

Contents

1	Examples on transmission and reception of texts when DRUCK, EMAS etc. are used	A-1
1.1	Example program – Sending data from the PLC (07 KT 92) to a terminal	A-1
1.2	Example program – Receiving data from a terminal	A-5
2	System reaction if an incorrect program is sent	A-10

For further information concerning the used function blocks see volume 7, CE Library.

1 Examples on transmission and reception of texts when DRUCK, EMAS etc. are used

1.1 Example program – Sending data from the PLC (07 KT 92) to a terminal

Load example program:

Insert diskette 1 of the programming software into drive A:

As usual, call the programming software under \ABB-SPS\CS31.

You will find the submenu point *Data restore* under the main menu point *Project man*. Call this point.

Specify *A:\DAT_SEND* as file to be saved. For target drive specify *C*:

Press <F1>.

Now the example program will be copied into the directory *C:\ABB-SPS\CS31\BSP*. When calling the example you must enter this path and the name *DAT_SEND* in the image *Project data* for *project*.

Using the example program, data are sent via the serial interface of the 07 KT 92 PLC to a terminal or a PC. The following data is transmitted as text:

"Hallo, data are sent to a terminal in this way."

The system cable 07 SK 91 is used as connection cable between the PLC and the terminal.

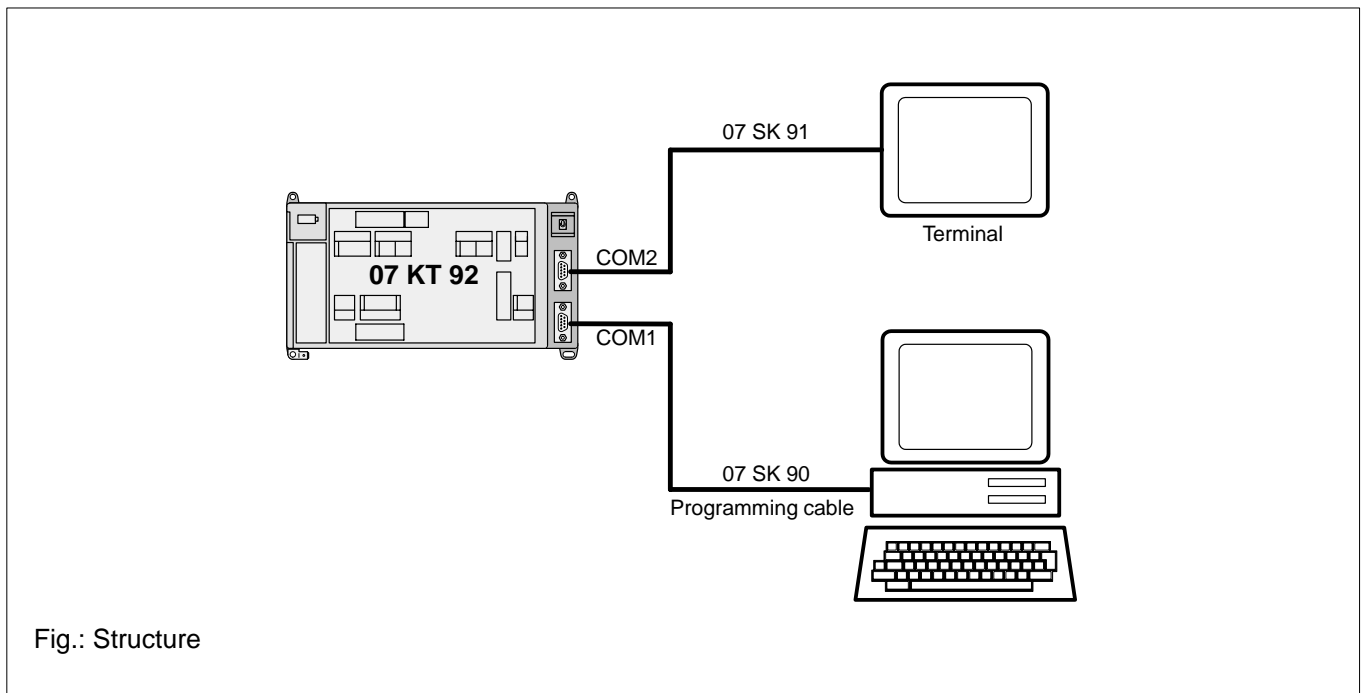


Fig.: Structure

Both, the interface of the terminal and the interface of the PLC must have the same settings (see segment plan1, INITIALIZATION). RTS/CTS is used for handshake (not XON/XOFF).

