SIMATIC S5

OPERATING INSTRUCTIONS

for Special Driver S5R006

Data Link (RK512)

with Parameter Assignment Facility

with 3964 / 3964R Protocol

without Reaction Message

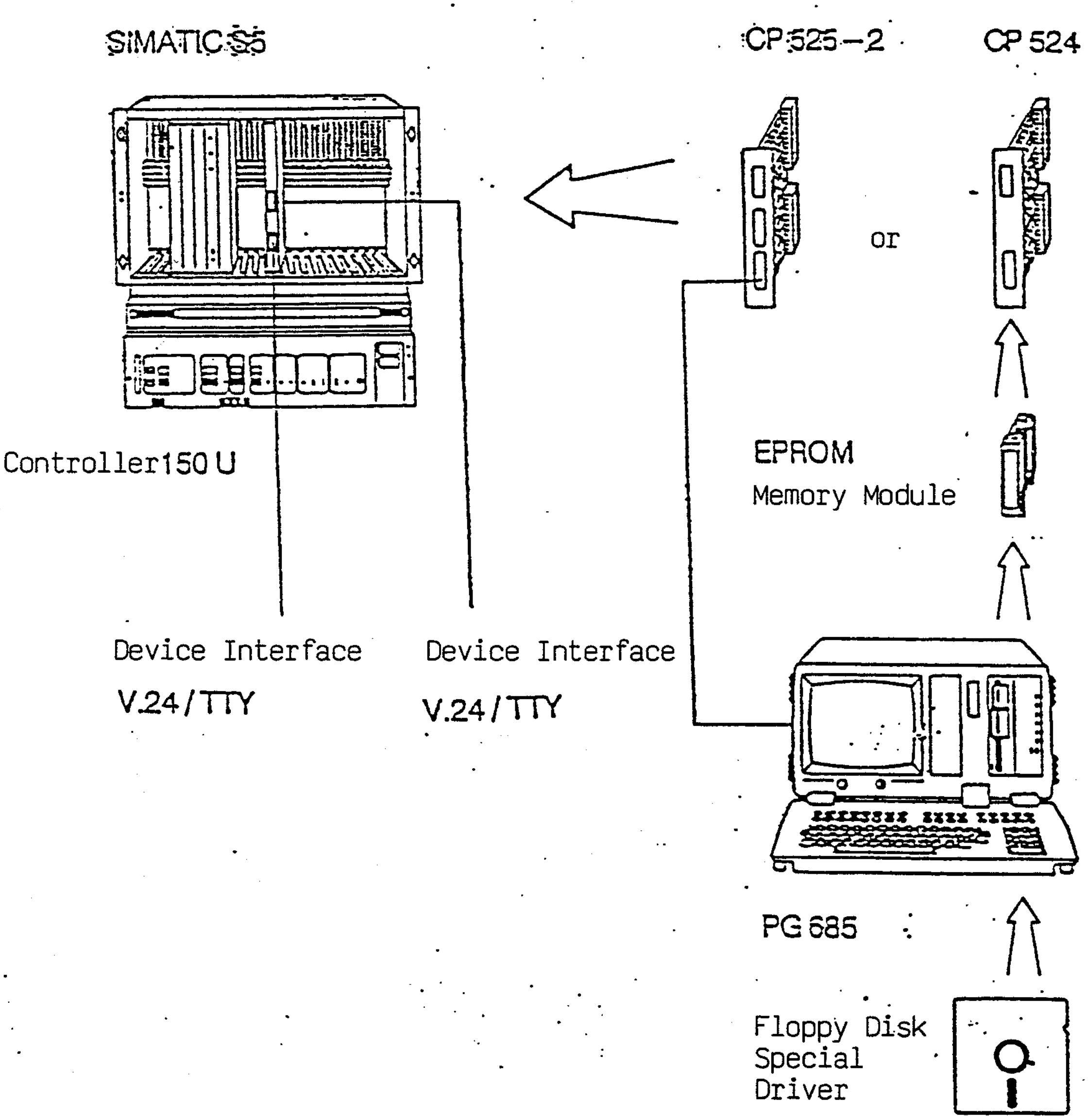
Operating System: S5-DOS Basis

Order No: 6ES5 897-2AB21-02

1 Introduction

The special driver S5R006 is a data link with parameter assignment facility for communications processors in SIMATIC S5 U-Series controllers. It is based on standard data link RK512 without reaction message and the S5 Procedure 3964/3964R.

