

Operating Manual

**ABB Procontic  
Programming System**

907 PC 331  
Programming and Test Software  
System-Specific Part  
ABB Procontic CS31  
Advant Controller 31

---

**ABB Schalt-  
und Steuerungstechnik**





# Preface

## Notes for the user

All commands entered by the user are shown in *italics*, keys are shown as `<Key>` and screen displays, e.g. instructions, appear in **bold**.

In the same way, you can use the *left mouse key* for the `<Enter key>` and the *right mouse key* for the `<space bar>`. For yes/no enquiries, you can use the *left mouse key* for yes and the *right one* for no.

Main menu: The selection of a main menu point is released or defined every time the `<right mouse key>` is pressed. The selected sub-menu point is carried out when the `<left mouse key>` is pressed.

Key combinations:

`<key>-key` or `<key>-key-key`.

In this case, press the first key and hold it down. At the same time, press the following keys.

Example 1: `<Alt>-2`

Press the `<Alt>` key and hold it down; at the same time, press the `<2>` key.

Example 2: `<Ctrl>-K-W`

Press the `<Ctrl>` key and hold it down; at the same time, press the keys `<K>` and `<W>` in turn.

### Calling up online commands with hotkeys

Hotkeys are used to rapidly select functions which can alternatively be executed via pop-up menus. E.g. `<Alt>-S` for "Start the PLC" (see Section 2, Work aid, "ONLINE commands").

### Selecting menu points in the pop-up menus

Rapid selection: by entering the high-lighted letters of the menu line; cursor positioning with `<arrow key up>` and `<down>` or the mouse not necessary.

## Keyboards

The designations of certain keys on keyboards with German, English or American character sets differ. The following table shows the differences:

German character set	English/American character set
<code>&lt;Strg&gt;</code>	<code>&lt;Ctrl&gt;</code>
<code>&lt;Einfg&gt;</code>	<code>&lt;Ins&gt;</code>
<code>&lt;Entf&gt; oder &lt;Löschen&gt;</code>	<code>&lt;Del&gt;</code>
<code>&lt;Bild ▲ &gt;</code>	<code>&lt;PgUp&gt;</code>
<code>&lt;Bild ▼ &gt;</code>	<code>&lt;PgDn&gt;</code>
<code>&lt;Pos 1&gt;</code>	<code>&lt;Home&gt;</code>
<code>&lt;Ende&gt;</code>	<code>&lt;End&gt;</code>
<code>&lt;WR&gt;</code>	<code>&lt;Return&gt;</code>
<code>&lt;Tab. rechts&gt;</code>	<code>&lt;RTAB&gt;</code>
<code>&lt;Tab. links&gt;</code>	<code>&lt;LTAB&gt;</code>
<code>&lt;▲ &gt;</code>	<code>&lt;Shift&gt;</code>

The `<Enter key>` is used in this document. Other standard designations for this key are `<Enter>`, `<Return>`, `<CR>` and `<WR>`.

## Abbreviations

AC31	Advant Controller 31
AWP	User program
IL	Instruction list
FBD	Function block diagram
LD	Ladder diagram
PLC	Programmable (logic) controller
CE	Connection element
<code>&lt;CR&gt;</code>	Carriage Return or Enter, Return
ZE	Central unit



# Contents

---

<b>1</b>	<b>Installation</b> .....	1-1
1.1	Installing programming software on the PC .....	1-2
1.1.1	Manual installation .....	1-3
1.2	Start programming software .....	1-5
<b>2</b>	<b>Notes on programming</b> .....	2-1
2.1	Connectable programming and service devices .....	2-1
2.2	Programming and test software 907 PC 331 .....	2-2
2.3	Programming .....	2-2
2.3.1	Procedure for creating a program ....	2-2
2.3.2	Entering/changing a program .....	2-2
2.4	Saving a program in the Flash EPROM .....	2-3
2.4.1	Program: RAM → Flash EPROM ...	2-3
2.4.2	Program: Flash EPROM → RAM ...	2-3
2.4.3	Commissioning phase .....	2-3
2.5	Special instructions .....	2-4
2.5.1	System constants with ABB Procontic CS31, Advant Controller 31 .....	2-4
2.5.2	Error diagnosis with ABB Procontic CS31, Advant Controller 31 .....	2-6
2.5.3	Intermediate flag, updating .....	2-6
<b>3</b>	<b>Configuration</b> .....	3-1
3.1	Project management .....	3-1
<b>4</b>	<b>ONLINE</b> .....	4-1
4.1	PLC communication .....	4-1
4.1.1	PLC communication 1 .....	4-1
4.1.2	PLC communication 2 .....	4-4
4.1.3	PLC communication 3 .....	4-5
4.1.4	PLC communication from FBD/LD and extended IL .....	4-6
4.2	Test .....	4-10
<b>5</b>	<b>"Compare" program</b> .....	5-1





## 1.1 Installing programming software on the PC

The programming software 907 PC 331 consists of 2 diskettes. If a version of 907 PC 331 already exists on your PC, please first read the instructions at the end of this page.

