SIEMENS

SIMATIC S5

Special Driver for CP 524 / CP 525-2 (S5-DOS)

OPEN DRIVER

for

Data Link to Host Systems

Operating Instructions

Order No.: 6ES5 897-2DC21 Version: 03

1. Introduction

The development of the "open driver" data link software package for communications processors CP524/CP525-2 makes it possible to establish a data link between host computers with simple transmission procedures and SIMATIC S5 U-series controllers (115U, 135U, 150U, 155U).

Transmission is asynchronous, half or full duplex and is carried out on an RS232 (= V.24), current loop (TTY 20 mA) or RS422 (for CP524 only) interface. Control signals of the V24 interface are not evaluated with this "open driver".

Using the "Open Driver" it is possible to send and receive data of any structure (all printable ASCII characters as well as all other characters from 00H to FFH).

By means of parameter assignment it is possible to support an XON/XOFF protocol in send direction. This is used by many printers to control the data flow.

Operation character by character with echo can also be parameterized.

The S5 user can keep the message structure open by transfer of the complete send message with all control characters (including possible start or end characters) to the CP. In the event of data reception the S5 user receives the complete received message. The structure of the send messages may differ from the structure of the received messages.

Transmission in both directions is carried out without handshake (without linking structure, without acknowledgements).

1.1 Message End in Send Messages

The length is determined by the "QLAE" parameter of FB SEND. It is possible to output an uneven amount of bytes from word-organized areas (e.g. data blocks) using the dummy "FFH".

1.2 Message End in Receive Messages

The criterion to recognize the end of a receive message can be parameterized as follows:

- The **length** of the receive message is always the same:
  Parameter assignment: **end criterion = fixed message length**
  specify length in bytes (max. 1024)

- There are **certain end characters** at the end of the message:
  This identifier (1 or 2 characters) must not appear in the user data (e.g. end character ETX or CR, LF when transmitting in ASCII format).
  Parameter assignment: **end criterion = one or two end characters**
  specify code of the end character(s)

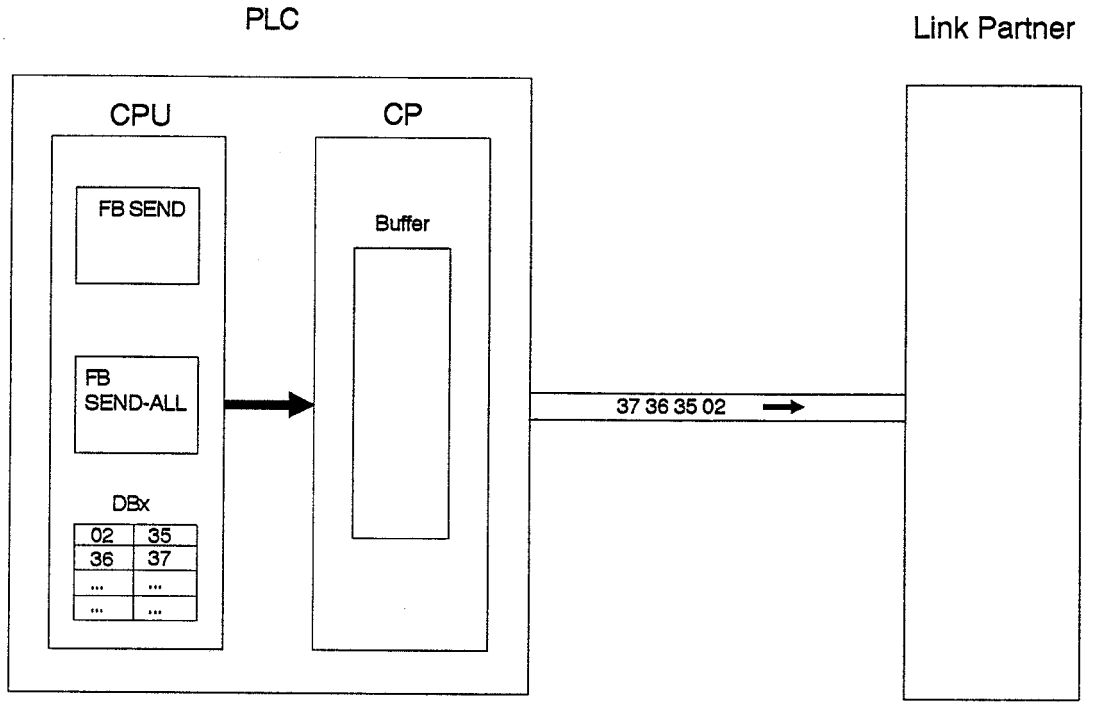
- If the message end is specified neither by a fixed length nor any control characters, it can be recognized by means of the **character delay time having elapsed**. In this instance the character delay time must clearly be shorter than the pause time between two messages.
  Parameter assignment: **End criterion = character delay time**

When parameterizing with end identifier or fixed length the character delay time ZVZ also serves the purpose of monitoring reception of the complete message. This time must not be exceeded between receiving a character and receiving the next character.

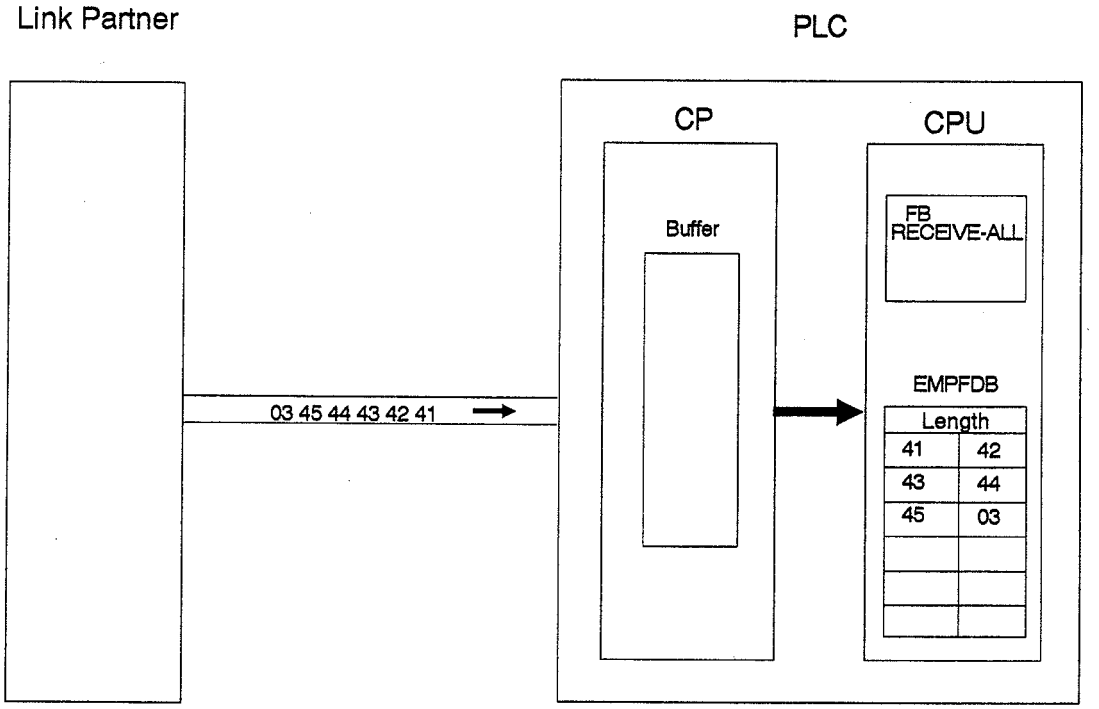
After recognizing the end criterion the complete message is transferred to the CPU. In addition to this the S5 user is advised of **the amount of received bytes in the first data word**. The data isentered by means of FB RECEIVE ALL which is why the receive data block must already have been stipulated during parameter assignment. Furthermore it is possible to specify by parameterization from which data word the entry is to be started.

The end criteria "character delay time" and "end characters" can also be linked to the indication of a part block length (max. 1024). Should this part block length be exceeded in a message, the user may receive one or two **part messages** (see 5.2.7 "TELLEN"). This makes it possible to work with small receive data blocks or with AS215 (Teleperm M with 128 byte blocks). A further advantage is that **there is no limit to the maximum message length**. If no part block length is specified, the maximum message length is 1024 byte including all control characters. The **minimum message length** is **1 byte**.

Transmission

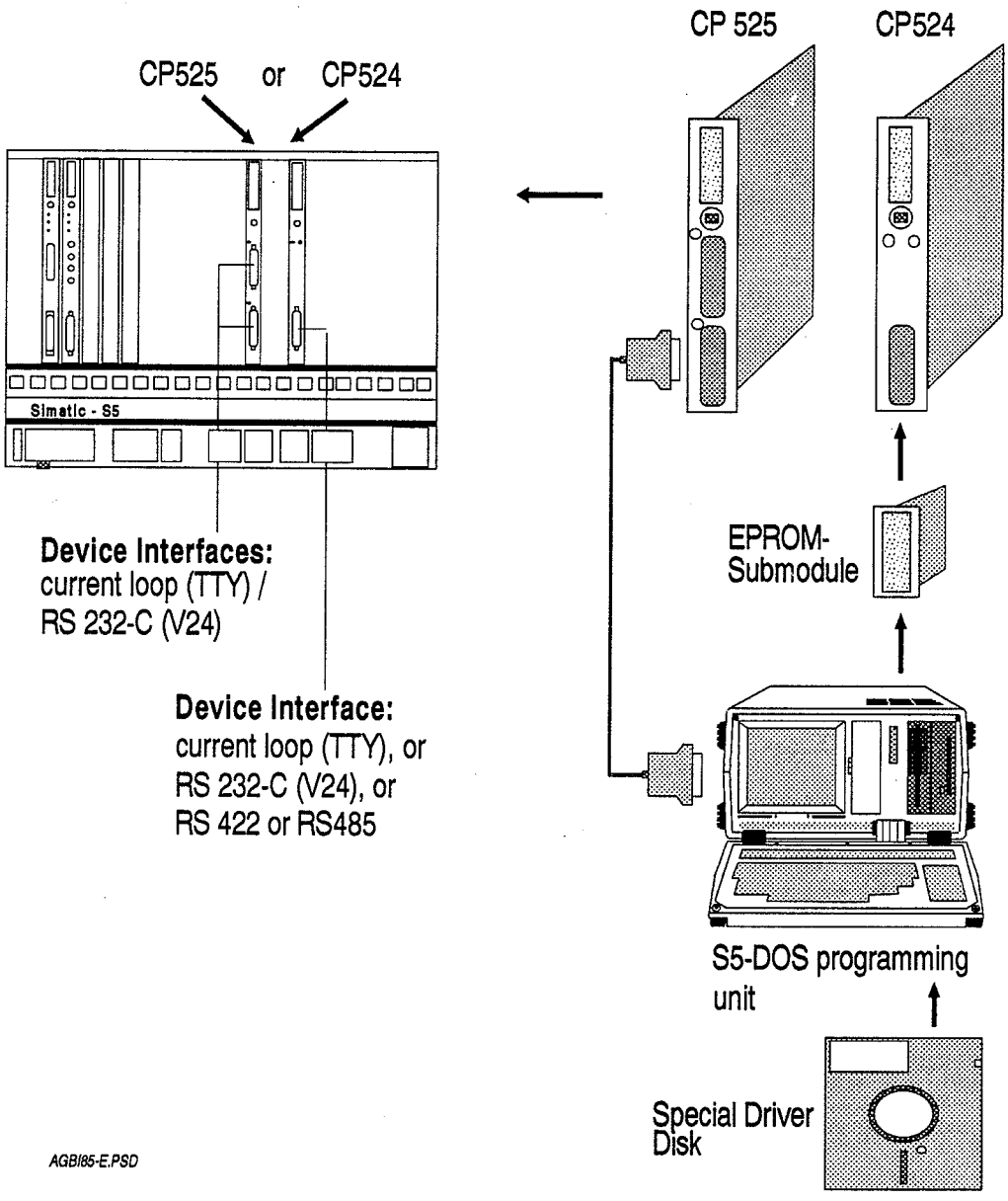


Reception



The supplied special driver disk can be processed by an S5-DOS programming unit. The driver (interpreter and procedure) is transferred from the directory to the user program using the COM525 software. An EPROM module can be programmed in OFF-line operation using the PROM525. This EPROM module can be used in CP524 or CP525. CP525 has two fixed device interfaces, which may be operated as a V24 (RS232-C) or TTY (current loop) interface. The CP524 can be operated with TTY (current loop), V.24 (RS232-C) or RS422 interface modules.

Possible System Configuration:



2. Installation of Special Driver on PG

The library "COMLIBD4.525" is located on the supplied floppy disk which contains the special driver "S5D004" consisting of interpreter and procedure.

I-S5D004 ⇨ interpreter
P-S5D004 ⇨ procedure

The installation of the driver is carried out on a programming unit (e.g. PG685) preceded by the installation of the STEP5 base package as well as COM525 programming software.

Starting from Drive B: user level 0 the library "COMLIBD4.525" is copied on to the winchester drive user level 0 as follows:

PIP B:=A:COMLIBD4.525[R]

Option R = copy check

To ensure the use of the special driver on all user levels, it is recommended to convert the library into a write protected system file by entering the following command:

SET COMLIBD4.525[SYS RO].

2.1 CP User Programs

Programs for CP525-2 and CP524 can only be created and processed using the new COM525 (S5-DOS Version).

The CP525 module with part number 6ES5 525 3UA11 (CP/M86 Version) cannot be programmed using the S5-DOS-COM525; programs which were created using COM525 (CP/M86 version) cannot be processed by S5-DOS-COM525.

A conversion of the programs is not possible.

3. Creation of CP User Program

After the copying procedure the Simatic programming packages must be called by entering

S5

followed by placing the cursor into the "**COM525...**" line and selection of the COM525 programming software by using function key F1 "PACKAGE".

The COM525 basic mask appears on the screen. By using F1 **"SELECT PROGRAM"** the next step is to branch into the "PROGRAM SELECTION" mask.
After specifying hard disk **"B"** as the drive and inputting the **program name** enter component **"CL"** which stands for computer link.

3.1 Copy Procedure from Library to User Program

In order to copy from the library to the user program, call the **"TRANSFER" from FD to FD** function by using F1-F2-F5.

The "TRANSFER" mask must be filled in as follows: Source is drive "B" as well as the library name **"COMLIBD4"**. The user program chosen in the "PROGRAM SELECTION" mask is automatically entered as the destination.

3.1.1 Transfer of Interpreter

Use function key F3 in order to obtain the INTERPRETER transfer mask followed by F7 which superimposes the interpreter to be transferred "I-S5D004" (component "CL") into this mask.

Function key F1 starts the transfer; the transfer end is indicated by the following message: "MESS.002": Completed!.

F8 "EXIT" leads to the return into the "TRANSFER" mask.

3.1.2 Transfer of Procedure

Use function key F4 to obtain the PROCEDURE transfer mask.

HELP function F7 superimposes the procedure **"P-S5D004"** to be transferred, F1 initiates the transfer.

After successful completion of the transfer, return to the "SELECTION" mask by pressing F8 "EXIT" twice.

3.2 Parameter Assignment of Procedure

The **procedure parameters** set via the "PARAMETER ASSIGNMENT OF PROCEDURE" mask **are not evaluated**. Setting is carried out via the Initialisation-Send (see Para. 5).

3.3 Creation of Job Blocks

Creation of Job Blocks is not required for the Special Driver.

4. Loading of Special Driver into CP

The loading procedure of the special driver S5D004 is identical to the procedure used for standard computer link CL512.

Transfer of the user program to CP525 and/or programming of an EPROM module is described in detail in

COM525 Manual Volume 2
Register 4 (Issue 06)
Paragraph 3.2.2

5. Parameter Assignment of Special Driver

5.1 Initial Start of Special Driver

After power failure, warm or cold re-start of the central controller, the CP waits for the **"SYNCHRON"** handling function which initializes the interface between CPU and CP. This means that the "SYNCHRON" handling function must be called in the relevant organization block. During the waiting time the interface specific **LED blinks three times** in 200 ms intervals.

After initializing the CPU-CP interface, the interface specific **LED blinks twice**; the special driver must now be initialized by triggering the special job "INIT"

The special driver must also be initialized again using the "INIT" special job after a cold re-start of the CP is carried out with a PG.

5.2 Special Job "INIT"

A "SEND-DIREKT" job with **A-Nr = 189** is interpreted by the CP as the special job "INIT". The "QTYP" parameter (= type of data source) must be defaulted with "DB". Defaulting the "QLAE" parameter (= length) is irrelevant, because the special driver reads out always a quantity of **eight** data words from the specified source data area.

A cyclically called SEND-ALL is required to process the "INIT"-Send-Direkt!

As mentioned before, it is essential to activate the special job "INIT" after power failure, warm or cold re-start of the CP. It is also possible to trigger "INIT" during a running operation. In this instance the procedure is not initialized until any reception running at that time is compleleted.

