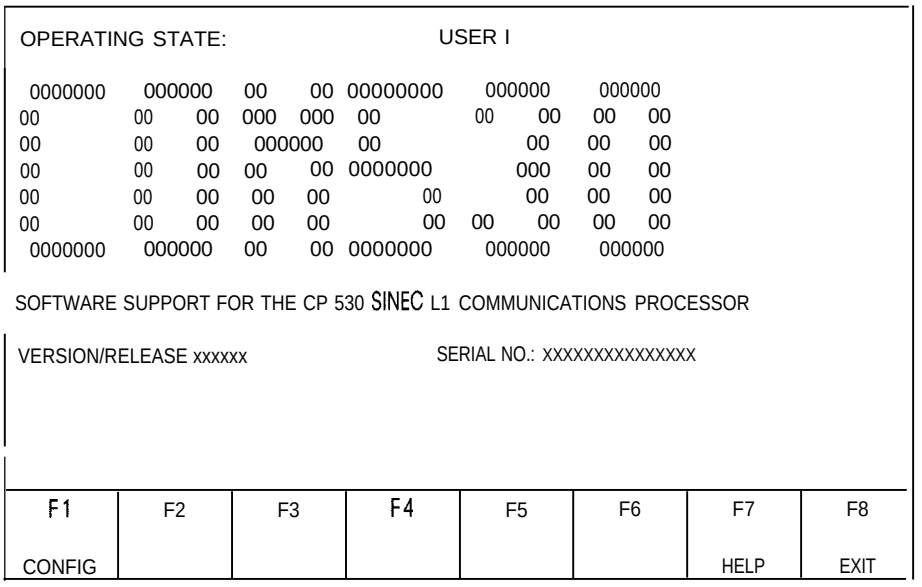
Insertion of a character at the cursor position

This key always has the same meaning as function key F8 EXIT (parallel key).
This results in the next-highest level in the form hierarchy being entered without the input data in the current screen form being evaluated.

This key means „ Store" if softkey F6 STORE has been defined in the form.
These are also parallel keys and have the same function.

HELP The HELP key has the same function as function keyF7– HELP if it has been defined in the form.

After COM 530 has been called, the following form appears:



Function keys:

F1: Selection of the configuration form.
In ONLINE mode contact is established with the CP 530. If the event message “CP does not reply” appears (after approx. 20 s), check the connecting cable.

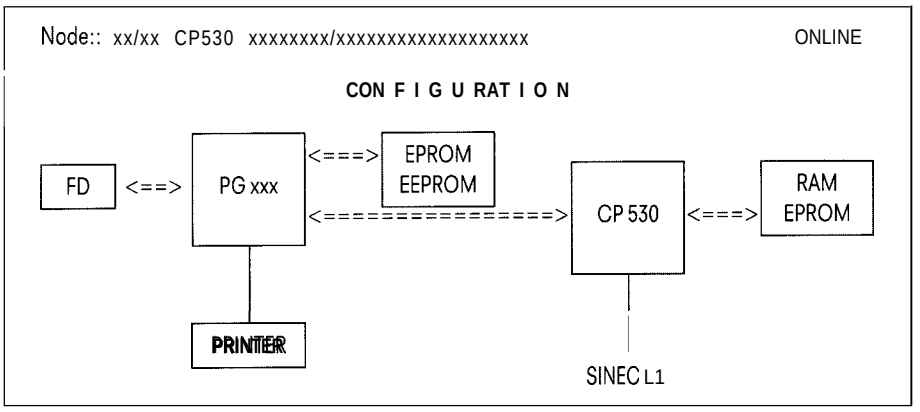
F7: Set the OFFLINE or ONLINE mode with HELP key. Default is OFFLINE.

F8: Takes you back into the S5-DOS operating system.

3.2 CONFIGURATION form

Either the CONFIGURATION (ONLINE) or CONFIGURATION (OFFLINE) form appears, depending on whether COM 530 is being **used on-line of off-line.**

(1) ONLINE programming:



CONFIGURATION (ONLINE) form

The softkey menu for the CONFIGURATION (ONLINE) form is in two parts:

Menu 1 a):

F1	F2	F3	F4	F5	F6	F7	F8
INPUT	DISPLAY	PRINT	TRANSFER	TESTAND START-UP	PAGE		EXIT

Menu 1 b):

F1	F2	F3	F4	F5	F6	F7	F8
INFO	DELETE	MODES			PAGE		EXIT

Meaning of the output fields in the CONFIGURATION (ONLINE) form:

The “Nodes” field in the header display a two-part node number, which is read out of the SYSID area of the CP 530.
For the master the second part is /00; for a slave it is a number greater than O (1-30).

The next two fields show the version number and symbolic designation of the CP 530 (only if the programmer is connected directly or indirectly to the CP 530, i.e. in ONLINE mode).

The identifier of the programmer on which COM 530 is currently running is displayed in the programmer symbol.

Function keys:

Menu 1 a)

F I : Entry of user data
The following are possible:
- Programming the system parameters (SYSID),
– Generation of a polling list,
– Generation of an interrupt (priority) list.
Selects the ENTRY form.

F 2: Display of user data
If available, SYSID identifiers, the polling list and the interrupt list can be displayed.
Selects the DISPLAY form.

F 3: Listing user data
The user data generated can be output for documentation purposes on a printer connected to the programmer.
Selects the PRINT form.

F 4: Transfer of SYSID identifiers, polling and interrupt lists
This function permits the transfer of bus parameters from the CP 530 onto mini-diskette. EPROMs and EEPROMS can also be programmed with the bus parameters.
Selects the TRANSFER form.

F 5: Selects the testing of data traffic on the SINECL1 (TEST form)
Note: A drive must be specified for a user diskette or a hard disk even in ONLINE mode.

F 6: Selects Menu 1 b

F 8: Initiates return to the COM 530 basic form

Menu Ib)

F 1: Selects the “Information” mode
Information is given on whether SYSID identifiers and/or polling list and/or interrupt list are in the CP 530 or EPROM/EEPROM submodule.
In addition, information is given on the programs stored on the user diskette (program names) and their contents (INFO form).

F 2: Selects the “Delete” mode
Deletes SYSID indentifiers, polling or interrupt lists (DELETE form)

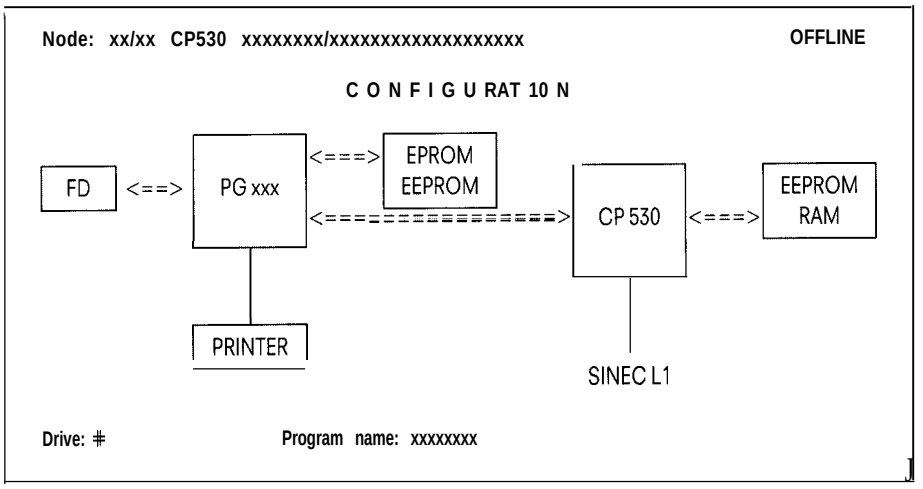
F 3: Selects mode setting (MODE form).

F 6: Selects menu Ia

F 8: Initiates return to the COM 530 basic form

3.2 CONFIGURATION Form

(2) OFFLINE programming:



CONFIGURATION (OFFLINE) form

The softkey menu for the CO NFIGURATI ON (OFFLINE) form is also in two parts:

Menu 2 a):

F1	F2	F3	F4	F5	F6	F7	F8
INPUT	DISPLAY	PRINT	TRANSFER		PAGE	HELP	EXIT

Menu 2 b):

F1	F2	F3	F4	F5	F6	F7	F8
INFO	DELETE				PAGE	HELP	EXIT

Entry and output fields:

PROGRAM NAME:
In OFFLINE mode, the user must enter a name here which he can use to identify new or already existing bus parameters (lists and SYSID). This specifies the file name under which the user data are read from, or written onto, the diskette. In addition, the drive containing the diskette or hard disk should also be specified.

The HELP function can be used to enter all program names on the diskette.

For the meaning of the output fields, see the CONFIGURATION (ONLINE) form. The “Version no. ” and “Symbolic designation” fields of the CP 530 are blank in OFFLINE mode.

The identifier of the programmer which is currently running COM 530 (e. g. PG 685) is displayed in the programmer symbol

Function keys:

Menu 2a)

F1, F2 F3: see Menu 1 a.

F4: Transfer of SYSID identifiers, polling and interrupt lists
This function permits bus parameters to be transferred from mini-diskette into the CP 530. EPROMs or EEPROMs can also be programmed with the bus parameters.
Selects the TRANSFER form.

F6: Selects menu 2 b

F7: HELP function
With the aid of the HELP function, the names of all programs (i.e. bus parameters) on the user diskette can be entered in the "PROGRAM NAME" field.

F8: Initiates return to the COM 530 basic form

Menu 2 b)

FI: Selects the "Information" mode
Information is given on whether SYSID identifiers and/or polling list and/or interrupt list are available in the EPROM or EEPROM. In addition, information is given on the programs stored on the user diskette (program names) and their contents. (INFO form)

F2: Selects the "Delete" mode
Deletion of SYSID identifiers, polling or interrupt list. (DELETE form)

F6: Selects menu 2 a

F7: HELP function
The HELP function can be used to enter the names of programs (i. e. bus parameters) on the selected drive in the "PROGRAM NAME" field.

F8: Initiates return to the COM 530 basic form

Possible error messages:

Error 2A: „ WARNING No EPROM driver on the system diskette. No EPROM calls!!!"
The attempt to program data direct into an EPROM/EEPROM or to read data out of an EPROM/EEPROM causes the program to crash ==> The "IN TERRUPTTRAP HALT" message appears.

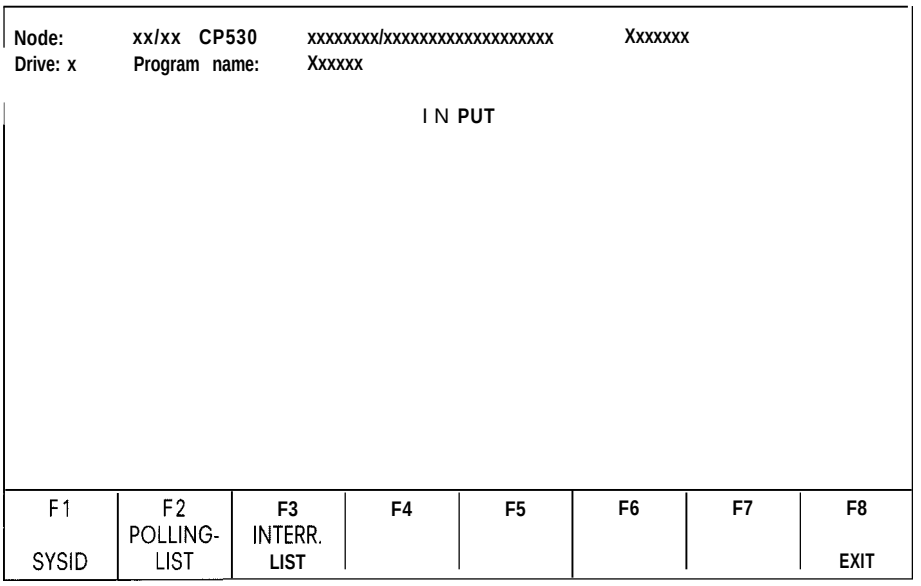
Remedy: Load the original S5-DOS system diskette and restart program.

Error 46: "Incomplete entry"

A freely selectable program name and the drive identifier must be specified in OFFLINE mode.

3.3 Entering (Programming) User Data

INPUT form



The following applies o the header in this and all subsequent forms:

The node numbers and identifiers are taken from the SYSID area of the CP 530 communications processor in ONLINE mode. If OFFLINE mode has been selected, these data are taken from the SYSID identifiers stored on the user diskette. If the SYSID identifiers stored on the user diskette under the program names specified in OFFLINE programming are not (yet) available, the fields in the header remain vacant.

The "Program name" field is only displayed when programming OFFLINE. It then contains the name specified in the CONFIGURATION (ONLINE) form.

Assignment of the function keys: Softkeys F2 and F3 only appear if the CP 530 is master in the SINEC LI network.

F1: Selects programming of the system parameters. (SYSID-ON form)

F2: Creation of the polling list (PO LL-INP form).
(Only possible when programming a master)

F3: Creation of the interrupt (priority) list (interrupt-INP form).
(Only possible when programming a master)

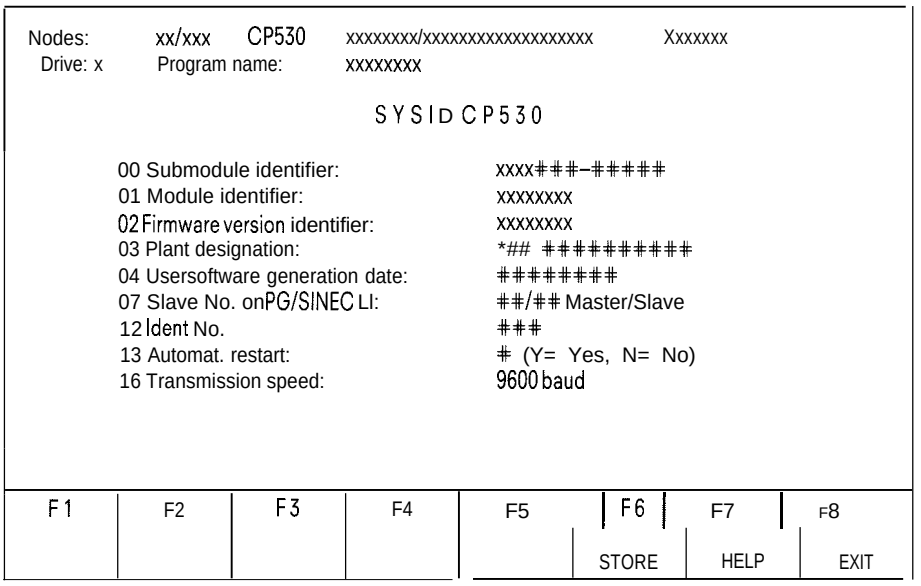
F8: Initiates return to the CONFIGURATION form.

