SIEMENS

SIMATIC S5

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

Adaptation MODBUS Protocol

S5 is Slave

Operating Instructions

Order No.: 6ES5 897 - 2QA21 Version 09

INDEX

1 Introduction

The data link software package "MODBUS-Protocol" for communications processors CP525-2/CP524 creates a data link of SIMATIC S5 controllers (115U, 135U, 150U and 155U) to "modbus-capable control systems (e.g. Honeywell TDC 3000 or Modicon controllers).

The transmission protocol is GOULD-MODICON-MODBUS with **RTU format.** Data transmission is carried out in accordance with the Master-Slave principle. The **slaves (SIMATIC S5)** send on request only, which means that the master (host computer) has the initiative. Message traffic from slave to slave is not possible.

Function codes 01, 02, 03, 04, 05, 06, 08, 15 and 16 can be used for communication between host system and CP. The communication area which all functions refer to is located on the CP.

The procedure is asynchronous, half duplex, code-transparent and may run on a 20 mA (TTY, line current), V24 (RS232-C), RS422 or RS485 interface (only for CP524). V24 auxiliary signals are not used in conjunction with this special driver when a V24 interface is in operation. In RS485 mode, the RS422/RS485 interface is multi-terminal capable, i.e it is possible to set up a network with one master and several slaves whereas the relevant hardware is required when any of the other interfaces are used (e.g. for Honeywell DHP).

The check sum of the messages is created by means of a Cycle-Redundance-Check. In order to keep the reply times of the slaves as short as possible this time consuming calculation is carried out in parallel to the Send procedure. This imposes a great load on the CP processor and uninterrupted transmissions can no longer be carried out at a Baud Rate of 19200 Bd. This results in an increased transmission time of a message by approximately 10 %.

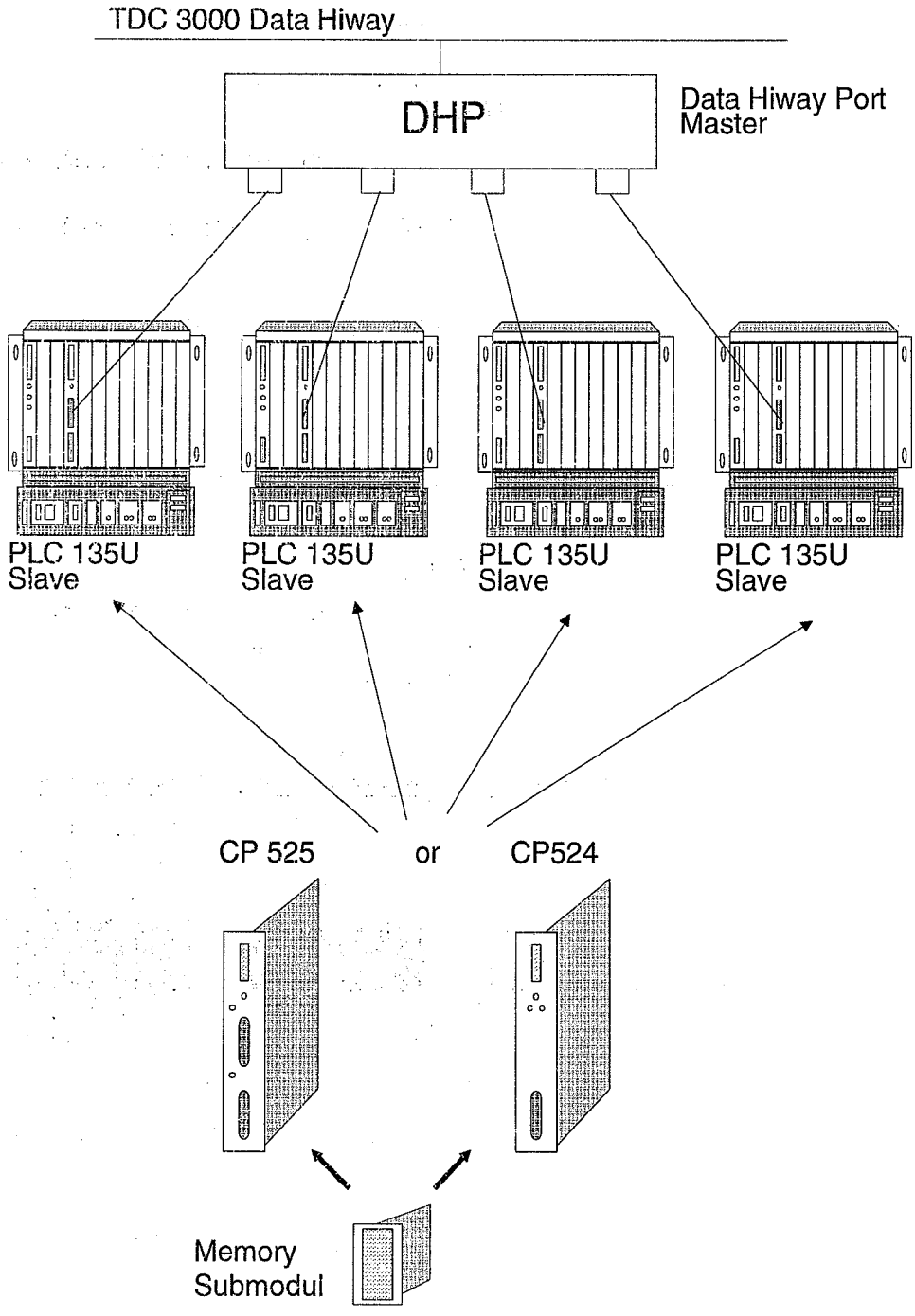
Please note that at Baud Rates from 9600 Bd (or for CP525 the total Baud Rate of the two interfaces) a character overflow in the interface block cannot be completely avoided during reception of messages from the master. In this instance the CP does not reply to the query from the master and enters error "77H" into SYSTAT. The Master must repeat the query. Fewer repetions can be achieved by using a lower Baud Rate.

A fast CPU (e.g. 928, 944, 946/47) puts an additional load on the CP processor. Caused by too slow processing of the CPU-handshake by the CP it is possible to encounter error "6" in the condition code word, "14H" in SYSTAT during all PLC jobs (Fetch-, Send-, Receive-Direkt). The required reaction to this error in the PLC program should be a repetition of the job in order to read from the communication area, or write into it. The time requirement on the CP can be reduced by using a smaller synchron blocking size (is equivalent to a smaller data transmission CPU-CP per data block).

The loading of the CP processor which has to carry out "simultaneous" operation of both the interface to the PLC and the interface to the link partner, can be reduced by the above mentioned measures (low baud rates, smaller blocking sizes) or by using a CP524 (one interface) instead of a CP525 (two interfaces).

Possible System Configuration:

Data Link to Honeywell TDC 3000 via DHP



2 Operating Modes of the Special Driver

The special driver can be adapted to any application by means of parameter assignment. Initially, the user should carry out a check as to how the following parameters are to be selected for his specific application:

The **length of the communication area** is

252 Data Words (= 1DB) ⇨ Chapter 2.1.1
504 Data Words (= 2DB) ⇨ Chapter 2.1.2

Is it possible to divide the communication area into a **writing area and a reading area** both S5 related and also master related?

* yes, areas seperate ⇨ Chapter 2.2.1

Case 1 operating mode "without co-ordination flag"

Case 2 operating mode "with co-ordination flag"
Co-ordination flag to report writing accesses of the master.

* no, areas not seperate ⇨ Chapter 2.2.2

Case 1 operating mode "without co-ordination flag"

Case 2 operating mode "with co-ordination flag"
Prior to overwriting, communication area must be disabled.

Case 2a operating mode "without waiting time"
In the event of an error, code 06 is sent immediately.

Case 2b operating mode "with waiting time"
Exception code 06 is sent only when the access of the master cannot be processed even after the waiting time has elapsed.

Does the master require information as to whether the **S5 device is in Stop status** (PLC monitoring by CP)? How long is the **PLC cycle time** (call of SEND-ALL)?

monitoring of a FETCH job ⇨ Chapter 2.3.1
monitoring of a SEND job ⇨ Chapter 2.3.2

Slave address used by the master to address the CP? ⇨ Chapter 3.2.4.3

2.1 Communication Area on CP

The communication area which all functions refer to, is transferred to the CP (= CP communication area). This area is the image of a DB or (depending on parameter assignment) of two DB's.

After a cold restart of the CP and/or the CPU, the parameterised S5 data area (max. 2 data blocks) is read and stored in the CP internal RAM memory. Both the connected host system as well as the CPU have reading and writing access to this communication area.

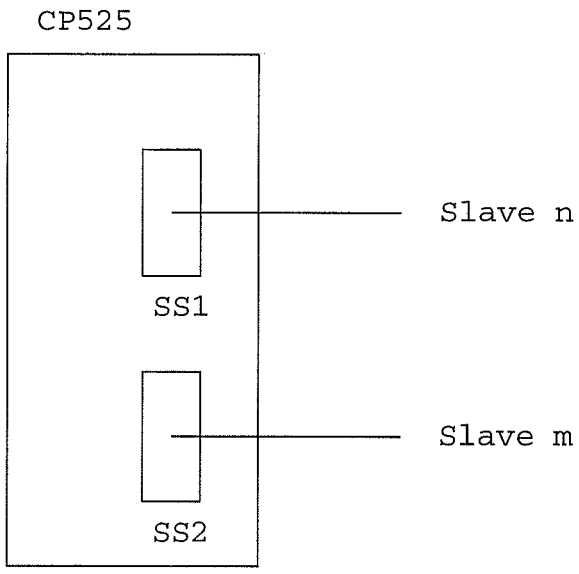
The data block(s) are required only for generating the communication area on the CP in the event of a cold re-start. During normal running of the special driver the DB's are not required or changed on CP initiative. Overwriting of the data blocks for the communication area is only carried out upon request by the S5 program (e.g. if the image of the communication area is carried in the DB's for the S5 user). The DB's can therefore be erased after start-up of the special driver.

This means that in the event of a function code received from the host device (see Para. 5 "Access to the CP Communication Area") it is not required to access the S5 data block area in order to carry out the reaction respective to the function code. This results in a reduction of the time required to generate a response to a minimum because the cycle time of the S5 program does not have to be taken into account.

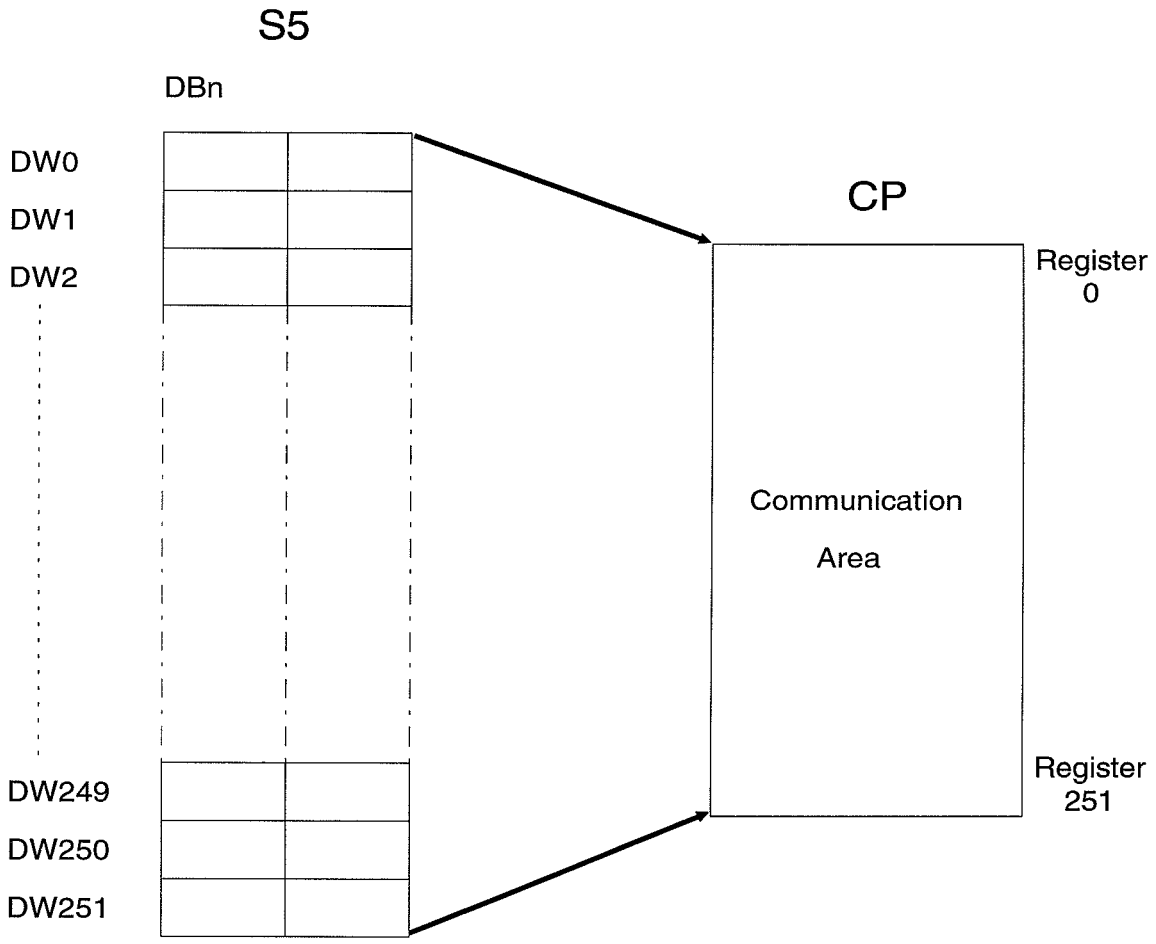
The S5 user can change the CP communication area by means of writing accesses (see Para 7.1 "Writing Accesses"). He can also copy the CP area into the S5 data area by means of reading accesses (see Para 7.2 "Reading Accesses").

When using both CP interfaces, one communication area per interface may be stored on the CP.

In this instance each interface operates as an independent slave.