You can also use a PC with 907 PC 331 instead of a terminal. Call the terminal emulation in 907 PC 331. If you only have one PC, proceed with 07 KT 92 in the same way as described with 07 KR 91.

Notes for the use of 07 KR 91:

The 07 KR 91 PLC has only one interface COM1. In order to use the example, the following changes are required for this PLC:

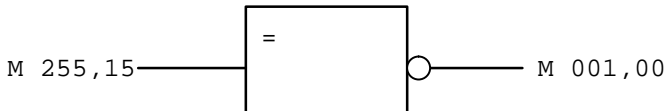
- For the parameter SSK of the blocks SINIT and DRUCK, 1 must be entered for COM1.
- The system cable 07 SK 90 must be used for the program transmission. After that, for the data display, this cable must be unplugged and the cable 07 SK 91 must be used instead.

Program description

Segment plan 1: INITIALIZATION

In this program part, the interface COM2 of the PLC is initialized in the following way:

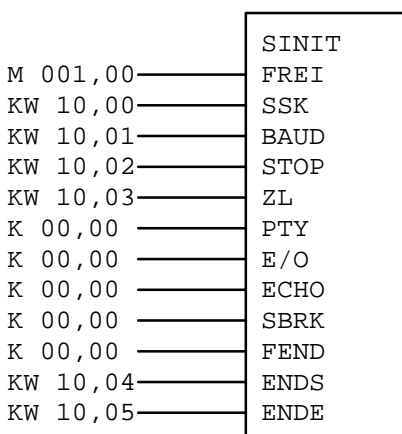
- Once at the start of the program
- Call SINIT function block
- Use M 255,15 flag
 - M 255,15 is 0 at the start of the program
 - Invert value and allocate M 001,00
 - Put value of 1 of M 001,00 to the enable input FREI of SINIT: the interface is now initialized.
 - Set M 255,15 to 1 at the end of the program → with the next program execution the SINIT block is not executed any more.



The parameters of the interface are set as follows:

Explanation	Input	Value
Interface No. 2 = COM2	SSK	2
Baud rate	BAUD	9600
Number of stop bits	STOP	1
Number of data bits per character	ZL	8
Transmission without parity bit	PTY	0
Odd parity bit (no effect because of PTY)	E/O	0
No echo	ECHO	0
Normal status of the send signal TxD	SBRK	0
No output of text termination character	FEND	0
Text termination character for sending direction (without effect)	ENDS	13 (=CR)
Text termination character for receiving direction	ENDE	13 (=CR)

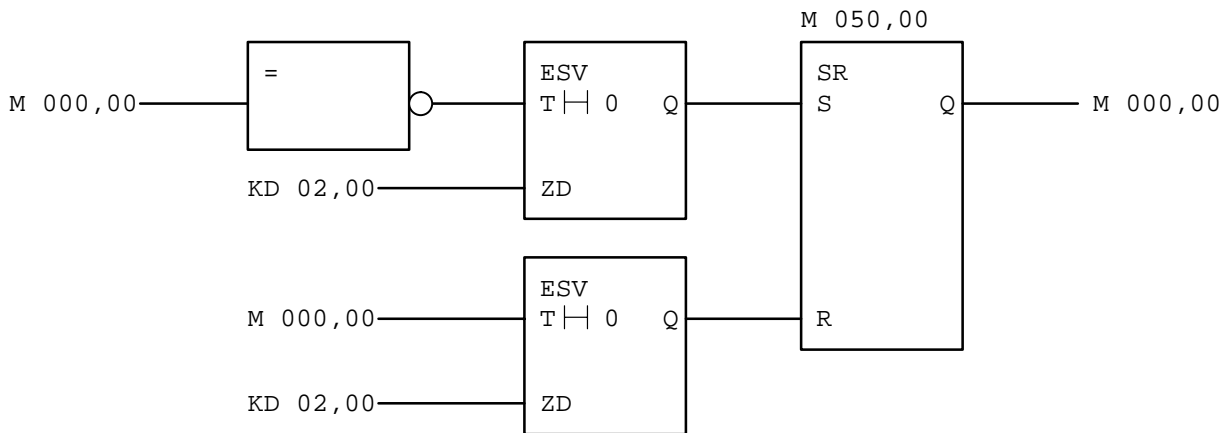
These values are entered by means of constants. They can be entered in the variable editor via the long text of the constants.