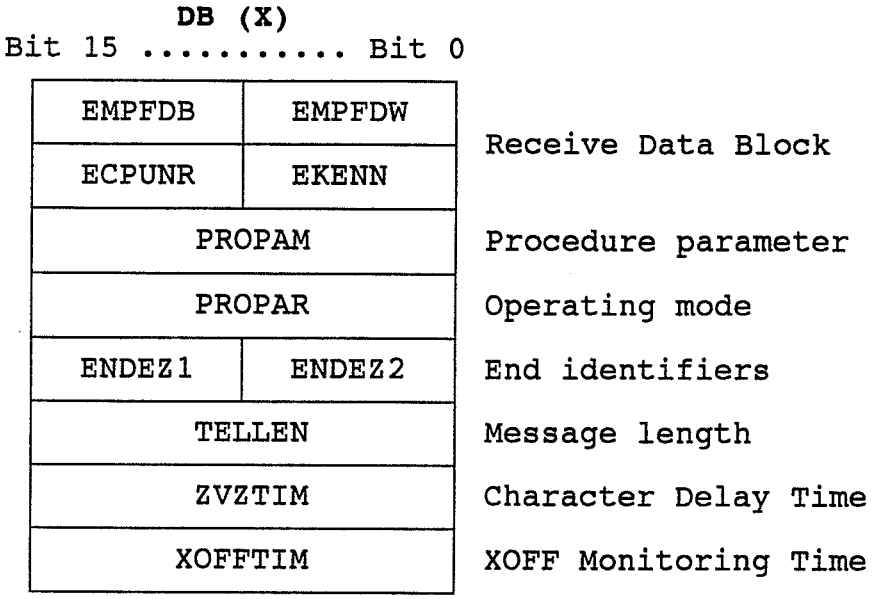
After the special driver has **recongnised the "INIT job**, it turns on the interface specific **LED. If no errors** were recognized during evaluation of the transferred Init-Data Block, the CP **switches off the LED** and returns to normal operation.

If the special driver **recognizes an error** during evaluation of the transferred parameters, the Init job is completed with error, and the driver waits for a new "INIT" job; in this case the specific **LED blinks four times.**

The error must be corrected in the S5 program or in the INIT DB in accordance with the error number entered in SYSTAT, and the INIT job must be called again.

5.2.1 Structure of the Initialisation Data Block

The Init Data Block transferred during triggering of the "INIT" special job contains all procedure specific characteristics as well as the Receive Data Block. It must be defaulted as follows:



5.2.2 Receive Data Block (Parameter EMPFDB/EMPFDW)

The Receive Data Block indicates the destination for received messages in the PLC. All received characters, including start and end characters, are entered here. The **CP enters the length** of the received data string in bytes into the specified **first data word**. Input of net data commences with the second data word.

If the received data string is shorter than the word length, the driver completes the remaining data RIGHT HAND SIDE DR using the dummy character 0FFH.

The parameter "**EMPFDB**" indicates the number of the receive data block. Numbers smaller 3 are not permitted.

The byte "**EMPFDW**" indicates the data word from which input of data is to be made.

Example:

EMPFDB

	LENGTH = 6 Byte		
EMPFDW n			--> input by driver
EMPFDW n+1	02	30	
EMPFDW n+2	31	32	nett data
EMPFDW n+3	33	03	

5.2.3 CPU Number, RECEIVE Identifier (ECPUNR/EKENN)

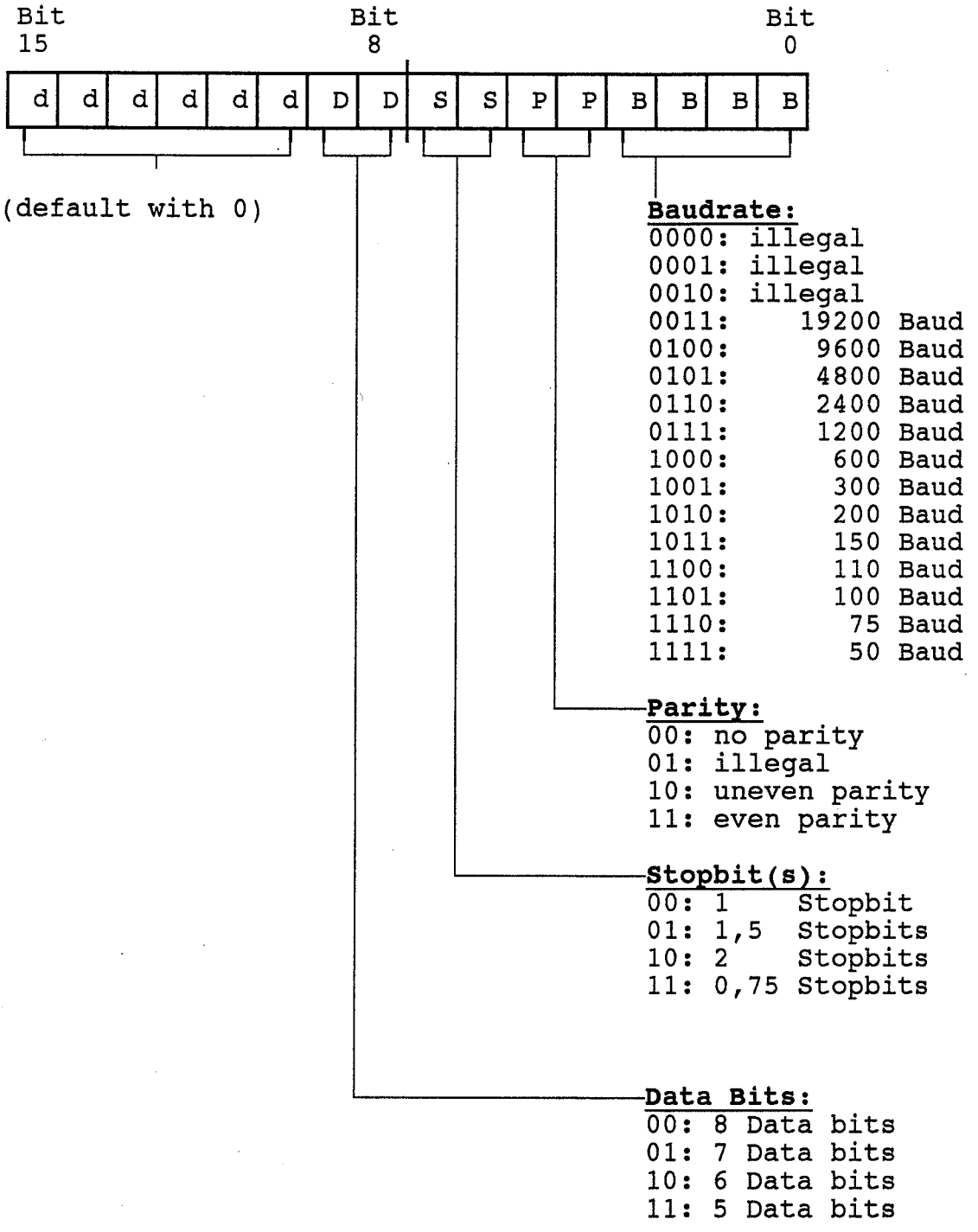
The parameter "**ECPUNR**" specifies the CPU number for which the received messages are intended. When working with single processor operation, 0 must be entered here, for multi processor operation the CPU number (1-4). If 0 is entered when working with multi processor operation, each CPU is ready to take data.

The "**EKENN**" field contains the parameter for an identifier (instead of an order number). After completed data transfer from the CP, this number is entered into the **Condition Code Word of the RECEIVE ALL** (see Para. 10.1.4 "Handling Function RECEIVE ALL). Permissible values are 1...223.

If the message from the link partner consists of several part messages which are blocked by TELLEN, the entry into the Condition Code Word for the Receive All is "EKENN+1" for the part messages, and "EKENN" for the last part message (please see also Para. 5.2.5.5 "End Criterion for Reception" and 5.2.7 "Message Length").

5.2.4 Procedure Parameter (PROPAM)

Structure of Data Word "PROPAM"



The Baud Rate is set in bits 0 to 3 (B). The bit combinations are defaulted as follows: -

0011 : 19200 Baud
0100 : 9600 Baud
0101 : 4800 Baud
0110 : 2400 Baud
0111 : 1200 Baud
1000 : 600 Baud
1001 : 300 Baud
1010 : 200 Baud
1011 : 150 Baud
1100 : 110 Baud
1101 : 100 Baud
1110 : 75 Baud
1111 : 50 Baud

The total Baud Rate of CP525 as well the maximum transmission speed of CP524 is 19200 Baud. Please also refer to Para. 5.4 "Maximum Baudrate".

Bits 4 and 5 (P) indicate the parity. The possibilities are as follows: -

00: no parity
10: odd parity
11: even parity

Bits 6 and 7 (S) indicate the amount of Stop Bits: -

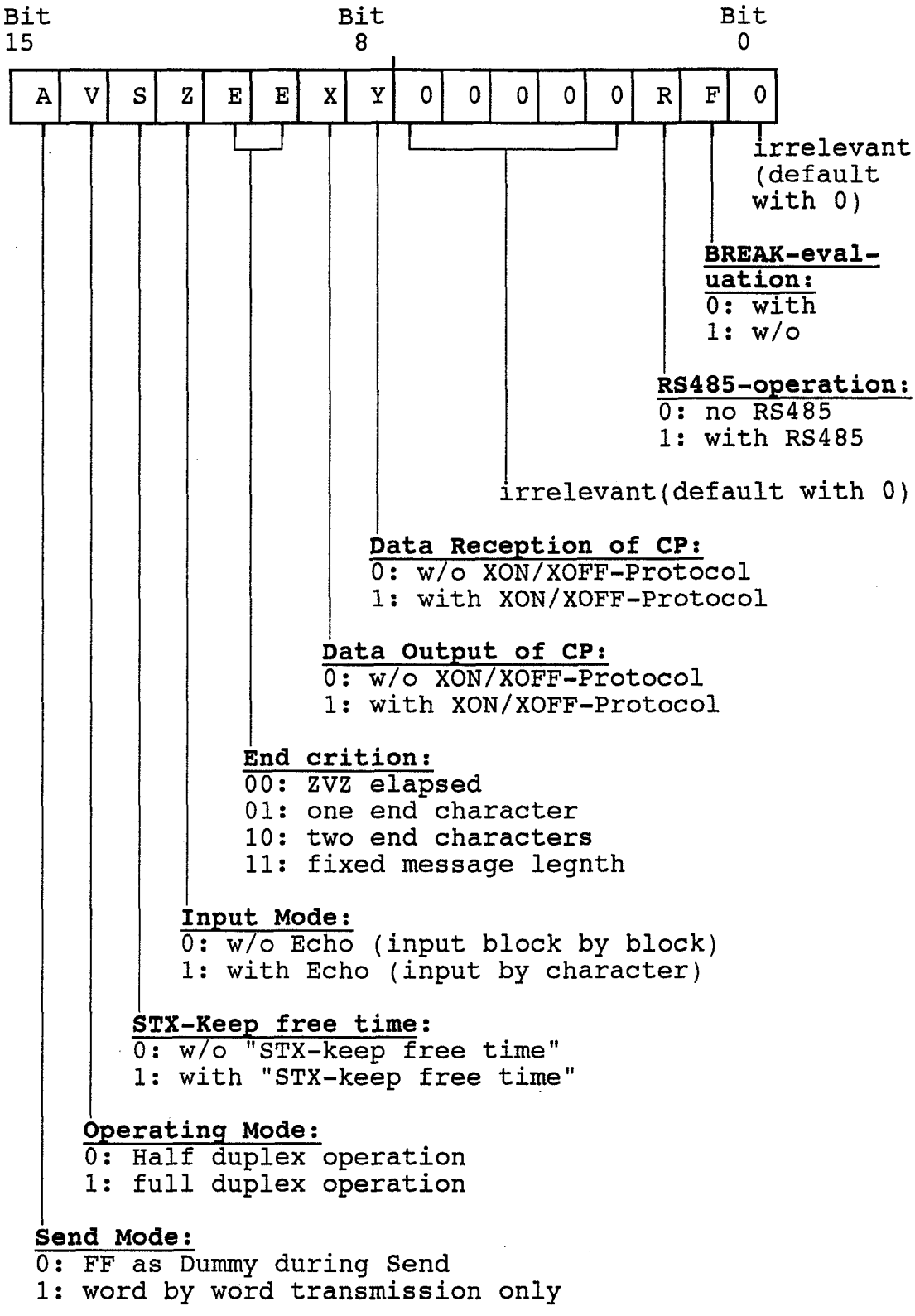
00: 1 Stop Bit
01: 1.5 Stop Bits
10: 2 Stop Bits
11: 0.75 Stop Bits

Bits 8 and 9 (D) define the amount of Data Bits: -

00: 8 Data Bits
01: 7 Data Bits
10: 6 Data Bits
11: 5 Data Bits

5.2.5 Operating Mode (PROPAR)

Structure of Data Word "PROPAR":



5.2.5.1 BREAK Evaluation (Bit 1)

It is recommended to normally operate with "BREAK" regocnition Bit 1 (F) = 0.

Bit 1 = 0: If BREAK occurs on the receiving line when a send job is being triggered, it is rejected by the CP with an error message.

There are certain exceptions, such as no receive line available, when it makes sense to switch off the BREAK **evaluation in the Condition Code word**:
Bit 1 = 1: In this operating mode, the "BREAK" status is ignored when a "SEND DIREKT" job is being triggered.
Parameter assignment does not influence the BREAK bit in SYSTAT. This bit can always be read with RECEIVE-221.

When **RS485 operation** is in use, the **BREAK** recognition must always be **turned off**, i.e. Bit 1 = 1.

5.2.5.2 RS485 Operation

When Bit 2 (R) = 1 is assigned, RS485 operation is in use, all other interfaces (V.24, TTY, RS422) must operate with parameter assignment Bit 2 = 0.

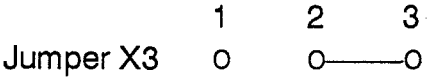
Parameter assignment with 1 is only possible with CP524 **(from Version 03)** and when the RS422/RS485 interface module is being used. It is permitted only when the specified operating mode is half duplex (software), **parameter assignment** is without XON/XOFF protocol, without keep free time and **without BREAK evaluation.**

Default for PROPAR: X000 XX00 0000 0110B (Bits 10 and 11 only, Bit 15 free selection).

Due to the fact that there is only one 2-wire line which is being switched over for transmission and reception, it is important to avoid simultaneous transmission by both partners (e.g. Master-Slave principle).

When using the RS485 interface, it is possible to connect up to 32 slaves to one network in half duplex operation.

For this purpose, **jumper X3** must be set as follows: -



When jumpers X10 and X11 are inserted, line R(A) (Pin 4) and R(B) (Pin 11) are preset via 1.2 kOhm resistors: -

R(A) - pull up +5V

R(B) - pull down 0V

A 120 Ohm terminating resistor must be fitted at the end of the network.

The special driver checks whether a CP524 with RS422/RS485 is in operation. The user must ensure by jumper setting that the RS422/RS485 hardware will run in half duplex mode.

There is only one hardware line which is switched over between transmission and receive operation.

BREAK evaluation is not possible (neither in ANZW nor in SYSTAT) in this operating mode.

5.2.5.3 XON/XOFF Protocol for Data Reception by CP (Bit 8)

When Bit 8 (X) = 1, the **XON/XOFF protocol is transmitted** (i.e. CP controls input of the link partner).

This operating mode is not possible with full duplex operation.

If the driver notices during a running input that it has only one free buffer, it sends <XOFF>. After that up to 124 further characters of the **same** message can be received.

As soon as another buffer becomes available through a RECEIVE ALL, the driver sends <XON>. After transmission of <XON> the driver expects the input to be continued within **twice the character delay time.**

5.2.5.4 XON/XOFF Protocol for Data Ouput by the CP (Bit 9)

When Bit 9 (X) = 1, reception of a XON/XOFF protocol is possible, i.e. control of the SEND job is carried by out by the link partner.

The XON/XOFF protocol is not allowed with full duplex operation.

If the <XOFF> character is received by the CP during a running output, transmission is interrupted. For hardware specific reasons it is possible to receive up to two further characters after the <XOFF> character.

The remaining data of the current data string is output when the <XON> character is transmitted by the link partner within the parameterized monitoring time XOFFTIM (Para. 5.2.9 "XOFF Monitoring Time").

If this monitoring time is exceeded, transmission is completed with an error message in the Condition Code word of the Send Job, and an error entry in the SYSTAT area (see also Para. 7.1 "Reception during Output in Half Duplex Operation). Any data still resident in the buffer are lost.

After reception of <XOFF> in neutral, the next SEND job is not output until <XON> is received, or until the monitoring time has elapsed.

5.2.5.5 End Criterion for Reception (Bit 10 and 11)

Bits 10 and 11 (E) determine the criterion for the end of a message.

Bit 11	Bit 10		
0	0	:	character delay time elapsed
0	1	:	1 end identifier
1	0	:	2 end identifiers
1	1	:	fixed message length

The ZVZTIM parameter (Para. 5.2.8 "Character Delay Time") must be specified with **each end criterion**.

The **end criteria "character delay time" and "end identifiers"** can also be linked to the indication of a **part block length in "TELLEN"** (Para. 5.2.7 "Message Length"). Both on recognition of the parameterized end criterion and on reaching the part block length is the data received up to that point transferred to the CPU.

If the **end identifiers** are being used, they must be entered in the ENDEZ1, ENDEZ2 fields (Para. 5.2.6 "End Identifiers").

If "**fixed message length**" was selected, the exact length of the receive messages must be entered in the "TELLEN" field (Para. 5.2.7 "Message Length").