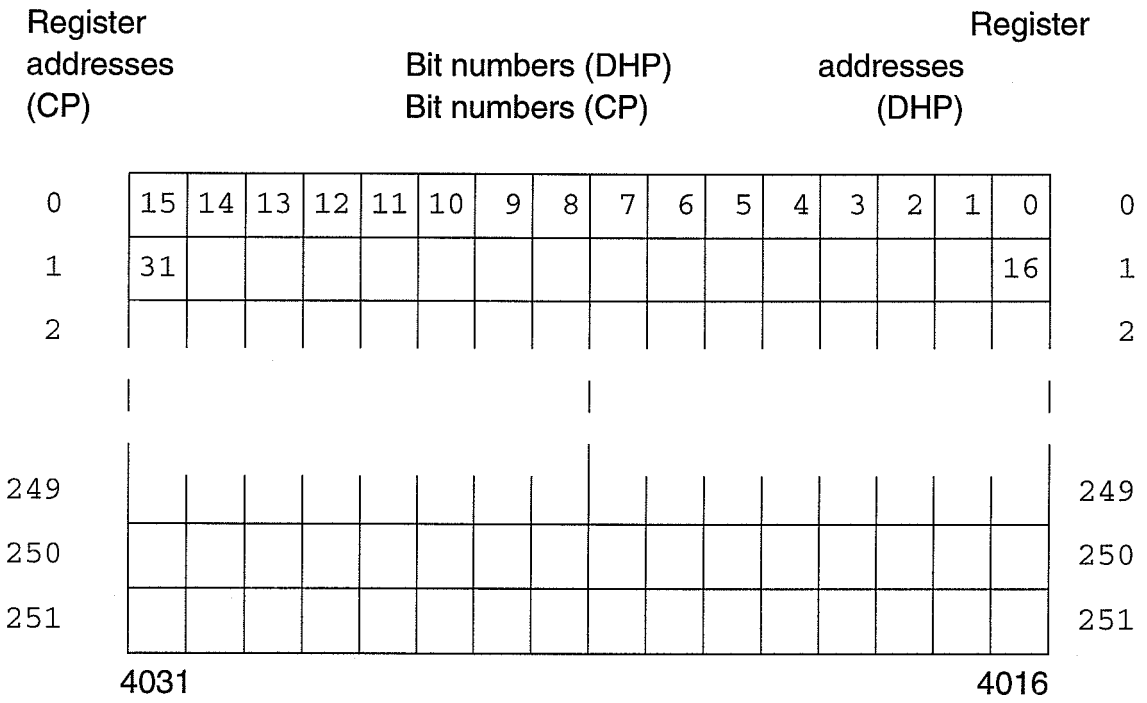


2.1.1 CP Communications Area for Operating Mode 1DB

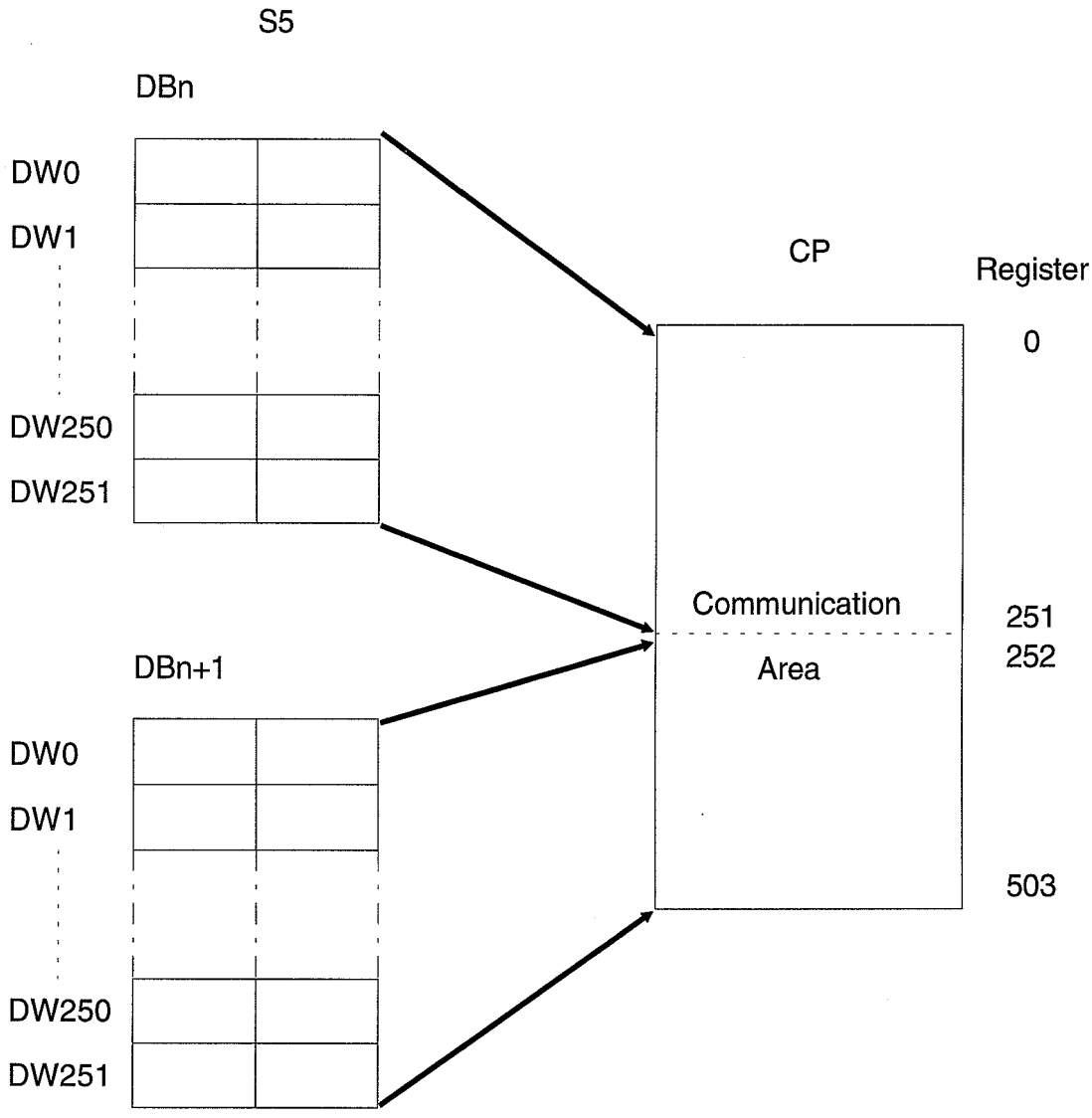


The following paragraph deals with the address area of the communication area located on the CP which is actually output to the serial interface by the host computer. The corresponding number in TDC 3000 must always be increased by one.

Operating Mode "1DB"

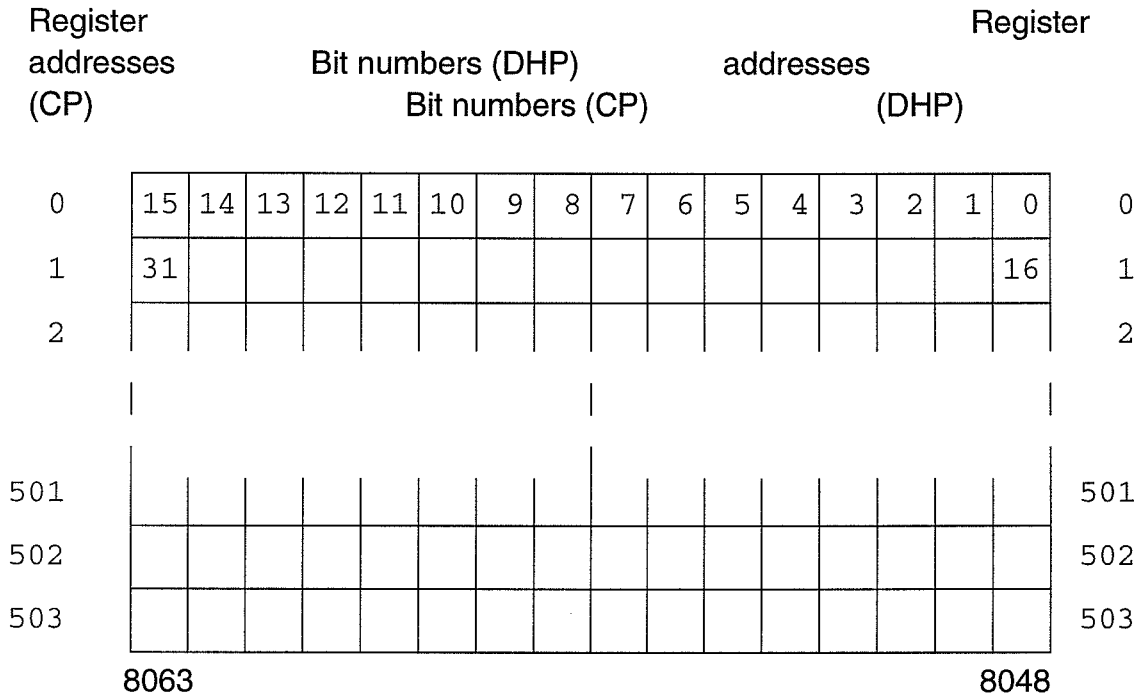


2.1.2 CP Communications Area for Operating Mode 2DB



The following paragraph deals with the address area of the communication area located on the CP which is actually output to the serial interface by the host computer. The corresponding number in TDC 3000 must always be increased by 1.

Operating Mode "2DB"



2.1.3 Generation of Communication Area

To generate the communication area, the handling function **"SEND-ALL" in the S5 program must be called at least once per cycle.** Depending on blocking size and when working with 2DB several "SEND-ALL" runs are necessary. The "SEND-ALL" must be called at the latest 3 seconds after starting the CP (cold re-start or SYNCHRON). It is, however, not possible to process it immediately after SYNCHRON. If several interfaces are in use, it is recommended to synchronize the MODBUS interface last, or, in the event of more than one MODBUS interface, to call a few "SEND-ALL" in between synchronsing.

The CP copy procedure is complete when the identifier "0001H" (operating mode "1DB") and/or "0002H" (operating mode "2DB") is dynamically displayed in the condition code word of the "SEND ALL" handling function.

In the event of errors during reading of the data area (e.g. DB does not exist or not at least 252 DW long; SEND-ALL not on time), each received function code is replied to with exception code 04 ("PLC in Stop"), and PLC jobs are rejected with error "9" in the condition code word ANZW.

2.1.4 Cold Re-Start of the CP

After a cold re-start of the CP (pressing the CP Operating mode switch) the specified S5 data area is read by the driver and stored in the CP as the communication area.

2.1.5 Initial Start of PLC

After a power failure, warm or cold re-start of the PLC, the handling block "SYNCHRON" (= synchronization of interface between PLC and CP) must be called in the relevant organization block for each interface in use. After processing the synchronization request from the PLC, the CP generates the CP communication area.

2.2 Division of the Communication Area

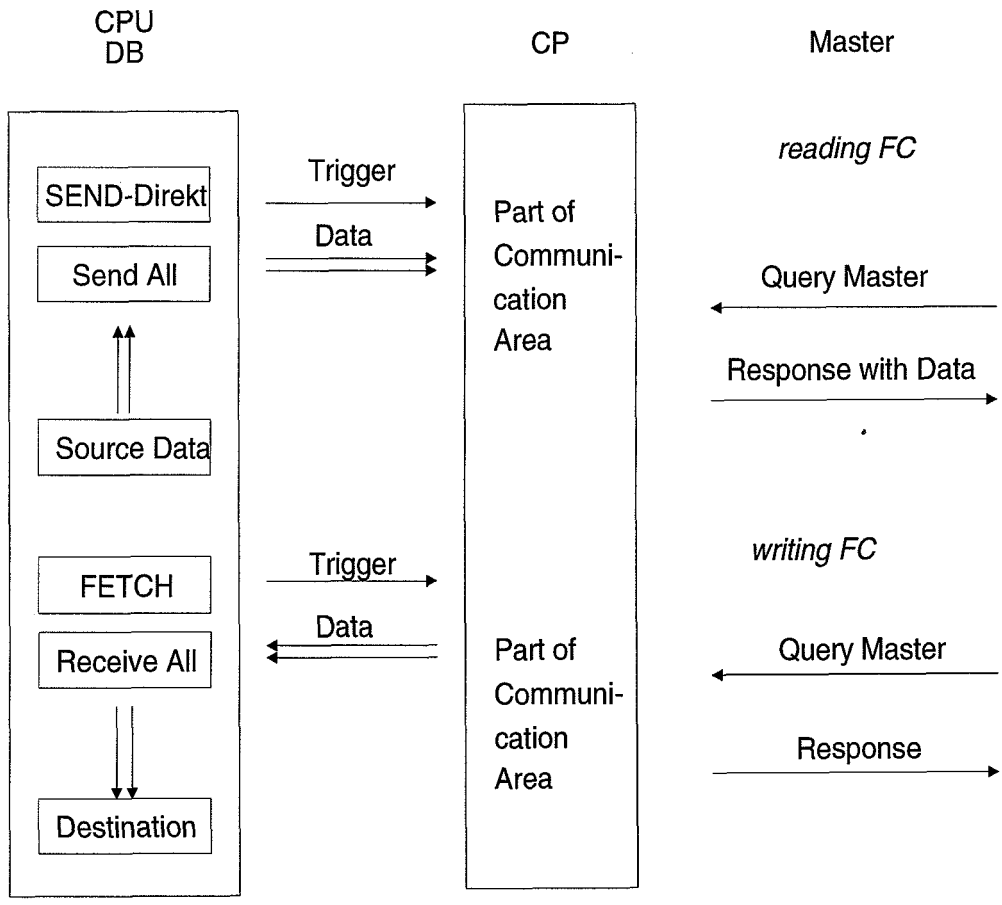
2.2.1 Seperate Areas for Writing and Reading

If it is possible to divide the communication area into one part where there is reading access only, and another part where there is writing access only, we recommend the following parameter assignment:

without exception code 06, without waiting time

In this instance it is also allowed to have several small areas for writing by the master and/or the S5, but there must be no single register to which the master and the S5 have writing access. Reading accesses on to the complete communications area are possible both by the master and/or by the S5 which is not relevant for the parameterisation.

Mode of Operation:



Case 1: **Parameter Assignment: without Co-ordination Flag**

SEND is only called in the S5 program if there has been a change in the source-DB data.

FETCH is called cyclically and up-dates the destination-DB.

Case 2: **Parameter Assignment: with Co-ordination Flag**

As "without exception code 06" has been parameterized, the co-ordination flag is used to report writing accesses of the master only; but it does not disable access to the communication area.

SEND is only called in the S5 program if there has been a change in the source-DB data.

FETCH must now be called conditionally when the co-ordination flag has been set, followed by reset of the co-ordination flag by S5 (see Para. 2.2.3 Co-ordination Flag).

2.2.2 Areas for Writing and Reading are not Seperate

Case 1: **Parameter Assignment: without Co-ordination Flag**

If it is not required to implement data-protection to the areas prior to overwriting by the CPU or by the master, parameter assignment can be "without co-ordination flag".

Case 2: **Parameter Assignment: with Co-ordination Flag**

If the areas are not seperate, and data protection is required prior to overwriting by the master, it must be possible to disable the communication area with the co-ordination flag (=COOR) for a certain period of time. This is necessary in order to avoid that the master and CPU write on to the same register "simultaneously", and to prevent the master making any further changes to the data between reading of the current status and re-writing of a changed value to the S5.

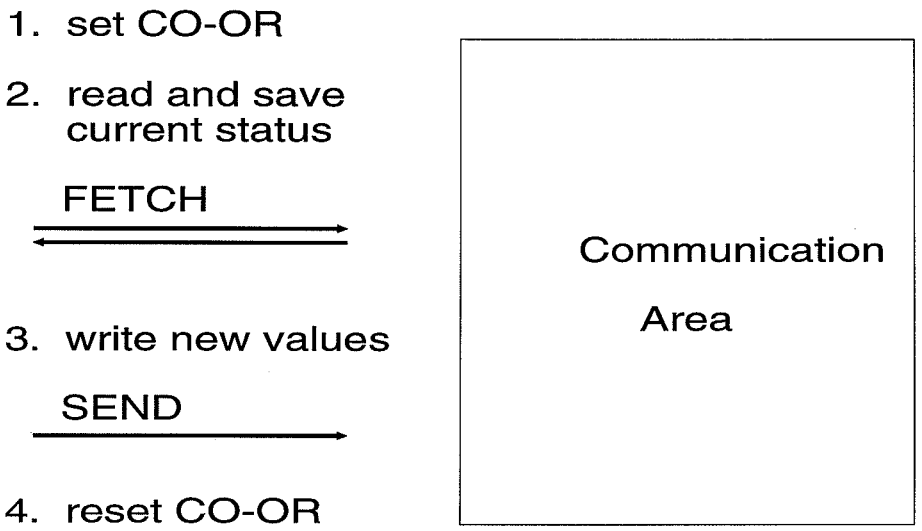
In this instance parameter assignment with Exception Code 06 is a mandatory requirement.