CP525-2 and CP524 can be used as communication processors.



2. Installation of Special Driver on PG685

The library "COMLIBR6.525" is located on the supplied 5.25" mini floppy disk which contains the special driver "S5R006" consisting of interpreter and procedure.

I-S5R006 -----> interpreter
P-S5R006 -----> procedure

The installation of the driver is carried out on a programming unit 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 "COMLIBR6.525" is copied on to the winchester drive user level 0 as follows: -

```
PIP B:=A:COMLIBR6.525[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  COMLIBR6.525[SYS RO].
```

CP User Programs

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

The CP525 module with part number 6ES5 525 3UA11 (CP/M 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 previous 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 "RK" which stands for Data Link.

3.1 Copy Procedure from Library to User Program

In order to copy from the library to the user program, call the "TRANSFER" 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 "COMLIBR6".
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-S5R006" (component "RK") into this mask.

Function key F1 starts the transfer; the transfer end is indicated by the following message: "MELD.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-S5R006" 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

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-S5R006" relating to the chosen program as well as the version number.

The input fields "character length", "parity" etc. can be freely assigned corresponding to the relevant system configuration.

The input field "PRIORITY" can be specified as "HIGHER" or "LOWER" and determines the priority in the event of an initialization conflict.

Use function key F6 to store the information.

3.3 Parameter Assignment of Job Block

The creation of job blocks is required only in the event of parameter assignment of the data link "with header" (see Paragraph 7.1.5).

This data link without reaction message allows only "SEND" as the job type. "FETCH" jobs are completed with error.

A detailed description on parameter assignment of the job block can be found in

COM Manual Volume 1
Register 7
Paragraph 2.2.

4 Loading of Special Driver into CP

The loading procedure of the special driver S5R006 is identical to the procedure used for Data Link RK512.

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 2
Paragraph 3.2.2.

5 Transmission Procedure

Transmission is asynchronous, half duplex and is carried out either via an RS232 (= V24) or TTY interface.
Control signals of the V24 interface are not evaluated.

The length of the data records which are transmitted and received, must not exceed 512 Byte. Data exchange from and to the Central Controller is word by word providing the transfer is carried out via a data block. If the end of the reception does not correspond with the end of a word, the left over byte is filled with 00H.

The procedure is based on the 3964(R) protocol. The code of the individual control characters can be parameterized.

A detailed description on the transmission procotol 3964R can be found in

COM525 Manual Volume 1
Register 7
Paragraph 5.

6 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 500 ms intervals.

After initializing the CPU-CP interface, the interface specific LED blinks twice; the next step is initializing the special driver before it returns to normal operation. This is carried out by triggering the special job "INIT" (see application example).

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 PG685 (see application example: FB12).

7 Parameter Assignment of the Data Link with Special Job "INIT"

A "SEND-DIREKT" job with A-Nr - 189 is interpreted by the CP as the special job "INIT".

With this "SEND" job it is possible to parameterize the type of data link, the times, the acknowledgement characters as well as the type of BCC creation.

The "QTYP" parameter (= type of data source) must be defaulted with "DB".

Defaulting the "QLAE" (= length) parameter is irrelevant, because the special driver reads out always a quantity of ten data words from the specified source data area.

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 such an event the initialization of the procedure is not carried out until reception is completed. It is important to note that messages may not be recognized or acknowledged incorrectly if the control characters are being changed.

After recognizing the special job "INIT" the special driver switches on the interface specific LED. If no errors are 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; also, the interface specific LED blinks four times.

The job number 189 is reserved for initialization and cannot be used for anything else. It is not required to program a job block for it.