5.2.5.6 Echo (Bit 12)

Bit 12 (Z) determines the transmission format of input messages. The standard setting (Bit 12 = 0) means that no echo is used (input block by block). If Bit 12 = 1, each character received from the CP is echo'ed, i.e. sent back to the link partner (input character by character).
Echo is only possible with half duplex operation.

5.2.5.7 Keep Free Time after <STX> Reception (Bit 13)

As a rule it is recommended to work without Keep Free Time Bit 13 (S) - 0.

Bit 13 (S) = 1 means that the Keep Free Time is activated after reception of <STX>, If this is received by the driver during a running output, it is understood by the CP as an input request from the partner. After output of the data string the CP waits for input from the link partner for the keep free time period of **FHZ = 60 s**. Send jobs are postponed by the CP until after reception of a message, or, if there is no message, until after the keep free time has elapsed. Half duplex operation is required here.

5.2.5.8 Full Duplex Operation/Half Duplex Operation (Bit 14)

Bit 14 (V) serves to differentiate between full and half duplex operation. In addition to the hardware related difference (full duplex: seperate transmission and receive lines, half duplex: RS485: only one line which is switched over for transmission and reception) there is also a software related difference:

BIT 14= 0: **Half Duplex Operation**: At one point in time the CP can either transmit or receive a message. The special driver can, however, still recognize individual characters on the receiving line (not RS485) (e.g. XON or XOFF, see Para. 7.1 "Reception during Output in Half Duplex Operation").

BIT 14 = 1: **Full Duplex Operation**: At one point in time the CP can transmit and receive a message. In full duplex operation, parameter assignment must be without echo, without XON/XOFF operation, and without RS485 operation. PROPAR for full duplex: X100 XX00 0000 00X0B - free selection only of bits 1, 10 and 11 and 15. **In full duplex operation, both link partners must be able to operate the transmission and receive lines independently from one another**. This means that the link partner must be able to process reception of a message from the CP whilst transmitting a message at the same time.

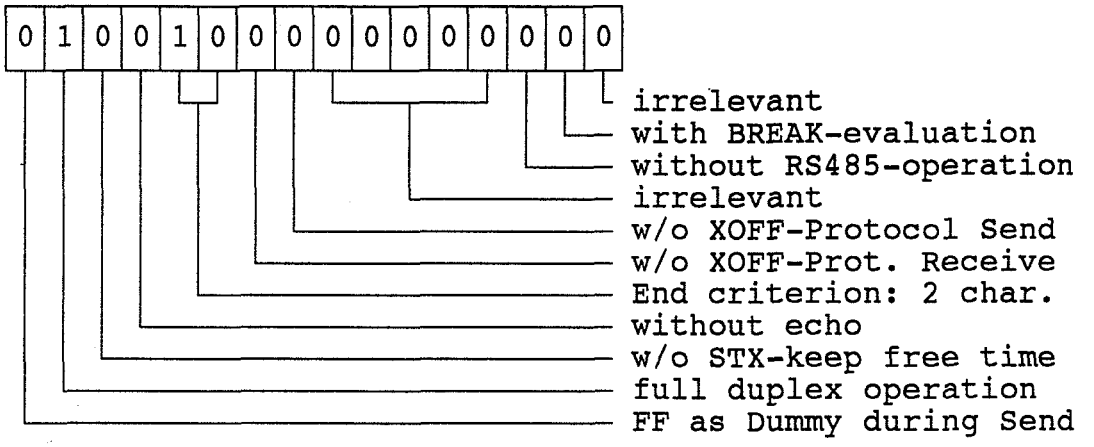
5.2.5.9 Transmission Word By Word - Dummy OFFH (Bit 15)

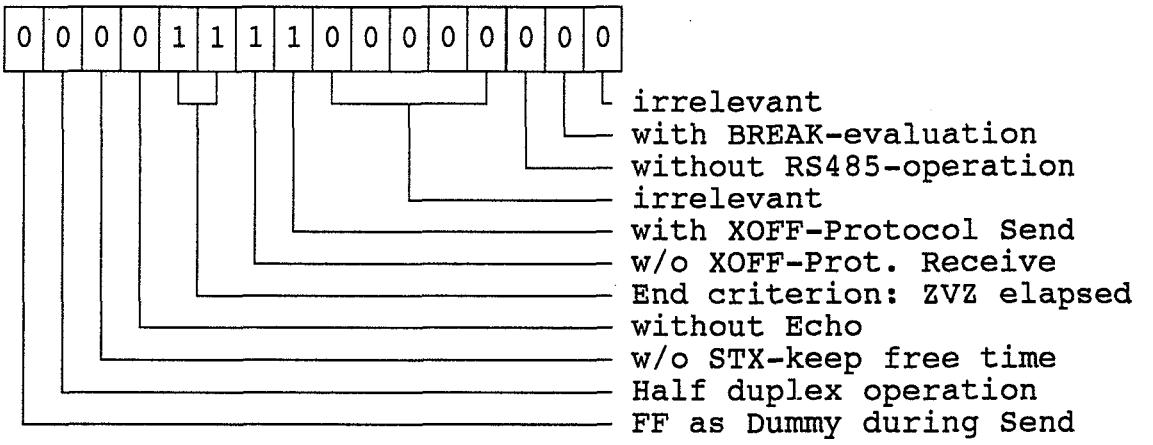
Bit 15 (A) serves to differentiate between operation with or without DUMMY character (OFFH).

BIT 15 = 0: If the **last character** is **0FFH** is a word organized SD5 source area (data block), it is interpreted as a DUMMY and is **not transmitted to the link partner**. (An uneven number of data bytes can be transmitted).

BIT 15 = 1: All data from the S5 source area, including OFFH as the last character, is transmitted to the link partner. (This means that only an even amount of data bytes can be output from a word organized area).

5.2.5.10 Examples for PROPAR





5.2.6 End Identifiers (ENDEZ1, ENDEZ2)

The HEX codes of the end identifiers must be entered into the **"ENDEZ1"** and **"ENDEZ2"** fields, providing the end criterion "with 1 or 2 end identifiers" was parameterized. If one end identifier is used, only "ENDEZ1" is evaluated.

5.2.7 Message Length (TELLEN)

If **"fixed message length"** was selected as the **end criterion**, the lengths of the receive messages must be entered in the **"TELLEN"** field. The lengths of the messages must tally exactly with the lengths specified there. Lengths of 1 to 1024 bytes are possible.

The indication of a **part block length in "TELLEN"** (max. 1024) can be linked to the **end criteria "character delay time" and "end characters"**.
Both on recognition of the parameterized end criterion and on reaching the part block length is the data received up to that point transferred to the CPU.
If this part block length is exceeded in a message, the user may receive one or several **part messages**. The driver carries on receiving until the end criterion, and then transfers the remainder of the message. This makes it possible to receive long messages even when working with small receive data blocks or with AS215 (Teleperm M with 128 byte blocks). A further advantage is that there is no limit to the maximum message length. **If Value 0 is specified in "TELLEN" operation is without part blocks**. The maximum message length is in this instance, 1024 byte including all control characters.

5.2.8 Character Delay Time (ZVZTIM)

The character delay time (ZVZ) can be parameterized using the word **"ZVZTIM"** in intervals of milliseconds. The smallest possible value to be selected is 4 milliseconds, the highest is 65.535 seconds.

The character delay time is the maximum time period which may elapse between reception of two characters.

Incoming characters within a message are monitored by means of this parameterized character delay time. If end characters were parameterized, and the time elapses, the CP aborts communication with error (-> ZVZ must be greater than the time between two characters (= min. character output time).

Reception errors are not reported until after the character delay time has elapsed, or after correct reception of the parameterized end characters.

When the end criterion is ZVZ, **the message end** is recognized when the character delay time has elapsed. ZVZ must safely elapse between two messages.

The character delay time must also be parameterized when end characters are being used.

The length of the ZVZ depends on the baud rate and on the speed of the link partner. In the event of flush transmission of the individual characters by the link partner, the following nominal values may apply: -

9600 Baud: 5 ms ... 100 ms
4800 Baud: 10 ms ... 200 ms
1200 Baud: 30 ms ... 800 ms
110 Baud:200ms ... 3 s

5.2.9 XOFF Monitoring Time XOFFTIM

The XOFF monitoring time XOFFTIM must be specified only when parameter assignment was "with XON/XOFF Protocol for Data Output by the CP". The time can be set in 1 ms intervals from 4 ms to 65.535s.

5.3 Parameter Assignment Errors

If an error is recognized during the Initialisation Send (e.g. incorrect bit combination in PROPAM, error during transfer of Init-DB), the interface specific LED on the CP blinks four times and the driver fails to start. The Send Job is completed with error and an appropriate error code is stored in the SYSTAT area.

When the LED blinks four times, the SYSTAT must be read and in accordance with the error number (see Para. 9.4.3 and 9.4.4) the error must be corrected in the INIT-DB or in the S5 program, and the INIT-SEND must be called again.

5.4 Total Baud Rate

The **total Baud Rate** of CP524 as well as the maximum transmission speed of CP524 is **19200 Baud.**

The "total baud rate" for the CP525 is calculated as the sum of baudrate of the first interface plus baud rate of the second interface.

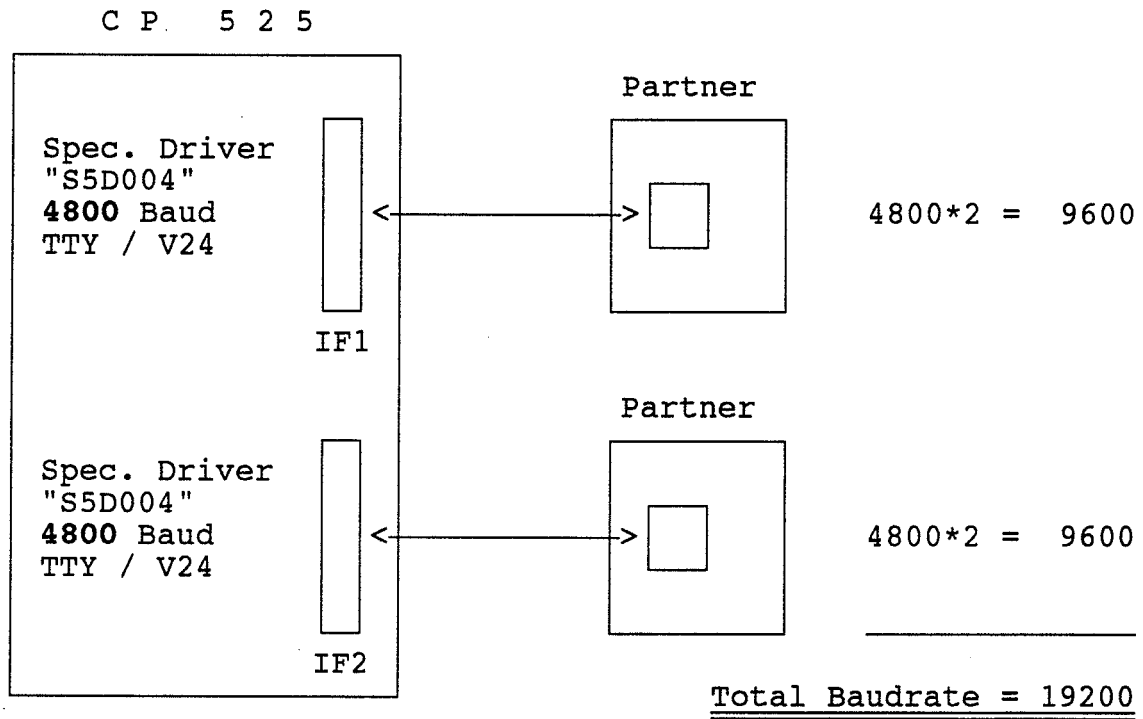
The above calculation is dependent on the operating mode in use "HALF DUPLEX/ FULL DUPLEX". In FULL DUPLEX operation (= special driver can transmit **and** receive messages at a given point in time) the set baud rate must be **multiplied by 2.**

When the **TTY interface** is in use, please note that the "parameterized baud rate must not exceed **9600 Baud.**

Examples: Total Baudrate

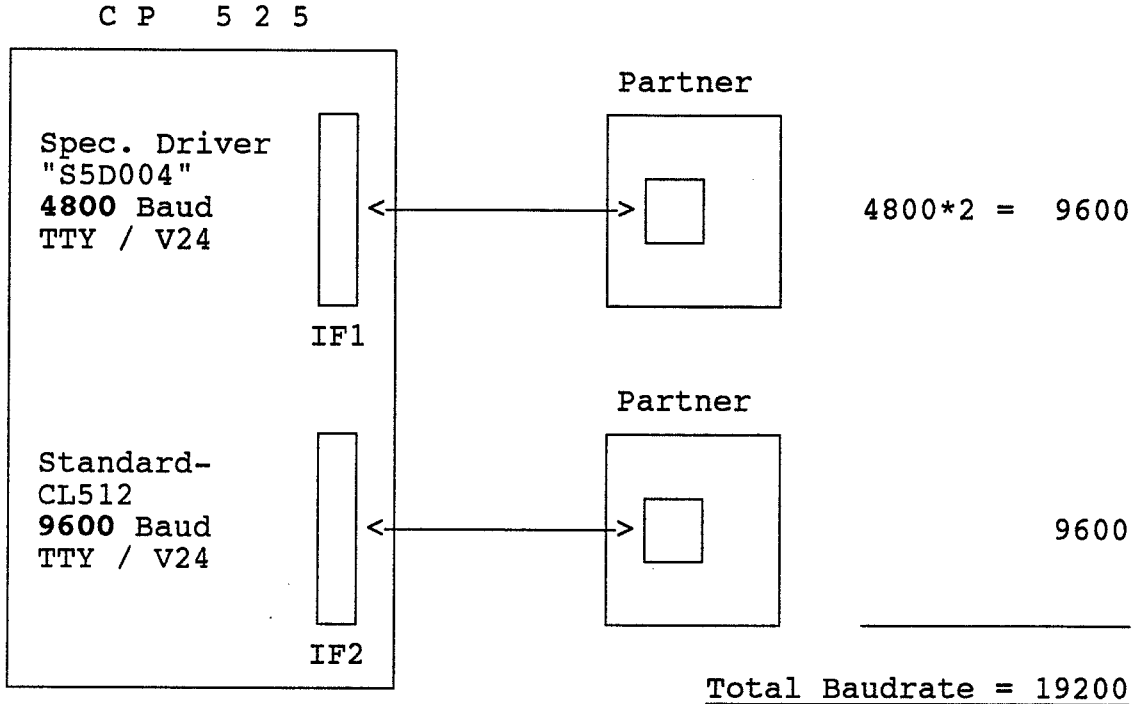
System Configuration:

Operating Mode:
FULL DUPLEX with two interfaces



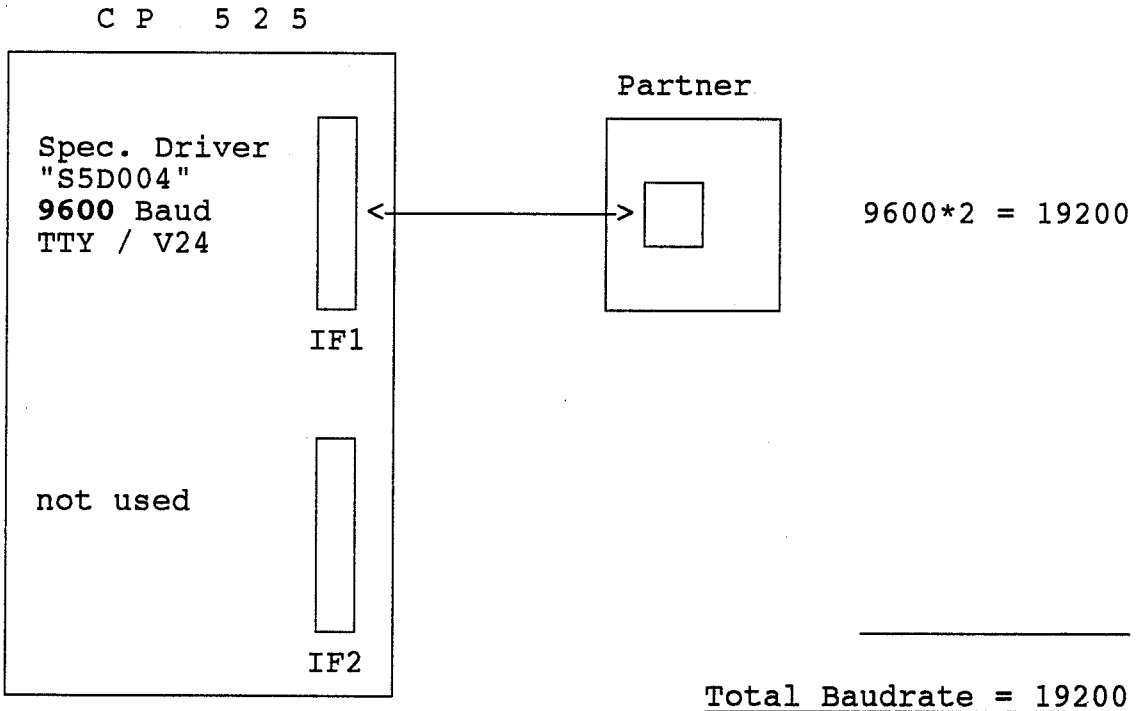
System Configuration:

Operating Mode:
1st Interface FULL DUPLEX
2nd Interface HALF DUPLEX



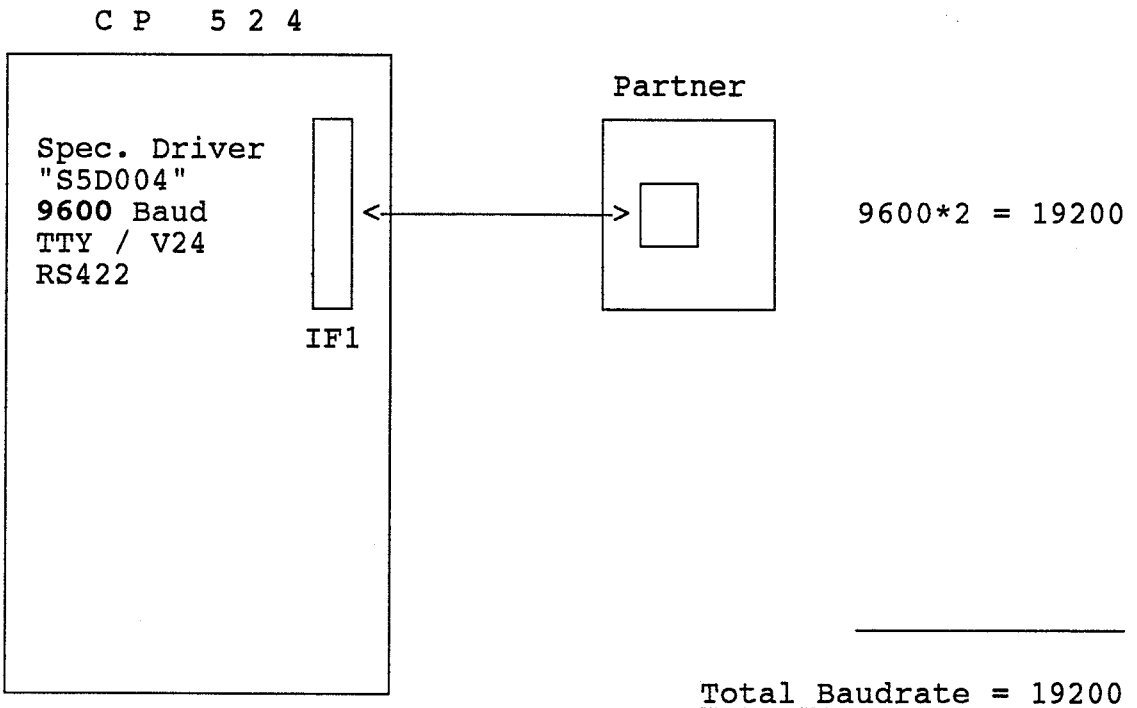
System Configuration:

Operating Mode:
One interface FULL DUPLEX



System Configuration:

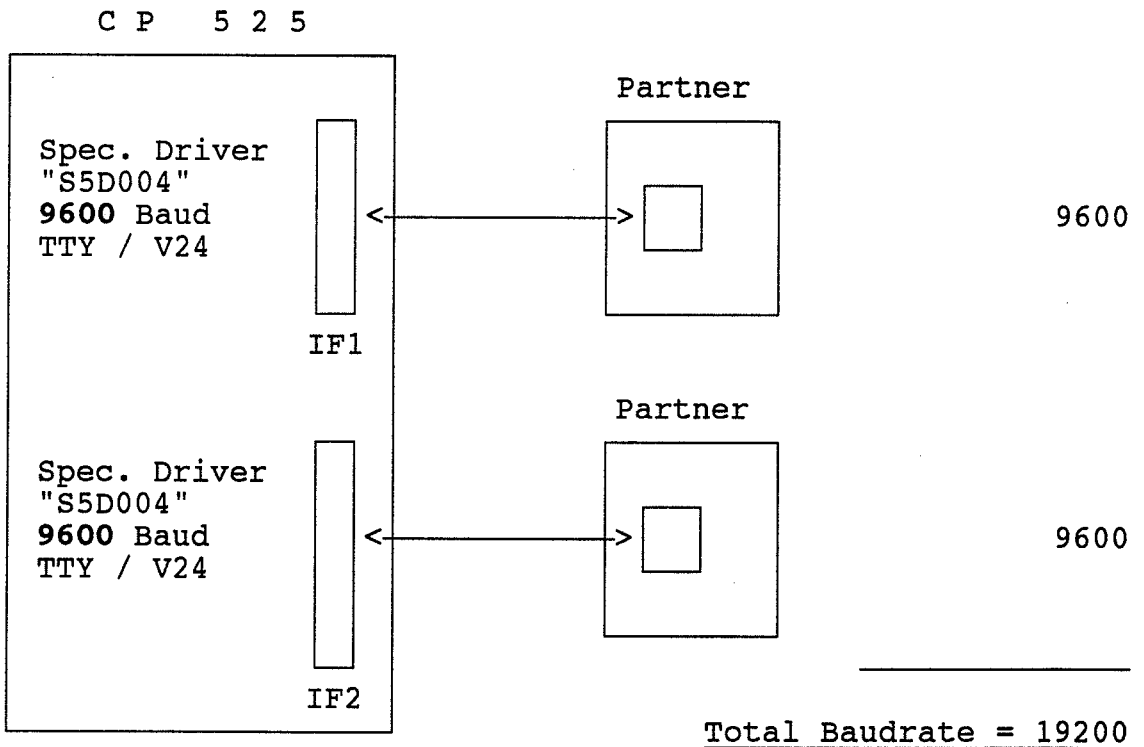
Operating Mode:
One interface FULL DUPLEX



Total Baudrate = 19200

System Configuration:

Operating Mode:
Two interfaces HALF DUPLEX



Total Baudrate = 19200

6. Transmission Procedure

Transmission is asynchronous, half or full duplex.

Data of any structure (all printable ASCII characters as well as all other characters from 00H to FFH) can be transmitted and received by the "open driver".

The code transparency of the procedure depends on which end criterion has been selected: -

- with one or two end characters ⇨ not code transparent

- end criterion: ZVZ or fixed length ⇨ code transparent

Code transparent means that any combination of characters (values from 00 to FF) may appear in the user data without recognising the end criterion.

If operation is with end character, transmission is not code transparent, and the end character(s) must be excluded from the user data. In order to avoid this the user data can be ASCII coded in the STEP 5 program, or the first end character can be duplicated.

Transmission is in both directions **without handshake** (without establishing the connection, without acknowledgements), and use of XON/XOFF protocol (transmission on/off) is optional.

6.1 Data Length

6.1.1 Data Length in Receive Direction

The maximum length of data blocks to be received, including end characters, is **1024 byte**. The CP stores the length in the first data word of the receive data block which means up to 1026 byte are entered into the data block.

If the **length** of the message to be received **exceeds 1024 byte**, it must be divided into **part blocks**. Blocking is carried out via the parameter "TELLEN" (see 5.2.7 "Message Length"). The message is then entered into the receive data block by means of several Receive All calls. Input of the individual data blocks is always made from the parameterized receive data word which means that the data in the receive data block must be protected accordingly.

6.1.2 Data Length in Send Direction

All data types may be used as the source.

The parameter QLAE (= source length) on the handling block is an amount of bytes provided the source area is organized in bytes (if not, it is an amount of words).

The following table lists all transferable data types with their parameter assignment possibilities on handling block "SEND DIREKT".

Details regarding the addresses depend on the CPU and do not always correspond when different types of CPU are used.

Particularly for absolute addresses it is strongly recommended to refer to additional CPU specific documentation.

Source, Send from CPU	Parameter Assignment on HDB in CPU			
	QTYP	DBNR	QANF	QLAE
Data Block	DB	3-255	0-2047	1-2048
Exp. D.Block	DX	3-255	0-2047	1-2048
DB 155U	DB	3-255	0-4090	1-4091
DX 155U	DX	3-255	0-4090	1-4091
Flags	MB	irrel.	0-255	1-256
Inputs	EB	irrel.	0-127	1-128
Outputs	AB	irrel.	0-127	1-128
Counters 115U	ZB	irrel.	0-127	1-128
Counters 135U	ZB	irrel.	0-127	1-128
Counters 150U	ZB	irrel.	0-255	1-256
Counters 155U	ZB	irrel.	0-255	1-256
Timers 115U	TB	irrel.	0-127	1-128
Timers 135U	TB	irrel.	0-127	1-128
Timers 150U	TB	irrel.	0-255	1-256
Timers 155U	TB	irrel.	0-255	1-256
Peripherals	PY	irrel.	0-255	1-256
Sy-Addr.135U	BS	irrel.	0-255	1-256
Sy-Addr.150U	BS	irrel.	0-511	1-512
Sy-Addr.155U	BS	irrel.	0-255	1-256
Absol.-Addr—	AS	irrel.	0-+32767 -32767	1-32767
exp. periph. applies to 150U only	QB	irrel.	0-255	1-256

7. Output of Data by CP

Data output is activated by the user by triggering handling block "SEND DIREKT" (see Para. 10.1.2).

The parameters may be assigned according to the table listed in para. 6.1.2.

When entering the data to be sent into the S5 source data area please note that the interpretation by the CP of the OFFH character as the last byte varies depending on parameter assignment of "PROPAR" (see Para. 5.2.5).

The CP does not add any special start and/or end identifiers to the data string transferred from the CPU to CP.

If "BREAK" occurs on the receiving line when triggering a send job and when operating mode "with BREAK evaluation" is in use, the CP rejects the triggered send job with an error message in ANZW.

The "BREAK" status is ignored , when a "SEND DIREKT" job is triggered with operating mode "without BREAK evaluation".

7.1 Reception during Output with Half Duplex Operation

Characters which are received during a running output, are ignored by the driver.

When operating mode "with XON/XOFF protocol during output" is in use, the special driver evaluates reception during a running output as follows: -
The driver only ignores characters unequal <XON> (=11H) or <XOFF> (=13H). If the <XOFF> character is received during a running output, transmission is interrupted. For hardware specific reasons, output of up to two more characters is possible after receiving the <XOFF> character. Providing the link partner transmits the <XON> character within the monitoring time "XOFFTIM", output takes place of the remaining data of the current data string.

If the monitoring time "XOFFTIM" is exceeded, transmission is completed with an error message in the condition code word of the Send Job and an error entry in the SYSTAT area.

If "with Keep Free time" is in operation, the <STX> character is evaluated. The <STX> character is understood by the CP as an input request from the partner. After output of the data string the CP waits for an input from the link partner for the time period called Keep Free time FHZ = 60 s. Send Jobs are not processed by the CP until after reception of an input message, or if no reception has taken place, after the Keep Free time has elapsed.

If "BREAK" occurs on the receiving line during output, the CP aborts communication and completes the current send job with an error message in the Condition Code Word, unless BREAK recognition is disabled by parameter assignment.

7.2 Example: Output Message

DB(x) DL	DB(x) DR	CP	
02		02H	Data String
	30	30H	
31		31H	
	32	32H	
03		03H	
	FF		

8. Input of Data by the Link Partner

An input message may commence with any character. The end of the received data string is recognized by the CP in accordance with the parameterized end criterion. The data is entered into the parameterized receive data block by means of a RECEIVE ALL.

The incoming characters are monitored with the parameterized **character delay time.** After this time has elapsed, and providing parameterized end identifiers or a fixed message legnth are used at the same time, the CP aborts communication with an error message in the SYSTAT area.

If "BREAK" is received on the receiving line during reception of a data string, the CP also aborts the connection with an error message.

If the CP recognizes transmission errors in a character, or if the data string exceeds 1024 bytes (applies only when parameter assignment is without part blocks see "TELLEN"), the driver **rejects** the message, and enters an error message into the SYSTAT area after recognizing the message end (= character delay time ZVZ elapsed or reception of the parameterized end sequence).

The CP operates internally with 16 buffers of 128 byte each. This means that several small messages which are received in very quick succession, can be buffered in the CP. The RECEIVE ALL, however, must be called as often as possible, in order to avoid the error "no input buffer free", or transmission of XOFF (see Para, 5.2.5.7).

8.1 Example: Receive Message

Link Partner		DB(x) DL	DB(x) DR	
		00	05	Data Length = 5 Byte
02H	Data String	02		
30H			30	
31H		31		
34H			34	
03H		03	FF	

9. Error Handling

Errors occuring during operation of the special driver can be caused anywhere in the program.

Once the CP has recognized the errors, the CPU is advised as accurately as possible. The user determines what reaction by the CPU is required by programming the STEP5 user program on the CPU accordingly. For instance the user can have the job repeated or a flag set.

Errors are indicated by:

- the light emitting diode **LED** on the CP

- an error number in the **PAFE**-byte of the handling block

- an error number in the condition code word **ANZW** of the handling block

- an error number in error message area **SYSTAT**

Follow the above sequence when trying to locate an error. You will be able to interpret and correct the error by means of the error descriptions listed on the following pages.

9.1 Error Messages on LED's

The CP has a red light emitting diode for each interface which indicates the CPU status and driver status during the start-up phase. For CP524, the left hand LED indicates software errors, the right hand LED indicates severe hardware errors after start-up.

The LED lights up when the driver has not yet been loaded on the interface, when no cold re-start has been carried out after loading, or when the switch on the CP is in the STOP/PGR position.

After the CP has completed the start-up phase, the driver status is indicated by a blinking action of the LED in 200 ms time intervals.

Amount of Blinking Actions	Meaning
twice	CP waiting for INIT-SEND
three times	CP waiting for SYNCHRON
four times	error with INIT-SEND; ⇨ read SYSTAT, correct error CP waiting for new INIT-SEND

After start-up of the driver has been completed without error, the LED is cancelled. Message traffic can now proceed.

If a hardware error is recognized during the start-up phase, the LED is not cancelled. In this instance, and as long as the SYNCHRON has not been processed correctly, the SYSTAT entry can only be read via system commands or "enquiry address" on the PG.

9.2 Error Numbers in PAFE-Byte of Handling Block

The PAFE-byte is a parameter which is specified by the user when calling a handling block. The handling blocks indicate when substantial errors in connection with the CP have occured or parameters have been specified incorrectly. A detailed description of PAFE errors can be found in

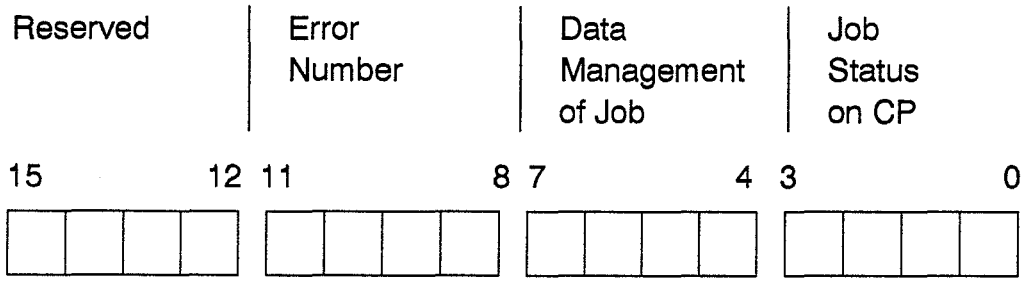
COM525 Manual Volume 2
Register 7 (Issue 06)
Para. 2.7.1.

9.3 Error Numbers in Condition Code Word ANZW

9.3.1 Assignment of Condition Code Word of Handling Block "SEND DIREKT"

Each job number of a "SEND" job defined in the PLC processor is allocated an individual condition code word (parameter ANZW). Using this condition code word it is possible to monitor processing of a job by the PLC processor and the CP.

Structure of the Condition Code Word



For the handling of bits 0-3, 4-7 and 12-15 please refer to the general description of the condition code word.

Meaning of Bits 0-3 for Job Status Display

Bit 0:		irrelevant for SEND Direkt
Bit 1:	job running	further jobs with the same A-NR are disabled by the CP ⇨ reset trigger flag
Bit 2:	job complete without error	data was sent to the link partner
Bit 3:	job complete with error	data could not be sent to the link partner ⇨ evaluate error number in bits 8 - 11 ; read SYSTAT

If an S5 output message could not be processed correctly, **the reason for the error is displayed in bits 8-11 of ANZW by means of an error number.**

IMPORTANT !
An error number is entered into the ANZW only when the error occured in connection with the processing of a Send-Direkt.

Each Send-Direkt completed with error is also entered into the SYSTAT area where the error situation is specified in more detail.

The possible error messages are explained on the following pages.

9.3.2 Error Numbers in ANZW of "SEND DIREKT"

Error Number Bit 8-11	Reason for Error
1...5	parameter assignment errors recongnized by CPU reported to CP
6	errors in data traffic between CPU and CP recognized by CP
7	job cannot be processed, errors in job parameters
C	switch in STOP/PGR position for job
E	monitoring time "X)FFTIM" elapsed after receiving <XOFF>
F	BREAK on receiving line

For a detailed description of error numbers 1...7 please refer to

COM525 Manual Volume 1
Register 7 (Issue 6)
Para. 8

9.4 Error Numbers in SYSTAT Area

The error message area SYSTAT is a data area in the dual port RAM which may be read by the CPU by means of the special jobs **RECEIVE DIREKT 200** or **RECEIVE DIREKT 221**.

The error message area of SYSTAT comprises three error message bytes and one additional status byte for each interface.

For a detailed description please refer to

COM525 Manual Volume 1
Register 7 (Issue 6)
Para. 7.