3.3.1 Programming the System Parameters (SYSID-INP Form)

The SYSID data area is a memory area in the CP 530 used for identification purposes.

Purpose of the SYSID:

– To uniquely describe the role of an intelligent module (e. g. CP 530) in a programmable controller system
– To transfer parameters to an intelligent module, causing certain reactions,
– To give information on the firmware and software status of the module.



Description of the SYSID INP form:

(1) ONLINE programming:

When the CP 530 is started, the operating system transfers the "Submodule identifier", "Module identifier", and "Firmware version identifier" parameters to the SYS I D area reserved in the RAM. Then it transfers the other parameters to the SYS! D area from the memory submodule connected (EPROM/EEPROM).
In ONLINE programming, the SYSID identifiers of the CP 530 appear in this form and can be modified by overwriting them. In order to do so, the CP 530 must be set to STOP (Section 3,10.1).

(2) OFFLINE programming:

The form also appears completed on the screen in OFFLINE programming if a SYSID file exists on the diskette under the program name specified (see CONFIGURATION form).
Otherwise the entry fields are vacant or defaulted with possible alternatives.
The contents of the SYSID (except for the first three parameters) can be generated OFFLINE on the programmer and transferred to an EPROM/EEPROM. This means that the EPROM s/EEPROMs can be identified at any time and can program the CP 530 on start-up.

3.3 Entering (Programming) User Data

Entry fields in the SYSID-INP form:

PLANT DESIGNATION:

Name freely selectable with up to 19 alphanumeric characters e.g.: “SHEDI PLA4 PC7°

USER SOFTWARE GENERATION DATE:

The generation date of the user software can be specified in a freely selectable format (e. g.: “31. 01. 85”)

SLAVE-NO. ON PG/SINEC L1 :

Specification of the node number of the CP 530 on the PG or SINECL1 bus.
Examples: „ /10” only node number for SINEC L1
„_8/_4” **two-part node number**
„_7/__” **. only node number for PG bus**

(„_” stands for a blank; the “/” is provided by the programmer)

Range: **Slave no. on PG** bus: 1...30
Slave no. on SINECL1: 1...30

Page frame no.:

The page frame number defines the position of the dual-port RAM in the address area.

Range: 1 ... 254

AUTOMAT. COLD RESTART:

The alternatives YES (Y) and NO (N) can be entered using the HELP function. In the SYSID area, YES is stored asAand NO as blank.

Default: “N”

TRANSMISSION SPEED:

The default transmission speed is 9,600 baud and cannot be altered.

Function keys:

F6: In ONLINE mode: Storing the SYSID identifiers in the CP 530
In OFFLINE mode: Storing the SYSID identifiers on the user diskette or hard disk under the program name specified.
After storage – return to the INPUT form.

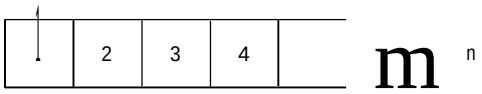
F7: Help functions by paging possible alternatives (only in the “Master/Slave identifier”, “Adressing mode” and “Automatic cold restart” fields.) Depressing the HELP function key causes a description to be displayed in the other Entry fields. This can be exited with F 8 (Exit), returning to this form.

F8: Return to the INPUT form without storing the SYSID identifiers.

3.3.2 Generation of a Polling List (PO LL-INP Form)

For minimal operation of the SINEC LI, only the polling list is required. This list contains 64 locations for entering slave numbers; the sequence thus laid down defines the order in which the slaves are to be addressed in LAN operation. The simplest case is the natural sequence; if all 64 places are used and certain slaves are specified several times, a certain priority can be achieved.

Simple case:



Assigning priority to slave 3:

1	2	3	4	5	3	6	7	3	...	...	3	n

POLL-INP form:

When entering a polling list for the first time, the following form appears:

ode: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxxx Xxxxxxx
rive: x Program name: Xxxxxxxx

POLLING LIST

=> ## ==>

F1	F2	F3	F4	F5	F6	F7	FE
DELETE	INSERT				STORE		EXIT

The entry field between the arrows symbolizes the first location in the polling list. The number of the first node in the list can be entered here. The first double arrow signifies the beginning of the list, the final double arrow the end of the list.

The F1, F2 and F6 function keys have no significance at this time as the list does not contain any nodes yet.

3.3 Entering (Programming) User Data

After a node number (e. g. 10) has been typed into the entry field, the form changes as follows:

```
Node:      xx/xx   CP530   xxxxxxxx/xxxxxxxxxxxxxxxxxx   xxxxxxx
Drive: x   Program name:   xxxxxxxx

                    POLLING LIST

  =>10 --> ## ==>
```

F1	F2	F3	F4	F5	F6	F7	F8
DELETE	INSERT				STORE		EXIT

It can be seen that, after a node has been entered in the list, a new entry field is offered for entering a further node and thus building up the list.
The slave numbers (1 to 30) are keyed into the entry fields in the sequence in which they are to be addressed. The same slave can appear several times, thus receiving higher priority than other slaves.

The polling list can have up to 64 entries.

After a certain number of slave numbers have been entered (e. g. 20) the form looks like this:

```
Node:      xx/xx   CP530   xxxxxxxx/xxxxxxxxxxxxxxxxxx   xxxxxxx
Drive: x   Program name:   xxxxxxxx

                    POLLING LIST

  ==> 10 --> 11 --> 12 --> 1 --> 2 --> 3 --> 10-->4 -->
  --> 5 --> 10 --> 6 --> 7 --> 8 --> 10 --> 9 --> 2 -->
  --> 13 --> 14 --> 10 --> 2 --> ## ==>
```

F1	F2	F3	F4	F5	F6	F7	F8
DELETE	INSERT				STORE		EXIT

The slave numbers can be changed once the respective field has been selected using the cursor control keys.

The function keys now have the following assignments:

F1: Operation of this key deletes the slave entry in the field on which the cursor is presently positioned (current entry field). All slaves following this entry field are than automatically moved one place forward. If the last field is deleted, the cursor automatically jumps to the first field.

F2: Operation of this key releases the current field, i. e. all slave numbers following this are moved one position to the right.
The key has no effect if the cursor is in the last (free) entry field or if 64 entries have already been made.

F6: Storing the polling list in the CP 530 (ONLINE mode) or on the user diskette or hard disk (OFFLINE mode) and return to the INPUT form.

F8: Return to the INPUT form without storing the polling list.

Possible error messages:

Error 01: "illegal input"!
– At least one slave number must be entered in the list,
– Possible slave numbers: 1 to 30.

Error 26: "Blank fields illegal"!
The polling list must not have blank fields. The cursor is in the first blank field that has been found.
Remedy: – Delete the field or
– Enter slave number in the respective field.

3.3.3 Generation of an Interrupt List (INTERRUPT-INPUT Form)

The interrupt list is only required if slaves can interrupt bus traffic with an interrupt request. The bus master then searches for the cause of interrupt in the sequence specified in the interrupt list.

Generation and display of the interrupt list is the same as for the polling list.

The following form therefore appears when an interrupt list is generated for the first time:

Node: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxxx Xxxxxxx
Drive: x Program name: Xxxxxxxx

INTERRUPT LIST

==> 10 —> ##

F1	F2	F3	F4	F5	F6	F7	F8
DELETE	INSERT				STORE		EXIT

Further operations and functions of the softkeys can be seen from the despriction of polling list generation (Section 3.3.2).

The numbers of the slaves (1 to 30) must be keyed into the entry fields in their order of priority in the event of a bus interrupt, The slave entered in the first field has the highest priority, the second the second-highest priority, etc.

Note: Each slave number may only be specified once!

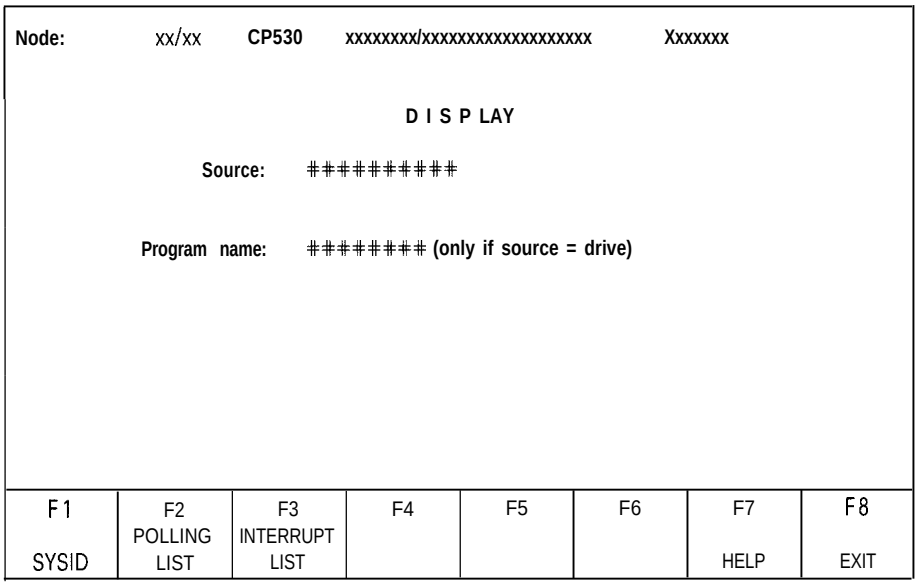
Possible error messages:

Error 27: „No double slave numbers in interrupt list!"
Each slave may only appear once in an interrupt list. The cursor is in the field in which the slave number appears for the second time.
Remedy: – Delete the respective field or
– Enter another slave number in the respective field.

See the POLLING-INP form for other error messages.

3.4 Output (Display) of the User Data

DISPLAY form



Entry fields in the DISPLAY form:

SOURCE:
The source can be specified here without changing the mode. An EPROM (EEPROM) can also be specified as source!

Possibilities:

Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP530, EPROM, EEPROM

* Depending on drives defined, e. g. A, B, C

These alternatives can be entered with the HELP function.

Default: ONLINE mode CP 530
OFFLINE mode FD

PROGRAM NAME:

If a diskette (FD) is specified as source, a program name must be entered under which the user data have been stored on the diskette.
The HELP function can be used to enter all program names stored on the user diskette.

Function keys: (Softkeys F 2 and F 3 only appear if the CP 530 is master!)

F I: Causes display of the system parameters (SYSID-DISP form)

F 2: Displays polling list (PO LL-DISP form)
(Only for display of master data)

F 3: Display of interrupt (priority) list (I NTERRUPT-DISP form)
(Only for display of master data)

F 7: HELP function for entering source and programm name
(if source = floppy disk)

F 8: Return to CONFIGURATION FORM

Possible error messages:

Error OI: "illegal Input!"
Source must be specified. The following possibilities exist:

Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP 530, EPROM, EEPROM

* Depending on drives defined, e. g. A, B, C

Error 31: "No polling list!"

Error 32: "No interrupt list!"

Error 33: "No SYSID identifier!"

One of these three error messages appears if the respective data type does not exist in the CP 530, the EPROM or in the programs specified on the user diskette.

Error 46: "Incomplete input"
A program name must be specified if the diskette (FD) is the source.

3.4.1 SYSID-DISP form

Node: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxxx Xxxxxxx
Drive: x Program name: xxxxxxxx

SYSID CP 530

Submodule identifier: Xxxxxxxxxxxx
Module identifier: xxxxxxxx
Firmware version identifier: xxxxxxxx
Plant designation: Xxxxxxxxxxxxxxxxxx
User software generation date: xxxxxxxx
Slave No. on PG/SINEC LI: xx / xx Master/Slave
Page frame No.: xxx
Automatic cold restart: x (Y= Yes, N = No)
Transmission speed: 9600baud

F1	F2	F3	F4	F5	F6	F7	F8
INPUT							EXIT

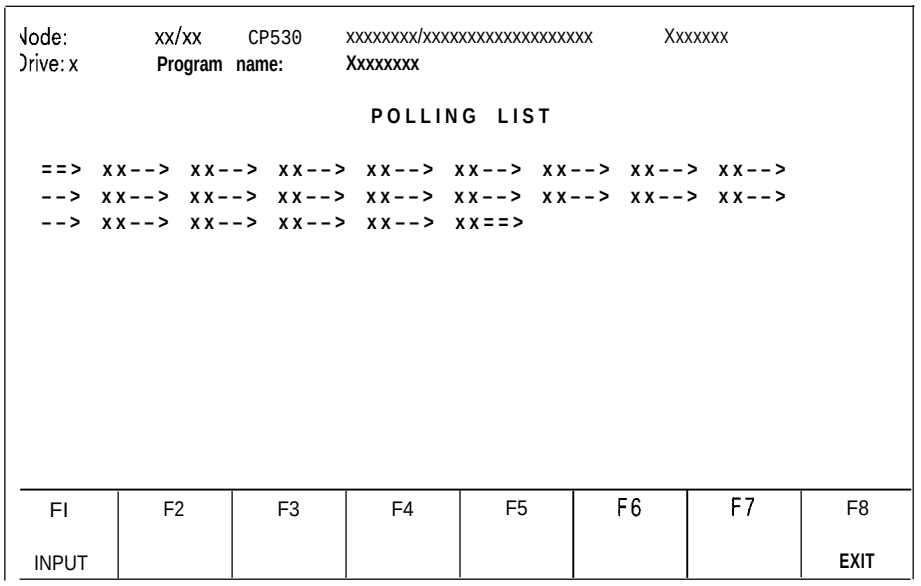
The significance of the data in the output fields can be seen from the SYSID-DISP form.
In output field 5 (ONLINE or OFFLINE) EPROM is specified as source if an EPROM is used.

Function keys: (Softkey F 1 remains unassigned if display is from an EPROM!)

F1: This key initiates display of the form for SYSID identifier input (see Section 3.3.1)

F8: causes return to the DISPLAY form

3.4.2 PO LL-DISP Form



Inoutput field 50ftheheader (ONLINEorOFFLINE) “EPROM’’a ppearsassource ifan EPROM isused.