7.1.1 Structure of Initialization Data Block

ZVZ		char. delay time in msec
QVZ		ack. delay time in msec
QUIPOS	QUINEG	ack. char. for establishing the connection
QVAPOS	QVANEG	ack. char. for connection clearance
STARTZ	ENDKEN	start char./No. of end char.
ENDZ1	ENDZ2	end character
BCC	QUIT	type of BCC / ack. Int. Procedure
KOPF	CPU-NR	with/without message header / CPU Addr. *)
DB-NR	DW-LOW	receive data block *)
BYTE	BIT	co-ordination flag *)

*) only relevant when parameterizing "without message header"

7.1.2 Parameter Assignment of Times

The character delay time "ZVZ" monitors the arrival of characters on reception. It can be parameterized in intervals of milliseconds. The smallest value it can be set to, is 4 msec.

The acknowledgement delay time "QVZ" specifies the time which elapses whilst waiting for an acknowledgement character. It can also be parameterized in intervals of milliseconds. If a smaller value than 256 msec is entered in the initialization DB, the job is rejected with error.

7.1.3 Parameter Assignment of Acknowledgement Characters

The Hex-code for the positive acknowledgement is entered into the "QUIPOS" field. It is transmitted as the reponse to a start character providing there was no error.

The "QUINEG" field contains the code for the negative acknowledgement which is transmitted as the reponse to a start character in the event of an error.

The Hex-code for the positive acknowledgement is entered into the "QVAPOS" field. It is transmitted as the reponse to a message, providing there was no error.

The "QVANEG" field contains the code for the negative acknowledgement which is transmitted as the reponse to a message in the event of an error.

The code of the start character is located in the "STARTZ" field. It is transmitted in order to establish the connection.

The number of end characters is parameterized in the "ENDKEN" field. If "1" is entered, only one end character is used. In this instance, only "ENDZ2" is evaluated. The value "2" results in two end characters being used.
If the transmission is carried out with only one end character, the procedure is not code transparent !!

The first end character "ENDZ1" is used for code conversion. If it appears in a data record, it is duplicated by the procedure. If the first end character is received twice, it results in the character being entered only once in the receive data block.

The "ENDZ2" field contains the second end character. The end character(s) are output after the useful data.

7.1.4 Parameter Assignment of BCC-Creation

The "BCC" field indicates whether transmission should be carried out with (type of creation) or without Block Check Character. If a "0" is entered into this field, operation is without BCC. Selecting from values 1 to 4 determines the modus of BCC creation, the modus corresponding to the selected number.

Modus 1:

Addition of all characters (without considering the overflow) inclusive of end sequence.

```
Example: 30H, 31H, 32H, 33H, 34H, 10H, 03H, 0DH
                                            |
                                            +- BCC Check
```

Modus 2:

Two' complement of the sum of all characters (without considering the overflow) inclusive of end sequence. Segmenting of the calculated checksum into HIGH - Nibble (= Bit 4...7) and LOW - Nibble (= Bit 0...3) and conversion into ASCII characters (ORing with 30H). When transferring the BCC check it is important to note that the HIGH - Nibble is transferred first followed by the LOW - Nibble.

```
Example: 30H, 31H, 32H, 33H, 10H, 03H, 3FH, 33H
                                       |     |
                                       |     +- BCC-LOW
                                       +------- BCC-HIGH
```

Modus 3:

ex-or logic operation of all characters inclusive of end sequence on even parity (= presetting the BCC character with 00H).

```
Example: 30H, 31H, 32H, 33H, 34H, 10H, 03H, 27H
                                             |
                                             +- BCC Check
```

Modus 4:

ex-or logic operation of all characters inclusive of end sequence on uneven parity (= presetting the BCC character with FFH).

```
Example: 30H, 31H, 32H, 33H, 34H, 10H, 03H, D8H
                                            |
                                            +- BCC Check
```

7.1.5 Parameter Assignment of Type of Data Link

It is possible to determine in the "QUIT" field whether the acknowledgement on reception is to be on the procedure or on the interpreter level. If "0" is entered, the received message is not acknowledged until it has been entered into the receive data block without error. If the message cannot be completely transferred to the central controller within the acknowledgement delay time, a negative acknowledgement is sent and transmission to the central controller is aborted (= interpreter level).
If "1" is entered into this field, the acknowledgement is issued on the procedure level. This means that a positive acknowledgement is sent immediately upon reception of the end character (and possibly BCC), providing no error has occurred during character reception.
If a co-ordination flag is used, the acknowledgement is always issued after entry of the data in the central controller.