All errors recognized by the CP are entered into the error message area of SYSTAT.
If the error in question is related to a "SEND DIREKT" job, an additional error number is entered into the condition code word.

For a description of the general error numbers 1-28, 2E, 4E, 4F, 53 and B0-D0 please refer to the COM525 manual. The other error numbers listed in the COM525 manual are not used by the special driver.

The error entries in SYSTAT are made in such detail that they are of particular use for exact error analysis during the commissioning phase.
In the event of any problems with the data link during normal operation the SYSTAT should always be consulted for reading and evaluation.

9.4.1 Error Codes in ANZW and SYSTAT for SEND-Jobs

ANZW	SYSTAT	Reason for Error
E	64H	monitoring time elapsed after reception of <XOFF>

9.4.2 Error Codes in SYSTAT for Receive Errors

SYSTAT	Reason for Error
65H	monitoring time elapsed after receipt of <X0FF> in neutral
70H	transmission error occurred in a character
71H	no input buffer was available
72H	received data string was greater than 1024 byte (applies for parameter assigment without TELLEN)
73H	character delay time exceeded
E3H	reception during transmission
FFH	BREAK during reception of a message

9.4.3 Error Codes in SYSTAT when Processing an INIT Send (A-NR.189)

ANZW	SYSTAT	Reason for Error
7	80H	A-NR is not 189 INIT-Send expected, but Send-Direkt with a job number unequal 189 recognized.
7	81H	Job type is not "SEND"
7	82H	Source Data type is not "DB"
7	83H	Send-Direkt disabled by user by Bit 7 in ANZW = 1
7	84H	Monitoring time elapsed for handshake on CP (e.g. SEND-ALL not within 3 seconds)
7	85H	Error display of CPU to CP, which cannot be interpreted by CP illegal reply by PLC (evaluate PAFE! use latest version of HDB
7	86H	source data block does not exist and/or. is illegal (DBNR: KY0,x)
7	87H	Source data block too short
7	88H	PLC reports PAFE 41: no access to area possible (use latest version of HDB's
7	89H	PLC reports PAFE 51: incorrect condition code word: group message for all errors relating to the condition code word (MW 0..196 recommended; DW 0..255 permitted, open DB)
7	8AH	current job was interrupted during cold re-start of CP or cold re-start of CP by PG CP in STOP

9.4.4 Error Codes in SYSTAT when Evaluating an INIT-DB

ANZW	SYSTAT	Reason for Error
7	8BH	Illegal Baudrate in PROPAM
7	8CH	Illegal Parity in PROPAM
7	8DH	PROPAR: full duplex only possible, without echo, w/o XON/XOFF-Protocol and w/o STX-keep free time
7	8EH	Character delay time "ZVZTIM" smaller 4ms
7	8FH	Number of Receive Data Block "EMPFDB" smaller 3
7	90H	CPU-Nummer greater 4 "ECPUNR"
7	91H	Illegal value for "EKENN" (1 to 223 permitted)
7	92H	Illegal value in "TELLEN" (1 to 1024 permitted)
7	93H	PROPAR: RS485-operation only possible, without echo, w/o XON/XOFF-Protocol, half duplex, w/o keep free time after STX and w/o BREAK-recognition
7	94H	RS485-operatiron, but no CP524 with RS485/RS422-sub-module
7	95H	"XOFFTIM" smaller 4 ms

10. Application Example

10.1 Correct Handling of Handling Blocks

10.1.1 Handling Function "SYNCHRON"

The "SYNCHRON" handling block synchronizes the interface between CPU and CP for a cold start (OB20), a manual re-start (OB21) or an automatic re-start after power failure (OB22).

"SYNCHRON" must be called for each interface of the CP in the initial start organization blocks of the CPU.

During synchronization the maximum blocking size for data transfer between CPU and CP is selected.

Parameter BLGR: KYx,y

Range of Values:	x = 0	
	y = 0 :	blocking size 256 byte
	y = 1 :	blocking size 16 byte
	y = 2 :	blocking size 32 byte
	y = 3 :	blocking size 64 byte
	y = 4 :	blocking size 128 byte
	y = 5 :	blocking size 256 byte
	y = 6 :	blocking size 256 byte*
	y = 7 to 254:	blocking size 256 byte
	y = 255:	blocking size 256 byte*

* In accordance with the Handling Block Operating Manuals a blocking size of 512 bytes is used for settings 0.6 and 0.255. However, the maximum blocking size carried out by the CP is 256.

ATTENTION:
When using **115U** PLC values 0.6 and 0.255 must not be used because in the event of message lengths > 256 bytes data might be overwritten.

10.1.2 Handling Function "SEND DIREKT"

"SEND DIREKT" with a job number 1...223 (exception: special jobs) initiates data output to the link partner.

The "SEND DIREKT" must be called with RLO=1. If the handling block is called with RLO=0, the only function carried out is "CONTROL" (up-dating of ANZW).

Any data type is permitted as the source. The maximum send length depends on the source type (see Para. 6.1.2 Data Length in Send Direction).
If "DB" is used as the source type, transmission is possible from any data block (3-255).
The maximum send length from a DB is 2048 bytes.

After calling a "SEND DIREKT" job, it is entered into the CP internal queue which may receive up to **ten** "SEND DIREKT" jobs. This means that up to ten different send jobs may run "simultaneously".

If there are already ten jobs queueing, each further job is rejected with "completed with error".

The handling block "SEND ALL" is required for the processing of each "SEND DIREKT" job.

10.1.3 Handling Function "SEND ALL"

When a "SEND DIREKT" job is triggered, the "SEND ALL" job (= parameter job number = 0) **carries out data exchange between CPU and CP.**

After recognizing a "SEND DIREKT" job, the CP temporarily stores the source parameters specified on the handling block, and sends a "SEND ALL" request to the CPU.

"SEND ALL" then transfers the requested data from the CPU source data area into the dual port RAM of the CP; the special driver then takes over data output to the partner.

For larger amounts of data, several "SEND ALL" calls may be necessary, depending on the blocking size, to transfer the data from the S5 source data area into the dual port RAM.

10.1.4 Handling Function "RECEIVE ALL"

In order to ensure that the **received messages** are entered into the data block in the central controller, the handling block "RECEIVE" must be cyclically called in the user program.

In this instance the "job number" parameter must be defaulted with **"0" (= RECEIVE ALL)**. The parameters "ZTYP", "DBNR", "ZANF", "ZLAE" on the handling block RECEIVE with job number "0" are irrelevant, because the destination information is preset by the special driver.

The CP waits for a "RECEIVE ALL" for a **maximum time period of 3 seconds**, after this time has elapsed, an error number is entered in the SYSTAT area.

The entry into the data block is displayed to the S5 user program by means of the condition code word of handling block "RECEIVE ALL". When the handling block is running idle (no data exists on CP), the used ANZW is "0000H". After entering the complete data block the parameterized job number **"EKENN"** is dynamically superimposed into the condition code word.

If the data block to be transferred is larger than the blocking size, several "RECEIVE ALL" functions are necessary to transfer the data from the CP into the S5 destination data area. In this instance, for a partial input the identifier "00FFH" is entered in the condition code word. The identifier "EKENN" is superimposed only after the last "RECEIVE ALL" of this data block.

If during parameter assignment, **end criterion ZVZ or end identifier was combined with TELLEN**, the message from the link partner may consist of several part messages. In this instance, two different identifiers are entered into the Condition Code word. The **last part message** is transferred with the parameterized job number **"EKENN"**. All previous **part messages** are transferred with **"EKENN+1"**.

10.1.5 Special Functions

The description for the special jobs for

- reading of error message area SYSTAT
- deleting of error message area SYSTAT
- reading of entire SYSTAT
- reading of SYSID
- reading/writing of date and time

may be found in

COM525 Manual Volume 1
Register 7 (Issue 6)
Para. 7.

The pseudo READ/WRITE function (job numbers 190...199) has not been realized for the special driver "S5D004", because dynamic presetting of CPU source parameters can be realized by means of indirect parameter assignment of "QTYP", ""DBNR", "QANF" and "QLAE".

10.2 Numbers of Handling Blocks

Function		Central Processing Unit 115U	135U R-Proc. 155U	150U
SEND	✔	FB 244	FB 120	FB 180
RECEIVE	✔	FB 245	FB 121	FB 181
FETCH		FB 246	FB 122	FB 182
RESET	✔	FB 248	FB 124	FB 183
CONTROL	✔	FB 247	FB 123	FB 184
SYNCHRON	✔	FB 249	FB 125	FB 185
SEN-A	✔		FB 126	
REC-A	✔		FB 127	

When used with special driver "S5D004" the handling blocks marked with "✔" must be used in the S5 program in the form previously described.

10.3 Demonstration Program

> The demonstration program is an example only, and not to be understood as a solution for customer specific system configurations.

In addition to the special driver COMLIBD4.525 the floppy disk contains the following files: PLC155ST.S5D, PLC150ST.S5D, PLC135ST.S5D, PLC115ST.S5D with an sample program for the individual PLC's and the CP525 or CP524 user program DEMOPROG.525.

The demonstration programs themselves have not been printed out in this Manual, but they can be printed out.

The comments in the demonstration program are available in German (AG...UST.S5D), English (PLC...ST.S5D) and French (API...ST.S5D).

10.3.1 Commissioning

Hardware Configuration

In order to work the demonstration program, the following hardware is required:

- S5-DOS programming unit (e.g. PG685) with the STEP5-package (version 3.0 or higher)
- PLC with CPU (115U, 135U, 150U or 155U)
- a CP525 with RAM- or EPROM-sub-module, or two CP524's with EPROM-sub-modules and interface sub-modules
- cable connector CP-CP e.g. 6ES5 726-1BD20 (TTY) or 6ES5 726-8BD20 (V.24)
- cable connector PLC-PG685 6ES5 731-1...0 or PLC-PG7XX 6ES5 734-2...0.
- cable connector PG685-CP 6ES5 726-0...0 or PG7XX-CP 6ES5 734-5...0, if a RAM-sub-module is used

Setting up of PG

Set up your programming unit. If you intend to work in a User area other than 0 within the PCP/M-86 operating system, this must be set up here.

Transfer of the program on to the hard disk is carried out by means of the PCP/M-command **PIP B:=A:*.*[RV]**.

Enter this command and acknowledge with the **Return-key**. Option [RV] ensures that also the SYS-files are being copied followed by a copy check. After the copy procedure has been completed without error, remove the floppy disk from the floppy drive.

Commissioning of CP

In the description for the demonstration program you will find the CP interfaces are referred to with their physical addresses, i.e. for the CP525 the top interface is IF 0, the bottom one is IF 1.

The user program is parameterized for interface numbers 0 and 1.

- Use the jumper settings on your CP(s) to set interface numbers SSNR=0, and SSNR=1.
- Insert a RAM or EPROM sub-module with 16kByte memory (this is sufficient) and the interface sub-module for CP524.
- The mode selector switch of the CP must be in the STOP position.

Commissioning of CPU

- Insert the CPU and CP into their appropriate positions in the PLC rack.
- After connecting the power supply, carry out an overall reset on the CPU as a precaution, and set the mode selector switch to the STOP position.
- Load the handling blocks (numbers see 10.2) into your PLC

Loading of COM-driver into the CP

The PG-CP cable connector carries out data transmission from PG to CP. Insert the plugs with the appropriate labels into the PLC interface of the PG and the bottom interface of the CP.

If an EPROM sub-module is used, the CP user program with the PROM525 package is loaded directly into the memory module, the PG-CP connection is

not required. A detailed description on the programming package PROM525 can be found in the COM525 manual Volume 2, Para, 6.

Select the S5-DOS command interpreter on your PG using the **S5** command.

The PG offers a number of STEP5 packages via the PACKAGE SELECTION MASK. The one to be selected is **COM525 Programming Package to CP**. After the selection has been carried out, the BASIC MASK of COM 525.appears on the screen.

Now use function key **F1** (select program).

Use function key **F7** to enter the program name *DEMOPROG* in the PROGRAM SELECTION MASK, the COMPONENT appears as *CL* which stands for Computer Link. Accept this setting using **F1**.

This is followed by the SELECTION mask where the driver can be loaded immediately by using **F2**. However, prior to loading the driver you have the additional option to delete any other interfaces which may have been loaded previously by using **F3.**

After using **F2** (Transfer), use **F3** to specify the copy direction (FD -> CP). The first step is transfer of the entire program (i.e. interpreter and procedure) for interface number **1**(= top SS) by using function key **F2**. After transfer has been completed without error, it is concluded with the message *MESS.002: Completed!* Now start the interface using **F1**, the *MESS.002: Completed!* message appears again.

The same method applies to the bottom interface: First select interface number **2**, transfer the entire program using **F2** and start it using **F1**. Do not disconnect the cable connector on the CP!

You can now leave the COM package: **F8** (Exit) -> **F8** (Exit) -> **F8** (Exit) -> **F1** (end program) -> **F8** (Exit), and return to the PACKAGE SELECTION mask.

- In the event of error messages during the copy procedure, it may possibly be necessary to carry out a reset of the interfaces and to repeat the transfer
- Also check the mode selector switch on the CP (which should still be in the STOP position) as well as the cable connectors
- CP-errors are also indicated on the Light Emitting Diodes on the CP (see Para. 9.1)
- Move the Mode Selector Switch on the CP to the RUN position
- Disconnect the cable connector to the PG.
- Use it to link the two CP interfaces together
- The LED's on the CP are flashing three times in an alternating rhythm - the CP is waiting for synchronisation from the PLC

Transfer of PLC Program

Choose the appropriate program PLCxxxUS5.S5D (the blanks xxx correspond to the PLC type 115, 135, 150, 155). Use the cable connector to establish a connection from PLC to PG which is required for the transfer.

The packages **LAD/CSF/STL** must now be selected in the PROGRAM PACKAGE mask. After acceptance the PRESET mask is called which must be filled in as follows: -

REPRESENTATION : STL
PROGRAM-FILE : B:PLCxxxUST.S5D [RW]
COMMMENTS : YES
OPERATING MODE : ON [CHGE IN CYCL]

Use **F6** to accept these pre-sets.

Use F7 (Help) -> F1 (Transfer) to transfer the program from the hard disk of the PG into the CPU RAM. The following parameters must now be assigned: *Source:* ***FD*** *Block:****B*** *To Dest.:* ***PLC***. Start the copy procedure using the **Transfer** key. The entire copy procedure can be checked with the DIR function (**F3**): *Source:* ***PLC*** *Block:* ***B***. A list is produced of the transferred blocks.

Finally you return to the FUNCTION SELECTION mask by using **F8** (Exit).

10.3.2 Sequence of Test Program in Normal Operation

Parameter Assignment of Special Driver

The parameter assignment of the special driver is carried out using the Initialization Data Block DB189 in the PLC-Programm:

DB189

Bit15.....Bit0

IF0	IF1			
DW 1	DW 11	EMPFDB	EMPFDW	Receive Data Block
DW 2	DW 12	ECPUNR	EKENN	
DW 3	DW 13	PROPAM		Procedure Parameter
DW 4	DW 14	PROPAR		Operating Mode
DW 5	DW 15	ENDEZ1	ENDEZ2	End Identifiers
DW 6	DW 16	TELLEN		Message Length
DW 7	DW 17	ZVZTIM		Character Delay Time
DW 8	DW 18	XOFFTIM		XOFF-Monitoring Time

For the interface 0 DW 1 to 8 are relevant and are assigned as follows:

DW 1: KY = 030,001 EMPF-DB-NR. 30, from DW 1
DW 2: KY = 000,001 To all CPU's, Receive Identifier = 1
DW 3: KM = 00000000 00110100 Procedure parameter:
Baudrate=9600 Baud, even parity,
1 Stopbit, 8 Data bits
DW 4: KM = 00000000 00000000 Operating Mode:
with BREAK-evaluation,
without RS 485 Betrieb,
without XON/XOFF-Protocol,
end criterion=ZVZ elapsed,
without Echo, without "STX-Keep Free time",
Halfduplex-Operation,
FF as Dummy for transmission
DW 5: KH = 0000 not used
DW 6: KH = 0000 not used
DW 7: KF = +00220 Character Delay Time = 220 ms
DW 8: KH = 0000 not used

For interface 1 DW 11 to 18 are relevant:

DW 11: KY = 031,001	EMPF-DB-NR. 30, from DW 1
DW 12: KY = 000,002	to all CPU's, Receive Identifier = 2
DW 13: KM = 00000000 00110100	Identical to IF 0
DW 14: KM = 00000000 00000000	"
DW 15: KH = 0000	"
DW 16: KH = 0000	"
DW 17: KF = +00220	"
DW 18: KH = 0000	"

Start of Test Program

Before changing the operating mode of the CPU to RUN, choose function Variable Control using **F4** (PLC-Fct) and **F6** (Cont Var).

If you are using STEP5 Version 3.0, use the image blocks aready stored containing the input masks.

Use of the Masks:

BB1 - empty mask; can be used to clear the screen

BB2 - transmission/receive check; this mask serves to monitor and analyse transmission and reception

BB3 - INIT-check; this mask serves to check the initialisation procedure of the interfaces

Use **F1** (Fetch) to call the image blocks, enter an appropriate number and accept using the **Accept-key**.

First load image block **BB3**.

After using the **Accept-Key** again, the signal stati of the operands are displayed - however, the PLC is still in STOP-status.

Change the operating mode of the CPU to RUN - This is displayed on the screen with the *PLC in Cycle* message.