Insert diskette 1 into drive A. Enter the following instruction:

A: INSTALL <Enter key>

The installation program now determines the drives available on your PC.

The following appears, for example:



You can reach the individual lines with the <arrow keys up> and <down>. Select one possibility in each case with the <arrow keys left> and <right>. Select the appropriate drives: In the first line, the drive containing the installation diskette; in the following line the desired hard disk. Also select the required language version. Start the installation by pressing the <Enter key>.

If the values FILES and BUFFERS must be adapted in the CONFIG.SYS file, this is done automatically by the installation program. An appropriate message for this appears. Installation is continued after the <Enter key> is pressed.

The following message appears after installation is completed (selected drive: C:):

**907 PC 331 was installed completely.**

**Remove installation diskette from the diskette drive and type**

**C:\AC31**

**to start.**

**If the CONFIG.SYS file was changed, first press <Ctrl><Alt><Del> in order to restart the PC.**

### Notes

At the start, the installation program checks whether an old version of 907 PC 331 is already installed. Installation can only take place if the old version is overwritten by the new one. **Therefore, if you have relevant user data in the sub-directory ABB-SPS\AC31, you must first save this. All files in directory AC31 are deleted and replaced by new files during installation.**

If difficulties arise during automatic installation, you can alternatively carry out the → manual installation.

## Notes on an update installation:

If you want to execute an **Update Installation** you must update your library.

Start the programming software as described under "1.2. Starting the programming software" and choose "Library", "Read Manufacturer Library". In order to update your SER90 library, enter "NEU" (if you want to work with the German version) resp. "NEW" (if you want to work with the English version). In order to update your SER50 library, enter "SER50NEU" (if you want to work with the German version) resp. "SER50NEW" (if you want to work with the English version). Please observe, that the library to be updated (AC31 or SER50) is entered in the project data.

The existing library is now updated. Your user CEs will be conserved.

### 1.1.1 Manual installation

In the case of manual installation, you must make a distinction as to whether new installation is taking place or whether an older version of 907 PC 331 already exists. In this case, installation can only be carried out by overwriting the older version with the new one. **Therefore, if you have relevant user data in the sub-directory ABB-SPS\AC31, you must first store this data.** All files in the directory AC31 are deleted and replaced by new ones during installation. The following steps describe manual installation of the programming software, divided up into new installation and update installation. Here, installation is described from drive A: to hard disk C:. Adapt this accordingly for other drive designations.

Enter the following commands from drive C:\. When doing so, terminate every line with the *<Enter key>*:  
Insert diskette 1 into drive A:.

#### New installation

```
C:\> COPY A:AC31.BAT
C:\> MD \ABB-SPS
C:\> CD \ABB-SPS
C:\ABB-SPS> MD AC31
C:\ABB-SPS> CD AC31
C:\ABB-SPS\AC31>
```

Then enter the following command:

```
C:\ABB-SPS\AC31> COPY A:AC31V3.?
```

Insert floppy disk 2 into drive A:. Then enter the following command:

```
C:\ABB-SPS\AC31> COPY A:AC31V3.?
```

Proceed with the following commands:

```
C:\ABB-SPS\AC31> A:PACKEN AC31V3.EXE AC31V2
C:\ABB-SPS\AC31> AC31V3
```

This takes a few minutes. Please answer enquiries following Overwrite with "Y" for yes. Then enter the following commands:

```
C:\ABB-SPS\AC31> DEL AC31V3.*
C:\ABB-SPS\AC31> MD BIB
C:\ABB-SPS\AC31> MD PROJEKT
C:\ABB-SPS\AC31> COPY EXAMP*.* PROJEKT
C:\ABB-SPS\AC31> DEL EXAMP*.*
C:\ABB-SPS\AC31> COPY ONLINE.OE PROJEKT
C:\ABB-SPS\AC31> DEL ONLINE.OE
```

#### Update installation

```
C:\> CD \ABB-SPS
```

```
C:\ABB-SPS> CD AC31
C:\ABB-SPS\AC31> DEL *.*
```

Then enter the following command:

```
C:\ABB-SPS\AC31> COPY A:AC31V3.?
```

```
C:\ABB-SPS\AC31> COPY A:AC31V3.?
```

Proceed with the following commands:

```
C:\ABB-SPS\AC31> A:PACKEN AC31V2.EXE AC31V3
C:\ABB-SPS\AC31> AC31V3
```

```
C:\ABB-SPS\AC31> DEL AC31V3.*
```

```
C:\ABB-SPS\AC31> COPY EXAMP*.* PROJEKT
C:\ABB-SPS\AC31> DEL EXAMP*.*
C:\ABB-SPS\AC31> COPY ONLINE.OE PROJEKT
C:\ABB-SPS\AC31> DEL ONLINE.OE
```