Segment plan 2: SEND DATA

In this program part, the sending procedure is realized. The DRUCK block is used for sending data. In order to trigger the sending procedure cyclically, a flip-flop is built first. This flip-flop provides a 0→1 edge at its output M 000,00 to enable the DRUCK block.

FLIP-FLOP for realization of the enabling edge for the DRUCK block. A 0→1 edge is generated periodically on a time base given in KD 02,00.

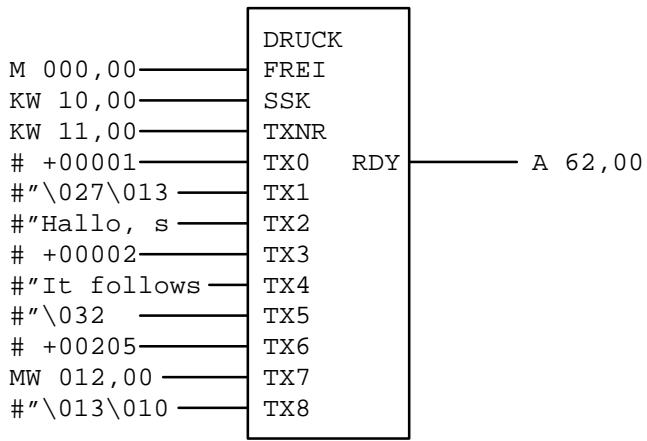


Sending procedure with the DRUCK block

There are two texts available for sending. The first text (fixed text) is connected to TX0...2. The second text (fixed text + value) is connected to TX3...8.

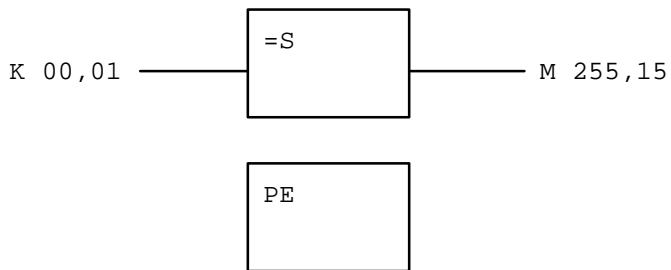
The parameters of the block are set as follows:

Explanation	Parameter	Value
Enabling the block with the 0→1 edge for outputting a text	FREI	0→1 edge
Interface No. (2 for COM2)	SSK	2
Number of the text to be output	TXNR	1 or 2
Text; the text starts with the text number (1st text: 1); after the text number the real text follows; each text starts with #"	TX0	1
first a device-dependent control sequence follows (\27 ... = <ESC> ...) for cursor positioning, then, the text to be displayed follows.	TX1	\027\013
2nd text; the text starts with the text number (2)	TX2	Hallo, so...
Text to be output	TX3	2
Space character before number to be output	TX4	It follows...
Type of format (205: 2 → word; 05 → five digits with leading zeros; for more information see DRUCK block documentation)	TX5	\032
Value to be output	TX6	205
Jump to the start of the next line: <CR><LF>	TX7	123
Output Ready: If RDY = 1 the block is ready for outputting a text	TX8	\013\010
	RDY	Ready message



Segment plan 3: PROGRAM END

M 255,15 is set to 1, and, beginning with the next program cycle, the SINIT block is no longer executed.



1.2 Example program – Receiving data from a terminal

Load example program:

Insert diskette 1 of the programming software into drive A:

As usual, call the programming software under \ABB-SPS\CS31.

You will find the submenu point *Data restore* under the main menu point *Project man*. Call this point.

Specify A:\DAT_EMPF as file to be saved. For target drive specify C:

Press <F1>.

Now the example program will be copied into the directory C:\ABB-SPS\CS31\BSP. When calling the example you must enter this path and the name DAT_EMPF in the image *Project data* for *project*.

Using the example program, data are received from a terminal via the serial interface COM2 of the 07 KT 92 PLC. It is dealt with 5 different telegrams.