The following display appears on the screen:

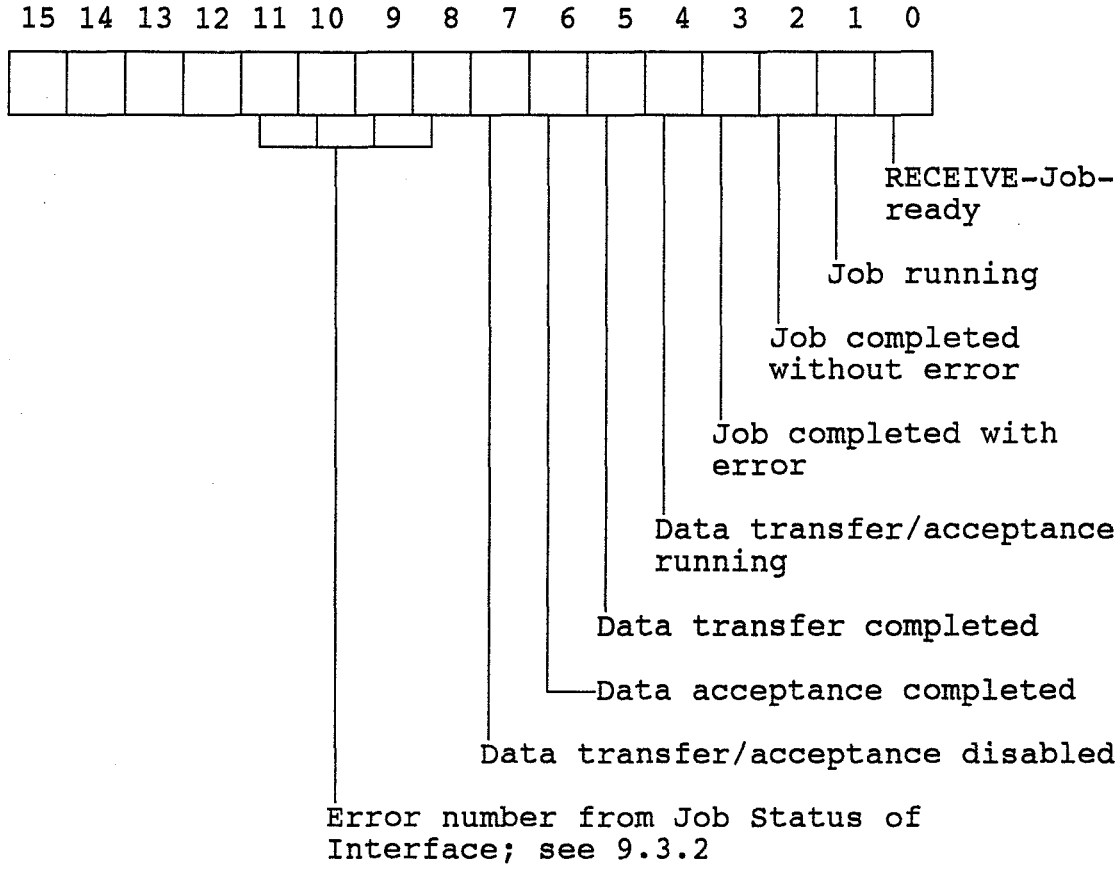
PLC IN CYCLE

Operands:	*Signal Stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000000*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 6	*KH=0024*	*ANZW INIT-Send*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DW 3	*KH=0000*	*PAFE INIT/free*
DW 4	*KH=0000*	*PAFE S.ALL/PAFE R.ALL*
DB 11		*Display-DB for IF1*
DW 6	*KH=0024*	*ANZW INIT-Send*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DW 3	*KH=0000*	*PAFE INIT/free*
DW 4	*KH=0000*	*PAFE S.ALL/PAFE R.ALL*

1184:Status processing running

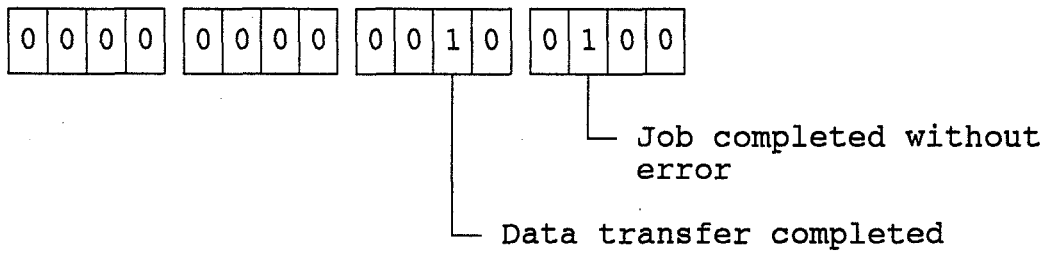
CONTROL VAR

The contents of of **DW6 (ANZW INIT-SEND**) supply confirmation of the completed initialisation. In this instance the bit positions of the condition code word have the following meaning: -



DW8 contains the following information: -

DW 6: KH=0024 , the bit representation means:



Transfer of Data between IF 0 und IF 1

For the data transfer the first requirement is a transmission line from IF 0 to IF1.

In order to call the single stati in **BB2**, press the **Abort key twice**, followed by **F6** (Cont Var), **F1** (Fetch), enter number **2** and confirm by using the **acceptance key twice**.

Use the **Abort Key** once, and you get to the control modus.

You can create transmission in one or two directions - this depends on whether you set M10.7 or M11.7, or both bits simultaneously.

Because of the special characteristics of the control function CONT VAR, the transmission procedure is carried out in two phases: first the data is merely transcribed from the PG to the PLC, followed by the actual data transfer between the interfaces.

If you wish to change data in DB20/21 or DB30/31 using Control Variable, you must therefore enter the changes in two stages: -

First enter the data into the appropriate data block in hexadecimal format, and acknowledge **twice** using the **Acceptance key**. The data is avaiable in the PLC.

Now use the **Abort Key** to return to Control Mode. Assign 1 to bits M10.7 und M11.7, and again press the **Acceptance key twice** - data exchange is carried out.

Image Block 2 now looks as follows: -

PLC IN CYCLE

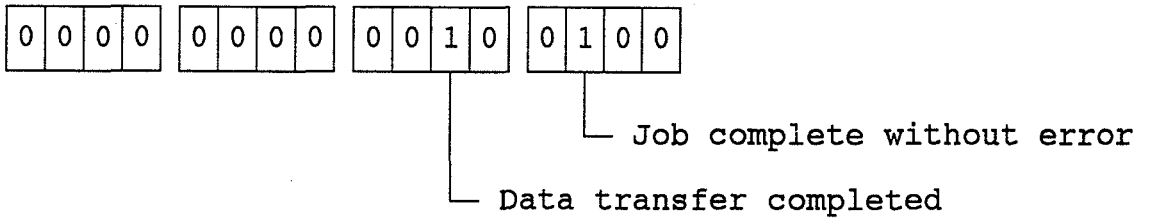
Operands:	*Signal Stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000000*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 8	*KH=0024*	*ANZW SEND-Direkt*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DB 11		*Display-DB for IF1*
DW 8	*KH=0024*	*ANZW SEND-Direkt*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DB 30		*Receive Data Block IF0*
DW 0	*KF=+1*	*Amount of Messages*
DW 1	*KF=+6*	*Message Length*
DW 2	*KH=CDEF*	*User data*
DB 31		*Receive Data Block IF1*
DW 0	*KF=+1*	*Amount of Messages*
DW 1	*KF=+2*	*Message Length*
DW 2	*KH=ABCD*	*User data*

1184:Status processing running

CONTROL VAR

An analysis shows: -

DW 8 (ANZW SEND Direkt) KH= 0 0 2 4



An indication on the amount of received messages can be found in receive data blocks DB30/31 in DW0, 1 message has been received so far. The length of the message in bytes can be found in DW2, interface 0 has received 6 bytes, interface 1 2 bytes. The first two received bytes are in data word 2.

The length of the transmitted messages is indicated in words in FB2/12 on FB SEND in parameter QLAE, it can also be changed there.

10.3.3 Sequence of Test Program with Error Simulation

The following paragraph deals with assumed (and very true to life) transmission errors, and their effect on the DW contents.
Furthermore we explain the analysis method.

Errors during Initialisation

If you do not get the message "Job completed without error" in DW6, use PAFE (DW3, DW4), the Condition Code Word (DW6) and SYSTAT (DD16) to narrow down the reason for the error. After correcting the error you can set the trigger flag (M10.0 and/or M11.0) and have the INIT-Send repeated. For this purpose you use the Abort key once, set the appropriate bit and accept the correction by pressing the Acceptance key twice. After a PAFE error, however, a new synchronization is required (e.g. Stop/Run on the CPU, or via the PG using PLC-FCT).

As long as the INIT-Send has not been completed without error and therefore the flags 10.1 and 11.1 are still set, no Send Jobs are possible for data transmission.

If the mode selector switch on the CP is still in STOP, the following display appears in **BB3**:

PLC IN CYCLE

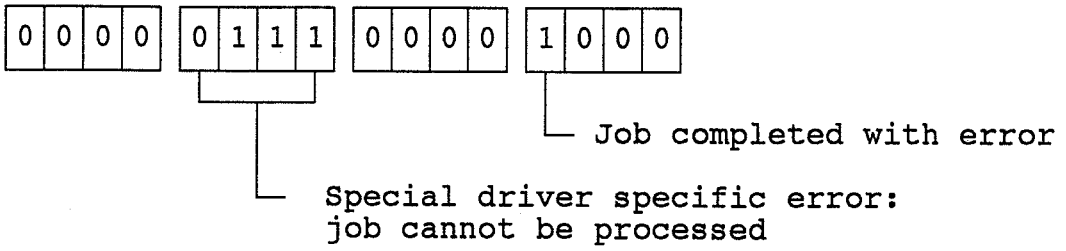
Operands:	*Signal Stati:*	
MB 10	*KM=00000010*	*Processing IF0*
MB 11	*KM=00000010*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 6	*KH=0708*	*ANZW INIT-Send*
DD 16	*KH=088A 0000*	*SYSTAT Contents*
DW 3	*KH=0000*	*PAFE INIT/free*
DW 4	*KH=0000*	*PAFE S.ALL/PAFE R.ALL*
DB 11		*Display-DB for IF1*
DW 6	*KH=0708*	*ANZW INIT-Send*
DD 16	*KH=088A 0000*	*SYSTAT Contents*
DW 3	*KH=0000*	*PAFE INIT/Free*
DW 4	*KH=00000*	*PAFE S.ALL/PAFE R.ALL*

1184:Status processing running

CONTROL VAR

DB 10/11 contains the following information: -

DW 6 (ANZW INIT- SEND) KH= 0 7 0 8



(see Operating Manual, Special Driver, Para. 9.3)

DW 16 (SYSTAT Contents)	0 8	8 A
DW 17 (SYSTAT Contents)	0 0	0 0

KH= 0 8 8 A

- 8 A: Nr. for SYSTAT
- 0 8: SYSTAT-Status byte (has lower priority in this context)

In accordance with Operating Manuual Special Driver Para. 9.4.3:
No. for ANZW: **07**, Nr. for SYSTAT: **8A**
Description: Current job was interrupted at the cold re-start of the CP, or cold re-start of the CP by the PG; CP in STOP
Remedy: Set Mode selector switch to RUN

Interface not loaded

The interface for IF2 was deleted, the signal stati of the operands look as follows in BB3: -

PLC IN CYCLE

Operands:	*Signal Stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000010*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 6	*KH=0024*	*ANZW INIT-Send*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DW 3	*KH=0000*	*PAFE INIT/free*
DW 4	*KH=0000*	*PAFE S.ALL/PAFE R.ALL*
DB 11		*Display-DB for IF1*
DW 6	*KH=0708*	*ANZW INIT-Send*
DD 16	*KH=08C9 0000*	*SYSTAT Contentst*
DW 3	*KH=0000*	*PAFE INIT/Free*
DW 4	*KH=0000*	*PAFE S.ALL/PAFE R.ALL*

1184:Status processing running

CONTROL VAR

Error analysis can be carried out based on DD16 of DB11:

DW 16 (SYSTAT Contents)	0 8	C 9
DW 17 (SYSTAT Contents)	0 0	0 0

```
KH=  0 8  C 9
       |   |
       |   └ Nr. for SYSTAT
       |
       └ SYSTAT-Status byte (has lower priority in
         this context)
```

In accordance with Operating Manuual COM525, Volume 1 (Issue 06), Register 7, Para. 8.2:
No. for SYSTAT: **C9**
Description: Hardware error on module: Interface not loaded.
Remedy: Load interface, if required.
Attention: Entry in SYSTAT is not deleted by loading, but only by RESET 200. This is called in the demonstration program at the re-start.

CP in STOP-Status

Flag bit 10.7 is set in order to send data from IF0 to IF1.
The CP-mode selector switch is in STOP.

The signal stati of the operands look as follows in **BB2**: -

PLC IN CYCLE

Operands:	*Signal Stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000000*	*Processing IF1*
DB 10		*Display DB for IF0*
DW 8	*KH=0C08*	*ANZW SEND-Direkt*
DD 16	*KH=0827 0000*	*SYSTAT Contents*
DB 11		*Display-DB for IF1*
DW 8	*KH=0000*	*ANZW SEND-Direkt*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DB 30		*Receive Data Block IF0*
DW 0	*KF=+0*	*Amount of Messages*
DW 1	*KF=+0*	*Message Length*
DW 2	*KH=0000*	*User Data*
DB 31		*Receive Data Block IF1*
DW 0	*KF=+0*	*Amount of Messages*
DW 1	*KF=+0*	*Message Length*
DW 2	*KH=0000*	*User Data*

1184:Status processing running

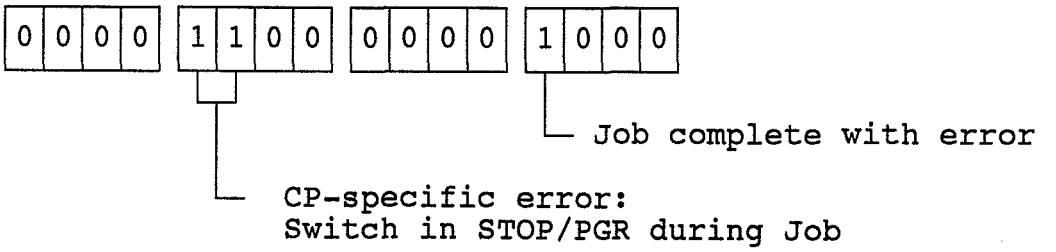
CONTROL VAR

The starting point for the evaluation is checking the condition code word of SEND-Direkt (DW 8.

NOTE: If the data link is not clear, STOP - RUN is sufficient to delete the receive data block and the job status.

DB 10 contains the following information: -

DW 8 (ANZW SEND Direkt) KH= 0 C 0 8



(see Operating Manual Special Driver, Para. 9.3)

DW 16 (SYSTAT Contents)	0 8	2 7
DW 17 (SYSTAT Contents)	0 0	0 0

KH= 0 8 2 7

- 2 7: No. for SYSTAT
- 0 8: SYSTAT-Status byte (has lower priority in this context)

In accordance with Manual COM525, Volume 1 (Issue 06), Register 7, para. 8.1 or 8.2: -
No. for ANZW: **0C**, No. for SYSTAT: **27**
Description: Switch in STOP/PGR during PLC-job
Remedy: Set the Mode Selector Switch to RUN
Amount of messages in DB31 DW0 is not increased, i.e. Interface 1 has not received a message.

If trigger flag M10.7 is set again, it merely results in a further entry of SYSYAT error 27 in DL of DW17. If all three error message bytes are full, and a further SYSTAT entry is necessary, bit 2^2 is set in DL of DW16 (entry 0C). This results in SYSTAT overflow. In order to recognize subsequent error messages, deletion of the SYSTAT area is required. This is carried out by bits M10.6 and/or M11.6 for interfaces 0 and 1.

Cable Connector IF0-IF1 interrupted

Both trigger flags M10.7 und M11.7 are set. Transmission is to be bi-directional. The cable connector was seperated from IF1. There was no prior deletion of the SYSTAT area.

The signal stati of the operands look as follows in **BB2**:

PLC IN CYCLE

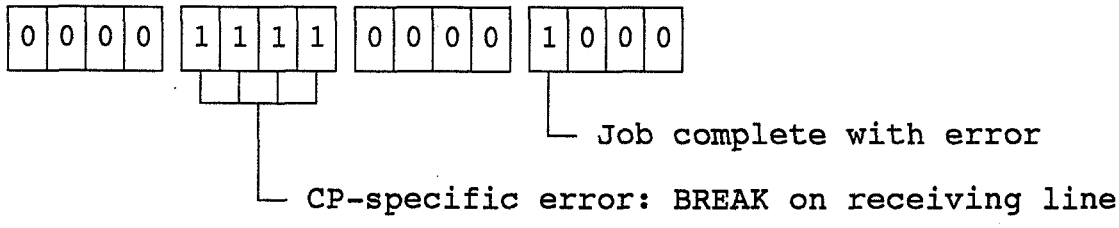
Operands:	*Signal Stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000000*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 8	*KH=0F08*	*ANZW SEND-Direkt*
DD 16	*KH=0A27 0000*	*SYSTAT Contents*
DB 11		*Display-DB for IF1*
DW 8	*KH=0F08*	*ANZW SEND-Direkt*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DB 30		*Receive Data Block IF0*
DW 0	*KF=+0*	*Amount of Messages*
DW 1	*KF=+0*	*Message Length*
DW 2	*KH=0000*	*User Data*
DB 31		*Receive Data Block IF1*
DW 0	*KF=+0*	*Amount of Messages*
DW 1	*KF=+0*	*Message Length*
DW 2	*KH=0000*	*User Data*

1184:Status processing running

CONTROL VAR

An analysis shows:

DW 8 (ANZW SEND Direkt) KH= 0 F 0 8



In accordance with Operating Manual Special Driver Para, 9.3.2:
No. for ANZW: **0F**
Description: BREAK, The receiving line to the partner is interrupted.