The Standard Procedure operates with acknowledgement on procedure level.

If "0" has been entered into the "KOPF" field (= header field), operation is without message header. The destination of the receive messages and a possible co-ordination flag is determined in the subsequent 5 bytes of the initialization data block.
The value "1" in the "KOPF" field means that operatiron is with S5 message header. In output direction the message header is generated by means of the job block relating to the Send Direkt, and it is output prior to the useful data. In receive direction it specifies the data destination.
If the data link has been parameterized to operate with header, the next 5 byte of the initialization DB are irrelevant.

7.1.6 Parameter Assignment of the Receive Data Block

The address of the processor (in the event of multi processor operation) into which the receive messages are to be entered, is defined in the "CPU-No." field (area 1 - 4). If "00" is enterd, "to all CPU's" is assumed.

The "DB-NR" field contains the number of the receive data block for messages from the link partner. This value must be greater than 2.

The field "DW-LOW" specifies from which data word in the parameterized receive data block the received messages are to be entered. The address for the DW HIGH is always set to zero.

The last two fields "BYTE" and "BIT" specify the co-ordination flag, which can be used to disable the input into the parameterized receive data block.

If operation is to be without co-ordination flag, the value "0" must be entered into the "BYTE" and "BIT" fields.

!! Attention !!
The flag 0.0 cannot be used as the co-ordination flag, because this is the identifier for "without co-ordination flag".

7.2 Example of a Message

```
   STARTZ   --------->          start character
            <--------- QUIPOS   pos. acknowledgement
   * KOPF   =========> *        header
1. ZEICHEN  --------->          1st character
            --------->
            --------->
            --------->
            --------->
n. ZEICHEN  --------->          nth character
   * ENDZ1  ---------> *        end character 1
     ENDZ2  --------->          end character 2
   * BCC    ---------> *
            <--------- QVAPOS / QVANEG
                                ack. char. for connection clearance
```

Parameters marked with * are output depending on parameter assignment.

8 Assignment of Condition Code Word of Handling Block "SEND-DIREKT"

Each job number of a "SEND" job defined in the CPU 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 CPU processor and the CP.

Structure of the Condition Code Word

```
Reserved      | Error        | Data           | Job
              | Number       | Management     | Status
              |              | of Job         | Display
              |              |                | on CP

15         12  11          8  7            4  3          0
+--+--+--+--+  +--+--+--+--+  +--+--+--+--+  +--+--+--+--+
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
+--+--+--+--+  +--+--+--+--+  +--+--+--+--+  +--+--+--+--+
```

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

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.

The possible error messages are described on the following pages.

8.1 Error Numbers in ANZW of "SEND-DIREKT"

Error Number Bit 8-11	Reason for Error
1...5	parameter assignment errors recognized by the CPU of the central controller which are reported to the CP
6	errors recognized by CP in data traffic between CPU and CP
7	job cannot be processed, error in job parameters
E	error in data traffic CP <--> partner
F	BREAK

For a detailed description of the error numbers please refer to

COM525 Manual Volume 1
Register 7
Paragraph 8.

9 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.

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
Paragraph 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.

The following chapters deal with special driver specific SYSTAT error messages only.
The remaining error numbers are described in the

COM525 Manual Volume 1
Register 7
Paragraph 8.

9.1 Error Code in ANZW and SYSTAT for Send Errors

ANZW	SYSTAT	Reason for Error
6H	60H	Fetch jobs are illegal
7H	25H	A send job with more than 512 byte has been triggered

9.2 Error Code in SYSTAT for Receive Errors

SYSTAT	Reason for Error
4BH	A message with more than 512 byte has been received
60H	A Fetch job has been triggered by the partner
61H	Applies to acknowledgement on interpreter level: a received message could not be transferred to the central controller within the acknowledgement delay time

9.3 Error Code in ANZW and SYSTAT when Processing the Special Job "INIT"

ANZW	SYSTAT	Reason for Error
7	80H	INIT-Send expected, but Send-Direkt with Job No <> 189 recognized
7	81H	Job Type <> "SEND"
7	82H	Source Data Type <> "DB"
7	83H	Area disabled by STEP5 program
7	84H	Monitoring time elapsed for handshake on CP, or illegal reply message from CPU during handshake
7	85H	Error display of CPU to CP which cannot be interpreted by CP
7	86H	Source data block does not exist
7	87H	Source data block too short
7	88H	Access to source data area not possible
7	89H	Incorrect condition code word
7	8AH	SYNCHRON during job processing