The master cannot access the area while it is disabled, and receives the CP busy exception code 06. This message is instantenous, or, if a waiting time is in operation, only if the co-ordination flag has not been reset on time.

Please find below an illustration of how the Co-ordination flag is handled.

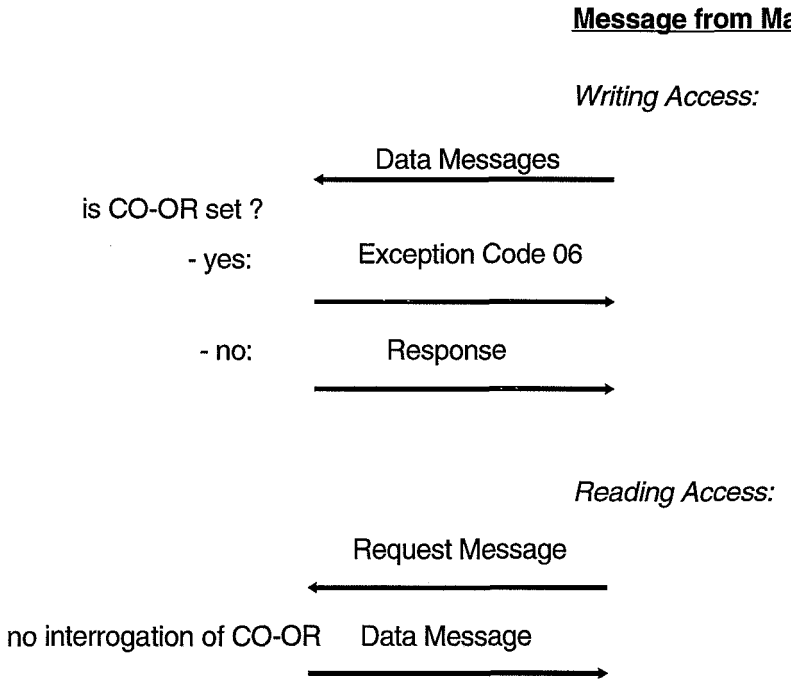
S5 Program:

writing access of S5:



reading access of S5:

If the co-ordination flag was set by the master by means of a writing access, FETCH must be called in the S5 program to read the new data. The co-ordination flag must now be re-set again by the S5.



Case 2a: **Parameter Assignment: without waiting time**

The master requires a reply immediately (e.g how many slaves in the network, short reply monitoring time), and processes the message 06 "CP busy" (e.g. master repeats after a certain time).

Case 2b: **Parameter Assignment: with waiting time**

This operating mode does not meet the requirement of very fast replies, and can only be parameterized, if the master can wait for a reply for a time > 100 ms.

The waiting time (see Para. 2.2.4 "Waiting Time KTTIM") is started in all instances where exception code 06 should be sent. This is the case for set co-ordination flag or simultaneous writing access of the CPU.

During the waiting time, the driver carries out continuous checks as to whether the function code can be processed (e.g. COOR set?). As soon as this is possible, the function code is processed. Should processing not yet be possible after the waiting time has elapsed (e.g. COOR not re-set), exception code 06 is sent.

2.2.3 Co-ordination Flag

The co-ordination flag is realized by the "interprocessor communication flag" area of the dual port RAM.

It is important to pay attention to the fact that the used flag is kept free in the S5 user program. Furthermore the relevant interprocessor communicatiom flag area of 32 byte length must be released on the CP (jumper settings slot SL27, see Operating Manual COM525).

Please find below a print out of the function block to be used for reading and/or writing of the co-ordination flag (=interprocessor communication flag).

Function Block "COO-FLAG"

Parameter: READ:"0" = Write
"1" = Read
SSBY:Interface Number, Flag Byte Number

```
FB9          SPRM-B                                   LEN=42  ABS
                                                        PAGE 1

SEGMENT 1
NAME  :COO-FLAG
BEZ   :READ  I/O/D/B/T/C: E    BI/BY/W/D: BI
BEZ   :SSBY  I/O/D/B/T/C: D    KM/KH/KY/KC/KF/KT/KZ/KG: KY

000B          :LW   =SSBZ      SAVE PARAMETERS "INTERFACE
000C          :T    FW200      NUMBER" AND "BYTE NUMBER"
000D          :
000E          :L    KHEEC8     LOAD ADDRESS "FY200"
0010          :L    KHFEFF     LOAD ADDRESS "VECTOR REGISTER"
0012          :TNB  1          WRITE SSNR ON TO VECTOR REGISTER
0013          :                replace with TNW for 150U!
0014          :L    KB238
0015          :T    FY200      DESTINATION ADDRESS IN FW200
0016          :
0017          :L    KHF200
0019          :L    FY201
001A          :+F              CALCULATE SOURCE ADDRESS
001B          :
001C          :L    FW200      LOAD DESTINATION ADDRESS
001D          :
001E          :A    =READ      "1" = READ ; "0" = WRITE
001F          :JC   =M001
0020          :TAK
```

```
0021 M001    :
0022         :TNB  1          TRANSFER
0023         :                replace with TNW for 150U!
0024         :BE
```

FB9 for PLC 155U:

Parameter: READ:"0" = Write
"1" = Read
BYTE:Flag Byte Number

```
FB9    SPRM-B                                          LEN=27  ABS
                                                         PAGE 1

SEGMENT 1
NAME  :COO-FLAG
BEZ   :READ      I/O/D/B/T/C:  E  BI/BY/W/D: BI
BEZ   :BYTE      I/O/D/B/T/C:  D  KM/KH/KY/KC/KF/KT/KZ/KG: KF

000B         :LW   =BYTE          "BYTE NUMBER"
000C         :ADD  DH 000F F200   ADDR.INTERPR.COMM. FLAG FYx
000D         :LW   =BYTE
000F         :ADD  DH 000E FC00   ADDRESS FLAG BYTE FYx
0010         :A    =READ          READ ?
0013         :JC   =READ          ==> READ
0014         :TAK                 ==> WRITE
0015 LESE    :TNW 1               TRANSFER 1 BYTE
0017         :BE
```

2.2.3.1 Handling of Co-ordination Flag

The co-ordination flag may be set either by the S5 user or by the CP. However,it can only be erased by the S5 program.

After receiving writing access (function code 05, 06, 15 and 16) the CP carries out an interrogation of the flag only if exception code 06 is being used. If, in this instance, the flag has been set, the CP rejects writing accesses with exception code 06 ("PLC busy/CP busy"). If the co-ordination flag has not been set, after processing the function code the flag bit is superimposed.

When operating without exception code 06, there is no interrogation of the co-ordination flag (no disabling) during reception of function codes 05, 06, 15 and 16, however, the flag bit is set after processing the function code.

When receiving reading accesses, the flag bit is not interrogated.

When operaring with exception code 06, the co-ordination flag should be set in the S5 user program before activating writing accesses (see Para. 7.1 "Writing Accesses") and it should be reset after completion of the handling function with or without error. This avoids simultaneous writing on to the CP communication area by the host computer and the PLC, because in this case the received function codes are rejected with exception code 06.

When triggering reading accesses, it is not necessary to superimpose the co-ordination flag relating to the PLC.

2.2.4 Waiting Time KTTIM

Parameter assignment of the waiting time KTTIM is only possible when exception code 06 is in operation. The waiting time is started in all those instances where exception code 06 should be transmitted. There are two possible causes for exception code 06:

1. After receiving a writing function code 05, 06, 15 or 16, the co-ordination flag is interrogated by the CP. If the flag bit is set, CP525 starts the parameterized waiting time KTTIM and carries out a cyclic test of the set co-ordination flag (amount of interrogations during the waiting time depends on further activities running on the CP and is therefore variable).

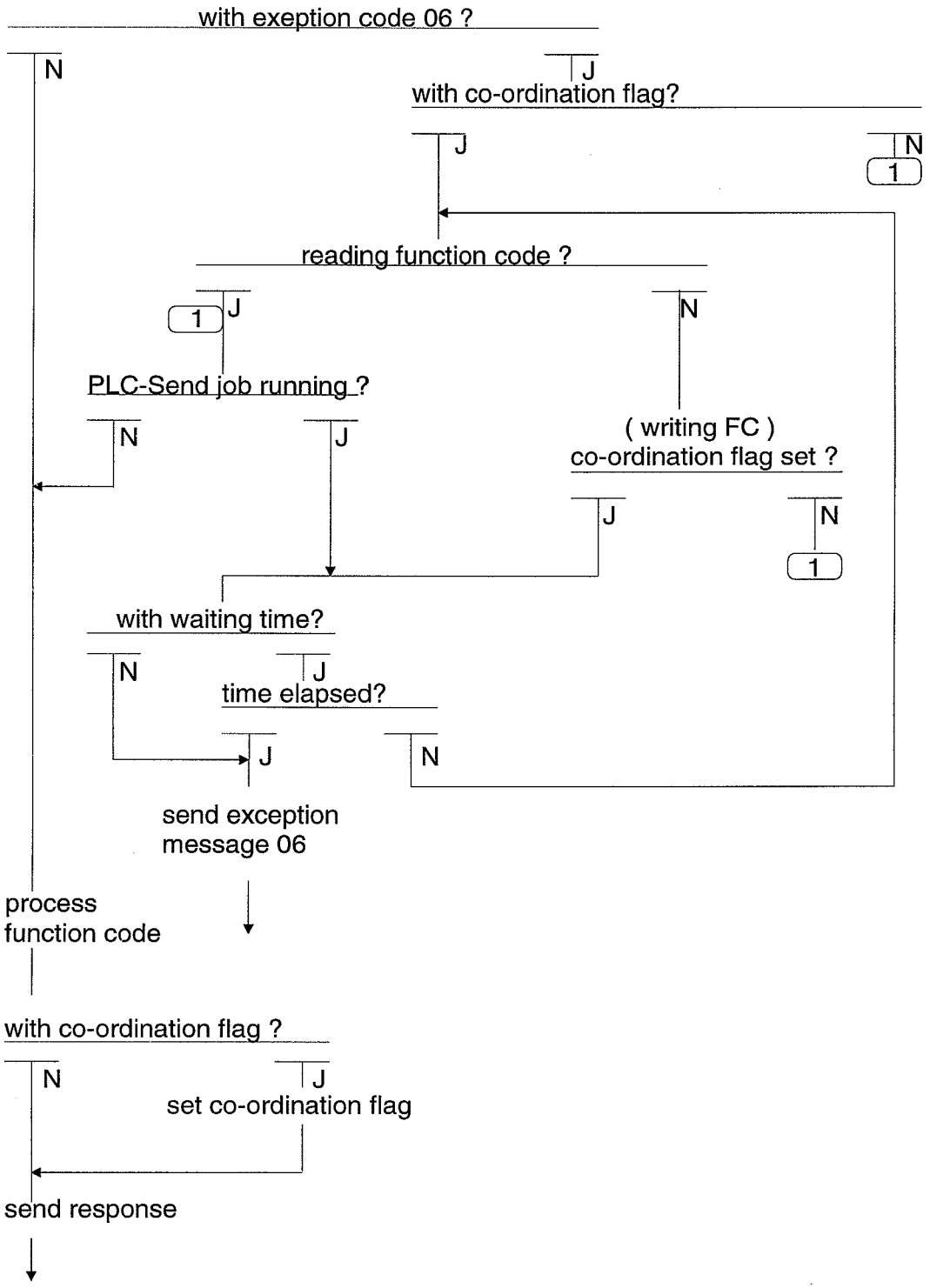
   If in the meantime the flag bit has been re-set, the function code is processed.

   If the flag bit is still set, and the waiting time KTTIM has elapsed, the received function code is replied to with exception code 06.

2. There is another occasion where the waiting time is started, and this is in the event of a message conflict, i.e. a writing or reading function code from the master is received while a PLC job SEND is running on the CP.

If another query is received while the waiting time is running, exception code 06 is sent at once as the reply.

Sequence in CP



2.3 Cycle Test by the CP

The monitoring time which has been parameterized according to Para. 3.2.4.2 is started by the CP after generating the CP communication area.

If during this first time period no handling function is recognized by the CP activated in the S5 program, it takes the initiative by requesting the "SEND-ALL" function which the CP monitors in 3 seconds (2nd time period) for firmware specific reasons.

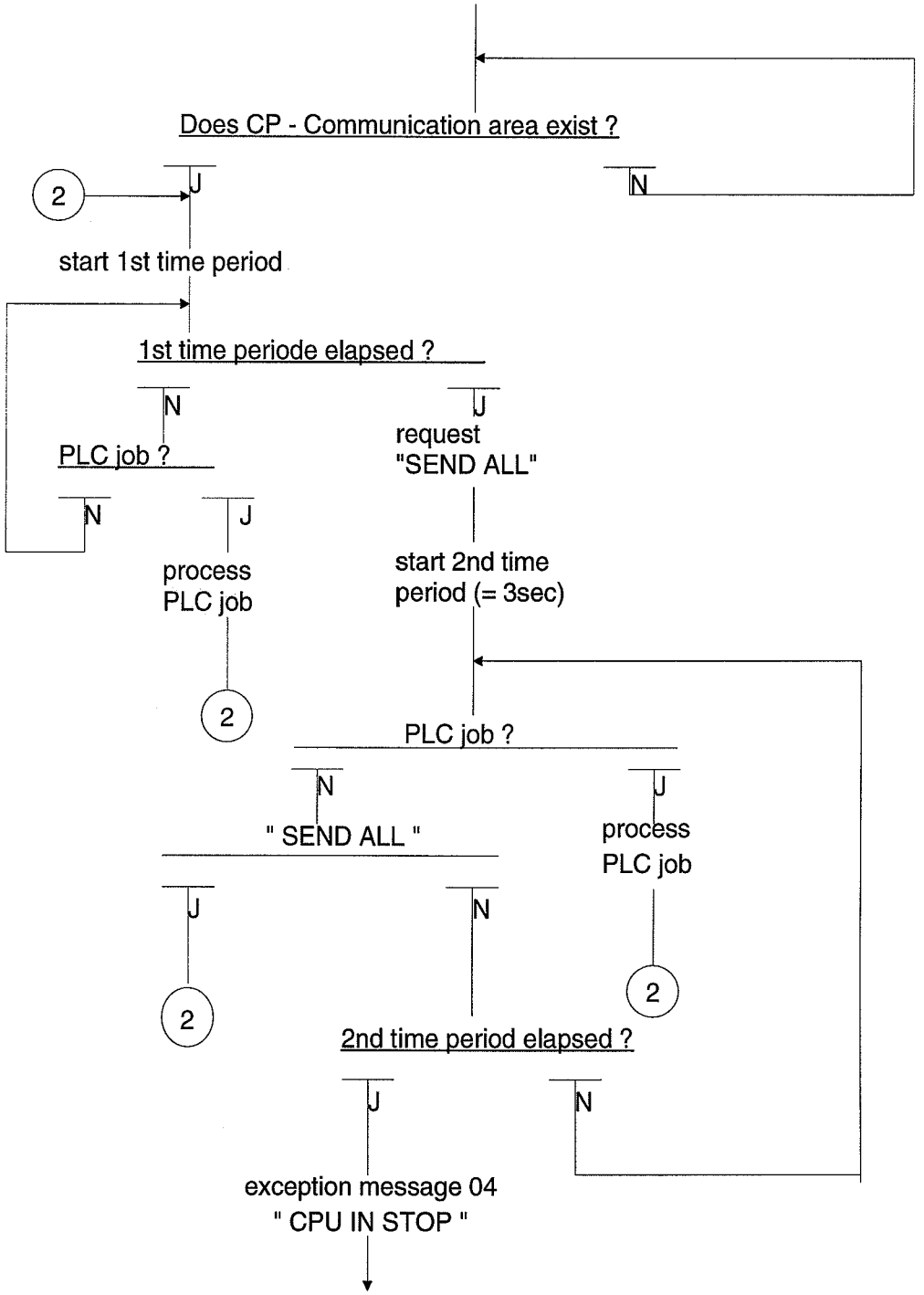
If no "SEND ALL" function is received during the second time period, or if the S5 program does not trigger a handling function during this time, the CP rejects all received function codes (exception: function code 08 "Loop Back Test") with exception code 04 ("PLC in Stop").