Remedy: Estalblish the connection between the units, and turn on the partner device. When operating with TTY, check if line current is present in neutral.

If BREAK is recognized during transmission, no SYSTAT entry is made. The error code of the last error is still in the SYSTAT area of interface 0. However, bit 2^1 is set in the SYSTAT area, which indicates the BREAK status on the line (see in BB2 DD16). On re-connection of the cable, 08 27 is displayed (BREAK bit is no longer set).

Receive Data Block does not exist

Delete SYSTAT area by setting the flag M10.6. Delete data block DB31 in the PLC, this is the receive data block for IF1. Now transmit data from IF0 to IF1 by setting the flag 10.7.

The signal stati of the operands look as follows in **BB2**:

PLC IN CYCLE

Operands:	*Signal stati:*	
MB 10	*KM=00000000*	*Processing IF0*
MB 11	*KM=00000000*	*Processing IF1*
DB 10		*Display-DB for IF0*
DW 8	*KH=0024*	*ANZW SEND-Direkt*
DD 16	*KH=0000 0000*	*SYSTAT Contents*
DB 11		*Display-DB for IF1*
DW 8	*KH=0F08*	*ANZW SEND-Direkt*
DD 16	*KH=0802 5300*	*SYSTAT Contents*
DB 30		*Receive Data Block IF0*
DW 0	*KF=+1*	*Amount of Messages*
DW 1	*KF=+2*	*Message Length*
DW 2	*KH=CDEF*	*User data*
DB 31		*Receive Data Block IF1*
DW 0	*KF= * DB missing*	
DW 1	*KF= * DB missing*	
DW 2	*KH= * DB missing*	

1184:Status processing running

CONTROL VAR

An analysis shows: -

DB10, DW8 (ANZW SEND Direkt) KH= 0 0 2 4

This means Job completed without error, data transmission completed. No error was recognized during the transmission of the message. This is correct.

The job status of the last Send-Direkt for IF1 is still in DB11, DW8. This is not relevant for the current evaluation.

DB11

DW 16 (SYSTAT Contents)	0 8	0 2
DW 17 (SYSTAT Contents)	5 3	0 0

```
KH=  0 8   0 2   5 3
       |       |
       |       └ No. for SYSTAT
       |
       └ SYSTAT-Status byte (has lower priority in
         this context)
```

In accordance with Manual COM 525, Volume 1 (Issue 06), Register 7, Para. 8.2:
No. for SYSTAT: **02 and 53**
Description:
02: DB does not exist and/or is illegal
53: when processing the link partner job: error during acceptance of data from the DPR, a more exact error number was entered before into the SYSTAT area
Remedy: create block

This means that the identifier "job complete without error" refers only to the transmission of a message, and does not indicate whether the other interface was able to receive the message without error or whether it transferred it to the PLC. Therefore it is always essential to evaluate the SYSTAT area in order to recognize receive errors.

10.3.4 Used Organisation, Function and Program Blocks in the Sample Program

OB1 Cyclic Operation
OB20 Cold Re-Start
OB21 Manual Re-Start
OB22 Automatic Re-Start

PB1 Processing of Interface 0
PB11 Processing of Interface 1

FB1 Initialisation IF 0
FB2 Transmission of IF 0 to IF 1
FB3 Reception IF 0
FB4 Processing of SYSTAT IF 0

FB11 Initialisation IF 1
FB12 Transmission of IF1 to IF 0
FB13 Reception IF 1
FB14 Processing of SYSTAT IF1

10.3.5 Used Data Blocks in Sample Program

DB10 Display IF 0
DB11 Display IF 1
DB20 Send Mailbox IF 0
DB21 Send Mailbox IF 1
DB30 Receive Data Block IF 0
DB31 Receive Data Block IF 1
DB189 INIT-DB IF 0 and 1

10.3.6 Flags Assigned in Sample Program

MB 10:

M 10.7 - Trigger Flag for SEND from IF 0 to IF 1
M 10.6 - SYSTAT Delete IF 0
M 10.1 - INIT IF 0 being processed
M 10.0 - Trigger Flag for INIT IF 0

MB 11:
M 11.7 - Trigger Flag for SEND from IF 1 to IF 0
M 11.6 - SYSTAT Delete IF 1
M 11.1 - INIT IF 1 in being processed
M 11.0 - Trigger Flag for INIT IF 1

MB 20: - Parameter Assignment Error

MW 22: - Auxiliary Flag Word for PLC 115U, because binary logic operations with the data word are not possible.

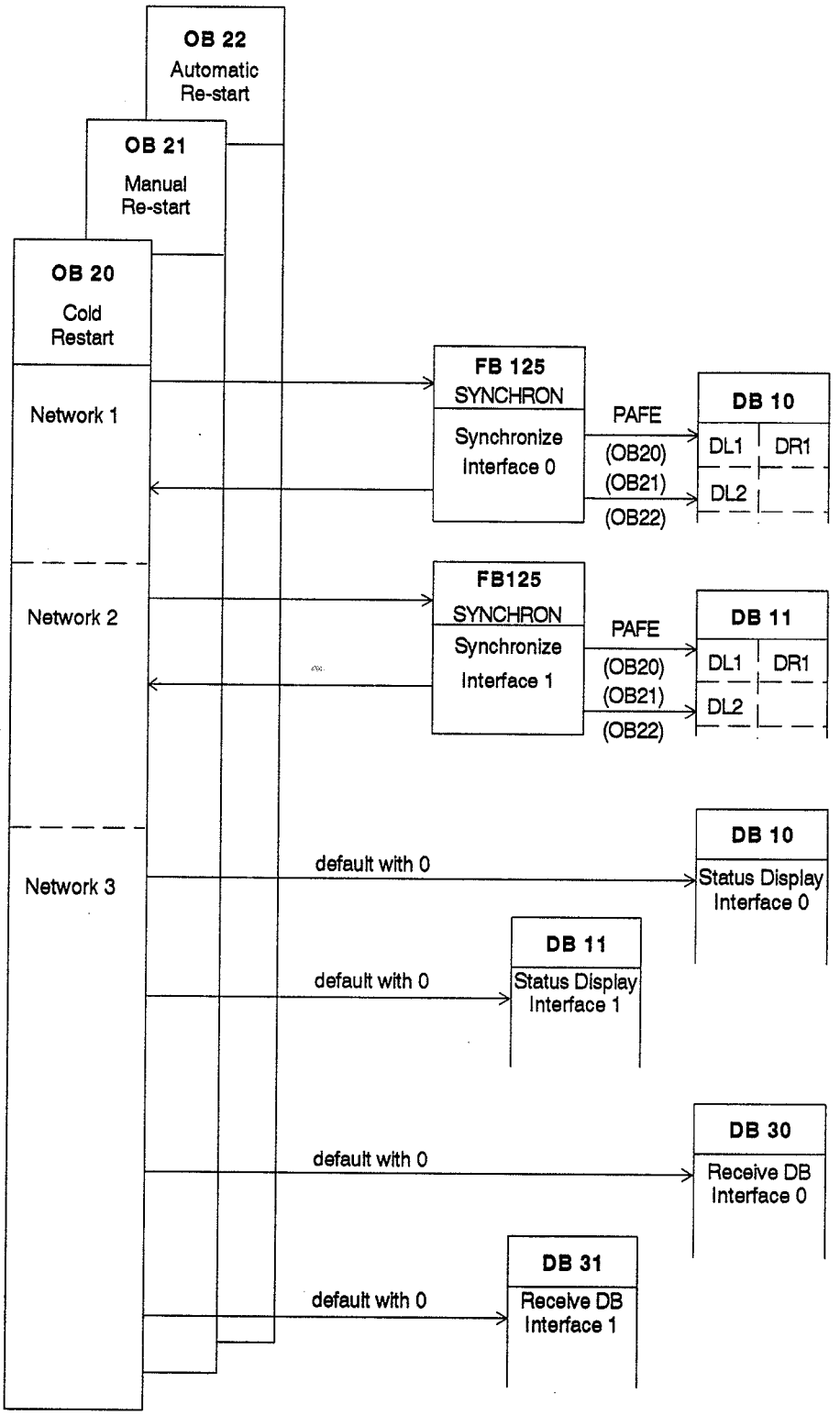
10.3.7 Structure of Data Blocks for Error Evaluation

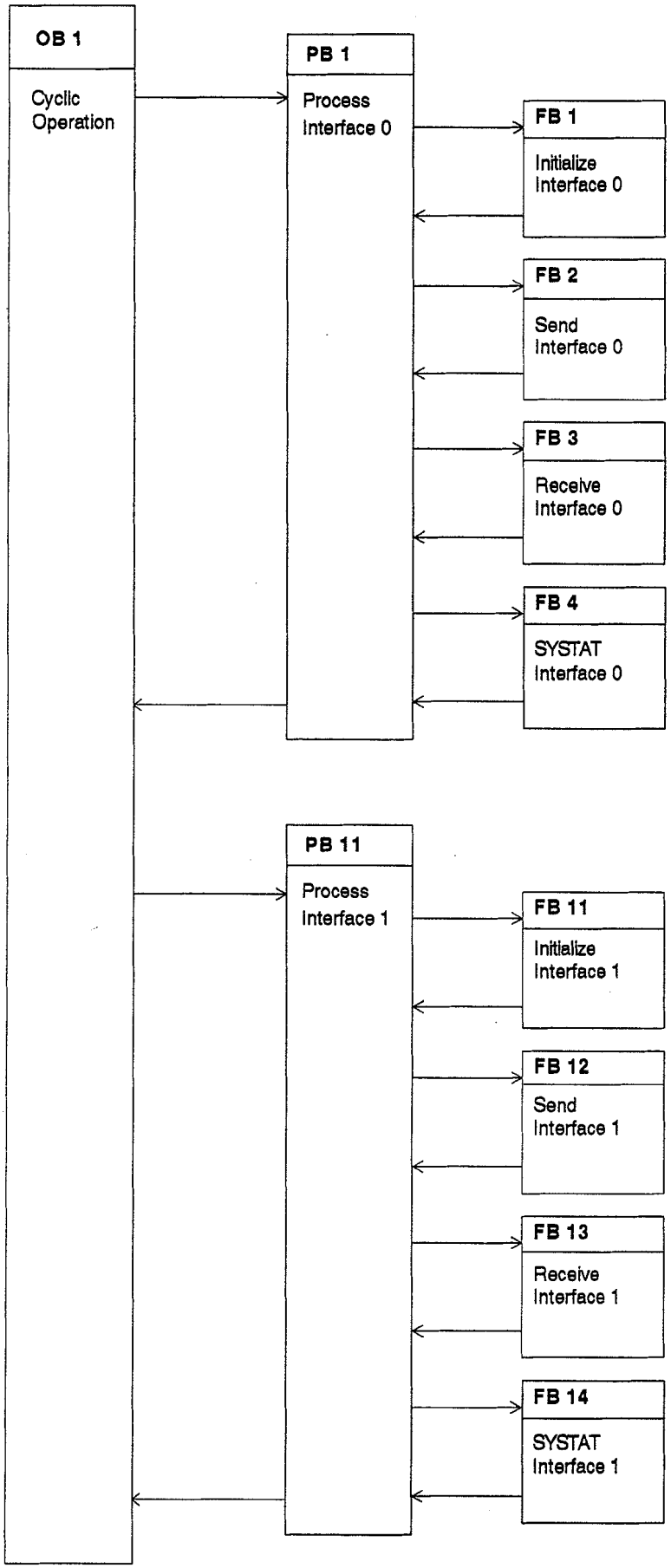
For Interface 0: DB 10, for Interface 1: DB 11

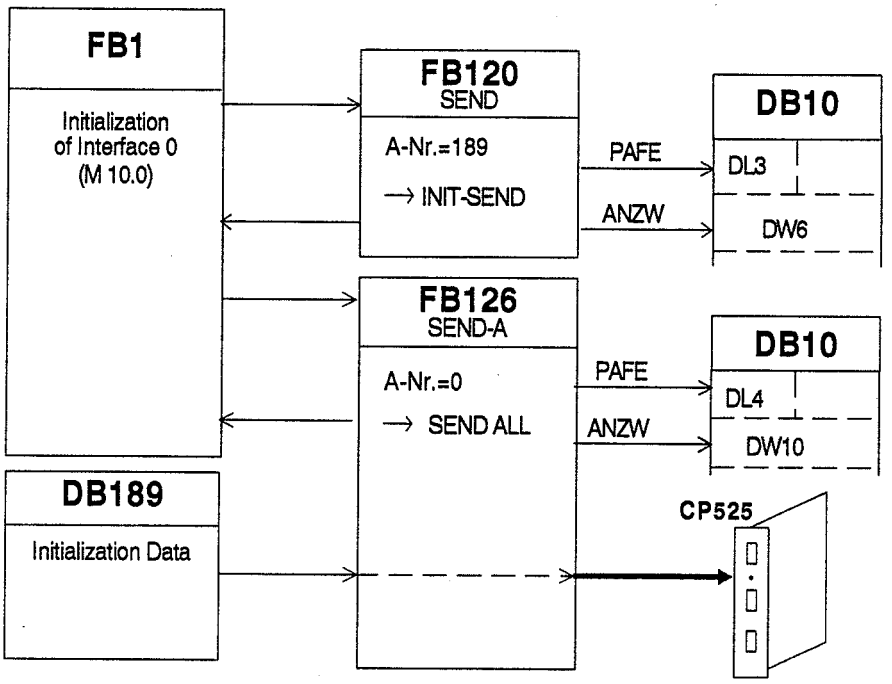
	DL	DR	
DW 0	free	free	
DW 1	PAFE SYNCH OB 20	PAFE SYNCH OB 21	
DW 2	PAFE SYNCH OB 22	free	
DW 3	PAFE INIT	PAFE SEND	
DW 4	PAFE SEND ALL	PAFE RECEIVE ALL	
DW 5	PAFE SYSTAT Read	PAFE SYSTAT Reset	
DW 6	ANZW INIT		These
DW 7	Reserved		DW
DW 8	ANZW SEND Direkt		are
DW 9	Reserved		deleted
DW 10	ANZW SEND ALL		during
DW 11	Reserved		Start
DW 12	ANZW RECEIVE ALL		Up
DW 13	Reserved		
DW 14	ANZW SYSTAT		
DW 15	Reserved		
DW 16	SYSTAT Contents		
DW 17	SYSTAT Contents		

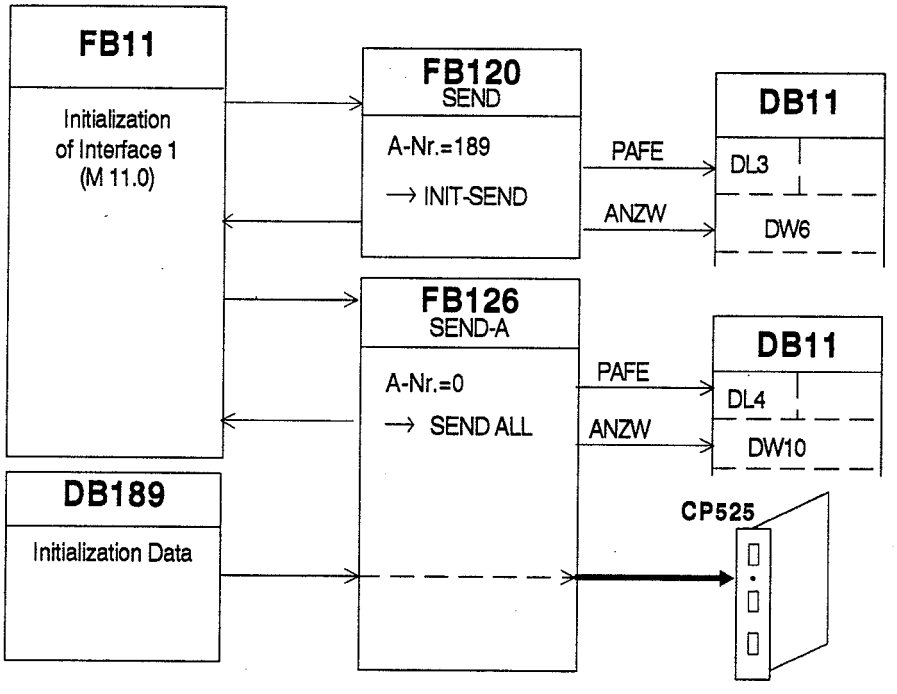
10.3.8 Overview PLC-Program (Example: PLC 135U)

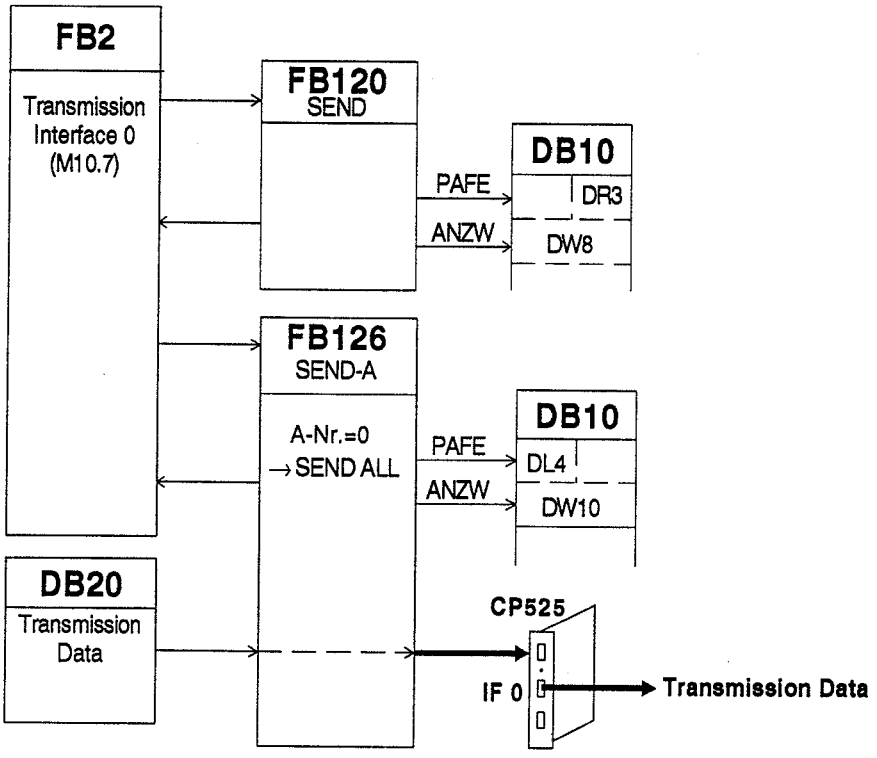
Other FB Numbers or standard FB's may have to be used for other PLC's (see 10.2 "Numbers of Handling Blocks").