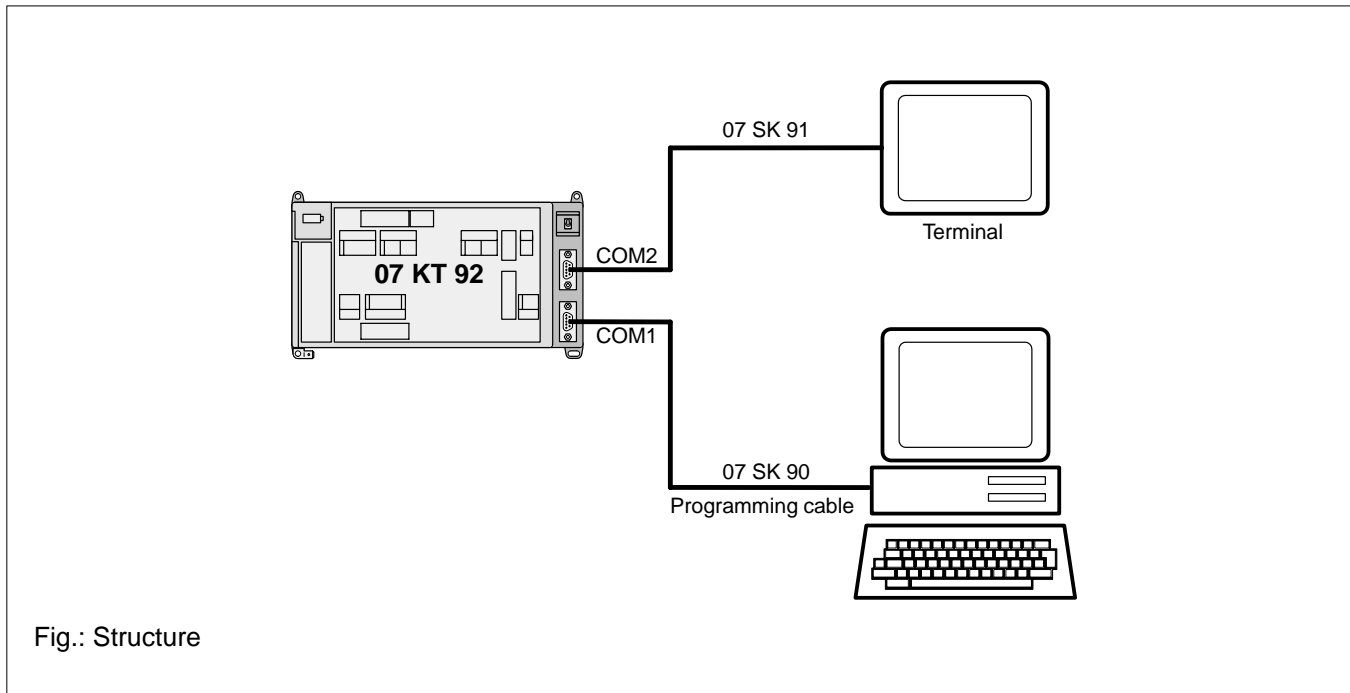


Fig.: Structure

Both, the interface of the terminal and the interface of the PLC must have the same settings (see segment plan1, INITIALIZATION). RTS/CTS is used for handshake (not XON/XOFF).

Check data reception:

If keys are pressed at the terminal, the PLC will receive data. This data reception can be checked. For this purpose, display the EMASm block in the FBD/LD editor, segment plan *RECEIVE DATA*. Switch on *ONLINE* then. In order to select the format of representation, press the <D> key and select the desired format in the displayed menu then. In addition, you can select the word represen-

tation (press the <X> key). You can check the received data at the outputs TELN and MW0...3.

Notes for the use of **07 KR 91**:

The 07 KR 91 PLC has only one interface COM1. In order to use the example, the following changes are required for this PLC:

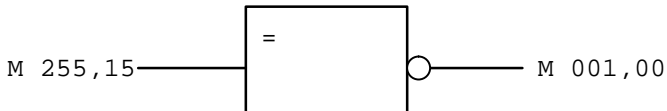
- For the parameter SSK of the blocks SINIT and EMASm, 1 must be entered for COM1.
- The system cable 07 SK 90 must be used for the program transmission. After that, for the data reception this cable must be unplugged and the cable 07 SK 91 must be used instead.