**If you want to use the German version, please enter the following:**

```
C:\ABB-SPS\AC31> COPY MPST_F_D.TEX MPST_F.TEX
C:\ABB-SPS\AC31> COPY MPST_F_D.TEX
MPST_F.TEX
C:\ABB-SPS\AC31> COPY SER50_D.VE? BIB\SER50.*
C:\ABB-SPS\AC31> DEL SER50*.*
C:\ABB-SPS\AC31> COPY SER90_D.VE? BIB\AC31.*
C:\ABB-SPS\AC31> DEL SER90*.*
C:\ABB-SPS\AC31> COPY SER50NEU.* BIB
C:\ABB-SPS\AC31> COPY NEU.* BIB
C:\ABB-SPS\AC31> COPY MPST_D.* MPST.*
C:\ABB-SPS\AC31> COPY MPST_D.* MPST.*
```

**If you want to use the English version, please enter the following:**

```
C:\ABB-SPS\AC31> COPY MPST_FG.TEX MPST_F.TEX      C:\ABB-SPS\AC31> COPY MPST_FG.TEX
                                                    MPST_F.TEX
C:\ABB-SPS\AC31> COPY SER50_GB.VE? BIB\SER50.*  C:\ABB-SPS\AC31> DEL SER50_*. *
C:\ABB-SPS\AC31> COPY SER90_GB.VE? BIB\SER90.*  C:\ABB-SPS\AC31> DEL SER90_*. *
C:\ABB-SPS\AC31> DEL AC31_*. *                  C:\ABB-SPS\AC31> COPY SER50NEW.* BIB
                                                    C:\ABB-SPS\AC31> COPY NEW.* BIB
C:\ABB-SPS\AC31> COPY MPST_GB.* MPST.*          C:\ABB-SPS\AC31> COPY MPST_GB.* MPST.*
```

**Only now are the following commands entered:**

```
C:\ABB-SPS\AC31> RENAME MPST_DA.EXE MPST_D.EXE  C:\ABB-SPS\AC31> RENAME MPST_DA.EXE
                                                    MPST_D.EXE
C:\ABB-SPS\AC31> DEL NEU.*                       C:\ABB-SPS\AC31> DEL NEU.*
C:\ABB-SPS\AC31> DEL NEW.*                       C:\ABB-SPS\AC31> DEL NEW.*
```

**New installation**

This terminates the installation.

Take the diskette 2 out of drive A:.

**Update installation**

This terminates the installation. Before starting the programming software, you must still check the values of FILES and BUFFERS in the file CONFIG.SYS. The values must each be at least 25 and must, if necessary, be set at this figure. When CONFIG.SYS is changed, restart the PC next.

Now start the programming software as described under "1.2. Starting the programming software" and choose "Library", "Read Manufacturer Library". In order to update your SER90 library, enter "NEU" (if you want to work with the German version) resp. "NEW" (if you want to work with the English version). In order to update your SER50 library, enter "SER50NEU" (if you want to work with the German version) resp. "SER50NEW" (if you want to work with the English version). Please observe, that the library to be updated (SER90 or SER50) is entered in the project data.

The existing library is now updated. Your user CEs will be conserved.

Take the diskette 2 out of drive A:.

## 1.2 Start programming software

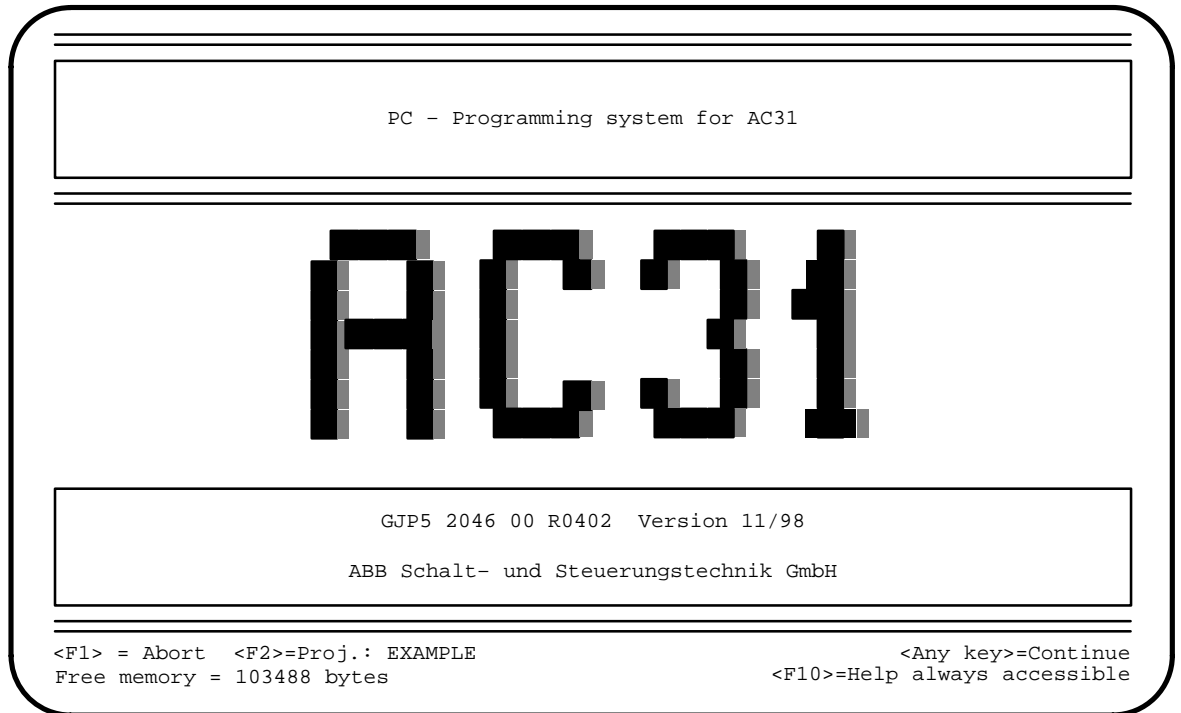
The following details apply to when you have installed the programming software under drive C:.