The polling list stored is displayed as generated (see Section 3.3.2) with the following differences:

- There are no entry fields.
- There is no vacant field at the end of the list.

The slave numbers appear in the output fields as generated.

Assignment of the function keys: (Softkeys does not appear if display is from an EPROM or EEPROM!)

F1: This key permits entries to be made in the polling list (see Section 3.3.2) for modification, etc.

F8: Causes return to the DISPLAY form

3.4.3 INTERRUPT-DISP Form

```
Node:      xx/xx   CP530   Xxxxxxxtixxxxxxxxxxxxxxxxxxx   Xxxxxxx
Drive:x    Program name:   Xxxxxxxx

                    INTERRUPT LIST

  ==> xx --> xx --> xx --> xx --> xx --> xx --> xx --> xx -->
  --> xx--> xx--> xx--> xx--> xx--> xx--> xx--> xx-->
  --> xx--> xx--> xx--> xx--> xx
```

F1	F2	F3	F4	F5	F6	F?	F8
INPUT							EXIT

In output field 5 of the header (ONLINE or OFFLINE) “EPROM” appears as source if an EPROM is used!

The interrupt list stored is displayed as generated (see Section 3.3.3), with the following differences:

- There are no entry fields.
- There is no vacant field at the end of the list.

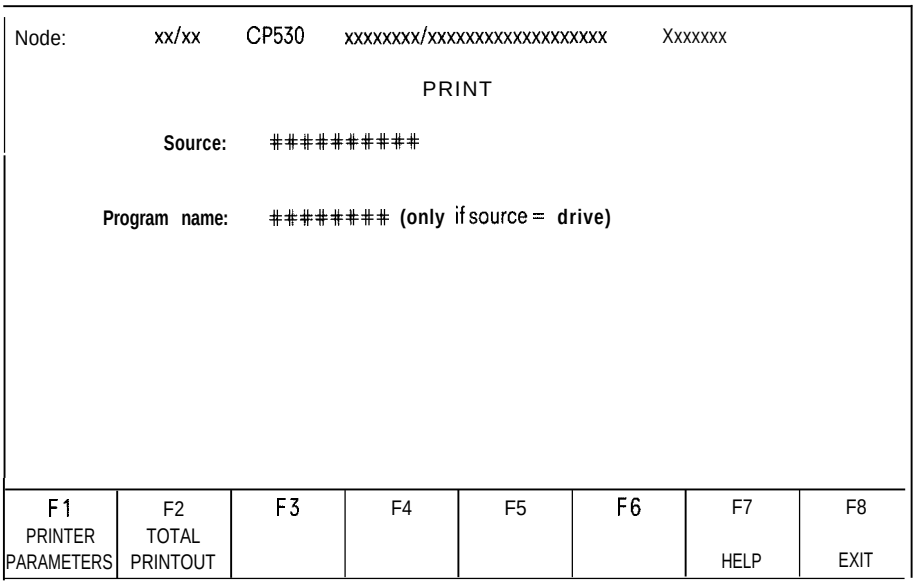
The slave numbers appear in the output field as entered.

Assignment of the function keys: (Softkey F 1 does not appear if display is from an EPROM or EEPROM!)

F1: This key permits entries to be made in the interrupt list (see Section 3.3.3) for modifications etc.

F8: Causes return to the DISPLAY form

3.5 PRINT Form



Entry fields in the PRINT form:

SOURCE:

Here it is possible to specify the source without changing the mode. An EPROM can also be given as source!

Possibilities:

Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP 530, EPROM, EEPROM

* Depending on drives defined, e. g. A, B, C . . .

These alternatives can be entered using the HELP function.

Default: ONLINE mode . . . . . CP 530
OFFLINE mode FD

PROGRAM NAME:

If a diskette (FD) is specified as source, a programm name must be entered here under which the bus parameters have been stored on diskette. The HELP function can be used to review all program names from the user diskette.

Function keys:

F 1: (PRINT-PAR) Causes transfer to the PRINTPAR form for assignment of parameters to the PG printer output.

The following menu appears:

F1	F2	F3	F4	F5	F6	F7	F8
YES	NO					HELP	EXIT

F 1: Print parameters are read from the user diskette before printer.

F2: No parameters are read and printing is started.

F7: HELP function for entering the source.

F8: Return to the CONFIGURATION form.

Printing of all user data from the CP 530 (in ONLINE mode) or user data from the EPROM or the user data in the user drive under the program name specified (in OFFLINE mode). The printed pages are provided with headers and footers (user texts as in F 1) and numbered.

Possible error messages:

Error OI: „Illegal input!"

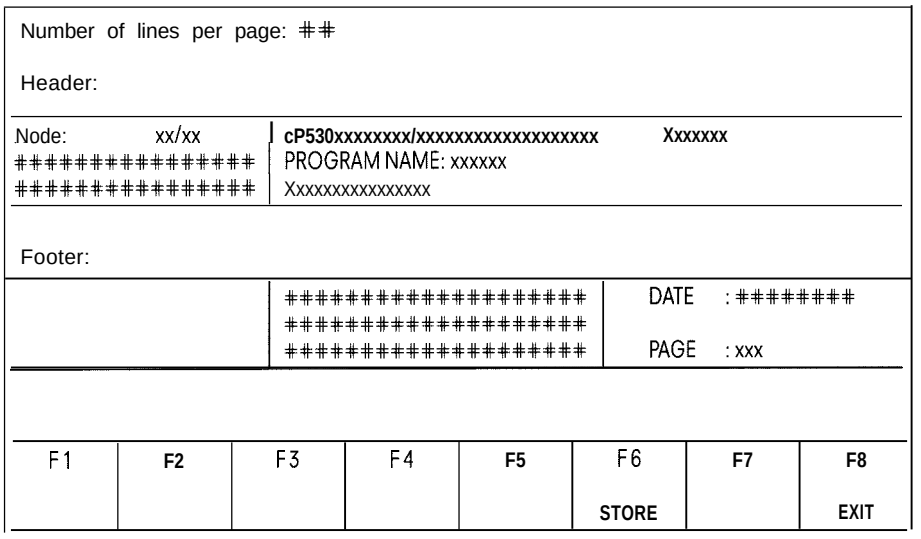
Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP530, EPROM, EEPROM

* Depending on drives defined, e. g. A, B, C

Error 46: "incomplete input"
If the diskette (FD) is stated as source, a program name must be specified.

3.5 PRINT Form

3.5.1 PRINTPAR Form



This form is also used to define the format of printouts.

The print parameters can only be stored once on diskette or hard disk. They then apply for all records.

Entry fields in the PRINTPAR form:

NUMBER OF LINES PER PAGE: Specification of number of lines per page for the programmer printer connected.
Range: 40...65

DATE: The current date can be entered herein any format.

The other entry fields can be filled with user text as required.

In this case, the output fields only indicate that dynamic texts appear in the printout here.
('SYSID', 'Polling list', 'Page number' etc.)

The entries in the ONLINE and OFFLINE printouts refer to the state of COM 530 when storing the print parameters.

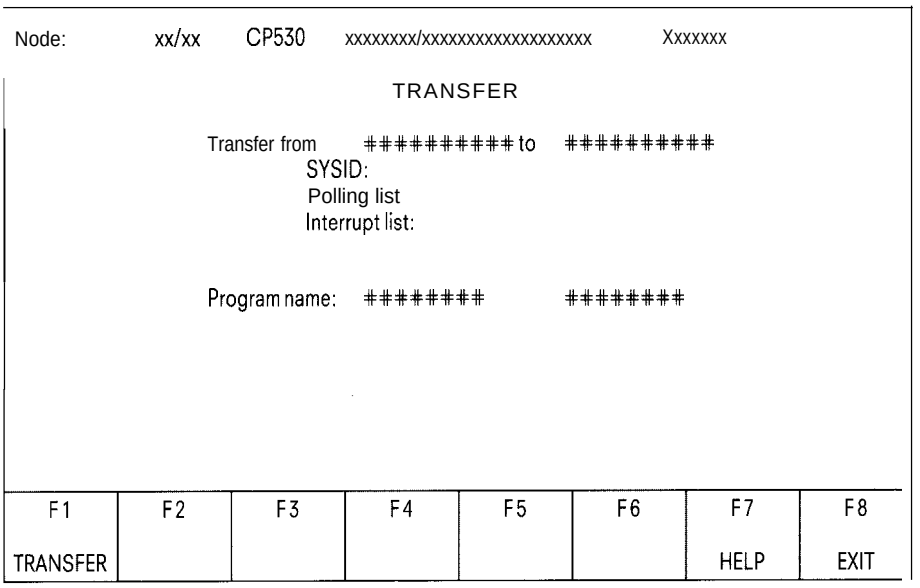
Function keys:

F 6: Operation of this function key causes the parameters generated for printout to be stored on the user diskette. They are not assigned to any program name.
The next time parameters are assigned to the printer, this form already has these parameters as defaults.

F 8: Return to the PRINT form

Possible error messages:

Error OI: "illegal input"
Permissible number of lines 40 to 65



Entryfields intheTRANSFERfrom:

TRANSFERFROM/TO:

In these two fields, the source and destination of the transfer are to be specified (This is also possible using the HELP key).

Possibilities:

Mode	Source	Destination
OFFLINE	FD, EPROM, EEPROM	FD, EPROM, EEPROM
ONLINE	FD, CP 530, EPROM, EEPROM	FD, CP 530, EPROM, EEPROM

Default: Source: 'DISKETTE' ('FD')
Destination: 'EPROM'

PROGRAM NAME:

If the diskette is specified as either destination or source, a program name must be entered under which the user data is stored on the diskette or are to be stored on it.
The HELP key can be used to review all program names on the user diskette.

Function keys:

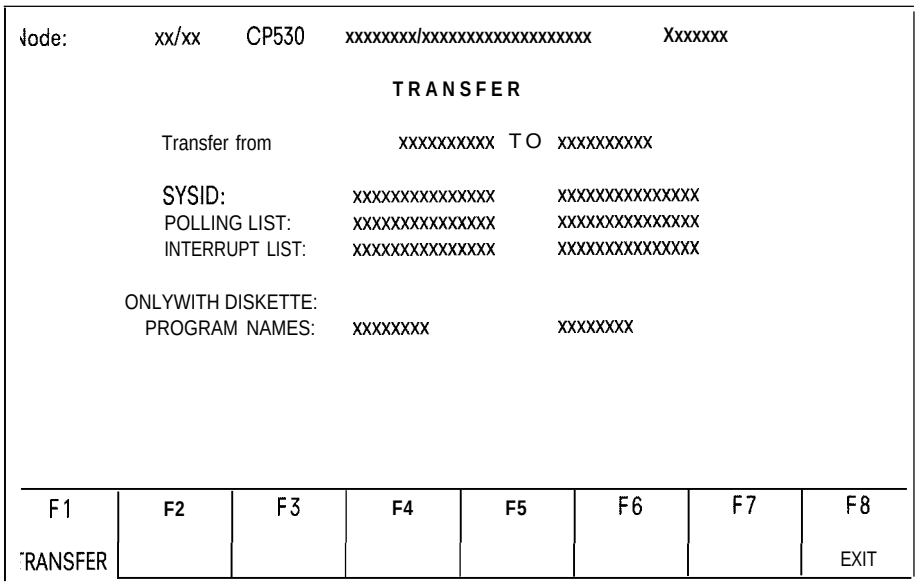
F I: The entry fields are converted into output fields. The data involved in each case also appear under the source and destination identifiers.

F 7: HELP function (paging through the alternatives) in the first five fields. In addition, the names of all programs on the user diskette can be reviewed in the "PROGRAM NAME" field.

F 8: Return to the CONFIGURATION form.

3.6 TRANSFER Form

On pressing F1: (TRANSFER)



Function keys:

F1: Initiation oftransfer.
The messages ''Actives", "Ended'' or ''Error messages" appear in thedisplayline.

F8: Return toTRANSFERfrom.

Possible error messages:

Error OI: "illegal input!"
Source and destination must be specified. The following possibilities are available:

Mode	Source	Destination
OFFLINE	FD, EPROM, EEPROM	FD, EPROM, EEPROM
ONLINE	FD, CP 530, EPROM, EEPROM	FD, CP 530, EPROM, EEPROM

Error 21: Source = Destination?
In "Transfer" mode, source and destination must not be identical. If FLOPPY is specified, it is sufficient to distinguish between the program names or drive designations. Possibilities:

Destination / Source	DISKETTE	CP530	(E)EPROM
FLOPPY	+	+	+
CP530	+	=	+
(E) EPROM	+	+	=

+ permissible – illegal

Error 36: "No SYSID identifier! Nothing transferred. "
This error message appears when F1 is pressed, although there are no data available in the source

Error 54: Illegal EPROM submodule type

3.7 Test and Startup

3.7.1 TEST Form

(Refer also to instructions GWA 4N5EB 8110545-02 for the L1 network, Section 3.3)

Node: xx/xx CP530 xxxxxxxx/xxxxxxxxxxxxxxxxxx Xxxxxxx

TEST AND STARTUP

READ DISKETTE FOR FORMATS AND STORE DATA: #~

F1	F2	F3	F4	F5 READ FORMATS	F6	F7 HELP	F8 EXIT

Note: The data diskette is required for storing or reading mailbox data (KH, KF etc)

Function keys:
After entering a drive which contains a diskette or a hard disk (possible with the F7 HELP key), the formats (KH, KC, ...) are stored,

F5: The following menu appears on the screen:

F1 STAT/ FORCE	F2 BUSTEST	F3 BUS CYCLE TIME	F4	F5	F6	F7	F8 EXIT

Function keys: (Softkeys F 2 and F 3 appear if the CP 530 is Master!)

F 1: Selects display of contents of mailboxes (STAT/FORCE form)

F 2: Selects bus test. The bus is stopped and further processing can then take place step by step in accordance with the polling list, (BUS TEST form)

F 3: Selects display of the bus cycle time. (CYCLE TIME form)

F 8: Return to the CONFIGURATION form

Possible error messages:

Error 35: "illegal return message from CP!"

Error 36: "CP USARTerror!"

Error 37: "No message from CP!"

Error 38: "US ARTerror PG side!"