Program description

Segment plan 1: INITIALIZATION

In this program part, the interface COM2 of the PLC is initialized in the following way:

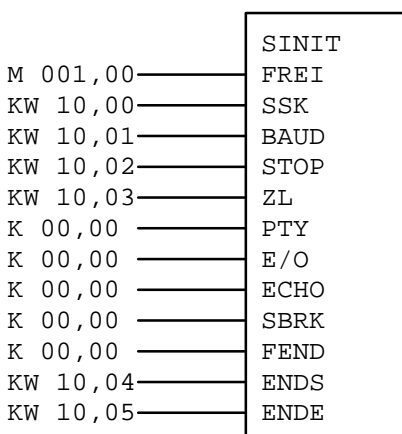
- Once at the start of the program
- Call SINIT function block
- Use M 255,15 flag
 - M 255,15 is 0 at the start of the program
 - Invert value and allocate M 001,00
 - Put value of 1 of M 001,00 to the enable input FREI of SINIT: the interface is now initialized.
 - Set M 255,15 to 1 at the end of the program → with the next program execution the SINIT block is not executed any more.



The parameters of the interface are set as follows:

Explanation	Input	Value
Interface No. 2 = COM2	SSK	2
Baud rate	BAUD	9600
Number of stop bits	STOP	1
Number of data bits per character	ZL	8
Transmission without parity bit	PTY	0
Odd parity bit (no effect because of PTY)	E/O	0
No echo	ECHO	0
Normal status of the send signal TxD	SBRK	0
No output of text termination character	FEND	0
Text termination character for sending direction (without effect)	ENDS	13 (=CR)
Text termination character for receiving direction	ENDE	13 (=CR)

These values are entered by means of constants. They can be entered in the variable editor via the long text of the constants.



Segment plan 2: RECEIVE DATA

In this program part, the receiving procedure is realized.

The EMASm block is used for receiving data. When the program is started, the value of 0 is at the QUIT input. Therefore, the block is able to receive a telegram.

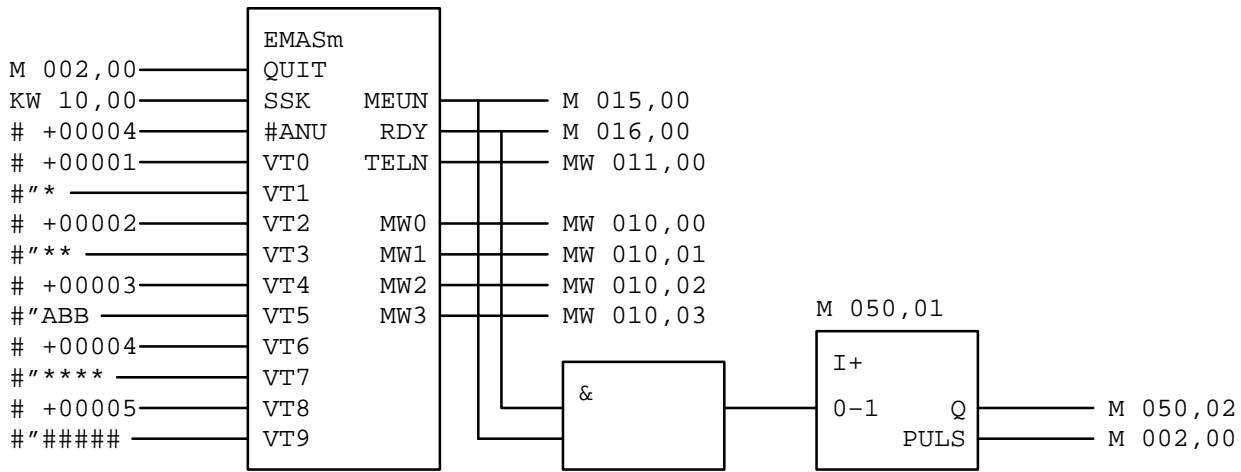
Receiving procedure with the EMASm block

The parameters of the block are set as follows:

Explanation	Parameters	Value
If 0, the block is always ready to receive ASCII characters from the interface	QUIT	0
Interface No. (2 for COM2)	SSK	2
Maximum number of user data outputs MWn; in this example: max. 4 ASCII characters	#ANU	4
Comparative telegram No. 1 1st comparative text: #"* i.e. 1 random ASCII character, terminated by <CR>	VT0 VT1	#1 1 ASCII char.
Comparative telegram No. 2 2nd comparative text: #"* i.e. 2 random ASCII characters, terminated by <CR> etc.	VT2 VT3	#2 2 ASCII char.
The output indicates whether the received data in MW0...3 are valid (0) or invalid (1)	MEUN	0 or 1
The output reports with a 1 that the telegram was received and processed	RDY	0 or 1
Number of the received and valid telegram The received characters from a valid telegram are stored here	TELN MW0...4	1...5 ASCII ch./number

5 different telegrams can be received in this example:

- Telegram 1: 1 random ASCII character, which was terminated by <CR>. MW 11,00 then contains "1", MW 10,01 contains the received ASCII character.
- Telegram 2: 2 random ASCII characters, which were terminated by <CR>. MW 11,00 then contains "2", MW 10,01...02 contain the received ASCII characters.
- Telegram 3: This telegram can only receive the characters "ABB", which were terminated by <CR>. Since this character string can definitely be assigned to telegram No. 3, only MW 11,00 contains a "3".
- Telegram 4: 4 random ASCII characters, which were terminated by <CR>. MW 11,00 then contains "4", MW 10,01...MW 10,04 contain the received ASCII characters.
- Telegram 5: A 5-digit number with sign (+32767...-32767), which were terminated by <CR>. MW 11,00 then contains "5", MW 10,01 contains the received 5-digit number (inclusive sign if the number is negative).



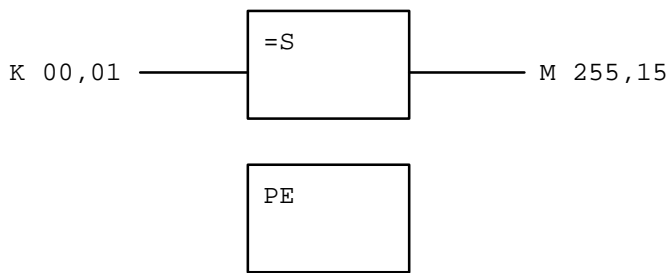
Procedure in case of invalid telegrams

The following example is valid: "ABC<CR>". It contains 3 characters followed by a <CR>. The 3 characters, however, do not match with ABB, the only possibility to send 3 characters (telegram No. 3). In such a case the EMASm block must be acknowledged before a new telegram can be received. For acknowledgement, the outputs MEUN and READY must be AND-combined.

If an invalid telegram has been received, these outputs MEUN and READY were set to 1. The input of the block I+ then gets signal 1, and the block generates an impulse. This impulse acknowledges the EMASm block with its 0→1 edge. EMASm is locked for one program cycle. In the next program cycle, EMASm is ready again for receiving data.

Segment plan 3: PROGRAM END

M 255,15 is set to 1, and, beginning with the next program cycle, the SINIT block is no longer executed.