A batch file AC31.BAT was filed in the root directory during the installation. This allows you to start the programming software as follows:

```
C:\AC31 <Enter key>
```

Start the programming software. The switch-on image becomes visible.

Display:

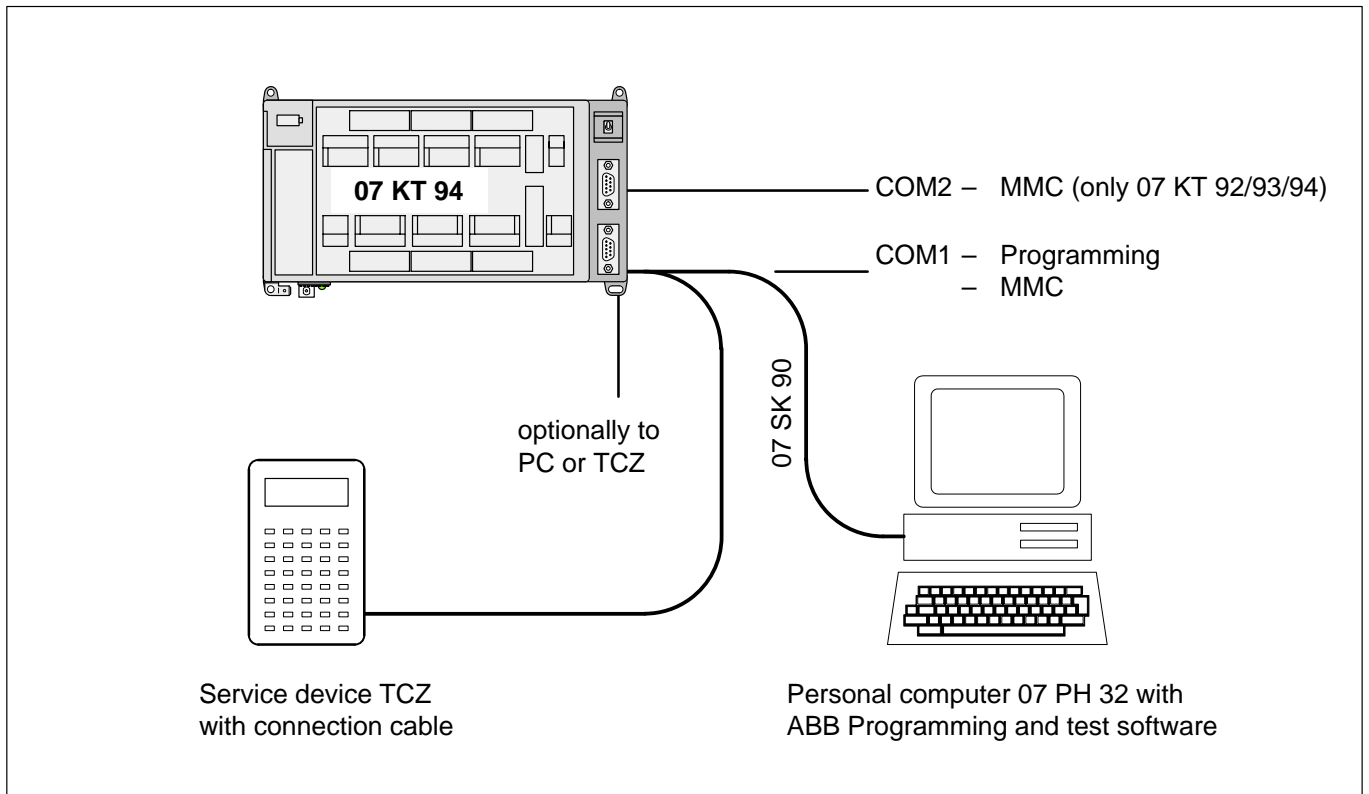


To call up the project *EXAMPLE*: Press <F2>. Enter AC31 <Enter key> as the password. This brings you into the main menu, from where you can process the project.



## 2 Notes on programming

This section gives you an overview of the procedure for programming.