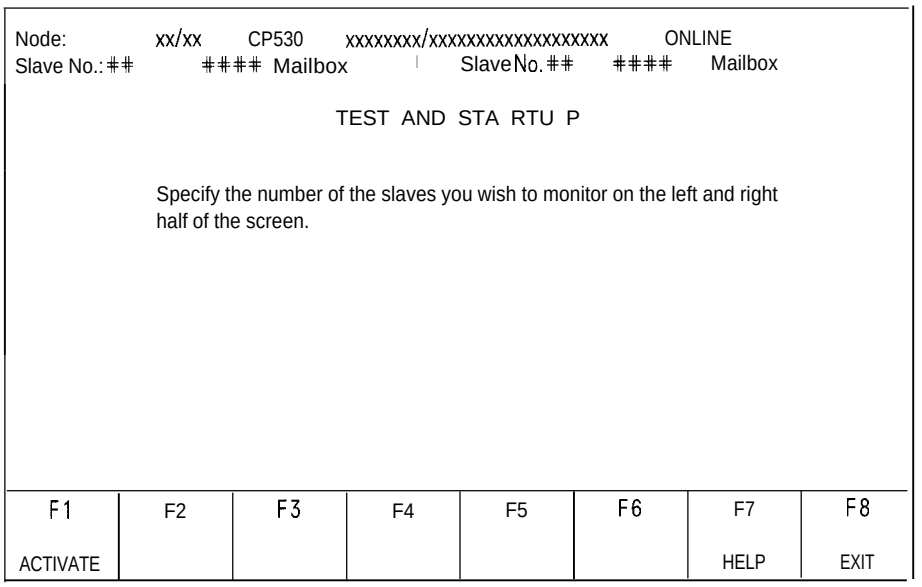
Error 39: "USARTerror PC side!"

Error 3A: "Interface not ready!"

Error3B: "Abort by CP!"

Note: In the following screen forms and their descriptions, the send and receive mailbox for a slave is always the same as the send and receive mailbox for the master CPS.

3.7.2 STAT/FORCE Form



Entry fields in the STAT/FORCE form:

SLAVE No. :

Specifies the numbers of the slaves to be tested

Range: 1...30

MAI LBOX:

Specifies which of the mailboxes (send or receive mailbox) is to be updated in the left or right half of the screen. SEND can be specified for the send mailbox or REC for the receive mailbox of the slave.

Default: SEND in both fields

Function keys:

F 1: The input data are transferred and the CP test function activated. The mailboxes specified are read (selects STATUS form).

F 7: SEND or REC can be entered in entry fields 2 and 4.

F 8: Return to the TEST form.

Possible error messages:

Error 01: **"Illegal input" !**
If a number outside the upper or lower limits is displayed in fields 1 and 3 or if there are blanks in these fields.
If something other than SEND, REC or blank appears in fields 2 and 4.

Error 37: "CP does not report!"

Error 38: "US ARTerror PG side!"

Error 39: "USARTerror PC side!"

Error 3A: "Interface not ready!"

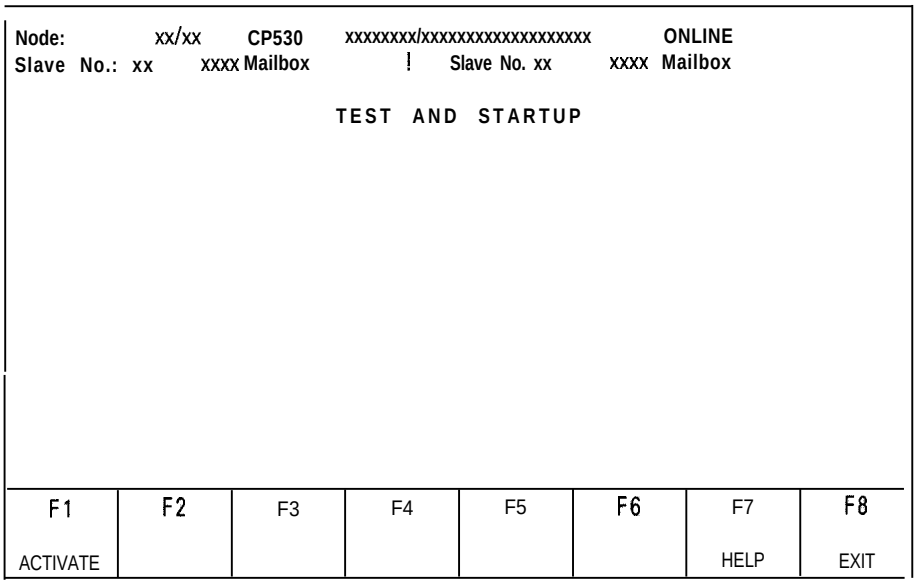
Error 3B: "Abort by CP!"

Error 58: "Slavexxfailed!"

Error 59: "Slave xx not on bus!"

In these last two error messages the number of the respective slaves is entered in the output fields.

If the CP 530 is configured as a slave, only its send and receive mailboxes can be monitored and the STAT/FORCE form appears as follows:



Output fields:

SLAVE No.:

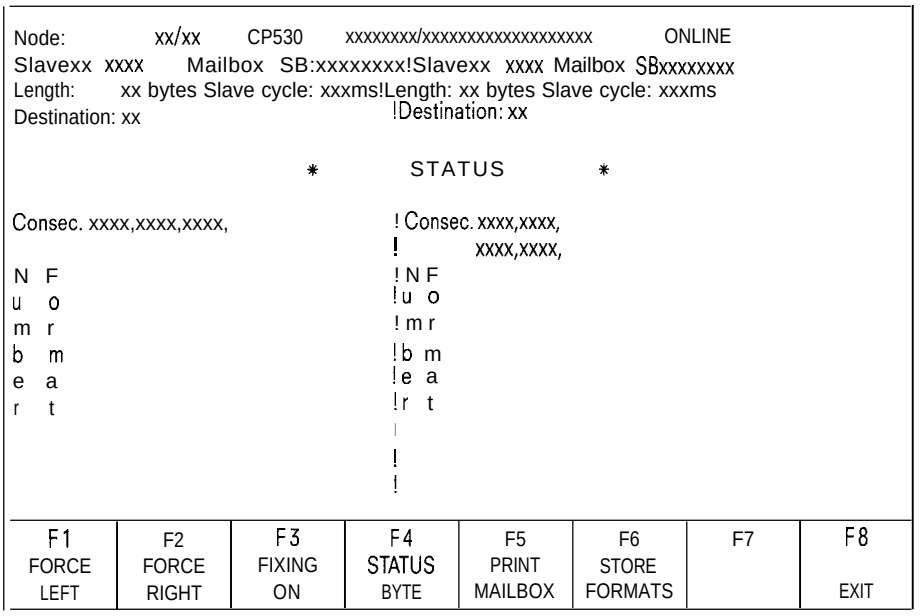
The SINEC L1 mode number from the SYSID appears here.

Range: 1...30

Function keys:
Same assignment as for the master

Possible error messages:
as for the master

3.7.3 STATUS Form



The following possibilities are available for menu labelling:

Mode left	Mode right	Possible softkeys
SEND	SEND	F 1–F6, F8
SEND	REC	F 1, F3 – F6, F8
REC	SEND	F2 – F6, F8
REC	REC	F3 – F6, F8
–	SEND or REC	F2 – F6, F8
SEND or REC	–	F2 – F6, F8

If the CP 530 is configured as a slave, the “FORCE LEFT” and “FORCE RIGHT” functions cannot be implemented

Output fields:

SLAVE: In the case of a receive mailbox, the number of the slave selected is entered. In the case of a send mailbox, 00 is entered as identifier for the master.

MAI LBOX: The data selected are taken from the previous form.

SB: The status byte of the respective slave is displayed in binary code. It can be examined in more detail with function key 4 (magnifier function).

LENGTH: The length of the mailbox in bytes.

SLAVE CYCLE: The response time of a slave until its mailbox is received (in milliseconds).

Destination: In the case of a receive mailbox, 00 is entered for the master in this field; in a send mailbox, this number is identical with that of the slave selected.

* : The appearance or disappearance of an asterisk in this field whenever a mailbox arrives indicates data traffic between the PG 675 and the CP 530.
The blinking frequency of the asterisk thus permits a rough estimate of the bus cycle time.

The part of the form between header and menu shows the contents of the mailboxes in the usual S 5 form divided into left and right mailboxes. The contents are constantly updated.

N. **B.:** When displaying a mailbox with a odd byte length, the irrelevant positions are omitted.
Example Mailbox with (byte) length of 9:

```
0:  KH = AAAA
1:  KF = +00255
2:  KY = 010,011
3:  KT = 100.3
4:  KH = FF
```

With the KH, KY, KC and KM formats, the last 2, 3, 1 or 9 positions are omitted. With the other formats, the data field is deleted.

Function keys F 1 to F 3 can be used to modify display formats or enter new contents in the send mailboxes.

Function keys:

F 1: Forcing the left mailbox. Both mailboxes are fixed, i. e. the display is frozen (not updated). The contents of the left mailbox can be modified.
Selects the FORCE form.

F 2: Forcing the right mailbox. Both mailboxes are fixed, i. e. the display is frozen (not updated). The contents of the right mailbox can be modified.
Selects the FORCE form.

F 3: Both mailboxes are fixed. The cursor is in the first format field of the left mailbox. Now the formats for the display can be modified (for both mailboxes).
The menu labelling for F 3 is changed to “FIXING OFF”. In addition, softkey F 4 enables the cursor to be positioned in the desired mailbox (see menu, next page).
Actuating F 3 once more causes a return to updating of the mailbox contents in the new formats. The form appears as shown in the diagram above.

F 4: This function enables the slave status byte, which appears as a binary pattern at the top right of each mailbox display, to be more closely examined. However, the status byte cannot be modified.
Causes the STATUS BYTE form to appear. (Section 3.7.5).

F 5: Listing of the contents of both mailboxes on the printer connected to the programmer. The display is frozen during printing.

F 6: The formats for the respective mailbox are stored in the user diskette. The display is frozen during the storage procedure. For each slave only one list of formats can be stored for the send mailbox and for the receive mailbox.

F 8: Return to the STAT/FORCE form

Menu after actuation of the "FIXING ON" key:

F1	F2	F3	F4	F5	F6	F7	F8
FORCE LEFT	FORCE RIGHT	FIXING OFF	FORMAT RIGHT	PRINT MAILBOX	STORE FORMATS		EXIT

F 4 is labelled "right" or "left", depending on whether the cursor is in the left or right mailbox.

Possible error messages:

Error 13: "Unknown data identifier!"
The formats can be modified in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted!"
Possible in conversion from all formats to the formats KT, KC, KS or KG.

Error 17: "Beginning – no further!"
If the mailbox is paged upwards but the first line has already been displayed.

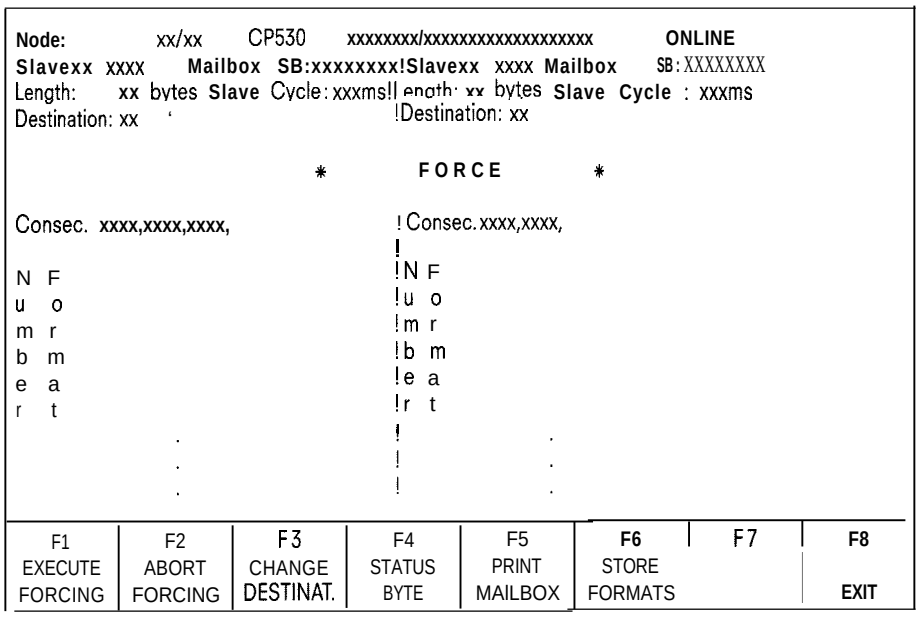
Error 18: "End – no further!"
If the mailbox is paged downwards and the last line has already been displayed.

Error 2C: "Left mailbox empty!"
If the interface module brings a left mailbox with length zero, this message is displayed.

Error 2D: "Right mailbox empty!"
If the interface module brings a right mailbox with the length zero, this error message is displayed.

Message 01: "Active"
If the F 6 function key, the Enter key or the F 5 function key or hardcopy key are actuated, this message appears.

3.7.4 FORCE Form



Output fields:
See STATUS form, Section 3.7.3

Function keys:

F 1: Forcing is executed, i. e. the modified send mailbox is sent to the destination node and the bus cycle continues. A return is made to the STATUS form and any newly entered formats taken over.

F 2: Forcing is not executed. The bus cycle continues.
Return to the STATUS form. New formats are not taken over.

F 3: The cursor jumps to the “Destination: xx” field which now becomes an entry field and permits a new destination slave to be entered.
Range: 1 ... 30

F 4: This function permits the status byte of the slave which appears as a binary pattern at the top right of each mailbox display and can be more closely examined and modified.
Causes the STATUS BYTE form to appear.

F 5: Printout of the contents of both mailboxes on the printer connected to the programmer.

F 6: The formats for the respective mailbox are stored on the user diskette.

F 8: Return to the STAT/FORCE form without forcing being executed.

Possible error messages:

Error 08: "Inhibited key!"

Error 12: "INSERT/ERASE" not possible here!"
The cursor is in the last line and an attempt has been made to erase or insert in the mailbox.

Error 13: "Unknown data identifier!"
The formats can be changed in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted!"
Possible on conversion from all formats to the KT, KC, KS or KG formats.

Error 17: "Beginning – No further!"
If the first line in the mailbox is displayed and an attempt is made to scroll upwards.

Error 18: "End – No further!"
If the last line in the mailbox is displayed and an attempt is made to scroll downwards.

Error 19: "Do not press function key!"
If the cursor is in the first field, i. e. where the editor is expecting an initial word address, it is not permitted to actuate a function key.