9.4 Error Code in ANZW and SYSTAT when Evaluating the Special Job "INIT"

ANZW	SYSTAT	Reason for Error
7	8BH	Value for "BCC" > 4
7	8CH	Amount of end identifier not 1 or 2
7	8DH	Acknowledgement Delay Time < 256ms
7	8EH	Character Delay Time < 4ms
7	8FH	Parameter "DB-NR" smaller 3
7	90H	Parameter "CPU-NR" greater 4
7	91H	Parameter "KOPF" or "QUIT" not 0 or 1
7	92H	Parameter "BIT" > 7

10 Handling Functions in Central Controller

10.1 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 block size for data transfer between CPU abd CP is selected.

Parameter BLGR: KYx,y

Range of Values: x = 0

y = 1 :	block size 16 byte
y = 2 :	block size 32 byte
y = 3 :	block size 64 byte
y = 4 :	block size 128 byte
y = 5 :	block size 256 byte
y = 6 :	block size 256 byte
y = 7 to 254 :	block size 256 byte
y = 255 :	block size 256 byte

10.2 Handling Function "SEND DIREKT"

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

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".

10.3 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 recognizig 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 block size, to transfer the data from the S5 source data area into the dual port RAM.

10.4 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 and/or the message header of the received message.

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 higher ranking byte of the used ANZW is "00H", after entering the complete data block an identifier is entered into the higher ranking byte of the condition code word. Providing operation is with co-ordination flag, this identifier is the byte number of the co-ordination flag. If operation is without header and no co-ordination flag is used, this identifier is "01H".

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

10.5 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
Para. 7.

The pseudo READ/WRITE function (job numbers 190...199) has not been realized for the link "without header", because dynamic presetting of CPU source parameters can be realized by means of indirect parameter assignment of "QTYP", "DBNR", "QANF" and "QLAE".

10.6 Numbers of Handling Blocks

Central Processing Unit

Function		115U	135U	150U
	1,		R-Processor	
SEND	+	FB244	FB120	FB180
RECEIVE	+	FB245	FB121	FB181
FETCH		FB246	FB122	FB182
RESET		FB248	FB124	FB183
CONTROL	+	FB247	FB123	FB184
SYNCHRON	+	FB249	FB125	FB185
SEN-A	+		FB126	
REC-A	+		FB127	

1, When used with special driver "S5R006" the handling blocks marked with "+" must be used in the S5 program in the form previously described!!!

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 User Program

The following pages illustrate an S5 user program for the 135U controller with R-processor.