### Contents

<b>2.1</b>	<b>Connectable programming and service devices</b> .....	2-1
<b>2.2</b>	<b>Programming and test software 907 PC 331</b> .....	2-2
<b>2.3</b>	<b>Programming</b> .....	2-2
	Procedure for creating a program .....	2-2
	Entering/changing a program .....	2-2
<b>2.4</b>	<b>Saving a program in the Flash EPROM</b> .....	2-3
<b>2.5</b>	<b>Special instructions</b> .....	2-4

### 2.1 Connectable programming and service devices

#### ● Programming and test unit 07 PH 32

IBM compatible notebook PC for the ABB programming and test software.

Technical data:

- 80386 SL processor
- Clock frequency 25 MHz
- 4 MByte RAM, can be extended up to 10 MByte
- Cache
- 84 MByte hard disk

- 3.5" floppy disk drive (1.44 MByte)
- 9.5" VGA monitor
- Battery operation for 3 hours
- Mains connection
- Mouse for rapid cursor control
- Second serial interface EIA RS-232
- Adaptor from 9-pole to 25-pole serial interface
- Dimensions 279 x 216 x 44 mm
- Weight ca. 2.7 kg
- Order No. for 07 PH 32: GJV 307 2104 R0002

#### ● TCZ service device

This hand-held device is suitable for servicing and for small alterations, e.g. for changing variable addresses, flags, times and counters.

Programming language:

- Instruction list (IL)

Technical data:

- 4 lines LED display with 16 characters each, 5 x 7 dots
- Keyboard with 40 keys
- Dimensions 80 x 155 x 31 mm
- Weight ca. 400 g
- Order No. for TCZ: FPR3200002R1001

## 2.2 Programming and test software 907 PC 331

Programming and test system for entering, testing and documenting programs.

Programming languages:

- Function block diagram (FBD)
- Instruction list (IL)
- Ladder diagram (LD)

Features:

- Easy handling due to modern menu surface
- Menu selection with mouse or keyboard
- Operator guidance through instruction and help texts
- Comments
- Symbolic variables
- ONLINE functionen such as status display bit/word, breakpoint, forces
- Comprehensive program documentation in different formats

## 2.3 Programming

The PLC can be programmed in:

- Instruction list (IL)
- Function block diagram (FBD)
- Ladder diagram (LD)
- Sequential function chart (SFC)

Programming in the instruction list corresponds to DIN 19239. This DIN defines the structure of the instruction and of the function block. The user program is entered into the PLC via the serial interface COM1 of the PLC.

When doing so, the ASCII characters from the instruction list are directly transferred to the PLC. The programming and test software, integrated in the PLC, translates the instruction list during the transfer into another format. This format (pseudo-code) is required for the processing. The software stores it in the user program RAM of the PLC.

The ASCII clear text interface revealed enables access to the PLC with different user-friendly aids. These range from the simple ASCII keyboard with digital display, beyond the terminal, up to the user-friendly programming and test system.

### 2.3.1 Procedure for creating a program

Creating an executable PLC program can be divided up as follows:

- Enter program, constants, system constants for operating modes and system constant for cycle time into the programming system
- Translate the program in the programming system
- Send program to the control system
- Send constants to the control system
- Save program in the Flash EPROM
- Start up and test the PLC program

For successful entering and altering of a program, please **pay special attention** to the following detailed description.

### 2.3.2 Entering / changing a program

A PLC program can be entered or changed via the serial interface COM1 of the PLC in the following statuses:

- "Aborted" status
- "Running" status

#### Entering / changing a program in the "aborted status"

In the "aborted" status, program changes can be made without restriction. *Before* starting a changed program, the operating modes must be set so that all operand ranges are initialized. It is thus ensured that all operands and historical values are initialized to "0", and values stored by the battery back-up do not lead to malfunctions.

#### Procedure when entering / changing a program in the "aborted status"

- Enter program / changes into the programming system
- Translate program / changes in the programming system
- Send program / changes to the control system
- Save program in Flash EPROM

Note:

For details really see also chapter 2.4 "Program saving in the Flash EPROM".

#### Entering / changing a program in the "running" status

- Changes can be made to a *running* PLC program.
- When the changes are taken over, there is *no* PLC standstill time.
- The user must exercise *extreme care* when making changes to a *running* PLC program, as the program changes immediately influence the running process without the possibility of tests. Programming errors can have grave results.

## Permissible program changes

- Any program changes are permissible as long as no *function blocks with historical values* are affected.
- *Function blocks with historical values* may only be added to or omitted at the *program end*.
- Any changes to indirect constants are permissible. This also includes changing the cycle time. If system constants are changed, chapter "system constants" must be taken into account.

## Forbidden program changes

- Deleting or adding to function blocks with historical values *within* the PLC program is not allowed.

## Procedure when changing a PLC program in the "running" status

- Carry out changes in the programming system
- Translate changes in the programming system
- Send changes to the control system
- Release changes
- Save changes in the Flash EPROM

Note:

For details really see also chapter 2.4 "Program saving in the Flash EPROM".