Error 1A: "Repetition factor to high!"
A repetition factor can be specified when editing a mailbox. If the total length of the mailbox, taking the repetition factors into consideration, is more than 32 words, the editor outputs this error message.

Error 2C: "Left mailbox empty!"
If the CP 530 supplies a left mailbox with the length zero, this message is displayed.

Error 2D: "Right mailbox empty!"
If the CP 530 supplies a right mailbox with the length zero, this error message is displayed.

Error 35: "illegal return message from PC!"

Error 36: "PC – USART error!"

Error 37: "CP does not report!"

Error 38: "US ART error, PG side!"

Error 39: "US ART error, PC side!"

Error 3A: "Interface not ready!"

Error 3B: "Abort by CP!"

Error 4A: "Slave already receiving data!"

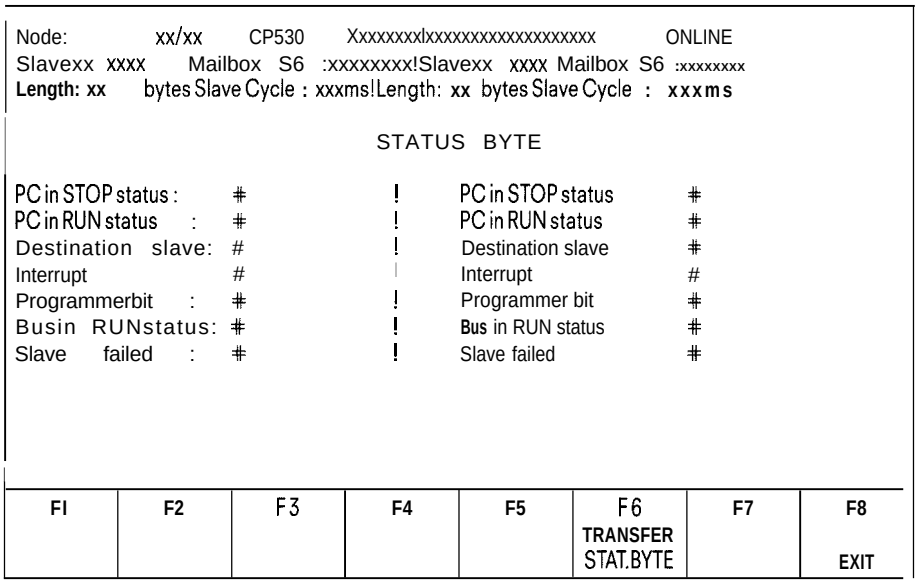
Error 57: "Error in slave cycle"
Slave 1) gives negative acknowledgement
2) supplies wrong data
3) has frame errors

Error 58: "Slave xx failed!"
Slave is in polling list on the bus, but does not reply to the master call.

Error 59: "Slave xx not on bus!"
In these last two error messages the number of the respective slave is entered in the output fields.

Message 01: "Active!"
If the F 6 function key, the transmit key or the F 5 function key or hardcopy key are actuated, this message appears.

3.7.5 STATUS BYTE Form



Output fields:

See STATUS form, Section 3.7.3

Entry fields:

The status byte is displayed for both slaves. By changing the bit statuses (in the entry fields), the status bytes are transmitted to the slaves. This, however, is only possible when changing from the FORCE form and actuating the F 6 key.

Function keys:

F6: Transmitting the status bytes as displayed on the form to the slaves.
Return to the FORCE form.

F8: Return to the FORCE form without changing the status bytes.

Possible error messages:

Error 01: "illegal input!"
If a digit other than O or 1 or a blank appears in the entry fields.
If the first two bits have the same status, i. e. if both are set or reset, this message also appears.

Example of STATUS FORCING:

```
Node:        23/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx  ONLINE
Slave No. 5      REC Mailbox                !  Slave No, 16 SEND Mailbox

                        TEST AND STARTUP

          Specify the number of the slaves you wish to observe on the left and right
          halves of the screen,

 F1      | F2 | F3 | F4 | F5 | F6 | F7   | F8
 ACTIVATE|    |    |    |    |    | HELP | EXIT
```

Actuating F 1 causes the STATUS form to appear.
The CP 530 then supplies, for example, the right mailbox first, displays it on the screen and then displays the left mailbox. Function key 3 is pressed while the CP 530 accesses the data of the slave.

The following display appears on the screen:

```
Node:        23/00  CP530  xxxxxxxx/xxxxxxxxxxxxxxxxxx  ONLINE
Slave 5 REC  Mailbox  SB:01001100        !Slave 00   SEND Mailbox SB : 10001110
Length: 48 bytes Slave Cycle : 159ms     !Length: 10  bytes Slave Cycle  : 234ms
Destination: 00                          !Destination: 16

                          STATUS
 0:     KH = ABCD                        !   O: KT= 735.2
 1:     KM= 0100110001110000             !   1: KY= 234,189
 2:     KH = 12CF                        !   2: KS= “;
 3: 5   KH = 3E6A                        !   3: KM= 1011100010101001
 8:     KG= +1423148-05                  !   3 :  K F =  - 2 1 5 5 5
 9:11   KS= §Q                           !   5:
20:     KM= 1110000111010110             !
21:     KC= 391                          !
22:     KM= 0101101001011010             !
23:     KY=  24,  1                      !
24:                                      !

 F1 | F2 FORCE RIGHT | F3 FIXING OFF | F4 FORMAT LEFT | F5 PRINT MAILBOX | F6 STORE FORMATS | F7 | F8 EXIT
```

3.7 Test and Startup

The cursor is positioned in the left mailbox. The formats can now be chanaed in the mailboxes. The followina formats are permissible: KH, KF, KS, KM, KT, KC, KG and KY.

A "?" appears for characters which cannot be converted to the format specified.

The next function selected is "FORCE RIGHT".

The following appears on the screen:

```
Node:          23/00     CP530   Xxxxxxxdxxxxxxxxxxxxxxxxxxx        ONLINE
Slave5 REC Mailbox SB:01001100    !Slave 00          SEND Mailbox SB 10001110
Length: 24 bytes Slave Cycle : 159ms  !Length: 10 bytes Slave Cycle: 234ms
Destination: 00                   !Destination: 16

                              FORCING        *
 O:     KH   = ABCD               !      O:  KT = 735.2
 1:     KM  = 0100110001110000    !      1:  KY= 234,189
 2:   KH   =    12CF              !      2:  KS=  ";
 3:     KH   = 3E6A               !      3: KM= 1011100010101001
 4:     KH   = 3E6A               !      4: KF =  -21555
 5:     KH   = 3E6A               !      5:
 6:     KH   = 3E6A               !
 7:   KH   =   3E6A               !
 8:     K G  = +1423148-05        !
 9:     KS   = §Q                 !
10:     KS   = §Q                 !
11:     KS   = §Q                 !
```

F1	F2	F3	F4	F5	F6	F7	F8
EXECUTE FORCING	ABORT FORCING	CHANGE DESTINAT.	STATUS BYTE	PRINT MAILBOX	STORE FORMATS		EXIT

The contents of the right mailbox can now be edited. Using the F 4 key, the status byte can be examined more closely and edited. Each bit has a certain function and can be set or reset.

For a description of the function keys, see Section 3.7.4.

3.7 Test and Startup

3.7.6 BUS TEST Function

The BUS TEST function of the COM 530 enables bus traffic to be executed step by step, whereby the data exchanged between CP 530 and the nodes can be observed in the form of mailboxes and, in certain cases, modified. When the bus test is called, the SINECL1 cycle is stopped at the end of the polling list.

Note: This function is only possible if the CP is configured as master

Operator procedure depends on whether or not interrupts triggered by slaves occur during bus processing.

1. Procedure without interrupts (master-slave traffic)

Each step in the bus test consists of two parts:

A) In the first part, the operator can examine and also modify the send mailbox of the CP 530 to the current slave (the slave currently being processed). This first part is completed by actuating the F 1 "SEND MAILBOX" softkey.

The following reaction is displayed on the programmer screen:
- Acknowledgement by the slave of the send mailbox of the CP 530
- The receive mailbox received by the slave and its acknowledgement.
- The cycle time for the entire traffic:
  Send-receive in master-slave traffic
  Send-receive-send . in slave-slave traffic
- Softkey F 1 now has the inscription "NEXT STEP", enabling the second part.

B) The second part, which is also initiated by softkey F 1 (now "NEXT STEP"), enables the minibus to poll the next slave in the polling list.

The following reaction is displayed on the screen of the programmer:
- The send mailbox of the CP 530 to the next slave in the polling list.
- Softkey F 1 has the label "SEND MAILBOX" once more, i.e. the first part is active again (see I.A.).

2. **If an interrupt occurs**

In this case, after the first part has been executed, an operator input must now be made:

A) The following reactions appear on the screen:
- The slave causing the interrupt is acknowledged:
- The send mailbox of the CP 530 to the interrupting slave appears.
- The send mailbox from the interrupting slave to another node appears.
- The "INTERRUPT CYCLE" message appears.
- Softkey F 1 ("NEXT STEP") permits the execution of the next step.

B) After this intermediate step, the previous procedure is followed as under 1.B. as long as no further interrupt occurs.
If a further interrupt occurs, the screen reaction is as in 2.A.

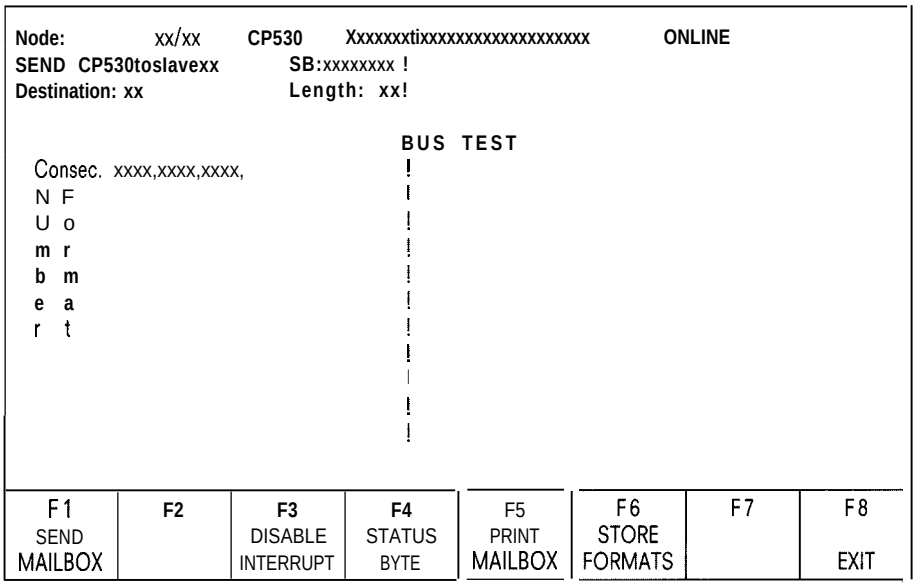
In the BUS TEST forms, send mailboxes of the CP 530 appear on the left half of the screen. Only these mailboxes can be modified. On the right half of the screen, the mailbox which is sent back to the CP 530 by the respective slave (direct traffic) appears
or
the mailbox which is sent from the current slave to another node (cross traffic) appears.

As the SINEC-L1 traffic is executed step by step in the BUS TEST, the occurence of frequent interrupts could prevent normal processing of the polling list. For this reason, a softkey function permits interrupts to be enabled or disabled.

The forms for the two parts of the operation for master-slave and slave-slave traffic follow:

BUS TEST forms 1-3

This form appears in the first step:



The left half of the screen shows the send mailbox of the CP 530 which is to be sent to the current slave.
Modifications can be made with the usual S5 operator input procedures.

Assignments of the function keys:

F I : Sending the send mailbox to the current slave. Causes the BUS TEST 2 form to appear for the second step.

F 3: The inscription of the key is alternately “DISABLE INTERRUPT” or “ENABLE INTERRUPT”. This key makes it possible to service the slaves in accordance with the polling list even if an interrupt occurs.

F 4: Function as in FORCING (Section 3.7.4)

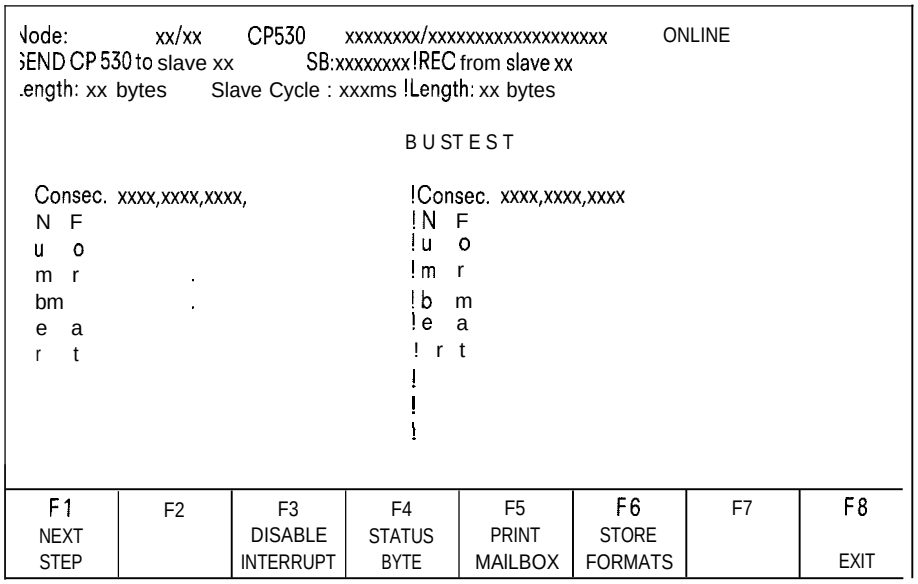
F 5: Function as in FORCING (Section 3.7.4)

F 6: Function as in FORCING (Section 3.7.4)

F 8: Exit BUS TEST function.
The bus remains in the STOP mode, and must be set again to the RUN mode with the MODES function.

BUS TEST 2 Form

This form is displayed in the second step:



The send mailbox from the previous form remains displayed. It can no longer be edited, however. The receive mailbox for the current slave appears on the right of the screen. The formats can be modified here.

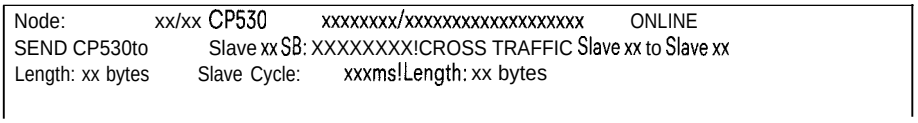
Assignment of the function keys:

F1: Polling the next slave. Return to the BUS TEST 1 form.