The next "sign of life" from the PLC re-activates the cycle test and exception message 04 is cancelled!

If the CP recognizes the requested "SEND-ALL" function or any handling function activated on PLC initiative (e.g. "SEND" job), the cycle test is started again commencing with the first time period.

In order to carry out the described test function, handling function "SEND ALL" must be cyclically called in the S5 user program. When calling the "SEND ALL" function, the CP reads the first data word of the parameterized source data block, however, without evaluation.

Please note at this point that the handling functions triggered in the PLC (e.g. "FETCH") receive a higher priority than the "SEND ALL" function requested by the CP in the second time period. This means that the CP revokes the "SEND ALL" request, the triggered handling function is processed, and after its completion the cycle test is started again beginning with the first time period.


Does CP - Communication area exist ?
J
N
2
start 1st time period
1st time periode elapsed ?
N
J
PLC job ?
request
"SEND ALL"
N
J
process
PLC job
start 2nd time
period (= 3sec)
2
PLC job ?
N
J
" SEND ALL "
process
PLC job
J
N
2
2
2nd time period elapsed ?
J
N
exception message 04
" CPU IN STOP "

The following points describe how the "SEND" and/or "FETCH" jobs activated in the S5 program may influence the cycle test initiated by the CP. In this instance the time of triggering (= first or second time period) is irrelevant!

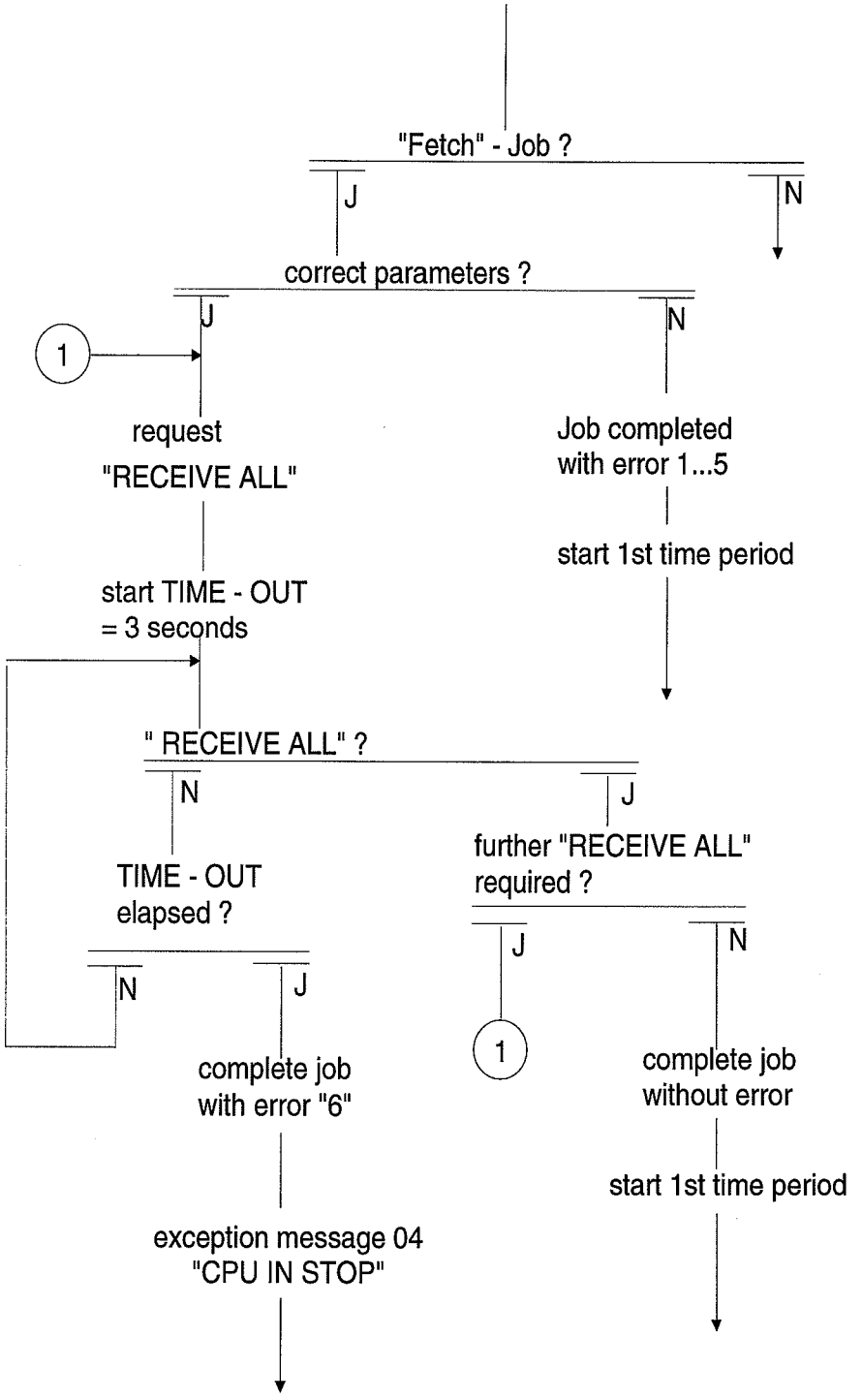
2.3.1 Monitoring of a "FETCH" Job

In order to completely process a "FETCH" job, the handling function "RECEIVE ALL" for data transfer CP -- PLC is called by the CP. The amount of "RECEIVE ALL" functions requested by the CP depends on the volume of data to be transferred and/or the specified blocking size.

After evaluating the transferred parameters the CP requests the "RECEIVE ALL" function and expects it within the next 3 seconds. Completion of the "FETCH" handling function without/with error results in a new start of the cycle test beginning with the first time period.

If the requested "RECEIVE ALL" function could not be processed within the TIME-OUT (e.g. PLC in STOP), the CP completes the handling function "FETCH" with exception code "6" in the condition code word. Furthermore, all function codes received from DHP from this point in time (exception: function code 08 "Loop Back Test") are rejected with exception code 04 ("PLC in stop").

The next sign of life from the CPU re-triggers the cycle test and exception message 04 is cancelled!.


"Fetch" - Job ?
J
N
correct parameters ?
J
N
1
request
"RECEIVE ALL"
Job completed
with error 1...5
start 1st time period
start TIME - OUT
= 3 seconds
" RECEIVE ALL" ?
N
J
TIME - OUT
elapsed ?
further "RECEIVE ALL"
required ?
N
J
J
N
1
complete job
with error "6"
complete job
without error
start 1st time period
exception message 04
"CPU IN STOP"

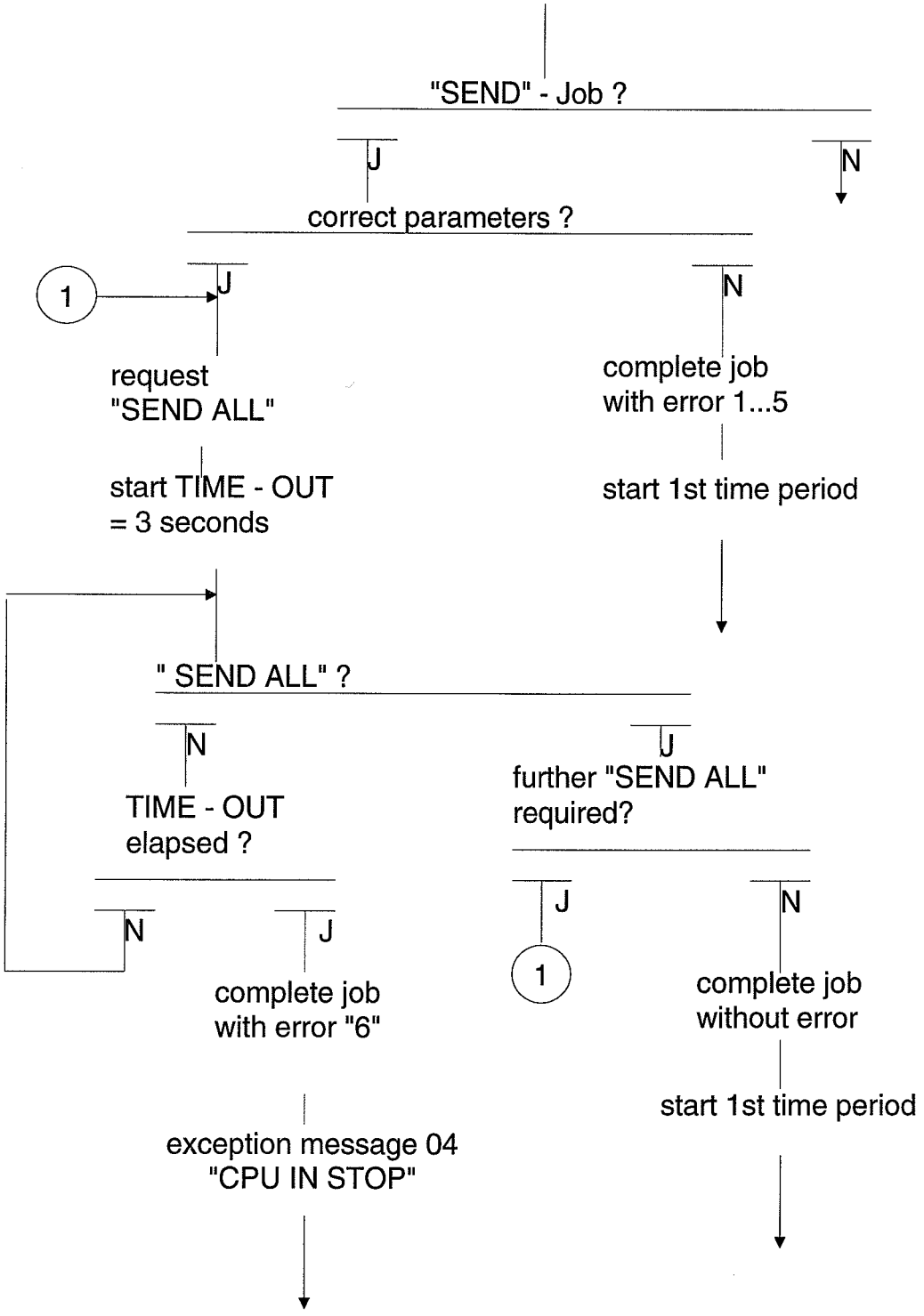
2.3.2 Monitoring of a "SEND" Job

In order to completely process a "SEND" job, the handling function "SEND ALL" for data transfer PLC -- CP is requested by the CP. The amount of "SEND ALL" functions requested by the CP depends on the volume of data to be taken over and/or the specified blocking size.

After evaluating the transferred parameters the CP requests the "SEND ALL" function and expects it within the next 3 seconds. Completion of the "SEND" handling function without/with error results in a new start of the cycle test, beginning with the first time period.

If the requested "SEND ALL" function could not be processed within the TIME-OUT (e.g. PLC in STOP), the CP completes the handling function "SEND" with error number "6" in the condition code word. Furthermore, all function codes received from DHP from this point in time (exception: function code 08 "Loop Back Test") are rejected with exception code 04 ("PLC in stop").

The next sign of life from the CPU re-triggers the cycle test and exception message 04 is cancelled!



2.3.3 Undefined PLC Jobs

Undefined PLC jobs, or PLC jobs not understood by the CP, also lead to a new start of the cycle monitoring.

3. Installation, Parameter Assignment and Loading of Special Driver

3.1 Installation of Special Driver on PG

The library "COMLIBR7.525" is located on the supplied 5.25" floppy disk which contains the special driver "S5R007" consisting of interpreter and procedure.

I-S5R007 ⇨ interpreter
P-S5R007 ⇨ procedure

The installation of the driver is carried out on an S5-DOS 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 "COMLIBR7.525" is copied on to the winchester drive user level 0 as follows:

PIP B:=A:COMLIBR7.525[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 COMLIBR7.525[SYS RO].

3.1.1 CP User Programs

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

The CP525 module 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.

3.2 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.

3.2.1 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 **"COMLIBR7"**. The user program chosen in the "PROGRAM SELECTION" mask is automatically entered as the destination.

3.2.1.1 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-S5R007"** (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.

3.2.1.2 Transfer of Procedure

Use function key F4 to obtain the PROCEDURE transfer mask.

HELP function F7 superimposes the procedure **"P-S5R007"**to be transferred, F1 initiates the transfer.

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

3.2.2 Parameter Assignment of Procedure

Starting from the "SELECTION" mask, branch into the "PARAMETER ASSIGNMENT OF PROCEDURE" mask by pressing F6-F2.

The "PARAMETER ASSIGNMENT OF PROCEDURE" mask displays the **procedure name "P-S5R007"** relating to the chosen program as well as the version number.

The input fields **"character length", "parity"** etc. may be filled in as required in accordance with the system configuration.

Use function key F6 to store the information.

3.2.2.1 Input Field "PRIORITY"

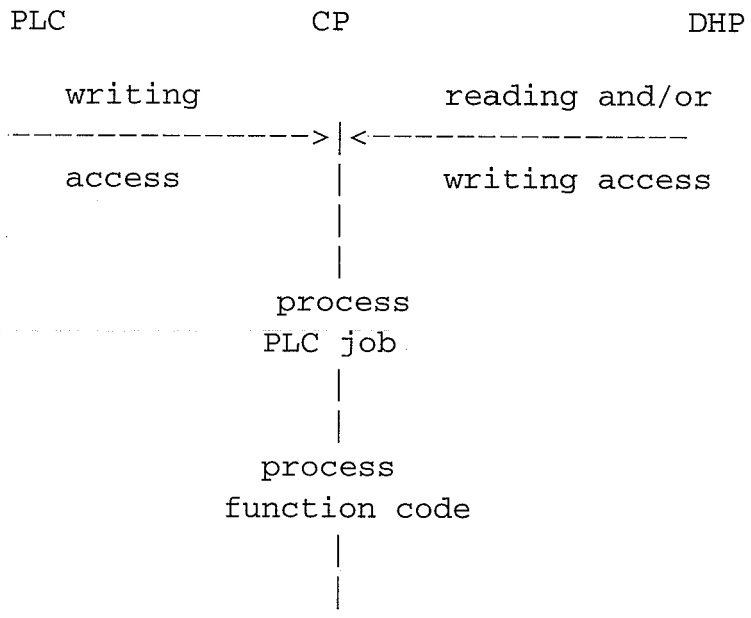
By means of the input field "PRIORITY" exception code 06 (PLC busy/CP busy) can be superimposed or blanked out.