## Rejecting a released change to a running PLC program and reactivating the old program status

### Not with AC31 Series 30, 40, 50

Using the "Reactivate old program" menu field, the old program status can be reproduced. I.e. the changes carried out to **and** released for a running PLC program are rejected again.

In addition, the PLC reproduces the old program status. The old program status is the status of the program **prior to** the program alteration.

After entering the "Reactivate old program" menu field, the reactivation of the old program status takes place without further interference by the user within approx. 1 ms.

The command can be used if the user realizes that the program changes carried out do not produce the results he hoped for.

Caution:

The reactivation of the old program status only takes place in the PLC. The user is expected to make undone the changes in the programming and test system himself.

## 2.4 Saving a program in the Flash EPROM

- The user can copy the PLC program from the program RAM to the Flash EPROM in order to save it in case of power fail or power off. To protect the program against

power off, a back-up battery is therefore not necessary.

### 2.4.1 Program: RAM → Flash EPROM

The program is saved in the Flash EPROM as follows:

- in the ABB programming and test system by calling the menu field  
"Progr. → EPROM"

or

- in the terminal emulation by entering the command  
`SP<CR>`

The Flash EPROM is deleted as follows:

- in the ABB programming and test system by calling the menu field  
"Delete EPROM"

or

- in the terminal emulation by entering the command  
`DEEP<CR>`

All configuration data which is in conjunction with the user program is saved inclusively when the PLC program is copied to the Flash EPROM.

### 2.4.2 Program: Flash EPROM → RAM

If a PLC program is stored in the Flash EPROM, this program is copied automatically into the program RAM in the following events:

- Power 'ON'
- RUN/STOP switch from STOP → RUN
- Program started with programming system

In order to make sure that an changed program is not overwritten by an old program version stored in the Flash EPROM, it is first necessary

- to store the changed program in the Flash EPROM

or

- to delete the Flash-EPROM

*before* any of the above mentioned actions.

### 2.4.3 Commissioning phase

If many program changes must be made during a system's commissioning phase, it is recommended to delete the Flash EPROM and to use a back-up battery for the RAM instead.

This has the following advantages:

- The time taken for repeated program storage is saved
- Inconsistencies between the programming device and the controller due to "forgotten" program saving cannot occur.

After commissioning has been completed, it is appropriate to save the program from the RAM into the EPROM.

## 2.5 Special instructions

When calling the connection elements, please always note the description of the corresponding element in the documentation.

### 2.5.1 System constants with ABB Procontic CS31 and Advant Controller 31

When a new project is opened, a default variable list containing reserved indirect constants is read in automatically. These constants serve to preset system parameters for the PLC.

ABB Series 90		Variable editor		EXAMPLE
E	Variable	Symbol		Long text
	K 00,00	BIT=0	0	
	K 00,01	BIT=1	1	
	KW 00,00	MAST_SLV	-2	
	KW 00,01	INIT_M	0	
	KW 00,02	INIT_MW	0	
	KW 00,03	INIT_MD	0	
	KW 00,04	INIT_S	0	
	KW 00,05	INIT_VW	0	
	KW 00,06	MODE_SST	0	
	KW 00,07	FK3_REAK	0	
	KW 00,08	ÜLAST_REAK	0	
	KW 00,09	HOCHFÄHR	0	
	KW 00,10	SLV_SEND	0	
	KW 00,11	SLV_REC	0	
	KD 00,00	ZYKL_ZEIT	10	
A		E	E	

Free memory for variables 65200 Bytes (100%)

#### Important:

If these constants are *not* contained in the user program, they are deleted after calling function "Delete variables (in the block)". The constants and their values must be then entered anew.

The default variable list can, however, also be read in with the Block command <Ctrl>KR. The following source must then be entered:

```
C:\ABB-SPS\AC31\SYS_CONS.SYM
```

The variable list contains the following entries:

#### Bit constant 0

K 00,00 Bit=0 (default value: 0)

#### Bit constant 1

K 00,01 Bit=1 (default value: 1)

#### Setting Master PLC, Slave PLC or Stand-alone PLC

KW 00,00 MAST\_SLV  
Default value: -2 i. e. Stand-alone

#### Setting initialization: binary flags

KW 00,01 INIT\_M  
Default value: 0 i. e. initialization of the entire range

#### Setting initialization: word flags

KW 00,02 INIT\_MW  
Default value: 0 i. e. initialization of the entire range

#### Setting initialization: double word flags

KW 00,03 INIT\_MD  
Default value: 0 i. e. initialization of the entire range

#### Setting initialization: step chains

KW 00,04 INIT\_S  
Default value: 0 i. e. initialization of the entire range

#### Setting initialization: historical values

KW 00,05 INIT\_VW  
Default value: 0 i.e. initialization of the historical values

#### Setting mode of use of the serial interface

KW 00,06 MODE\_SST  
Default value: 0 i.e. mode of use is determined by the status of pin 6 of the interface connector.