2 System reaction if an incorrect program is sent

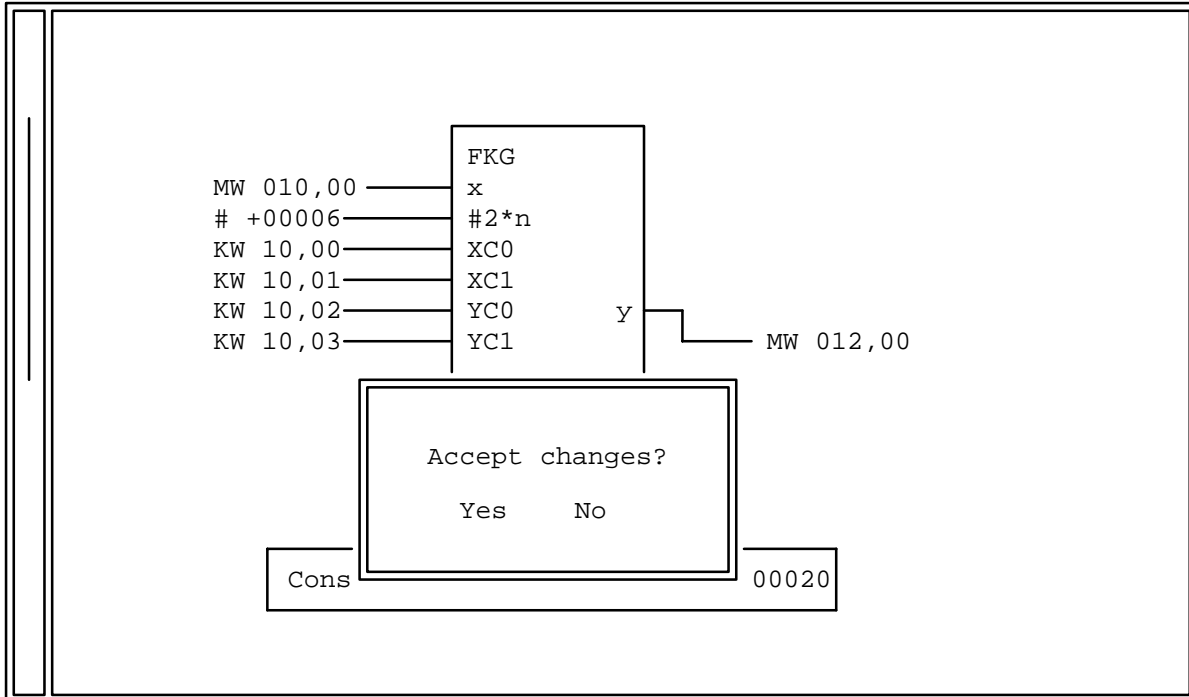
a) Sending of an incorrect program with the status "PLC is running"

In the following example, a program is sent which contains the FKG function block. The input #2*n has a wrong specification. The PLC checks the correctness only with the enabling, i.e. the error cannot be detected during the sending procedure.

PROJECT: TEST

SP No.:002 ABS

TX: TY:



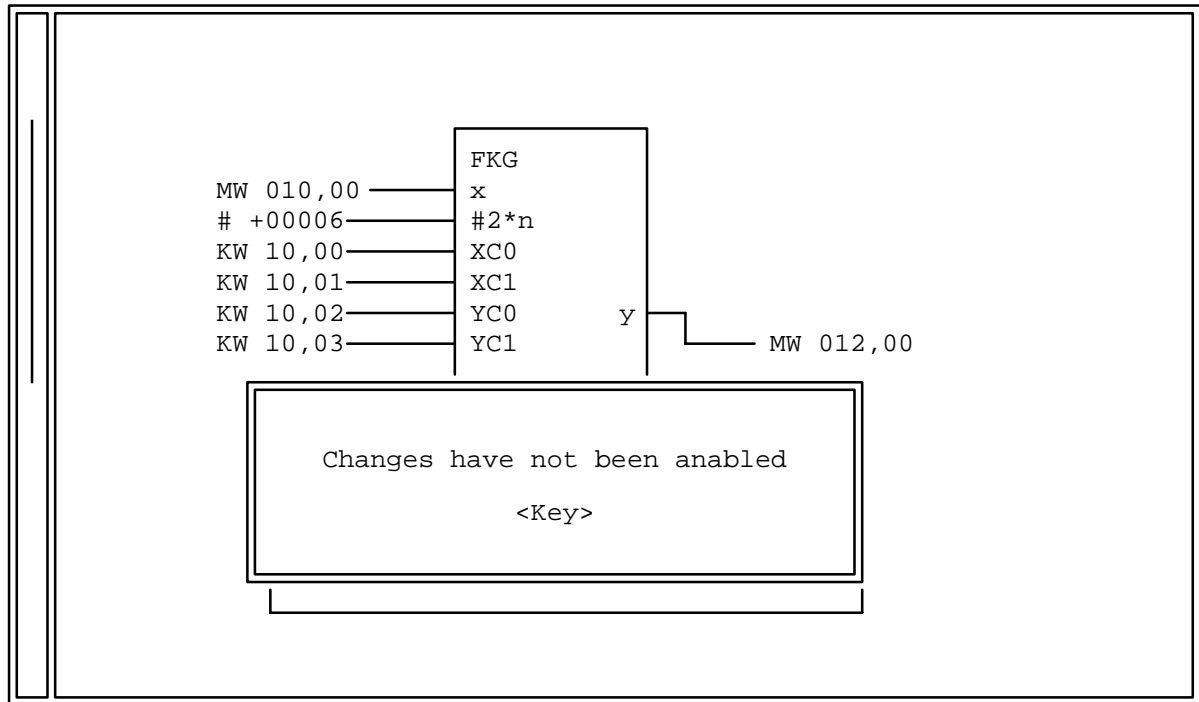
Constants have been sent completely.