For all other function key assignments, see BUS TEST 1 form.

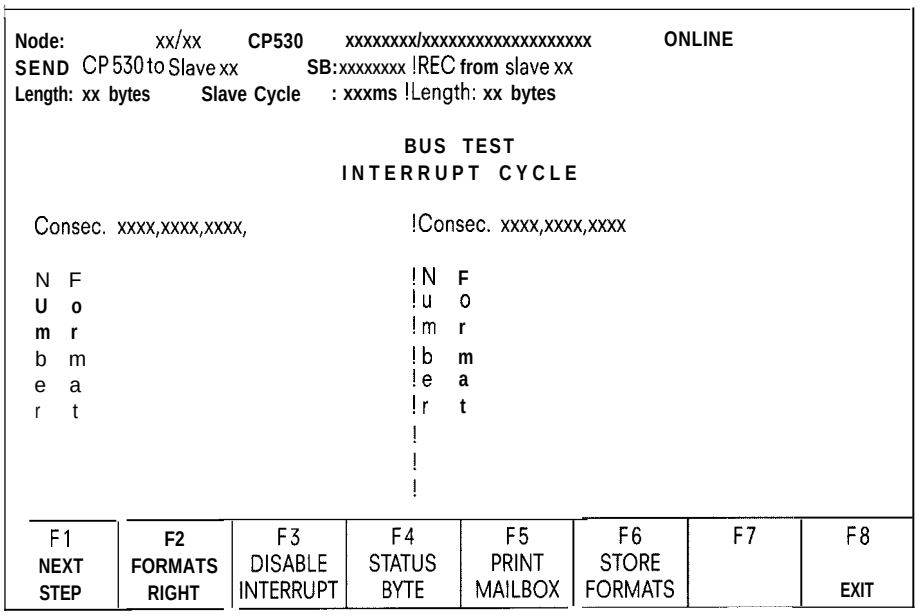
On the right of the screen, the mailbox from the slave to the master appears or the mailbox of the slave to another slave appears (slave-slave traffic).

In the case of cross traffic, the following form header appears:



BUS TEST3 Form

This form appears (in the second step) if an interrupt occurs:



The left half of the screen displays the send mailbox from the CP 530 to the slave sending the interrupt. On the right, the send mailbox from the current slave to the master or another slave appears.
Editing is not possible in either mailbox, but the formats can be changed.
Whether or not slave-slave traffic is taking place is indicated in the header.

Function keys:

F 1: Bus traffic continues. If no further interrupt occurs, the BUS TEST 2 form appears once more. The original send mailbox for the slave whose turn it should have been is displayed on the left and on the right the respective receive mailbox appears.
If a further interrupt occurs, however, the BUS TEST 3 form reappears with the mailboxes to and/or from the slave sending the interrupt.

F 2: This key must be actuated to position the cursor for changing the formats in the required mailbox. The inscription is alternately “FORMATS RIGHT” and “FORMATS LEFT”.

For further function keys, see the BUS TEST 1 form.

Possible error messages:

Error OB: "Inhibited key!"

Error 12: "!NSERT/ERASE not possible here!"
The cursor is in the last line and an attempt has been made to insert or erase in the mailbox.

Error 13: „"Unknown data identifier!"
The formats can be changed in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 16: "Data cannot be interpreted"
In connection with conversion from all formats to the formats KT, KC, KS or KG.

Error 17: "Beginning – no further!"
If the mailbox is paged upwards but the first line is already displayed.

Error 18: "End – no further!"
If the mailbox is paged downwards and the last line is already displayed.

Error 19: "Do not press function key!"
If the cursor is in the first field, i. e. where the editor is expecting an initial word address, it is not permitted to press a function key.

Error 1A: "Repetition factor too high!"
A repetition factor can be specified when editing a mailbox. If the total length of the mailbox, taking the repetition factor into account, is greater than 32 words, the editor outputs this error message.

Example of BUS TEST:

The bus test function has been selected. The mailbox of the master to the slave, e. g. slave 5, is displayed.

```
Node:              8/00     CP530     xxxxxxxx/xxxxxxxxxxxxxxxxxxx     ONLINE
SEND CP 530 to slave 5          SB:01001100  !
Destination: 5                  Length: 21 bytes !

                                        BU ST EST

 0:     KH = ABCD                       !
 1:     KM = 1011001110001111           !
 2:     KH = 2345                       !
 3:     KH = 2345                       |
 4:     KH = 2345                       !
 5:     KH = 2345                       !
 6:     KH = 2345                       !
 7:     KH = 9876                       !
 8:     KG = +1423148-05                !
 9:     KF = +32767                     !
10:     KY = 17,30                      !
11:     KC = 789                        |
```

F1 SEND MAILBOX	F2	F3 DISABLE INTERRUPT	F4 STATUS BYTE	F5 PRINT MAILBOX	F6 STORE FORMATS	F7	F8 EXIT

The mailbox can be edited. If function kev F 1 is actuated, the mailbox is sent to the slave and the receive mailbox is read.
Three different cases can now occur:

a) Slave sends mailbox to master

b) Slave sends to another slave

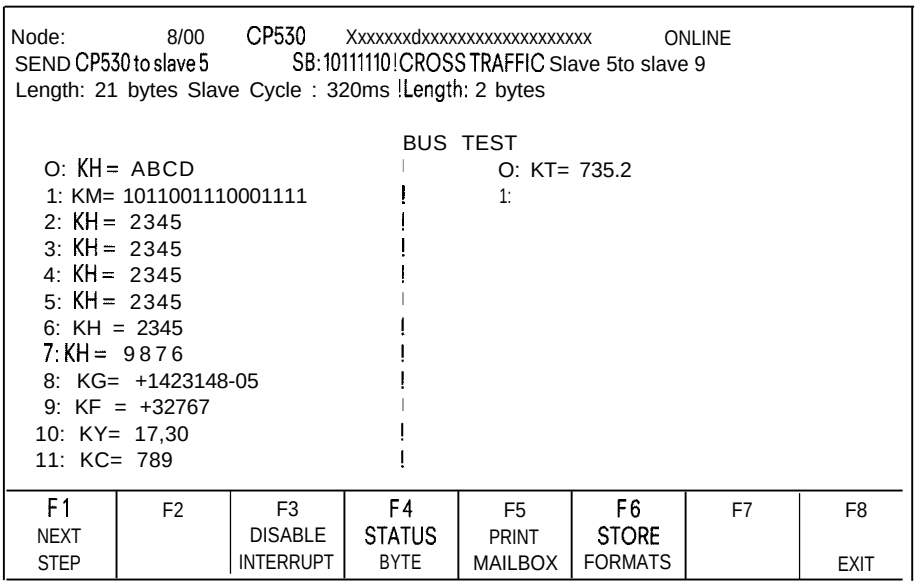
c) Interrupt

If case a) occurs, the form has the following appearance:

```
Node:          8/00    CP530    Xxxxxxxxkxxxxxxxxxxxxxxxxxx        ONLINE
SEND CP530toslave5          SB:10111110 !REC from slave 5
Length: 21 bytes    Slave Cycle :  320ms !Length: 6 bytes

                                        BUS TEST
  O: KH = ABCD                    !        0: KT= 735.2
  1: KM= 1011001110001111         !        1: KY= 234,189
  2: KH = 2345                    !        2: KS= ";
  3: KH = 2345                    !        3:
  4: KH = 2345                    !
  5: KH = 2345                    !
  6: KH = 2345                    !
  7: KH = 9876                    |
  8: KG= +1423148-05              !
  9: KF = +32767                  !
 10: KY= 17,30                    !
 11: KC= 789                      !
```

F1	F2	F3	F4	F5	F6	F7	F8
NEXT STEP		DISABLE INTERRUPT	STATUS BYTE	PRINT MAILBOX	STORE FORMATS		EXIT

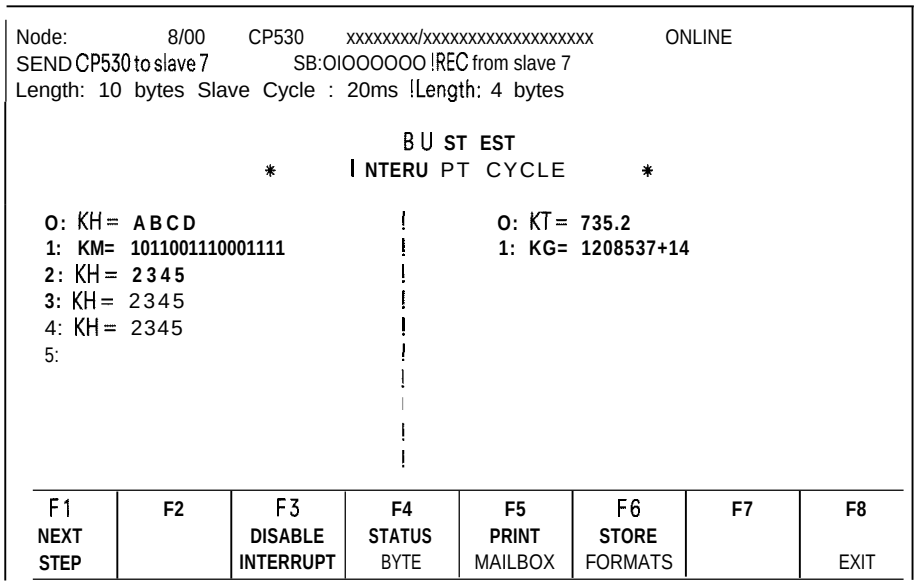
If case b) occurs, the form appears as follows:



In both cases, function key F 1 “NEXT STEP” can be used to display the send mailbox of the master to the next slave in the polling list. However, only the formats can be changed here.

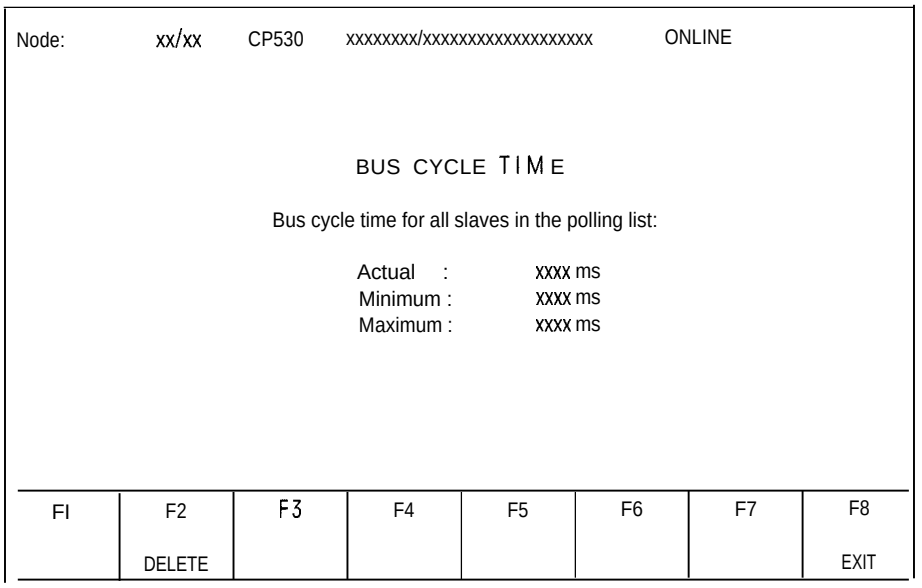
Case c) Interrupt

An interrupt has occurred, i. e. the send mailbox of the masterto the slave sending the interrupt and the mailbox of theslavesending the interrupt to the master or to another slave (slave-slave traffic) are displayed.



Now the left or right formats can be changed.
Actuation of the F 1 key causes a return to case a) orb) if no further interrupt occurs.

3.7.7 CYCLE TIME Form



Output fields:

ACTUAL: In this field the current bus cycle time is displayed. It is updated approximately every half second.

MINIMUM: A non-return pointer shows the lowest value.

MAXIMUM: A non-return pointer shows the highest value.

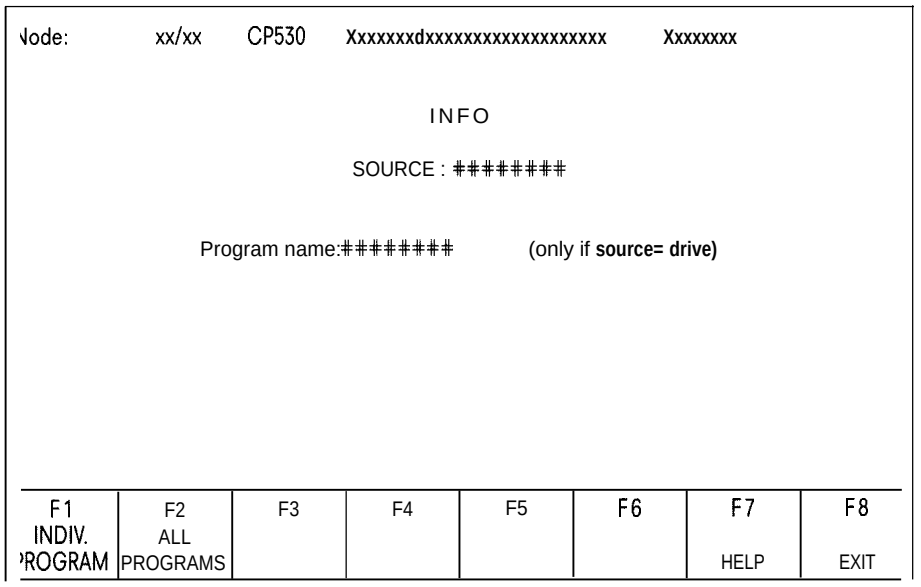
Function keys:

F2: The interface module receives the command to delete the bus cycle times.

F8: Return to the TEST form.

Note: This function is only possible if the CP is configured as master

3.8 INFO Form



Entry fields in the INFO form:

SOURCE:

Here it is possible to specify the source without changing the mode. An EPROM can also be given as source!

Possibilities:

Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP 530, EPROM, EEPROM

* Depending on drives defined, e. g. A, B, C,

These alternatives can be entered using the HELP function.