****

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

****

```
OB20                                    LENGTH = 16        ABS
                                                          PAGE     1
SEGMENT 1
0000       :SPA FB125        SYNCHRONIZE INTERFACE
0001 NAME  :SYNCHRON
0002 SSNR  :    KY 0,0
0003 BLGR  :    KY 0,6       BLOCKING SIZE: 256 BYTE
0004 PAFE  :    MB 1
0005       :UN  M      1.0   IF NO PAFE, TRIGGER
0006       :S   M     99.0   SPECIAL JOB "INIT"
0007       :BE
```

```
OB21                                    LENGTH = 11        ABS
                                                          PAGE     1
SEGMENT 1
0000       :SPA FB125        SYNCHRONIZE INTERFACE
0001 NAME  :SYNCHRON
0002 SSNR  :    KY 0,0
0003 BLGR  :    KY 0,6       BLOCKING SIZE: 256 BYTE
0004 PAFE  :    MB 2
0005       :UN  M      2.0   IF NO PAFE, TRIGGER
0006       :S   M     99.0   SPECIAL JOB "INIT"
0007       :BE
```

```
OB22                                    LENGTH = 11        ABS
                                                          PAGE     1
SEGMENT 1
0000       :SPA FB125        SYNCHRONIZE INTERFACE
0001 NAME  :SYNCHRON
0002 SSNR  :    KY 0,0
0003 BLGR  :    KY 0,6       BLOCKING SIZE: 256 BYTE
0004 PAFE  :    MB 3
0005       :UN  M       3.0  IF NO PAFE, TRIGGER
0006       :S   M      99.0  SPECIAL JOB "INIT"
0007       :BE
```

```
0B1                                        LENGTH=39           ABS
                                                               PAGE    1
SEGMENT 1        0000
0000       :U    M    99.0     CALL SPECIAL JOB "INIT"
0001       :SPB FB189          CONDITIONALLY
0002 NAME  :INIT
0003       :
0004       :U    M    21.2     IF SPECIAL JOB "INIT"
0005       :U    M    99.7     WAS COMPLETED WITHOUT ERROR
0006       :SPA FB 120         START DATA OUTPUT
0007 NAME  :SEND
0008 SSNR  :     KY 0,0
0009 A-NR  :     KY 0,1
000A ABZW  :     MW   40
000B QTYP  :     KC DB
000C DBNR  :     KY 0,30       SOURCE-DB = DB30
000D QANF  :     KF +0         SOURCE-DW = DW0
000E QLAE  :     KF +255       LENGTH = 255 WORDS
000F PAFE  :     MB   39
0010       :U    M    41.1     WHEN JOB RUNNING
0011       :R    M    99.7     RE-SET TRIGGER FLAG
0012       :U    M    41.2     WHEN JOB COMPLETED WITHOUT ERROR
0013       :S    M    99.7     SET TRIGGER FLAG AGAIN
0014       :U    M    41.3     WHEN JOB COMPLETED WITH ERROR
0015       :SPB FB 12          EVALUATE ERROR (FB12)
0016 NAME  :ANZW07
0017       :***

SEGMENT 2        0018
0018       :SPA FB126          "SEND-ALL"
0019 NAME  :SEND-A
001A SSNR  :     KY0,0
001B A-NR  :     KY0,0
001C ANZW  :     MW 60
001D PAFE  :     MB 59
001E       :
001F       :SPA FB 10          RECEPTION OCCURRED?
0020 NAME  :RECCP
0021       :BE
```

```
FB10                                LENGTH=24          ABS
                                                      PAGE   1

SEGMENT 1      0000
NAME :RECCP

0005      :SPA FB127
0006 NAME :REC-A
0007 SSNR :    KY0,0
0008 A-NR :    KY0,0
0009 ANZW :    MW 80
000A PAFE :    MB 79
000B      :L   MB 81        COMPARISON OF "HIGH"BYTE OF
000C      :L   KB1          CONDITION CODE WORD WITH "1" OR CO-ORD.FLAG
000D      :><F              IF NO RECEPTION OCCURRED
000E      :BEB              BLOCK END, OTHERWISE
000F      :                 ****************************
0010      :                 EVALUATE RECEPTION
0011      :                 ****************************
0012      :BE
```

```
FB12                                LENGTH=55          ABS
                                                      PAGE    1
SEGMENT 1      0000
NAME :ANZW07

0005      :L   MB  40
0006      :L   KB 7        WAS SEND JOB COMPLETED
0007      :><F             WITH ERROR NUMBER "07H"?
0008      :SPB =M001       SPW NO -> NO ERROR EVALUATION
0009      :
000A      :0   M    0.0    ERROR NUMBER= "07H" -->
000B      :0N  M    0.0
000C      :SPA FB 121      READ SYSTAT-AREA
000D NAME :RECEIVE
000E SSNR :    KY 0,0
000F A-NR :    KY 0,200
0010 ANZW :    MW  100
0011 ZTYP :    KC DB
0012 DBNR :    KY 0,8
0013 ZANF :    KF +0
0014 ZLAE :    KF +2
0015 PAFE :    MB  97
0016      :L   DR   0
0017      :L   KB 128
0018      :!=F             SYSTAT ERROR = "80H" ??
0019      :SPB  =M002      SPW YES --> CP-COLD RE-START SUBMITTED
001A      :L    DR   0
001B      :L    KB 22
001C      :!=F             SYSTAT ERROR = "16H" ??
001D      :SPB  =M002      SPW YES --> CP-COLD RE-START SUBMITTED
001E M001 :
001F      :0    M    0.0
0020      :0N   M    0.0
0021      :SPA  FB 124     DELETE SYSTAT-AREA
0022 NAME :RESET
0023 SSNR :    KY 0,0
0024 A-NR :    KY 0,200
0025 PAFE :    MB  98
0026      :A   DB   8
0027      :L   KH 0000
0029      :T   DW   0
002A      :T   DW   1
002B      :BEA
002C M002 :
002D      :0   M    0.0
002E      :0N  M    0.0
002F      :S   M   99.0    TRIGGER SPECIAL JOB "INIT"
0030      :SPA =M001
0031      :BE
```

```
FB189                                      LENGTH=50          ABS
                                                              PAGE    1

SEGMENT 1        0000
NAME :INIT

0005       :U    M    21.1      DO NOT TRIGGER IF
0006       :SPB =M001           JOB ALREADY RUNNING
0007       :
0008       :0    M      0.0
0009       :ON   M      0.0
000A       :SPA FB 120          TRIGGER SPECIAL JOB "INIT"
000B NAME  :SEND
000C SSNR  :     KY 0,0
000D A-NR  :     KY 0,189
000E ANZW  :     MW  20
000F QTYP  :     KC DB
0010 DBNR  :     KY 0,189       INIT-DB = DB189
0011 QANF  :     KF +0
0012 QLAE  :     KF +10
0013 PAFE  :     MB  19
0014       :U    M    19.0      WHEN PAFE
0015       :BEB                 ---> BLOCK END
0016 M001  :
0017       :SPA  FB 123         UP-DATE CONDITION CODE WORD
0018 NAME  :CONTROL
0019 SSNR  :      KY 0,0
001A A-NR  :      KY 0,189
001B ANZW  :      MW  20
001C PAFE  :      MB  19
001D       :U     M    21.1     IF JOB STILL RUNNING
001E       :BEB                 ---> BLOCK END
001F       :
0020       :U     M    21.3     ERROR OCCURRED ??
0021       :SPB   =M002         IF YES, CARRY OUT ERROR
0022       :                    HANDLING
0023       :0    M                0024       :ON   M      0.0
0025       :R    M     99.0     RE-SET TRIGGER FLAG
0026       :BEA
0027 M002  :
0028       :                    **** ERROR HANDLING ****
0029       :0    M     0.0      AND TRIGGER SPECIAL JOB "INIT"
002A       :ON   M     0.0      AGAIN, IF REQUIRED
002B       :S    M    99.0
002C       :BE
```