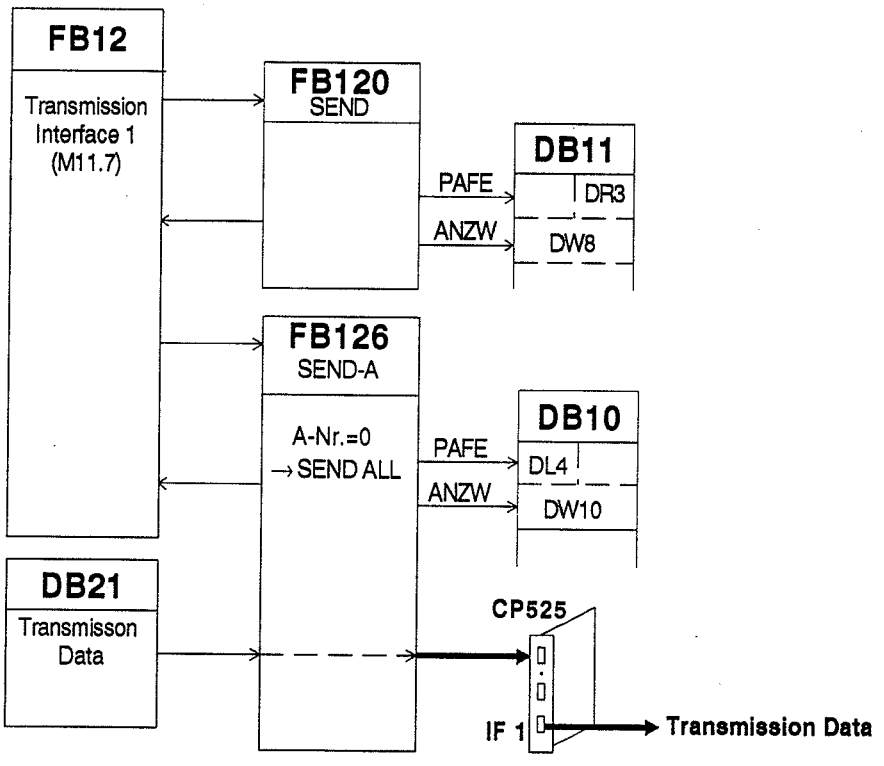


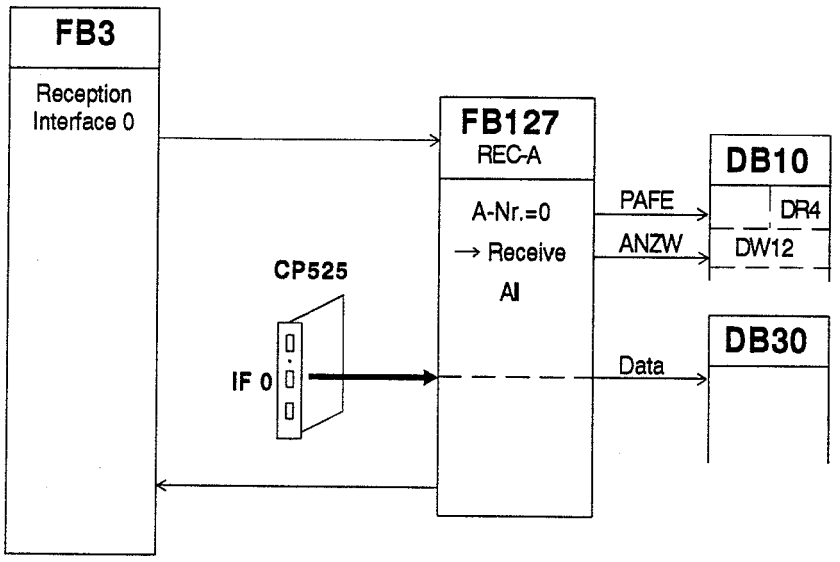

OB 1
Cyclic Operation
PB 1
Process Interface 0
FB 1
Initialize Interface 0
FB 2
Send Interface 0
FB 3
Receive Interface 0
FB 4
SYSTAT Interface 0
PB 11
Process Interface 1
FB 11
Initialize Interface 1
FB 12
Send Interface 1
FB 13
Receive Interface 1
FB 14
SYSTAT Interface 1

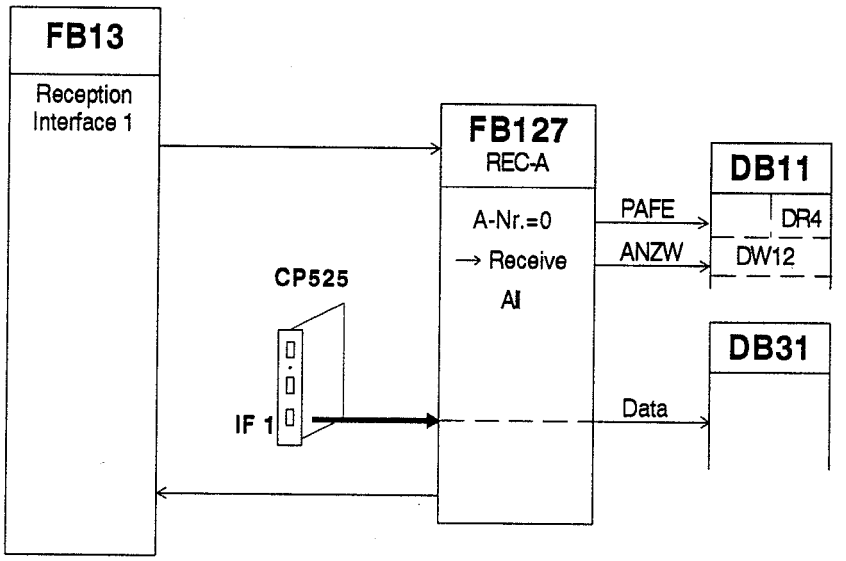

FB1
Initialization of Interface 0 (M 10.0)
FB120
SEND
A-Nr.=189
→ INIT-SEND
PAFE
ANZW
DB10
DL3
DW6
FB126
SEND-A
A-Nr.=0
→ SEND ALL
PAFE
ANZW
DB10
DL4
DW10
DB189
Initialization Data
CP525

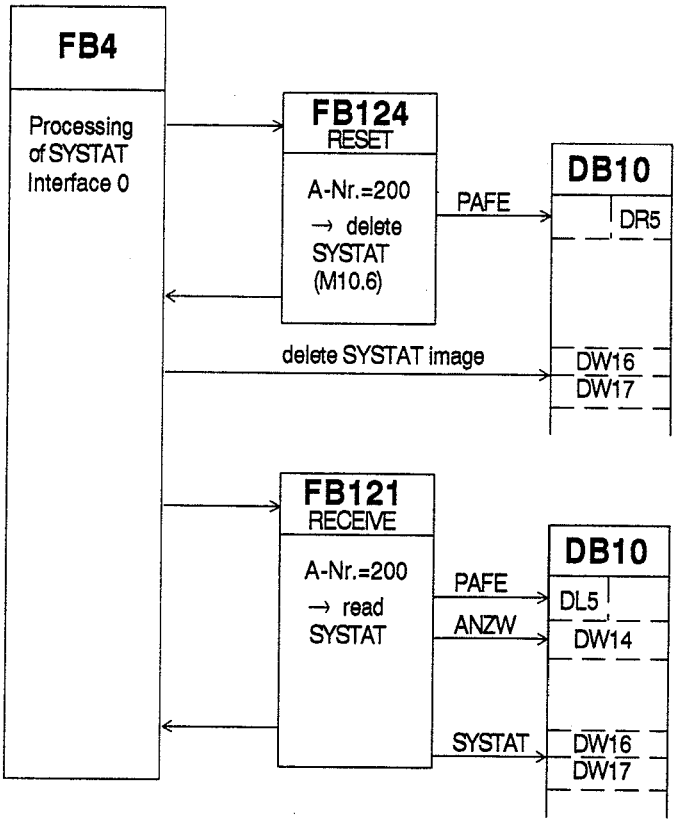

FB11
Initialization of Interface 1 (M 11.0)
FB120
SEND
A-Nr.=189
→ INIT-SEND
PAFE
ANZW
DB11
DL3
DW6
FB126
SEND-A
A-Nr.=0
→ SEND ALL
PAFE
ANZW
DB11
DL4
DW10
DB189
Initialization Data
CP525

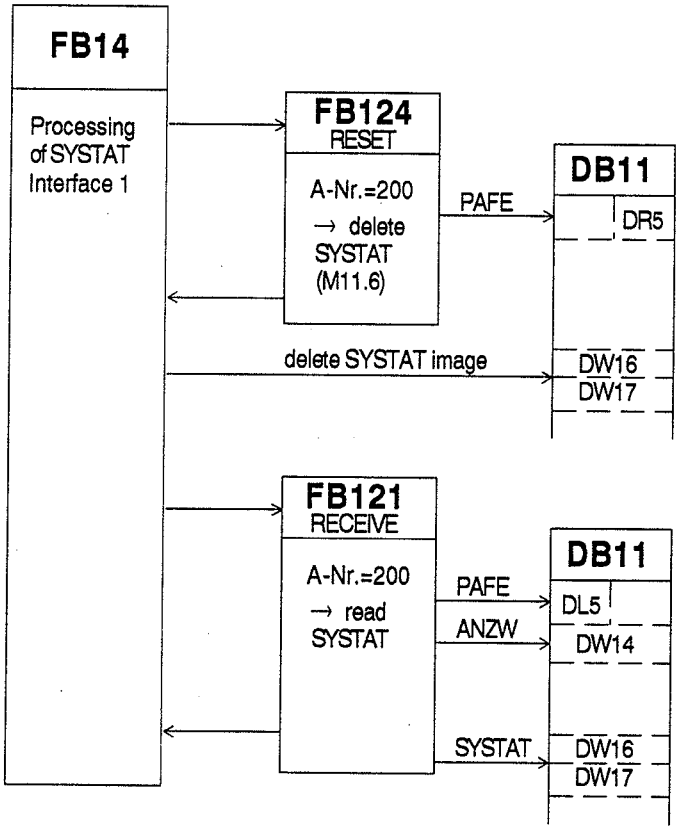

FB2
Transmission Interface 0 (M10.7)
FB120
SEND
PAFE
ANZW
DB10
DR3
DW8
FB126
SEND-A
A-Nr.=0
→SEND ALL
PAFE
ANZW
DB10
DL4
DW10
DB20
Transmission Data
CP525
IF 0
Transmission Data


FB12
Transmission Interface 1 (M11.7)
FB120
SEND
PAFE
ANZW
DB11
DR3
DW8
FB126
SEND-A
A-Nr.=0
→SEND ALL
PAFE
ANZW
DB10
DL4
DW10
DB21
Transmisson Data
CP525
IF 1
Transmission Data


FB3
Reception
Interface 0
FB127
REC-A
A-Nr.=0
→ Receive
AI
CP525
IF 0
PAFE
ANZW
Data
DB10
DR4
DW12
DB30


FB13
Reception
Interface 1
FB127
REC-A
A-Nr.=0
→ Receive
AI
CP525
IF 1
PAFE
ANZW
Data
DB11
DR4
DW12
DB31


FB4
Processing of SYSTAT Interface 0
FB124
RESET
A-Nr.=200
→ delete SYSTAT (M10.6)
PAFE
DB10
DR5
delete SYSTAT image
DW16
DW17
FB121
RECEIVE
A-Nr.=200
→ read SYSTAT
PAFE
ANZW
SYSTAT
DB10
DL5
DW14
DW16
DW17


FB14
Processing of SYSTAT Interface 1
FB124
RESET
A-Nr.=200
→ delete SYSTAT (M11.6)
PAFE
DB11
DR5
delete SYSTAT image
DW16
DW17
FB121
RECEIVE
A-Nr.=200
→ read SYSTAT
PAFE
ANZW
SYSTAT
DB11
DL5
DW14
DW16
DW17

11. CP Information

Additional information regarding CP525-2/CP524 and COM525 (connector pin assignment, COM525 handling etc.) may be found in the following manuals:

COM525
Programming Package for Communications Processors
CP524 and CP525 (S5-DOS)
Volumes 1 and 2.

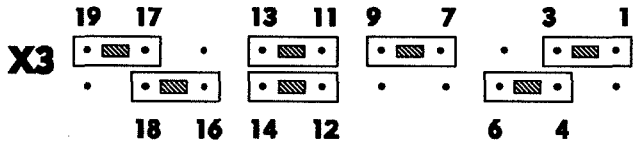
12. Notes

Jumper settings on the RS422-A/RS485-module 6ES5 897-0AA43

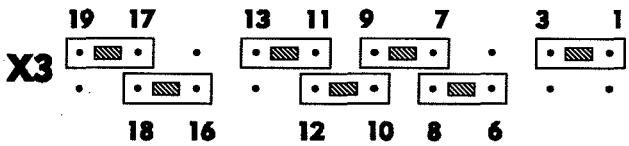
The RS422-A/RS485 - module with order-no. 6ES5 897-0AA43 is a development of the module -0AA42. Lower power input makes it possible to use the new module without fan. The location of the jumpers and the jumper settings have changed in contrary to the old module.

The manuals of special drivers and the COM525-manual describe the jumper settings of the module -0AA42.
Please find below the jumper settings of the module -0AA43 to preset the two-wire-line R and to switch over the data direction on the two-wire line R.

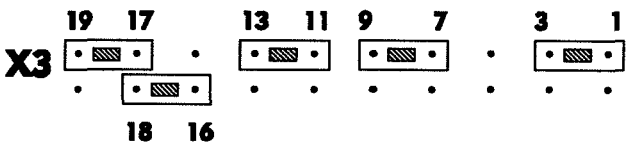
Presetting of the two-wire-line R for recognizing the break status



With jumpers 12-14 and 4-6 the two-wire-line R has the control signal, the break status will be regonized (default).
Pin 4 (R(A)) of the front connector is connected to +5 V via resistor.
Pin 11 (R(B)) of the front connector is connected to ground via resistor.

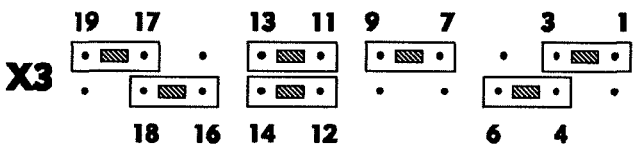


With the jumpers 10-12 and 6-8 the two-wire-line is preset as follows:
Pin 4 (R(A)) of the front connector is connected to ground via resistor.
Pin 11 (R(B)) of the front connector is connected to +5 V via resistor.
The break status **cannot** be recognized.

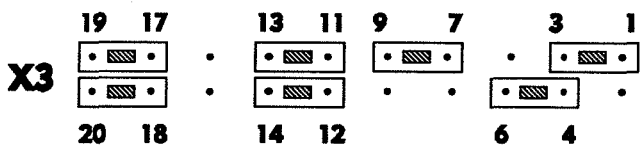


If the jumpers 12-14 and 4-6 (or 10-12 and 6-8) are dropped, the two-wire-line R is not preset. The recognition of break status cannot be guaranteed.

Data direction on the two-wire-line R



jumper 16-18 inserted
setting for **full duplex mode**
Data can only be received on the two-wire-line R (default).



jumper 18-20 inserted
setting for **half duplex mode**
Data can be transmitted or received on the two-wire-line R (special driver necessary).