Default: ONLINE mode . CP 530
OFFLINE mode FD

PROGRAM NAME:

If a drive (FD) is specified as source, the program name under which the user data are stored on the minidiskette or hard disk must be entered here. All program names on the diskette/disk can be entered with the aid of the HELP function.

Function keys:

F1: Information is given on whether a SYSID identifier and/or polling list and/or interrupt list is available either in the memory submodule of the CP 530 or in the memory submodule of the CPU or whether these have been stored on the user diskette under the program name specified. Selects the INFO 1 form.

F 2: All program names on the user diskette are listed. Selects the INFO 2 form to appear. (Only possible for "FD" source!)

F 7: HELP function for entering the source and available program names.

F 8: Return to the CONFIGURATION form.

Possible error messages:

Error OI: "illegal input!"
Source must be specified. The following are possible:

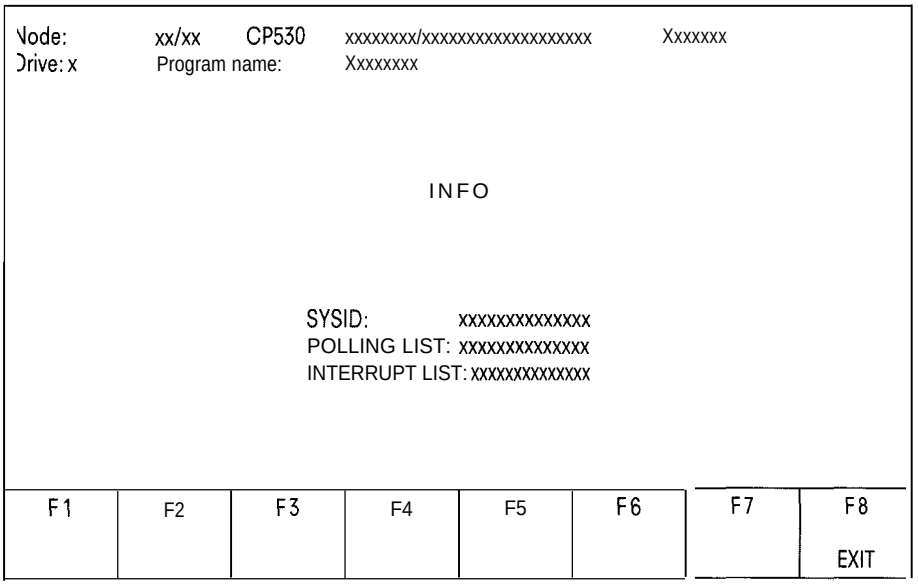
Mode	Source
OFFLINE	FD*, EPROM, EEPROM
ONLINE	FD*, CP 530, EPROM, EEPROM

* Depending on the drives defined, e. g. A, B, C, . . .

Error 29: "No program name for EPROM and EEPROM!"
If the EPROM and CP 530 is designated as source, no information can be received on "all programs" (i.e. all program names on the user diskette) as only one user program can be stored in the EPROM or CP530.

Error 46: "Incomplete input"
If a diskette (FD) is given as source, a program name must be specified.

INFO 1 Form

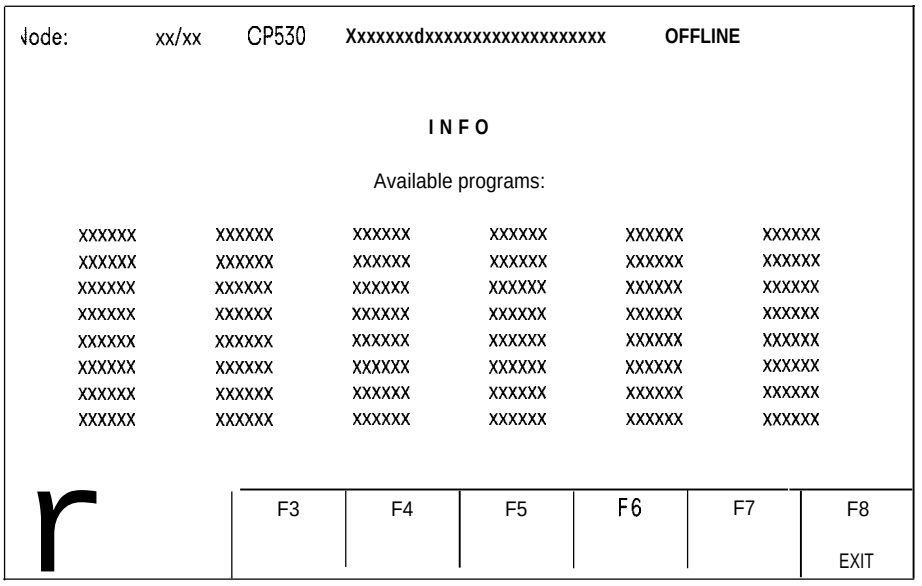


"AVAILABLE" or" NOT AVAILABLE" appear in the output fields.

F8: Return to the CONFIGURATION form.

3.8 INFO Form

INFO 2 Form



All available CP 530 programs on the selected disk/diskette are listed.

Function keys:

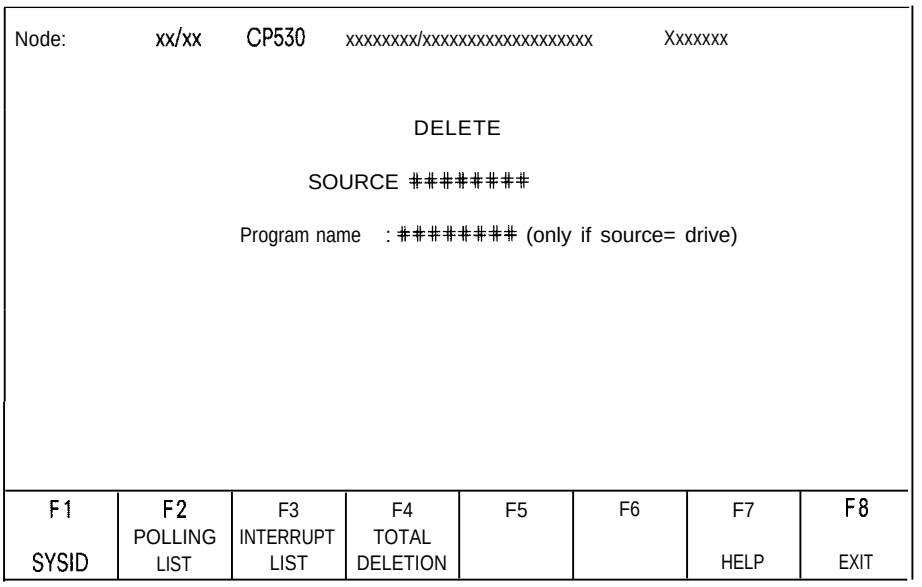
FI : Listing on the printer and return to the INFO form

F8: Return to the INFO form

In “Delete” mode, user data stored in the CP 530 or on diskette, hard disk or EEPROM under a certain program name can be deleted.

The following possibilities are available:

· Individual deletion (i. e.: Deletion of SYSID, polling or interrupt lists)
– Total deletion (i. e.: Deletion of SYSID, polling and interrupt lists)



Entry fields in the DELETE form:

SOURCE:

It is possilbe to specify the source here without changing the mode.

Possibilities:

Mode	Source
OFFLINE	FD
ONLINE	FD, CP 530

The alternatives can be entered using the HELP function.

Default: ONLINE mode . . . . . CP 530
OFFLINE mode FD

PROGRAM NAME:

If diskette (FD) is specified as source, a program name under which the bus parameters are stored on the minidiskette or hard disk must be entered here.

All program names on the user diskette can be entered with the aid of the HELP function.

3.9 DELETE Form

Function keys:

F1: Deletion of the SYSID identifier in the source specified:
Deletion is indicated with the "Active!" and "SYSID deleted!" displays.

F 2: Deletion of the polling list in the source specified:
Deletion is indicated by the "Active!" and "Polling list deleted!" messages.

F 3: Deletion of the interrupt list in the source specified:
Deletion is indicated by the "Active!" and "Interrupt list deleted!" messages.

F 4: **Delete** all user data:
The SYSID, polling list and interrupt list in the source specified (i. e. in the CP 530 or on diskette under the name stated) are deleted.
In the error message line the message "Delete all?" appears. The user is prompted to acknowledge.

Menu for acknowledgement:

F1	F2	F3	F4	F5	F6	F7	F8
YES	NO						EXIT

The deletion is indicated by the "Active!" and "SYSIS deleted!" or "Polling list deleted!" or "Interrupt list deleted!" messages.

F 7: HELP function for entering the source and the program names on the floppy diskette

F 8: Return to the CONFIGURATION form.

Possible error messages:

Error 01: "Illegal input!"
Source must be specified. The following are possible:

Mode	Source
OFFLINE	FD
ONLINE	FD, CP 530

Error 31: "Polling list does not exist!"

Error 32: "Interrupt list does not exist!"

Error 33: "SYSID identifier does not exist!"
One of these three error messages appears if the respective data type is not available on the CP 530 or on the user diskette under the program name specified.

Error 46: "Incomplete input"
If minidiskette or hard disk (FD) is given as source, a program name must be specified.

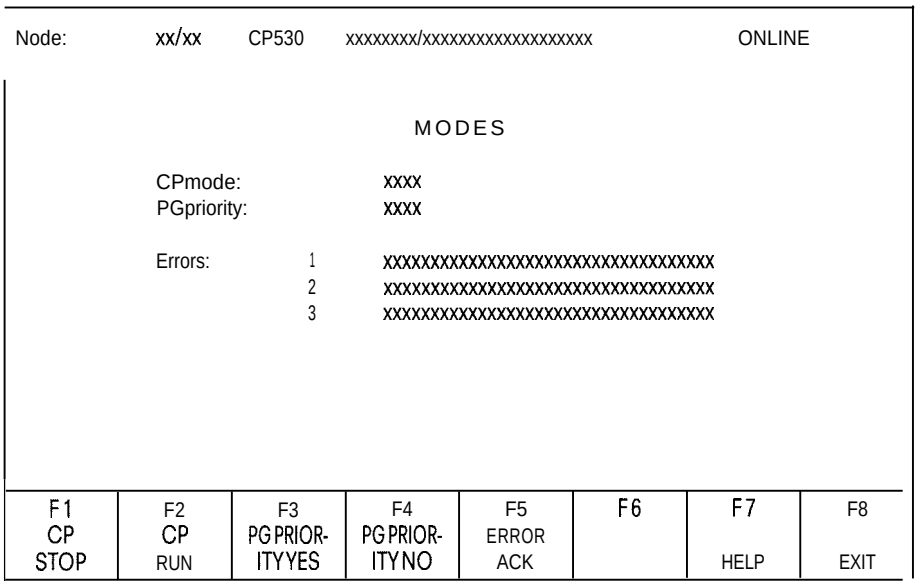
3.10.1 Operating Modes

The following operating modes can be set:

- Starting the CP 530 (CP RUN),
- Stopping the CP 530 (CP-STOP),
- PG priority (ON and OFF),

PG priority ON means that the programmer connected directly or indirectly to the PG (programmer) interface has absolute priority.

After selecting the form, the status information is constantly requested by the CP in the “Status” function and updated in the dis-



Output fields in the MODES form:

CP MODE:

Here the current mode of the CP 530 is displayed: The possibilities are “STOP” or “RUN”. The “STOP” or “RUN” modes can be forced with function keys F 1 or F 2.

PG PRIORITY:

The “PG PRIORITY” field also shows the current CP mode (possibilities: YES and NO).
These settings can be changed with the F 3 and F 5 function keys.

ERRORS:

The error messages collected in the CP 530 are displayed in the “ERROR” fields and updated approximately every half second.

After function key F 1 to F 4 have been actuated, the “CAUTION: DANGEROUS STATE – Operate keys again” message is output and only the relevant function keys appear in the menu.

3.10 Setting the Operating Mode (MODE Form)

Function keys:

F 1: The CP 530 is brought to the STOP mode at the end of the current cycle through the polling list.
As a dangerous system status can occur, the user is prompted to acknowledge with the following menu:

F1	F2	F3	F4	F5	F6	F7	F8
CP **STOP**	CP **RUN**						**EXIT**

F 1: After positive acknowledgement, the CP 530 is brought to STOP:
This is indicated as follows:
- The "Executed!" message is output and
- in the "CP MODE" field the word "STOP" appears.

F **2:** The operating mode is not changed as the CP 530 is already in the RUN mode. The "CP IN RUN MODE" message appears.
Only
a) Positive acknowledgement with F 1 or
b) Negative acknowledgement with F 8 are meaningful.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

F 2: The CP 530 is brought to the RUN mode. As a dangerous system status can occur, the operator is promted to acknowledge with the following menu:

F1	F2	F3	F4	F5	F6	F7	F8
CP STOP	CP RUN						EXIT

F 1: The operating mode is not changed as the CP 530 is already in the STOP mode. The "CP IN STOP MODE" message appears.
Only
a) positive acknowledgement with F 2 or
b) negative acknowledgement with F 8 are meaningful.

F 2: After this positive acknowledgement, the CP 530 is set to RUN.
This is indicated as follows:
- The "Executed!" message is displayed and
- in the CP mode field the word RUN appears.

F 8: A negative acknowledgement means that the operating mode is not executed. Return to main menu.

F3: The CP 530 is brought to the "PG check YES" mode.
As a dangerous system status can occur, the operator is prompted to acknowledge with the following menu:

FI	F2	F3 PG PRIORITY YES	F4 PG PRIORITY NO	F5	F6	F7	F8 EXIT

F 3: After this positive acknowledgement the CP 530 is brought to the "PG check YES" mode.
This is indicated as follows:
- The "Executed!" message is displayed
- "YES" appears in the "PG PRIORITY" field

F 4: The mode is not changed as the CP 530 is already in the "PG check NO" mode. The "PG PRIORITY OFF" message appears.
Only
a) Positive acknowledgement with F 3 or
b) Negative acknowledgement with F 8 are meaningful.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

F4: The CP 530 is brought to the "PG check NO" mode. As a dangerous system status can occur, the operator is prompted to acknowledge with the following menu:

F1	F2	F3 PG PRIORITY YES	F4 PG PRIORITY NO	F5	F6	F7	F8 EXIT

F 3: The operating mode is not changed as the CP 530 is already in the "PG check YES" mode. The " PG PRIORITY ON" message appears.
Only
a) Positive acknowledgement with F 4
b) Negative acknowledgement with F 8 are meaningful.