#### PLC response to errors of class 3

KW 00,07 FK3\_REAK  
Default value: 0 i.e. only signal errors

**PLC response to overload/short circuit at the direct outputs A 62,00 ... A 62,07**

KW 00,08 ÜLAST\_REAK  
Default value: 0 i.e. automatic acknowledgement of the overloaded output

**Initialization of the AC31 system after power ON**

KW 00,09 HOCHFAHR  
Default value: 0 i.e. user program is started immediately

**Size of the transmitting area of the slave PLC**

KW 00,10 SLV\_SEND  
Default value: 0

**Size of the receiving area of the slave PLC**

KW 00,11 SLV\_REC  
Default value: 0

**Setting of the cycle time**

KD 00,00 ZYKL\_ZEIT  
Default value: 10

The cycle time of the user program is preset by the double word constant KD 00,00. Only integral multiples of 5 msec are permitted as the cycle time. The entry must be made in the unit of time [ms]. The value is entered in the long text of the constant in the variable editor.

**Setting the system constants so that these constants are not deleted when using the function 'Delete variables in block'**

The user can configure the constants as follows:

1. Variable list

The system constants with the relevant, required value (in the long text) can be transferred to the variable list as they are or can be changed. They are transferred to the PLC with function "Send constants". In this case, the system constants are deleted by command "Delete variables (in the block)".

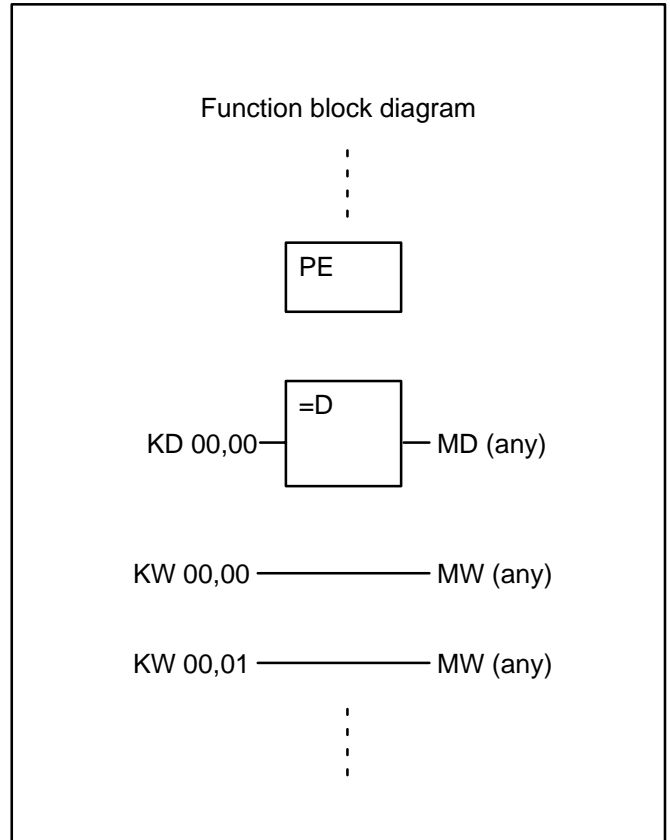
2. User program

The system constants are planned in the user program. The value is entered in the long text of the constant.

In this case, the system constants are retained even after "Delete variables (in the block)".

It is practical to preset by a connection element (CE) or an instruction after the end of the program (PE).

Example function block diagram:



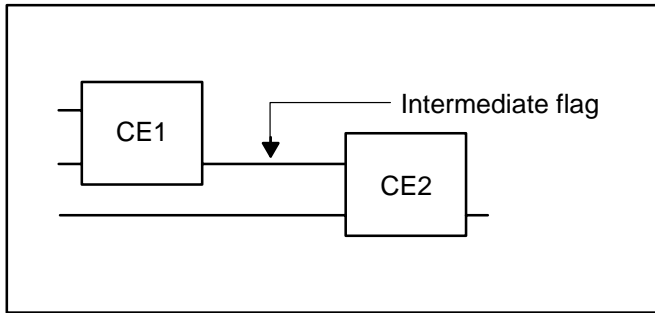
### 2.5.2 Error diagnosis with Advant Controller 31

Using the default online list, it is possible to perform error diagnosis for Advant Controller 31. This online list can be read in with function Read block. The online list (which is blank at the start) must be called for this purpose. The default online list is read in with command <Ctrl>KR and by entering:

```
C:\ABB-SPS\AC31\PROJEKT\ONLINE
```

### 2.5.3 Intermediate flag, updating

If an output of a connection element (CE) is connected directly to the input of another CE in the function block diagram, the programming and test system 907 PC 33 automatically assigns an intermediate flag for this connection.