*** Initialization Block DB189 ***

```
The special driver was parameterized as follows: -
(Standard values)

ZVZ = 200 ms              character delay time
QVZ = 550 ms              acknowledgement delay time
QUIPOS = <DLE> 10H        pos. ack. establishment of connection
QUINEG = <NAK> 15H        neg. ack. establishment of connection
QVAPOS = <DLE> 10H        pos. ack. connection clearance
QVANEG = <NAK> 15H        neg. ack. connection clearance
STARTZ = <STX> 02H        start character
with 2 end characters
ENDZ1  = <DLE> 10H        end character 1
ENDZ2  = <ETX> 03H        end character 2
with BCC generation XOR logic operation on even parity
acknowledgement on procedure level
without message header
to all CPU's
receive data block DB255 from DW0
without co-ordination flag
```

```
DB189                                  LENGTH=13
                                                      PAGE      1
          0:     KF = +220;            ZVZ
          1:     KF = +550;            QVZ
          2:     KH = 1015;            QUIPOS / QUINEG
          3:     KH = 1015;            QVAPOS / QVANEG
          4:     KH = 0202;            STARTZ / ENDKEN
          5:     KH = 1003;            ENDZ1  / ENDZ2
          6:     KH = 0301;            BCC    / QUIT
          7:     KH = 0000;            KOPF   / CPU-NR
          8:     KY = 255,000;         DB-NR  / DW-LOW
          9:     KY = 000,000;         BYTE   / BIT
         10:
```

Index of Versions

Version	Change	Date
0.1	first release	July 88
0.2	-INIT-Send 189 mandatory requirement for start-up -method of BCC generation can be parameterized -differentation between acknowledgement characters establishment or clearance of connection -BREAK handling correct	Oct 88

13 Notes

Index