F 4: After this positive acknowledgement, the CP 530 is brought to the "PG check NO" mode.
This is indicated as follwos:
- The "Executed!" message is displayed
- "NO" appears in the "PG PRIORITY" field.

F 8: Negative acknowledgement means that the operating mode is not changed. Return to main menu.

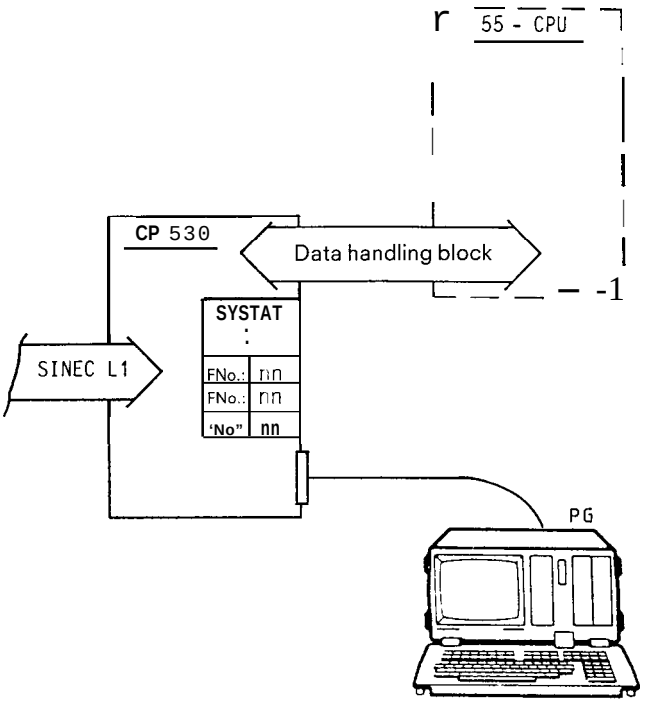
F 7: The HELP function causes a description of the operating modes which can be set with F 1 to F 4 to be displayed on the screen. This description can be exited with function key F 8 (EXIT).

F 8: Return to the CONFIGURATION form

3.10 Setting the Operating Mode (MODES Form)

3.10.2 Error Messages

Errors occurring in data traffic on the SINEC LI LAN and between the CP and the programmable controller are entered in the "Error" field and displayed in the "MODES" form.



The error number permits a breakdown of the errors into classes; the appended text provides more detailed information on the particular error or fault (see table of Error Classes).

a) Description of a group error number in more detail, e. g. the result of the self text produces class 1 errors and these can only be reported to the works if the module is replaced or repaired. Example: Error 7: "Hardware error 7".

b) Description relating to number of job containing error, e. g. Error 54: "lob 177 not defined".

c) Description of a slave in connection with which the error occurred, e. g. Error 60: "Slave 3 not in polling list".

Error Classes:

Advantages of error classes
- more efficient error messages
- the user can define a general reaction to a failure class.

There are also other errors but these are not shown in this way since the failure relates to those parts of the system necessary to transmit the error number.

Class	Error No.	General Description	Operator reaction	Level of service
1	10–29	Possible hardware fault	Check/replace/ repair Hardware	1. Operator 2. Service
2	30–49	Operational errors	Check/replace Switches Submodule	Operator
3	50–69	Parameter assignment programming error	Diagnostics byPG necessary S5-SW changes	Programmer, Configuring Engineer
4	70–90	Status messages	Record	Operator

Error List SYSTAT

Class	Error No.	Ext.	
	decimal	representation	
I	10	x x	ERROR 10: HARDWARE ERROR NO. XX
	11	x x	ERROR 11: INTERNAL ERROR MESSAGE NO. XX
II	30	0	ERROR 30: WAITING FOR SYNCHRON
	31	0	ERROR31 : WRONG CP MODULE
	32	0	ERROR 32: PG FUNCTION OPERATIVE
	33	0	ERROR 33: CP IS IN STOP: NO SLAVE SEND IS POSSIBLE
	34	0	ERROR 34: CP IS NOT IN THE STOP MODE
	35	0	ERROR 35: THE CP MAY NOT RUN: SWITCH IS SET TO STOP
III	50	0	ERROR 50: SYSID ERROR
	51	0	ERROR51 : POLLING LIST ERROR
	52	0	ERROR 52: INTERRUPT LIST ERROR
	53	x x x	ERROR 53: JOB XXX NOT DEFINED
	54	x x x	ERROR 54: JOB DESCRIPTION XXX NOT RECOGNIZED
	55	x x x	ERROR 55: JOB NO, XXX ONLY FOR RECEIVE
	56	x x x	ERROR 56: JOB NO. XXX ONLY FOR SEND
	57	x x x	ERROR 57: JOB NO. XXX ONLY COMPATIBLE WITH THE MASTER
	58	x x x	ERROR 58: JOB NO. XXX ONLY COMPATIBLE WITH THE SLAVE
	59	x x x	ERROR 59: JOB TOO LONG
	60	x x x	ERROR 60: SLAVE XXX NOT IN POLLING LIST
	61	x x x	ERROR61 :THE JOB NUMBER XXX USES AN UNDEFINED LIST
	62	x x x	ERROR 62: JOB NUMBER XXX SBR NOT ALLOWED
Iv	70	0	ERROR 70: BUS ERROR
	71	x x x	ERROR71 : CONNECTION TO SLAVE HAS AN INTERFERENCE
	72	x x x	ERROR 72: THE WRONG SLAVE (NO. XXX) HAS REPLIED
	73	x x x	ERROR 73: SLAVE NO. XXX HAS FAI LED

4. Appendix

4.1 COM 530 Error List

The following message texts appear in the message line; the numbers are internal references only and should not appear.

Error Ol: Illegal input

The COM 530 executes validity checks for the data entered: The cursor blinks in the field in which an illegal entry has been made.
- In the field only certain limit values are permissible
- In the field only certain alternatives are permissible
- For further details, see Operating Manual.

Error 02: Illegal key!

The key which has just been actuated is illegal in the field in which the cursor is blinking. The following are possible:
- Only digits and letters
- Only letters
- Only digits
- Only special characters
- Only digits and "+" and "-"
- Only digits and "+" and "-" and "."
- Only hexadecimal characters
- Only digits between O and 3
- Only digits O and 1

are permissible.

Error OB: Inhibited key!

Error OD: Programmer memory overflow!

Action: The programmer must be switched off with the power switch and switched on again and the COM 530 must be reloaded. If this error occurs frequently, the manufacturer must be contacted.

Error OE: System file not in drive O!

A system file required for correct execution of COM 530 is not in drive O. Action: Insert system diskette in drive O.

Error OF: Program load error!

A system file required for correct execution of COM 530 cannot be loaded. Action: Use original system diskette.

Error 10: Starting address Length of mailbox!

Error 11: Mailbox empty!

Error 12: INSERT/ERASE not possible here!

The cursor is in the last line and an attempt has been made to erase or insert data in the mailbox.

Error 13: Unknown data identifier!

The formats can be changed in the mailboxes. This error occurs if formats other than KH, KF, KS, KT, KC, KM, KY or KG are entered.

Error 15: Input missing!

Error 16: Data cannot be interpreted!

Possible in conversions from all formats to formats KT, KC, KS or KG.

Error 17: Beginning – No further!

If the mailbox is paged upwards and the first line is already displayed.

Error 18: End – No further

If the mailbox is paged downwards and the last line is already displayed.

Error 19: Do not press function key!

If the cursor is in the first field, i. e. where the editor is expecting an initial word address, no function key maybe pressed.

Error 1A: Repetition factor too high!

When a mailbox is edited, a repetition factor can be specified. If the entire length of the mailbox taking the repetition factor into account, is greater than 32 words, the editor outputs this error message.

Error 21: Source = Destination?

In "Transfer" mode, source and destination devices must not be identical. Under FLOPPY it is sufficient to distinguish by specifying the program name or drive labels.

Possibilities:

Destination / Source	DISKETTE	CP 530	EPROM
DISKETTE	+	+	+
CP 530	+	-	+
EPROM	+	+	–

+ permissible
– iillegal

DISKETTE User diskette or hard disk
CP 530 SINECL1 communications processor
EPROM Plug-in EPROM-/EEPROM submodule

Error 23: Transmission error

This error message indicates an error in all transmission types:

TRANSMIT PG ==> CP 530
– No transmission cable plugged in
– Transmission cable incorrectly plugged in
– Wrong transmission cable
– An EPROM submodule is inserted in the CP 530, all accesses forwriting ordeletion are not possible
– CP 530 not in operation
– Transmission time exceeded

PG ==> FD
Please contact manufacturer
– Hardware fault
– COM 530 system error

Error 24: Read SYSID identifier!

Error 25: SYSID, polling and interrupt lists non-existent!

In "Print all" mode, no data can be listed as no user data are stored in
– the CP 530 memory (ONLINE mode) or
– on the diskette under the program name specified (OFFLINE mode),

Error 26: No blank fields permitted!

Blank fields are not permitted in polling lists or interrupt lists.
The cursor is in the first vacant field found.
Action:
– Delete the respective field or
– Enter a slave number in the respective field.

Error 27: No double slave number in interrupt list!

Each slave may only appear once in an interrupt list.

4. Appendix

4.1 COM 530 Error List

Error 28: No EPROM/EEPROM submodule plugged in!
An EPROM/EEPROM submodule must be plugged into the receptacle for it on the programmer.
See SINEC L1 Operating Instructions.

Error 29: No program names for EPROM and CP 530!
"info" mode:
Where EPROM or CP 530 is specified as source, no information can be given for "All programs" (= all program names on diskette), as only **one** user program can be stored in the CP 530 or EPROM.
Source = CP 530 or EPROM:
Only " Individual program" mode is possible in which information is given on whether the SYSID identifier and/or polling list and/or interrupt list exist or not.

Error 2A: WARNING: No EPROM driver on the system diskette. No EPROM calls!!!
The COM 530 draws the user's attention to the fact that there is no EPROM driver on the system diskette being used.
WARNING: An attempt to program data into an EPROM or to read data from an EPROM causes a program crash. The "IN TERRUPTTRAP HALT" message appears in the form.
Action: Put original S5-DOS system diskette into a drive and start program once more.

Error2C: Left mailbox empty!
If the interface supplies a mailbox (left) with length O, this error message appears.

Error2D: Right mailbox empty!
If the interface supplies a mailbox (right) with length O, this error message appears.

Error 31: Polling list does not exist!

Error 32: Interrupt list does not exist!

Error 33: SYSID identifier does not exist!
These three error messages appear in
– output
– Delete
– Transmit
modes if the respective data type does not exist in the CP 530 or in the EPROM or on the user diskette under the program name specified.

Error 35: Illegal return message from PC!

Error 36: PC-Usart error!

Error 37: CP does not report!

Error 38: USART error on programmer side!

Error 39: USART error on CP side

Error 3A: Interface not ready!

Error3B: Abort by CP!

Error 3C: List non-existent!

Error3D: Diskette directory full!
The diskette directory of the user diskette or hard disk is full. A new file cannot be initialized.
Action: Use new diskette
Delete files not required

Error 3E: User diskette full!
The space available on the diskette or hard disk being used is insufficient for storing the file.
Action: Use another diskette
Delete files not required

Error 41: EPROM not erased!
The EPROM must be completely erased before it is programmed.

Error 42: VPP error!

The supply voltage level (for the EPROM) is not within the permissible range. This suggests a hardware fault on the EPROM or at the PG interface.
Action: Service or repair

Error 43: Comparison error!

After bus parameters have been programmed into an EPROM from diskette or the CP 530, the contents of the EPROM are compared with the data on the diskette or CP 530.
if the data are not identical, this error message is output.
Action: Erase EPROM and reprogram

Error 45: Address out of range!

Error 46: Incomplete input!

In the
– output
– Transmit
– Print
– Info
– Delete
modes, a program name and drive identifier must be given for the source diskette (the cursor is in the PROGRAM NAME field).

Error 47: Slave failed!

Error 48: Wrong mode!

Error 49: Slave not in polling list!

Error 4A: Slave already receiving data!

Error 54: EPROM/EEPROMsubmodule type illegal!

The wrong EPROM/EEPROMsubmodule is being used.

See SINECL1 Operating Instructions

Error 56: Mode change, CP stopped

Error 57: Error in slave cycle

Error 58: Slave xx failed!

Error 59: Slave xx not on bus!

In both of these error messages, the number of the respective slave is entered in the output fields.

Error 60: Only hexadecimal characters permitted!

Error 61: –32.768 fixed-point +32.767!

Error 62: FORMAT: aaa,bbb;aaa,bbb only up to 255!

Error 63: ASCII characters only!

Error 64: FORMAT: aaa, b; b ...0 to 3!

Error 65: Wrong counter word!

Error 66: Only keys O and 1 permissible!

Error 67: FLOATING-POINT SYNTAX: + 1234567–89!

Error 70: Incorrect CP module!

Error 71: Error when programming the EEPROM on the CP!

4. Appendix

4.1 COM 530 Error List

Error 72: It is not possible to set the CP to RUN since the Stop/Run switch is at STOP!

Error 73: CP 530 is a slave

Error 75: Slave failed

Error 77: Programmer data has been overwritten.

Error 78: Slave not in polling list

The following error messages may appear after accessing a drive (diskette or hard disk):

Drive not defined
Is there a diskette in the drive selected?

Defective external memory
Defective drive or diskette. Is the diskette correctly formatted?

External memory write-protected
Read-only access to selected drive.

File write-protected
Read-only access to the selected program

External memory full
No free memory space available on the selected diskette or hard disk. This message is also displayed when the directory is full.

List does not exist
The selected list (SYSID, polling list, interrupt list) is not available in the selected program.

SIEMENS AKTIENGESELLSCHAFT

Bestell-Nr.: GWA4NEB 811 0730-02a
Printed in the Federal Republic of Germany
AG 07871.0 FL 60 en

Siemens AG
Gerätewerk Amberg, TDI
Postfach 1954
Werner-von-Siemens-Straße 48-52

D-8450 Amberg
Federal Republic of Germany

Sender (Please fill out)

Name

Company/Department

Address

Telephone

Suggestions: Corrections:

Manual **SINEC LI, 5th Edition (6ES5998-7LA21)**

Have you found any typographical errors while reading this manual? Please use this form to tell us about them.
We would also welcome any ideas and suggestions you may have.