After the user has enabled the Online program changes, the PLC detects the wrong definition at the FKG block and sends an error message to the programming system. The changes will not be enabled. The old program still runs.

PROJECT: TEST

SP No.:002 ABS

TX: TY:



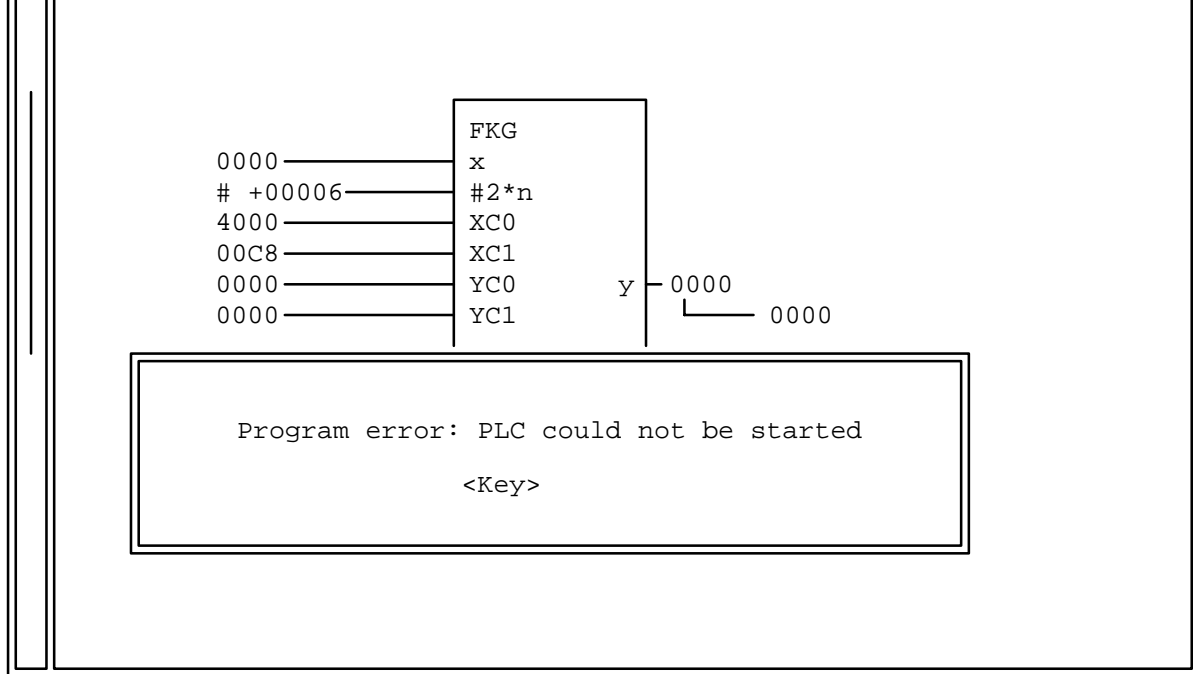
BLOCK ERROR 00129 FKG AT ADDRESS: 00137 <KEY>

After the error has been remedied, the entire program must be sent to the PLC.

b) Sending of an incorrect program with the status "PLC is aborted"

The error at the FKG block is detected by the PLC on the transition from the status "aborted" to the status "running". If the PLC has received the "Start" command in Online from 907 PC 331, the error message shown below appears.

```
PROJECT: TEST                SP No.:002 ABS    ONLINE    TX:03 TY:007  
=SPS:$ PRG:1 = aborted      Variable status
```



```
BLOCK ERROR 00129 FKG AT ADDRESS: 00137 <Key>
```




Printed on chlorine-free bleached paper

ABB Schalt- und Steuerungstechnik GmbH
Eppelheimer Straße 82 Postfach 10 50 09
D-69123 Heidelberg D-69040 Heidelberg

Telephone +49 6221 777-0
Telefax +49 6221 777-111
E-Mail desst.help@de.abb.com
Internet <http://www.abb-sst.de>

Printed in the Federal Republic of Germany (08.99)