PRIORITY	Function
Higher	with Exception Code 06
Lower	without Exception Code 06

Function: without Exception Code 06

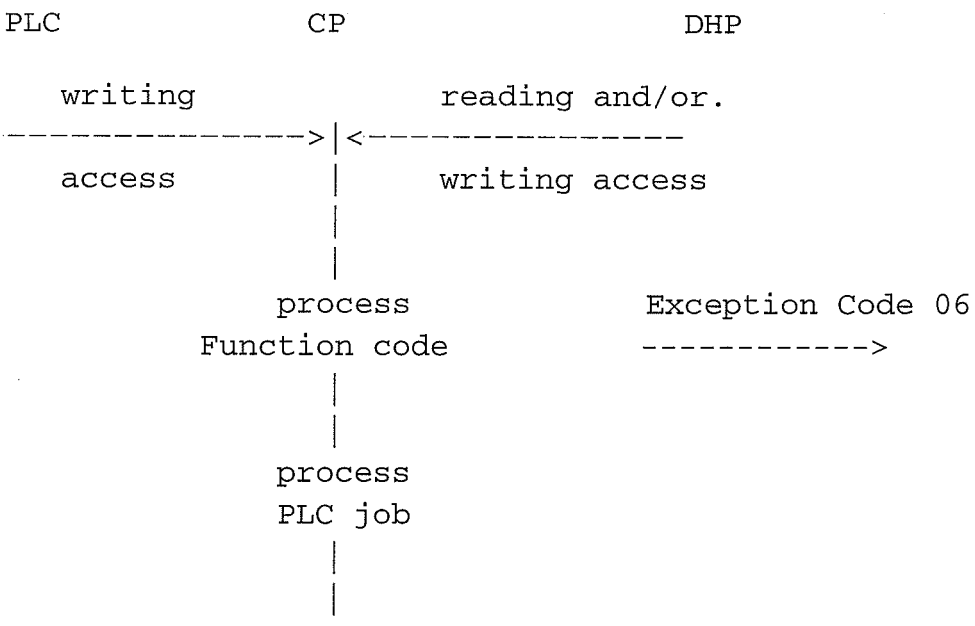
In the event of the CP receiving simultaneously a writing request from the PLC and a writing and/or reading function code from DHP, **the PLC job is processed first followed by the relevant function code.**

This means that access from DHP to the CP communication area is of a higher priority, because it overwrites the data transferred from the PLC in the event of access to the same registers!



Function: with Exception Code 06

In the event of a writing access having been activated by the PLC, and if prior to its execution the CP receives a reading or writing function code from DHP it replies to this message with exception code 06, and processes the PLC job afterwards.



3.2.3 Creation of a Job Block

Starting from the "SELECTION" mask, programming of a job block is initiated by means of function key F1.

A job number (1...223) is entered into the mask "PROGRAMMING OF JOB BLOCK".

Function key F5 leads back to the "JOB PROGRAMMING" mask.

3.2.3.1 Meaning of Job Number A-NR

In order to generate the CP communication area, a data block area must be created for the CP which is read by the CP in the event of a cold re-start of the CP (see Para. 2.1.4) or initial start of the PLC (see Para. 2.1.5) and which is stored in the RAM memory as the communication area.

The job number A-Nr (1...223) allocated to the programmed job block states **the data block to be read** when using **the "1DB" operating mode** (see Para. 2.1.1 CP Communications Area for Operating Mode 1DB).

When using **two data blocks** the parameter A-NR determines the **first data block to be selected**, the subsequent one is used as number two.

From each data block the data words 0...251 are taken over and stored on the CP as the CP communication area.

3.1.4 JOB(x) PROGRAMMING

In the "JOB PROGRAMMING" mask the input fields **"Job:"** and **"Job Type:"** must be filled in as follows: -

Job:	SEND
Job Type:	DATA BLOCK

The input field "**CPU-NR.**" must be specified only for multi processor operation.

In the event of a cold re-start of the PLC and/or the CP, the parameterized source data area is read from the CPU specified here and stored in the CP; additionally this CPU is used to carry out the cycle test as described in Para. 2.3.

For single processor operation the input field "CPU-NR." should be skipped.

3.2.4.1 Changed Meaning

When used with the special driver S5R007 the remaining input fields have a changed meaning as against their standard function.

3.2.4.2 Input Field "DB-NR." (PLC Cycle Monitoring)

The sequence of the PLC cycle monitoring is desribed in Para. 2.3. The PLC cycle is monitored by means of two time periods started by the CP. The length of the **first time period** can be set in 20 ms intervals using the "DB-NR." parameter of the programmed job block.

DB - NR.	first time period
3	0 ms
4	20 ms
5	40 ms
:	:
255	5040 ms

If **0 ms** is specified as **the monitoring time**, the CP does **not check the PLC cycle**!

3.2.4.3 Input Field "Destination - Word Address" (Slave Address)

The **slave address** is preset with the **"destination-word address"** of the programmed job block. The slave address for each CP interface may be located within the range of 1-255.

3.2.4.4 Input Field "Co-ordination Flag Optional"

When using the special driver with a co-ordination flag, **the used flag byte** is pre-set using **the byte value** of the "Co-ordination Flag Optional" parameter. The CP uses **bit 0 as the flag bit**.

The functions
- amount of data blocks
- with/without co-ordination flag
can be parameterized by using the **bit number** of the "Co-ordination Flag Optional" parameter.

Bit number	Amount DB	Coordination Flag
0	1	without
1	1	with
2	2	without
3	2	with

For bit numbers greater three the CP stores an error number in the SYSTAT area.

The maximum amount of registers and/or bit numbers depends on the amount of used data blocks.

Amount DB	max. Register numbers	max. Bitnumbers
1	0...251	0...4031
2	0...503	0...8063

3.2.5 PROGRAMMING OF JOB (x+1)

As from Version 6 of the Special Driver, it is possible to parameterize **RS485 operation**, the **waiting time KTTIM** and a **factor for character delay time**. This requires programming of a **second job** the job number of which must be the direct subsequent number to the first job number. The three parameters of this job block can be selected independently of each other, they do not require each other (e.g. also KTTIM only, without RS485 operation).

If operation without waiting time, without RS485 and with standard character delay time is required, programming of this job is not required!

The input fields "Job:" and "Job Type:" must be filled in as follows in the "JOB PROGRAMMING" mask:

Job : SEND

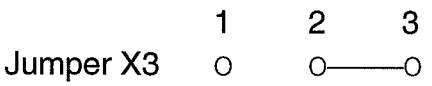
Job Type : DATA-BLOCK

The input field "CPU-NO." is irrelevant.

3.2.5.1 Input Field "DB-NR" (RS485 Operation)

If the "DB-Nr" field is preset with "3", operation is with RS485, any other input means that operation is without RS485. Assignment of the parameter with "03" results in the driver operating in RS485 MODE. This is only possible when using CP524 and the RS422/RSS485 interface sub-module. When the RS485 interface is in use, it is possible to connect up to 32 slaves to one network in half duplex operation.

For this purpose **jumper X3** on the module 6ES5 752-0AA42 must be set as follows:



For the inserted **jumpers X10** and **X11** leads 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

It is required to fit a 120 Ohm terminating resistor at the end of the network.

The special driver checks whether a CP524 is in use with an RS422/RS485 sub-module. It is up to the user to ensure that the RS422/RS485 hardware runs in half duplex operation (jumper setting).

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

In this operating mode BREAK is not evaluated (neither in ANZW nor in SYSYAT)

3.2.5.2 Input Field "Destination-Word Address" (Waiting Time KTTIM)

This input field is used for parameter assignment of the waiting time KTTIM. This waiting time starts prior to transmission of exception code 06 (see Para. 2.2.4). The waiting time KTTIM is entered value * 100 ms. If the parameter is preset with " 0", operation is without waiting time. It is only possible to parameterize a waiting time if operation is with exception code 06.

3.2.5.3 Input Field "Co-ordination Flag Optional" (Factor for Character Delay Time)

Assignment of the byte-field "Co-ordination Flag" is irrelevant. The bit field serves to set a multiplication factor for the character delay time. The character delay time should only be changed if the link partner cannot meet the required times. The resulting character delay time = factor * table value (see Para. 4.1 "Transmission Procedure").

If the field is left blank, or value 0 is entered, the factor is 1.

3.2.6 Programming Example

J O B

Job - Nr:	010	
Job:	SEND	
Job Type:	DATA BLOCK	
CPU - Nr:		
DB - Nr:	003	
Dest. Word Address:	00025D	0019H
Co-ordination Flag Optional:	010.3	

In the example above, the parameters were set as follows:

- data blocks to be read DB10 and DB11
- without cycle monitoring
- slave address = 25
- co-ordination flag FY10.0
- operation with 2 DB's
- operation with co-ordination flag

no 2nd job block necessary because

- no RS485 interface
- no waiting time KTTIM
- standard character delay time

3.2.7 Parameter Assignment Error Job Block

If the CP recognizes one of the following parameter assignment errors during parameter assignment of the COM job block, it enters an error code into the SYSTAT area; furthermore the **interface specific LED** of the CP **blinks** in 100ms intervals!

- no "SEND" job block programmed with COM525
- job type unequal "data block"
- bit number of "Co-ordination Flag Optional" parameter greater three
- RS485 operation parameterized, but no CP524 with RS422/RS485 module (2nd job block DB-No = 03)
- waiting time KTTIM parameterized for operating mode "without exception code 06"

3.2.8 STORE JOB(x)

The job block is accepted and stored using function keys F6-F8-F6.

3.3. Loading of Special Driver into CP

The loading procedure of the special driver S5R007 is identical to the procedure used for standard computer link CL512.

The transfer of the user program into the CP and/or the programming of an EPROM module is described in detail in

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

4 Communication TDC3000 -- CP

4.1 Transmission Procedure

Data transmission is carried out without handshake.

The master (=host computer) has the initiative, the slaves (=CP) send on request only. Message traffic from slave to slave is not possible.

Data traffic begins with the slave address (from 1-255) followed by the function code (01, 02, 03, 04, 05, 06, 08, 15,or 16), the address field, the data, and a CRC check sum.

The entire message received by the CP consists in the event of function codes 01-08 always of 8 bytes; in the event of function codes 15 and/or 16 it consists of a maximum of 264 and/or 263 bytes.

The message end is recognized by the CP when no transfer has occurred during the time required to transmit three and a half characters (cf. MODBUS Protocol Reference Guide Pages 1-8). This means that the TIME-OUT is baud rate dependent!

Used times:

Baudrate	TIME-OUT
19200	4ms
9600	4ms
4800	8ms
2400	16ms
1200	32ms
600	64ms
300	128ms
200	193ms
150	256ms
110	350ms
100	386ms
75	512ms
50	772ms

To enable the CP to recognize the error "character delay time" despite this end criterion, the driver calculates the message end in accordance with the received message parameters (= CRC - check "High") and after reception of the last character it starts the baud rate independent TIME-OUT. If no further characters are received by the CP within this time the received data string is considered to be correct and is evaluated. If a further character is received, the CP ignores the received character sequence and returns to basic setting.

The maximum time which may elapse between receiving two characters (character delay time ZVZ) is also equivalent to the time required to transmit three and a half characters. If no further character is received by the CP within this time, the message piece received so far is considered invalid. If the link partner does not meet the required times, the character delay time can be changed using the 2nd job block (see Para. 3.2.5.3).

If illegal function codes are transmitted, exception message number 01: "Illegal Function Code" is sent to the master.

The CRC check character is calculated from the following polynominal:

$$X^{16}+X^{15}+X^{2}+1$$

The result is added to the message during transmission (first the low byte, then the high byte). During reception all data is subject to the same CRC check. On correct transmission, the received CRC sum and the internally generated CRC sum should tally. Only then is an action initiated. If the CRC word is incorrect, no reaction is made.

Query from DHP

Meaning		HEXA
Slave-Address 5		05
Function - Code 03		03
Start Address	High	00
32 (0020H)	Low	20
Amount of Words	High	00
1 (0001H)	Low	01
Check sum CRC-Check	Low	84
"	High	44

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 03		03
Byte - Counter (2 Bytes)	02	
Contents DW 32	High	8E
Contents DW 32	Low	C3
Check Sum CRC-Check	Low	6C
"	High	75

4.1.1 Broadcast Message (Slave Address 0)

If the slave address is Zero, the driver addresses all slaves on the bus. Therefore there are no responses from the CP to the host computer.

Only writing function codes are allowed with a broadcast message: -

- Function Code 05 ---> Modify Coil Status
- Function Code 06 ---> Modify Register Content
- Function Code 15 ---> Force Multiple Coils
- Function Code 16 ---> Preset Multiple Register

Function codes other than those above are ignored when the slave address is Zero.

5 Access to CP Communication Area

The connected host system can access the CP communication area by means of function codes.

Function-code	Function
01	- Read Coil Status
02	- Read Input Status
03	- Read Holding Registers
04	- Read Input Registers
05	- Modify Coil Status
06	- Modify Register Content
08	- Loop back Test
15	- Force Multiple Coils
16	- Preset Multiple Registers

Function codes 03, 04, 06 and 16 address the CP communication area by register, starting from 0....251 and/or 0....503 respectively.

Function codes 01, 02, 03 and 15, however, access the CP communication area by bit, bit numbers starting from 0....4031 and/or 0....8063 are permitted.

In the event of a non-existent or not sufficiently long CP communication area being accessed during a query from DHP, an exception message is output.

The maximum length of user data (= message without header and two byte CRC check) of a message requested by DHP is 255 bytes (= 2040 bits) for function codes 01 and 02.

For function codes 03 and 04 the maximum length of user data is 254 byte (= 127 words).

If a longer message is requested by DHP, an exception message with exception code 03 "Illegal Data Value" is output by the CP.

The following description of the individual function codes only deals with the special features resulting from connection of the CP to the MODBUS. Basic knowledge of data traffic and message structure is a pre-requisite.

5.1 Function Code 01 --- Read Coil Status

This function enables the user to read the CP communication area bit by bit starting from bit number 0....4031 and/or 0....8063 respectively.

The number of the first bit and the length of the area to be read in bits are transferred to the CP.

Any value between 1 and 2040 is permitted. The maximum amount of bits is pre-determined by the MODBUS protocol, because the byte counter in the response is displayed by a byte (maximum value: FFH=255).

The errors listed below result in an exception message from the CP:

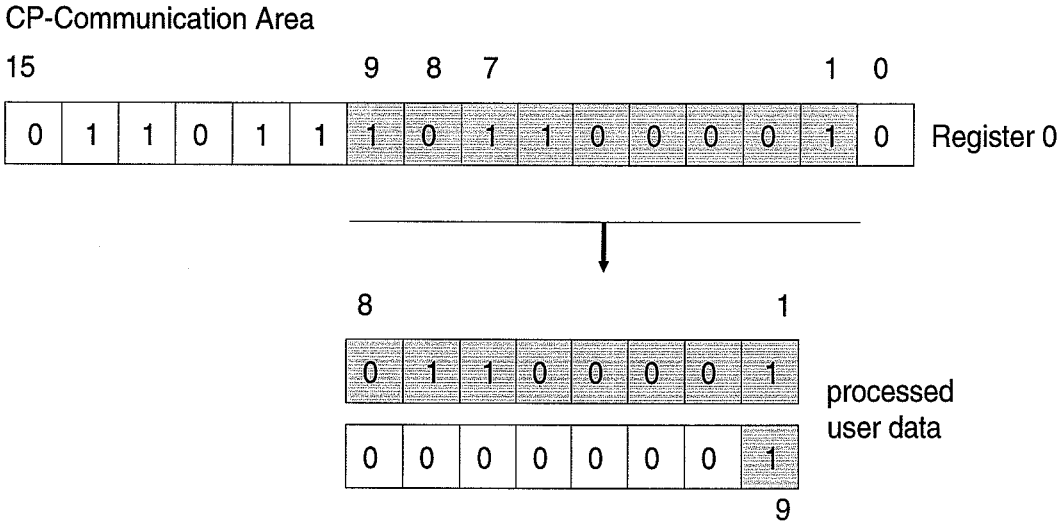
Exception code	Reason for Error
02	Start bit number greater 4031 and/or 8063
03	Amount of bits greater 2040
03	Amount of bits = 0
03	Start bit number + amount of bits greater 4032 and/or 8064

Reading by bit may commence from any bit number, and the addressed bit is the first bit on the transmission line. If the number of the bits to be read cannot be divided by eight (= no byte limit), zeros are added to the remaining bits of the last byte to be transmitted.

Example: Bit number = 1
Bit number = 9

Query from DHP

Meaning		HEXA
Slave Address 5		05
Function - Code 01		01
Start Address	High	00
01 (0001H)	Low	01
Amount of bits	High	00
9 (0009H)	Low	09
Check sum CRC-Check	Low	AC
"	High	48



In the example above two bytes with the values 61H and 01H are transmitted by the CP to the link partner in the form of a response.

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 01		01
Amount of Data Bytes (1 - 255)		02
Contents Byte 1		61
Contents Byte 2		01
Check Sum CRC-Check	Low	29
"	High	FC

5.2 Function Code 02 -- Read Input Status

Due to the fact that communication in the CP is exclusively carried out via the CP communication area already mentioned, this function is equivalent to function 01 described above.

5.3 Function Code 03 -- Read Holding Registers

This command enables the user to read register addresses 0...251 and/or 0...503 of the CP communication area.

The first number of the register and the required amount of registers (16 bit) are transferred in the query. A maximum of 127 registers (1 register = two bytes) may be sent simultaneously, because the byte counter in the response is displayed by a byte (maximum value: FFH=255).

The errors listed below result in exception messages from the CP:

Exception Code	Reason for Error
02	Start Register Number greater 251 and/or 503
03	Amount of Registers greater 127
03	Amount of Registers = 0
03	Start Register Number + Amount of Registers greater 252 and/or 504

Query from DHP

Meaning		HEXA
Slave-Address 5		05
Function - Code 03		03
Start Address	High	00
64 (0040H)	Low	40
Amount of Words	High	00
02	Low	02
Check Sum CRC-Check	Low	C4
"	High	5B

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 03		03
Amount of Data Bytes (2 - 254)		04
Contents Register 64	High	21
"	Low	23
Contents Register 65	High	00
"	Low	00
Check Sum CRC-Check	Low	44
"	High	05

5.4 Function Code 04 -- Read Input Registers

Due to the fact that communication in the CP is exclusively carried out via the CP communication area already mentioned, this function is equivalent to function 03 described above.

5.5 Function Code 05 -- Modify Coil Status

It is possible to alter any bit of a register, but only one bit per call.

When altering individual bits in the CP communication area, the address and the new value are transferred. The bit is set, when the value FF00H (= 65280) is transferred. In the event of the value 0000H the bit is re-set.

The register address in the CP communication area is calculated from the following formula:

(Bit address / 16) = Register address,
remainder is bit number

E.g. The stated bit address was number 30. This results in register address 1 with bit number 14 in the CP communication area.

Bit numbers greater 4031 and/or 8063 are rejected with exception code 02 ("invalid address field"), data fields unequal FF00H and/or 0000H are rejected with exception code 03 ("invalid data value").

Query from DHP

Meaning		HEXA
Slave-Address 5		05
Function - Code 05		05
Start Address for Reg./Bit	High	00
25 (0019H)	Low	19
Data Field	High	FF
65280 (FF00H)	Low	00
Check Sum CRC-Check	Low	5C
"	High	79

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 05		05
Address for Word/Bit	High	00
"	Low	19
Data field	High	FF
"	Low	00
Check Sum CRC-Check	Low	5C
"	High	79

5.6 Function Code 06 -- Modify Register Content

One word only per message is entered into the CP communication area. The allocation of the register addresses corresponds to the allocation for function code 03.

When addressing registers greater 251 and/or 503, the CP sends an exception message with exception code 02.

Query from DHP

Meaning		HEXA
Slave-Address 5		05
Function - Code 06		06
Register Address	High	00
0032 (0020H)	Low	20
Data field	High	12
4660 (1234H)	Low	34
Check Sum CRC-Check	Low	84
"	High	F3

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 06		06
Register Address	High	00
"	Low	20
Daten Field	High	12
"	Low	34
Check Sum CRC-Check	Low	84
"	High	F3

5.8 Function Code 08 -- Loop Back Test

This function code is only processed by the procedure of the driver, and serves to check the connection. The received message is sent back un-altered.

5.8 Function Code 15 -- Force Multiple Coils

Using this function the user may change up to 2040 bits with one query.

In this instance, the message parameters sent by DHP have the following meaning:

- "Address" = Bit Start Address
- "Quantity" = Amount of Bits to be changed
- "Bytecount" = Amount of following "Data Coils"
- n x "Status Coil" = Bitstati 1 = ON
  0 = OFF

The parameter "byte count" is checked by the CP according to the following regulations:

Bytecount = INT(Quantity / 8)
Rest = 0 ⇨ Bytecount = Bytecount
Rest <>0 ⇨ Bytecount = Bytecount + 1

If the received parameter "byte count" does not tally with the byte count calculated by the CP, the CP sends an exception message with exception code 03 "Illegal Data Value".

The CP also generates an exception message in the event of the errors listed below:

Exception Code	Reason for Error
02	Bit Start Address greater 4031 and/or 8063
03	Amount of Bits(= "Quantity") greater 2040
03	Amount of Bits = 0
03	Bit Start Address + Amount of Bits greater 4032 and/or 8064

Example:

Bit Start Address = 07
Amount of Bits = 13
Bytecount = 02
Status Coil 07-14 = 01100100
Status Coil 15-19 = xxx00100

The bits of status coil 2 identified with 'x' are not evaluated by the CP, because only 13 bits of the CP communication area are to be changed.

CP-Area before execution of function code:

15							7							0
1	0	1	0	1	0	0	1	1	1	0	1	0	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1

CP-Area after execution of function code:

15							7							0
0	**0**	**1**	**1**	**0**	**0**	**1**	**0**	**0**	1	0	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	0	1	**0**	**0**	**1**	**0**

Query from DHP

Meaning		HEXA
Slave-Address 5		05
Function - Code 15		0F
Addess	High	00
07 (0007H)	Low	07
Quantity	High	00
13 (000DH)	Low	0D
Bytecount 02		02
Status Coil 7-14		64
Status Coil 15-19		64
Check Sum CRC-Check	Low	FC
"	High	10

Response from CP

Meaning		HEXA
Slave-Address 5		05
Function - Code 15		0F
Address	High	00
07 (0007H)	Low	07
Quantity	High	00
13 (000DH)	Low	0D
Check Sum CRC-Check	Low	24
"	High	4B

5.9 Function Code 16 -- Preset Multiple Registers

Using this function code it is possible to write up to 127 slave registers of the CP communication area.

The message parameters sent from DHP are evaluated by the CP as follows: -

- "Address" = Register Start Address
- "Quantity" = Amount of Registers
- "Bytecount" = Amount of following Data Bytes
- n * "Data" = Register New Values (n * 2Byte)

The CP generates an exception message in the event of the following errors listed below:

Exception Code	Reason for Error
02	Register start address greater 251 and/or 503
03	Amount of registers greater 127
03	Amount of registers = 0
03	"Bytecount" unequal amount of registers * 2
03	Register start address + amount of registers greater 252 and/or 504

Query from DHP

Meaning		HEXA
Slave Address 5		05
Function Code 16		10
Address	High	00
01 (0001H)	Low	01
Quantity	High	00
02 (0002H)	Low	02
Bytecount 04		04
Data Register 1	High	FF
	Low	00
Data Register 2	High	FF
	Low	00
Check sum CRC-Check	Low	56
"	High	B7

Response from CP

Meaning		HEXA
Slave Address 5		05
Function Code 16		10
Address	High	00
01 (0001H)	Low	01
Quantity	High	00
02 (0002H)	Low	02
Check sum CRC-Check	Low	11
"	High	8C

6 Error Messages to the Master

If a message is received incorrecty (e.g. CRC errors or switch on CP in STOP position), the job is ignored. There is no error message to the master.

If errors are reported, the highest bit is set in the function code and an exception message with the following structure is sent back:

Exception Message from CP

Meaning		HEXA
Slave Address 5		05
Function Code 05		85
Exception Code (1-6)		02
Check Sum CRC-Check	Low	F3
"	High	50

The above example illustrates exception 02 ("Illegal data address"), which occurred during function code 05.

In the event of slave address 0 there is never a response, hence no exception message.

6.1 Possible Exceptions and Reasons

Exception Code 01 : Illegal Function
This error is reported when an illegal function code has been sent.

Exception Code 02 : Illegal Data Address
This error occurs when an illegal register address and/or bit number has been addressed in the CP communication area.

Exception Code 03 : Illegal Data Value
This error occurs when, for instance during function code 05, values other than 0000H and/or FF00H have been entered in the data field, or when the amount of net data to be sent exceeds 255.

Exception Code 04 : Failure in associated device
This error occurs when
- the cycle test could not be processed successfully
- when errors have occured during reading of the S5 data area. In this instance each message is replied to with exception code 04 until a cold-restart of either the CP or the S5 is carried out.
- when function codes have been received during the reading of the S5 data area.
- when after a SEND Direkt there is no call of a SEND ALL within 3 sec (also when operation is without cycle test).

Exception Code 06 : Busy, rejected message
This error occurs when
- the writing access is disabled by the co-ordination flag
- a writing access of the CP is being processed at the moment. For a further description refer to Para. 3.2.2.1 "Input Field 'PRIORITY'".

7 Communication S5 --- CP

The S5-user can alter the CP communication area by writing accesses and/or take over the CP communication area by reading accesses into PLC data blocks.

7.1 Writing Accesses

These functions are triggered by calling the "SEND" handling block.

Any job number can be allocated to the parameter "A-NR". The parameter "QTYP" (= source type) must be preset with "DB".

The right hand data DR of the first transmitted word is interpreted as a function code (permitted function codes: 05 and 16).

Word two states the address field.

The following data word(s) display the data field.

An access which has been correctly processed may be checked by the identification "job complete without error" in the condition code word of the triggered send job.

7.1.1 Function Code 05 -- Modify Coil Status

When triggering function code 05 (a bit in the CP communication area is changed) three data words (function code, address field and data field) must be transferred to the CP.

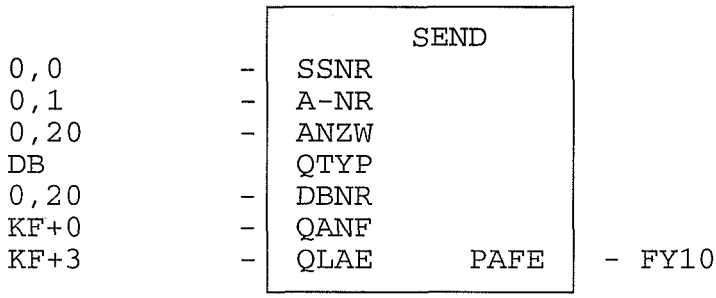
The addressed bit is set, when the value FF00H (=65280) is transferred. For value 0000H the bit is re-set.

The register address in the CP communication area is calculated from the following formula:

(bit address / 16) = register address, the remainder is the bit number

Example: Function code 05

Set Bit 1 of Register Address 2



DB20

	DL	DR	
DW0	00H	05H	Function code
DW1	00H	21H	Bit address
DW2	FFH	00H	Data

7.1.2 Function Code 16 -- Preset Multiple Registers

The registers of the CP communication area can be written on block by block.

By triggering the handling block "SEND", the register start address and register values are transferred.

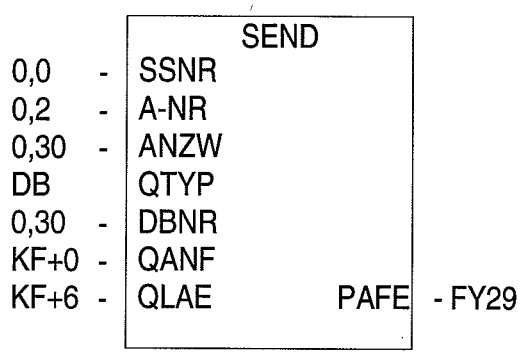
Irrespective of operating mode "1DB/2DB's" a maximum of 126 registers can be written on with one "SEND" job.

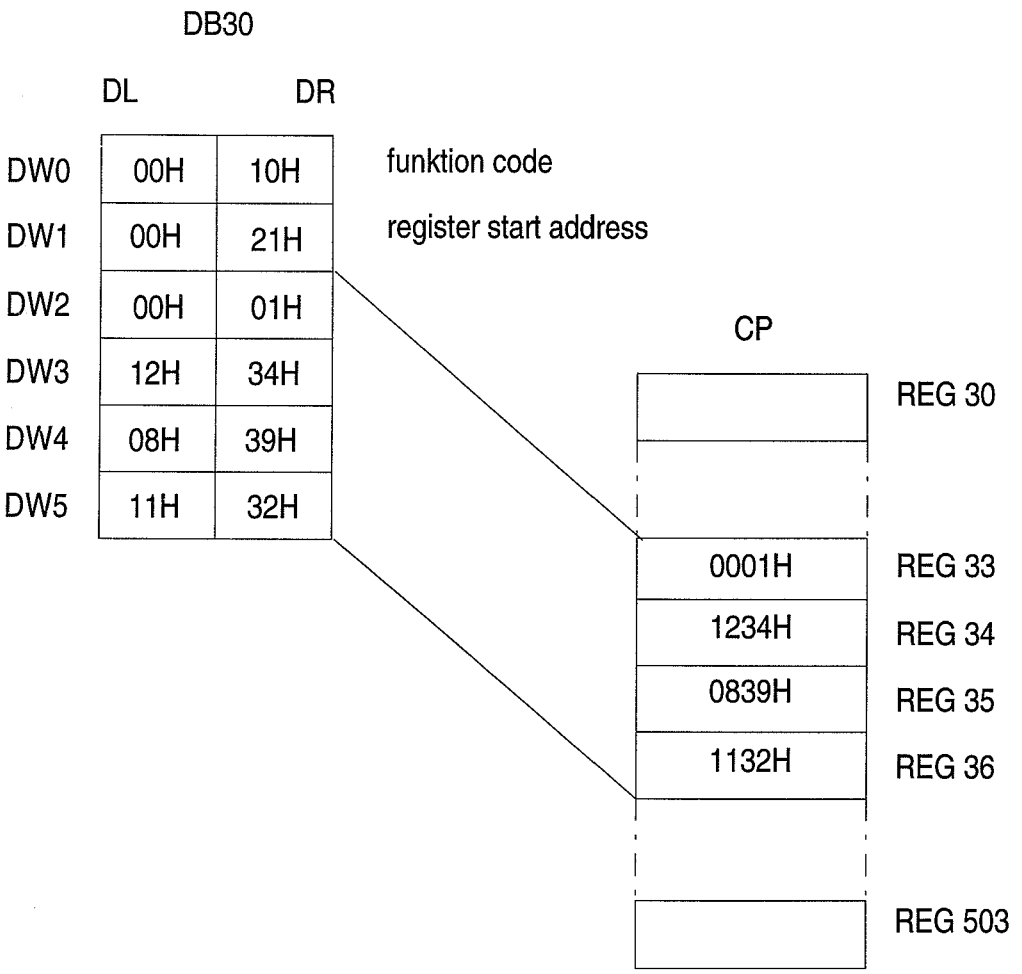
The amount of registers does not have to be explicitly transferred by the S5 program because the CP calculates this value from the parameter "QLAE" (length in words) of the handling function "SEND":

Amount of registers = "QLAE" - 2

Example: Function Code 16

Write on registers 33,34,35 and 36





7.2 Reading Accesses

By triggering the handling function "FETCH", the entire CP communication area or parts thereof can be taken over into the S5 data area. Any job number may be allocated to the parameter "A-NR".

The CP interprets the parameters "DBNR", "QANF" and "QLAE" of the activated handling block as follows: -

Parameter	Function
DBNR	Destination Data Block(s) in PLC
ZANF	- Register Start Address of CP-Communications Area - Destination Data Word in PLC
ZLAE	Amount of Registers

Parameter "DBNR" always states the first destination data block!

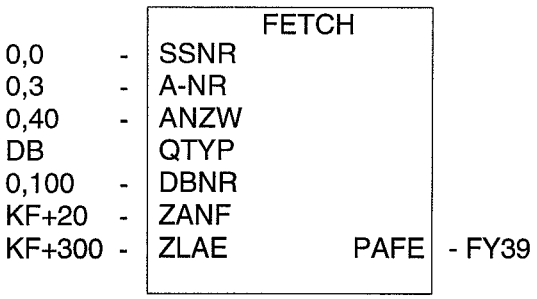
Therefore, the following applies to register start addresses greater 252:

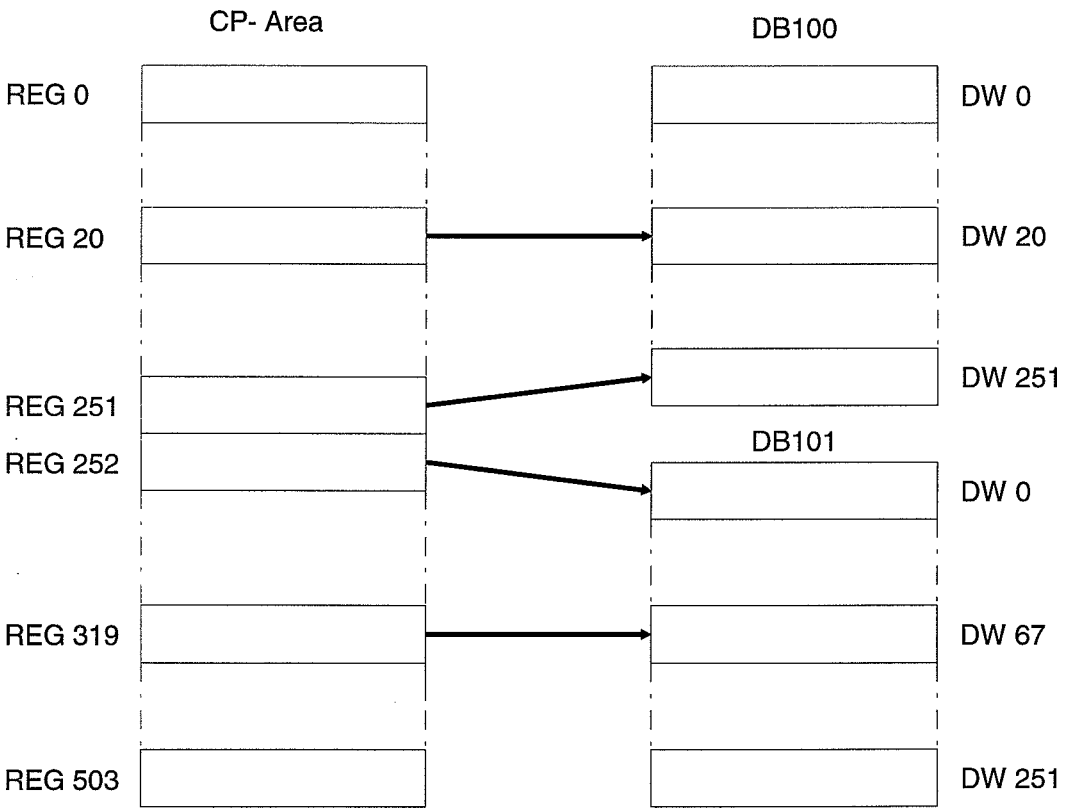
- destination data block = Parameter "DBNR" +1
- destination data word = Parameter "QANF" -252

The specified CP communication area is copied into the S5 data area when the identification "job complete without error" appears in the condition code word of the triggered "FETCH" job.

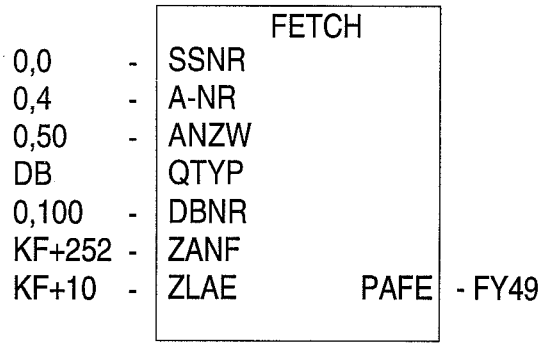
Examples: reading access

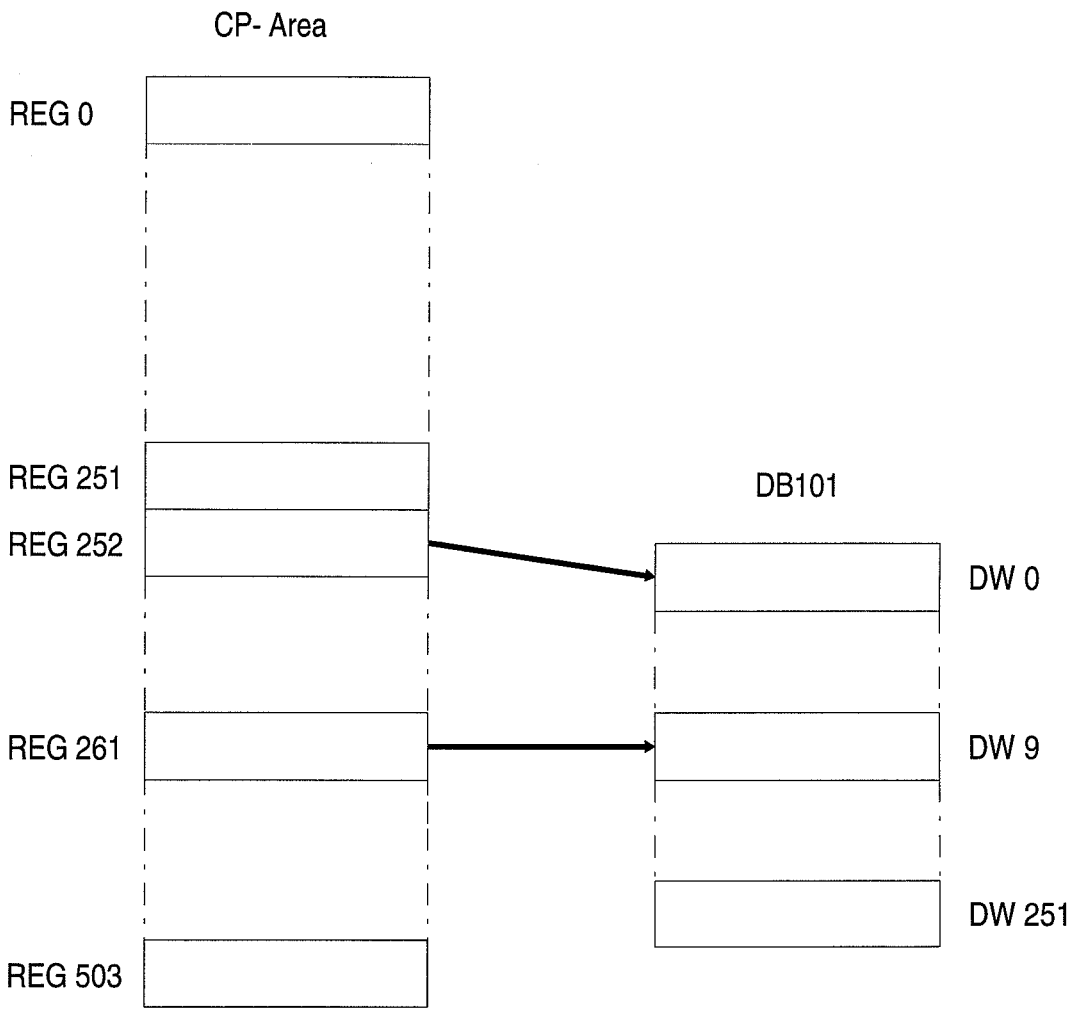
300 registers are to be read starting from register address 20.





10 registers are to be read starting from register 252.





8. 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.

8.1 Error Messages on LED's

The CP525 has a red light emitting diode for each interface which indicates the CP status and driver status during the start-up phase. When using CP524 please note that software errors are indicated on the left hand LED, whereas after start-up only serious hardware faults are displayed on the right hand LED.

The LED's light 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. When the switch is in the STOP position, received messages are not acknowledged.

The LED blinks if an error is recognized in the job block during CP start-up.

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.

8.2 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

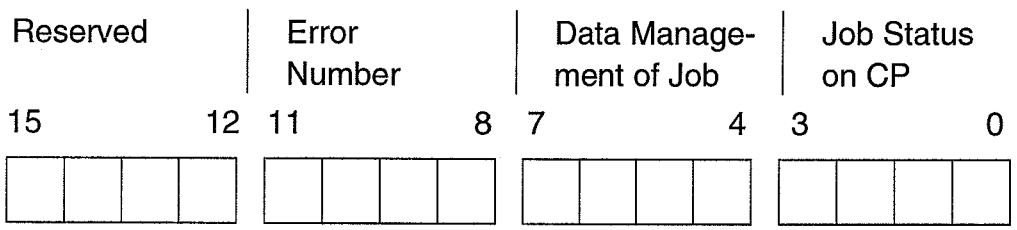
COM525 Manual Volume 2
Register 7 (Issue 06)
Para. 2.7.1.

8.3 Error Numbers in Condition Code Word ANZW

8.3.1 Assignment of Condition Code Word of HDBs "SEND" and "FETCH"

Each job number of a "SEND" or "FETCH" 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 transferred to CP/ accepted by CP
Bit 3:	job complete with error	data could not be transferred to CP/accepted by CP ⇨ evaluate error number in bits 8 - 11 ; read SYSTAT

If a "SEND" and/or "FETCH" job could not be processed correctly, **the reason for the error is displayed in bits 8-11 of ANZW by means of an error number.** An error number is entered into the ANZW only if the error occurred in conjunction with the processing of a Direkt job.

Each Direkt job where an error has occurred, receives an additional error entry in the SYSTAT area where the error situation is specified in more detail. The possible error messages are explained on the following pages.

8.3.2 Error Numbers in ANZW for "SEND" and "FETCH"

Error Number Bit 8-11	Reason for Error
1...5	Parameter assignment errors recogized by the CPU of the PLC which are reported to the CP
6	Errors recognized by the CP in data traffic between CPU and CP
7	job cannot be processed, error in job parameters
8	free
9	no CP communication area exists
A	free
B	function code unequal 05 and/or 16
C	address field error switch on STOP/PGR during job
D	data field error
E	addition error (address field and data field)
F	free

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

COM525 Manual Volume 1
Register 7 (Issue 06)
Para. 8

8.4 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 job **RECEIVE DIREKT 200 or 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 06)
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 "DIREKT" job, an additional error number is entered into the condition code word.

The description of general error numbers 1-28, 2E, 4E, 4F, 53 and B0-D0 can be found in the COM525 manual. The other error numbers of the COM525 manual are not used by the special driver.

The error entry in SYSTAT is made in such detail that it is especially suitable for exact error analysis during the commissioning phase.
If data link problems occur during operation, it is always recommended to read and evaluate the SYSTAT.

8.4.1 Error Codes in ANZW and SYSTAT for PLC Job "SEND"

ANZW	SYSTAT	Reason for Error
7	25H	the amount of transferred data bytes is - unequal 06 for function code 05 - smaller 04 and/or greater 256 for function code 16
B		function code unequal 05 and/or 16
C	60H	bit number greater 4031 and/or 8063 for function code 05
C	61H	register address greater 251 and/or 503 for function code 16
D	70H	data field not correct for function code 05
E	80H	register start address + amount greater 252 and/or 504 for function code 16

8.4.2 Error Codes in ANZW and SYSTAT for PLC Job "FETCH"

ANZW	SYSTAT	Reason for Error
C	62H	register start address greater 251 for operating mode "1DB"
C	63H	register start address greater 503 for operating mode "2DB"
D	71H	register amount greater 252 for operating mode "1DB"
D	72H	register amount greater 504 for operating mode "2DB"
E	81H	register start address and register amount greater 252 for operating mode "1DB"
E	82H	register start address and register amount greater 504 for operating mode "2DB"

8.4.3 Error Codes in SYSTAT for Parameter Assignment Errors

SYSTAT	Reason for Error
9AH	no "SEND"-job block programmed with COM525
9BH	job type unequal "Data Block""
9CH	Bit number of parameter "Co-ordination Flag Optional" greater three
9DH	RS485 operation parameterised but no CP524 with RS422/RS485 sub-module
9EH	waiting time KTTIM parameterised, but operation is not with Exception Code 06

8.4.4 Error Codes in SYSTAT for Incorrect CP Communication Area

SYSTAT	Reason for Error
90H	Error display from CPU to CP which cannot be interpreted by the CP
91H	ORG-format illegal
92H	Source-DB does not exist
93H	Source-DB too short
94H	no access to source DB
95H	incorrect condition code word
96H	illegal reply from CPU during Handshake
97H	Monitoring time elapsed for Handshake on CP (no SEND-ALL programmed)
98H	Area disabled
99H	SYNCHRON during generation of CP communication area

8.4.5 Error Codes in SYSTAT for Incorrect Queries

SYSTAT	Reason for Error
74H	Character delay elapsed
75H	Amount of received bytes not correct
76H	Received CRC check character does not tally with the character generated internally
77H	Transmission error recognized (during the message and on message start)
78H	Message end not correct (see Para. 4.1) Character delay time not elapsed on message end
79H	correct new message received before response was sent (7CH was already entered)
7AH	no input buffer free
7CH	Reception prior to transmission of response (also faulty reception)
FFH	BREAK during message reception

9. Handling Functions in Programmable Controller

9.1 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 Programmable Controller **115U** values 0.6 and 0.255 must not be used because in the event of message lengths > 256 bytes data might be overwritten in the event of FETCH jobs.

9.2 Handling Functions "SEND DIREKT"/"FETCH DIREKT"

The handling functions "SEND" and "FETCH" (job number 1...223 with the exception of the special jobs) have already been explained in detail in Paragraphs 7.1 and 7.2.

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

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

9.3 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.

The "SEND ALL" handling function is also required in order to generate the CP communication area after cold re-start of the CP and/or initial start of the CPU.

Depending on the operating mode "1DB/2DB's" the identifiers "0001H" and/or "0002H" are dynamically superimposed into the condition code word after taking over the S5 data area.

If the "SEND ALL" function is active for the cycle test, the CP superimposes the identifier "0004H" into the condition code word of the handling block.

The parameters "QTYP", "DBNR", "QANF", and "QLAE" on the handling block are irrelevant as the source information is pre-determined by the CP.

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.

9.4 Handling Function "RECEIVE ALL"

In order to ensure that the "FETCH" handling function can be processed, the handling block "RECEIVE" must be cyclically called in the user program.

In this instance the "job number" parameter must be defaulted with **"0"**.

The parameters "ZTYP", "DBNR", "ZANF", "ZLAE" on the handling block RECEIVE with job number 0 are irrelevant, because the destination information is pre-determined by the special driver.

When the handling block is running idle (no data taken over from the CP), the ANZW is "0000H", when entering data into the data block the job number of the respective "FETCH" job is superimposed dynamically into the condition code word.

For larger amounts of data several "RECEIVE ALL" functions may be necessary, depending on the blocking size, to transfer the data from the CP into the S5 destination data area.

9.5 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 06)
Para. 7.

The pseudo READ/WRITE function (job numbers 190...199) has not been realized for the special driver "S5R007", because dynamic presetting of CPU source parameters can be realized by means of indirect parameter assignment of "QTYP", "DBNR", "QANF" and "QLAE".

9.6 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 "S5R007" the handling blocks marked with "✔" must be used in the S5 program in the form previously described. RESET must only be used with A-NR.=200 and RECEIVE-Direkt only with A-NR.=200 or 221 for processing of SYSTAT.

10 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

11 User Program

The following pages illustrate an S5 user program for the 135U controller with R-processor. For this particular programming example, the "S5R007" special driver has been parameterized as follows: -

- with exception code 06
- without cycle test CP525 related (TIME OUT = 00)
- operation with co-ordination flag FY10.0
- operation with two data blocks

The S5 program is an example only, it is not to be understood as a solution for customer specific system configurations.

```
OB20                                                    LEN=11
                                                        PAGE  1

SEGMENT    1
0000           :JU    FB125SYNCHRONIZE INTERFACE
0001    NAME:SYNCHRON
0002    SSNR:         KY 0,0
0003    BLGR:         KY 0,6       BLOCKING SIZE: 256 BYTE
0004    PAFE :        FY 5
0005           :BE
```

```
OB21                                                    LEN=11
                                                        PAGE  1

SEGMENT    1
0000           :JU    FB125SYNCHRONIZE INTERFACE
0001    NAME:SYNCHRON
0002    SSNR:         KY 0,2
0003    BLGR:         KY 0,6       BLOCKING SIZE: 256 BYTE
0004    PAFE :        FY 6
0005           :BE
```

```
OB22                                                    LEN=11
                                                        PAGE  1

SEGMENT    1
0000           :JU    FB125SYNCHRONIZE INTERFACE
0001    NAME:SYNCHRON
0002    SSNR:         KY 0,2
0003    BLGR:         KY 0,6       BLOCKING SIZE: 256 BYTE
0004    PAFE :        FY 7
0005           :BE
```

```
OB 1                                                    LEN=40
                                                        PAGE 1

SEGMENT     1     0000
0000              :A    F 99.0      TRIGGER FLAG FC16
0001              :AN   F 31.1      FC05 NOT TRIGGERED
0002              :AN   F 41.1      "FETCH"-JOB NOT RUNNING
0003              :JC   FB16
0004    NAME      :FC16
0005              :
0006              :A    F 99.1      TRIGGER FLAG FC05
0007              :AN   F 21.1      FC16 NOT TRIGGERED
0008              :AN   F 41.1      "FETCH"-JOB NOT RUNNING
0009              :JC   FB5
000A    NAME      :FC05
000B              :
000C              :A    F 10.0      CO-ORDINATION FLAG SET
000D              :AN   F 21.1      FCT-CODE 16 NOT ACTIVATED
000E              :AN   F 31.1      FCT-CODE 05 NOT ACTIVATED
000F              :JC   FB7         ---> READ CP- AREA
0010    NAME      :CP-AREA
0011              :
0012              :JU   FB127       "RECEIVE ALL" - FUNCTION
0013    NAME      :REC-A
0014    SSNR      :     KY0,2
0015    A-NR      :     KY0,0
0016    ANZW      :     FW50
0017    PAFE      :     FY49
0018              :
0019              :JU   FB126       "SEND ALL" - FUNCTION
001A    NAME      :SEND-A
001B    SSNR      :     KY0,2
001C    A-NR      :     KY0,0
001D    ANZW      :     FW60
001E    PAFE      :     FY59
001F              :
0020              :JU   FB10
0021    NAME      :COO/READ         READ COOR-FLAG
0022              :BE
```

FB 5 LEN=52
PAGE 1

```
SEGMENT 1          0000
NAME :FC05
0005          :A   F 31.1     WHEN JOB RUNNING --->
0006          :JC  =F 001     NO NEW TRIGGER
0007          :
0008          :L   KF+2       SET CP-ORDINATION FLAG,
000A          :T   FY10       CARRY ALONG AND WRITE
000B          :JU  FB9
000C    NAME  :COO-FLAG
000D    LESE  :    F 10.1
000E    SSBY  :    KY2,10
                                here it is recommended to insert a FETCH
                                to save the current values
000F          :JU  FB120
0010          :SEND           EXECUTE FUNCTION CODE 05
0011    SSNR  :    KY0,2
0012    A-NR  :    KY0,5
0013    ANZW  :    FY30
0015    QTYP  :    KSDB
0016    DBNR  :    KY0,30      DB-NUMBER = DB30
0017    QANF  :    KF+0        DW-NUMBER = DW0
0018    QLAE  :    KF+3        LENGTH = 3 WORDS
0019    PAFE  :    FY29
001A          :A   F 29.0      WHEN PAFE
001B          :BEB             -->BLOCK END
001C    M001  :
001D          :JU  FB123       UPDATE CONDITION CODE WORD
001E    NAME  :CONTROL
001F    SSNR  :    KY0,2
0020    A-NR  :    KY0,5
0021    ANZW  :    FW30
0022    PAFE  :    FY28
0023          :A   F 31.1      WHEN JOB NOT YET COMPLETE
              :BEB             -->BLOCK END
0024          :
0025          :L   KF+0        DELETE CO-ORDINATION FLAG
0027          :T   FY10        CARRY ALONG AND WRITE
0028          :JC  FB9         Example of FB 9 in Para. 2.2.3
0029    NAME  :COO-FLAG        "Co-ordination Flag"
002A    LESE  :    F10.1
002B    SSBY  :    KY2,10
002C          :
002D          :R   F99.1       RE-SET CO-ORDINATION FLAG
002E          :BE
```

FB 16 LEN=53
PAGE 1

```
SEGMENT 1          0000
NAME :FC16
0005          :A    F 21.1     WHEN JOB RUNNING --->
0006          :JC   =F 001     NO NEW TRIGGER
0007          :
0008          :L    KF+1       SET CO-ORDINATION FLAG,
000A          :T    FY10       CARRY ALONG AND WRITE
000B          :JU   FB9
000C    NAME  :COO-FLAG
000D    LESE  :     F 10.1
000E    SSBY  :     KY2,10     here it is recommended to insert a FETCH
                               to save the current values
000F          :JU   FB120
0010    NAME  :SEND            EXECUTE FUNCTION CODE 16
0011    SSNR  :     KY0,2
0012    A-NR  :     KY0,16
0013    ANZW  :     FW20
0014    QTYP  :     KSDB
0015    DBNR  :     KY0,20     DB-NUMBER = DB20
0016    QANF  :     KF+0       DW-NUMBER = DW0
0017    QLAE  :     KF+5       AMOUNT OF REGISTERS="QLAE"-2=3
0018    PAFE  :     FY19
0019          :A    F 19.0     WHEN PAFE
001A          :BEB             -->BLOCK END
001B    M001  :
001C          :JU   FB123      UPDATE CONDITION CODE WORD
001D    NAME  :CONTROL
001E    SSNR  :     KY0,2
001F    A-NR  :     KY0,16
0020    ANZW  :     FW20
0021    PAFE  :     FY18
0022          :A    F 21.1     WHEN JOB NOT YET COMPLETE
0023          :BEB             -->BLOCK END
0024          :
0025          :L    KF+0       DELETE CO-ORDINATION FLAG
0027          :T    FY10       CARRY ALONG AND WRITE
0028          :JC   FB9
0029    NAME  :COO-FLAG
002A    LESE  :     F10.1
002B    SSBY  :     KY2,10
002C          :
002D          :R    F99.0      RESET TRIGGER FLAG
002E          :
002F          :BE
```

```
FB 7                                                    LEN=44
                                                        PAGE 1

SEGMENT 1          0000
NAME :CP-AREA

0005          :A    F 41.1      WHEN JOB RUNNING --->
0006          :JC   =F 001      NO NEW TRIGGER
0007          :
0008          :JU   FB122
0009    NAME  :FETCH
000A    SSNR  :     KY0,2
000B    A-NR  :     KY0,1
000C    ANZW  :     F40
000D    QTYP  :     KSDB
000E    DBNR  :     KY0,100     DB-NR = DB100 +DB101
000F    ZANF  :     KF+0        START ADDRESS = REGISTER 0
0010    ZLAE  :     KF+504      LENGTH = 504 REGISTERS
                                (DB100: DW0-251; DB101: DW0-251
                                see also Para. 2.1.1)
0011    PAFE  :     FY39
0012          :A    F 39.0      WHEN PAFE
0013          :BEB              -->BLOCK END
0014    M001  :
0015          :JU   FB123       UPDATE CONDITION CODE WORD
0016    NAME  :CONTROL
0017    SSNR  :     KY0,2
0018    A-NR  :     KY0,1
0019    ANZW  :     FW40
001A    PAFE  :     FY38
001B          :A    F 41.1      WHEN JOB NOT YET COMPLETE
001C          :BEB              -->BLOCK END
001D          :
001F          :L    KF+0        DELETE CO-ORDINATION FLAG
0020          :T    FY10        CARRY ALONG AND WRITE
0021          :JC   FB9
0022    NAME  :COO-FLAG
0023    LESE  :     F10.1
0024    SSBY  :     KY2,10
0025          :
0026          :BE
```

FB 10

LEN=23
PAGE 1

```
SEGMENT    1      0000
NAME  :COO/READ

0005            :AN   F 22.1      FCT - CODE 16 NOT ACTIVATED
0006            :AN   F 31.1      FCT - CODE 05 NOT ACTIVATED
0007            :AN   F 41.1      CP - AREA NOT READ
0008            :JC   =M001
0009            :BEA
000A    M001    :
000B            :L    KB2         READ CO-ORDINATION FLAG
000C            :T    FY10
000D            :JC   FB9
000E    NAME    :COO-FLAG
000F    LESE    :     F10.1
0010    SSBY    :     KY2,10
0011            :BE
```

DB20

LEN=10
PAGE 1

```
      0:    KH=  0010;
      1:    KF=  +00251;
      2:    KH=  FFFF;
      3:    KH=  FFFF;
      4:    KH=  FFFF;
      5:
```

DB30

LEN=8
PAGE 1

```
      0:    KH=  0005;
      1:    KF=  +08063;
      2:    KH=  0000;
      3:
```

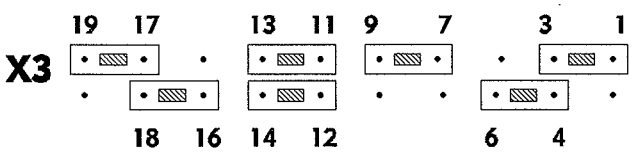
12. Notes

Jumper settings on the RS422-A/RS485-module 6ES5 752-0AA43

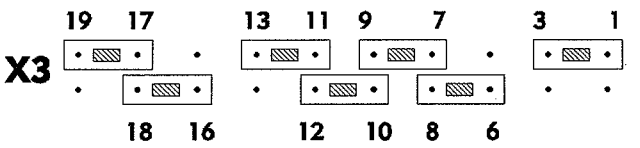
The RS422-A/RS485 - module with order-no. 6ES5 897-0AA43 is a development of the module -0AA42. Lower power input makes it possible to use the new module without fan. The location of the jumpers and the jumper settings have changed in contrary to the old module.

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.

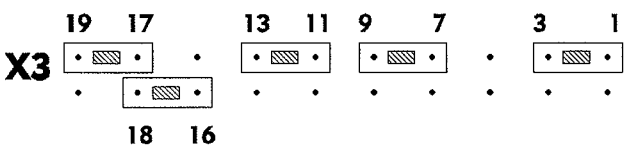
Presetting of the two-wire-line R for recognizing the break status



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.

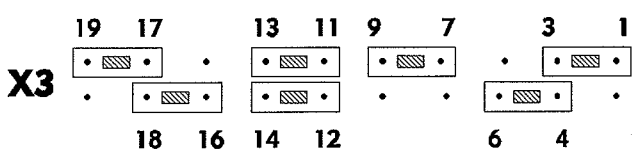


With the jumpers 10-12 and 6-8 the two-wire-line is preset as follows:
Pin 4 (R(A)) of the front connector is connected to ground via resistor.
Pin 11 (R(B)) of the front connector is connected to +5 V via resistor.
The break status **cannot** be recognized.

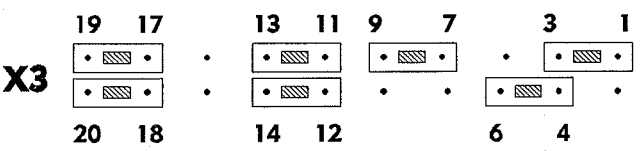


If the jumpers 12-14 and 4-6 (or 10-12 and 6-8) are dropped, the two-wire-line R is not preset. The recognition of break status cannot be guaranteed.

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).