This intermediate flag is used several times in the program unless it is assigned "globally". Owing to this multiple use, new values are continually assigned to the intermediate flag. In the case of some connection elements (CEs), the outputs and therefore the associated intermediate flags are not updated in every program cycle. Inter-

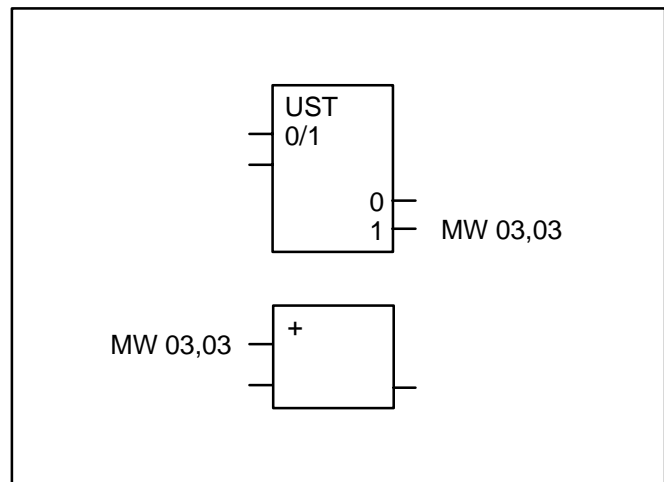
mediate flags not updated by the CE1 can lead to faulty results with CE2. These faults are often difficult to recognize and find.

Remedy:

1. As a result, the outputs may not be directly connected with inputs of the subsequent CEs by lines in the case of CEs which do not update their outputs. In this case, the "connection" to the next CE must be made by the explicit definition of a flag which is not in the range of the intermediate flags.
2. Alternatively, it is also possible to assign the intermediate flags globally in the concerned partial plan by adding @GLO as a text at the start of the partial plan.

Example:

No updating of the CE outputs; consequently, assign a flag explicitly.





## 1 Project data

907 PC 331 displays the following after calling menu option *Project data*:

ABB Series 90		Project data		EXAMPLE	
PLC manufacturer: ABB		PLC name _____: AC31		PLC version	
Project____: C:\ABB-SPS\VAC31\PROJEKT\EXAMPLE				Series 90	
Library____: BIB\VAC31				07KP62	
PLC type_____: Series 90		Timeout:		Series 30, 40, 50	
Responsible_____:		Last update:		T320 V8	
Project designation____:					
Project description____:					

The *Project data* menu contains information relevant to creation, processing and documentation of the user program.

Refer to 907 PC 33, chapter 5, for explanation of the parameters which can be entered.

### PLC type

The first step is to select the PLC type field with the cursor. Press the space bar to set and select the PLC type for which the user program is intended.

Series 90:

Make this selection when you program a 07 KR 91, 07 KT 92, 07 KT 93 or 07 KT 94.

Series 30,40,50:

Make this selection when you program a 07 KR 31, 07 KT 31, 07 CR 41, 07 CT 41, 07 KR 51 or 07 KT 51.

07 KP 62:

Make this selection when you program the communication processor 07 KP 62 of the system ABB Procontic T200.

T320 V8:

Make this selection when you are processing a project which you have transferred with data transfer from 907

PC 32. This allows you to process and adapt the operands which, for example, exceed the limits of Series 90.

This activates the PLC type-dependent syntax monitoring facility for the program input. The relevant range limits of the operands of Series 30, 40, 50 and 90 are listed in Volume 4.

**Note: The flag ranges (M and MW) are not limited by the Series 30, 40, 50 choice. The user has to take care not to use an invalid variable. Otherwise a transmission error will occur when the program is sent to the central unit.**

There is no control type-dependent monitoring for the input of the connection elements (CEs) in the FBD/LD editor. All CEs contained in the CE library can be entered with every control type setting.

All further settings in the above screen display are explained in the operating manual 907 PC 33, General Part.

The choice of Series 30, 40, 50 selects the library with connection elements (CEs) available for these central units.

**With the Series 30, 40, 50 the library of SER50 has to be chosen instead of the SER90 library. The library of SER50 contains all connection elements (CEs) available for Series 30, 40, 50.**

**IMPORTANT:**  
 Depending on the nature of the **ONLINE** functions, these have a direct influence on the PLC program and/or the I/O signals and thus on the process. Therefore, in all cases before running the function, make sure that personal injury and machine damage are impossible!

907 PC 331 provides a series of functions for commissioning and checking the PLC program. These include the main menu options *PLC Commun. 1...3* and the **ONLINE** functions that can be called in the FBD/LD editor, extended IL and the variable editor in a menu. To do this, your PC's serial interface COM1 must be connected to the PLC's serial interface with the cable 07 SK 90 R1, FPTN404948R0002 or 07 SK 50. You will find a further description of this cable in the system description of the Advant Controller 31, Volume 2 – Hardware.

You can use either the interface COM1 or COM2 of your PC for communication with the PLC. The interface is set in the main menu *Config.PC33, COM port for prog..* When delivered, 907 PC 331 is set to the interface COM1.

## 4.1 PLC communication

### Monitoring of the access path

If the programming system loads a PLC program in the PLC, the project name is also transferred to the PLC and stored.

The programming system automatically checks whether the access path leads to the required user program, before each access operation to the PLC.

For this purpose, a check is conducted in order to establish whether the project name in the PLC is the same as the project name in the programming system.

This ensures that the programming system is connected to the correct user program.

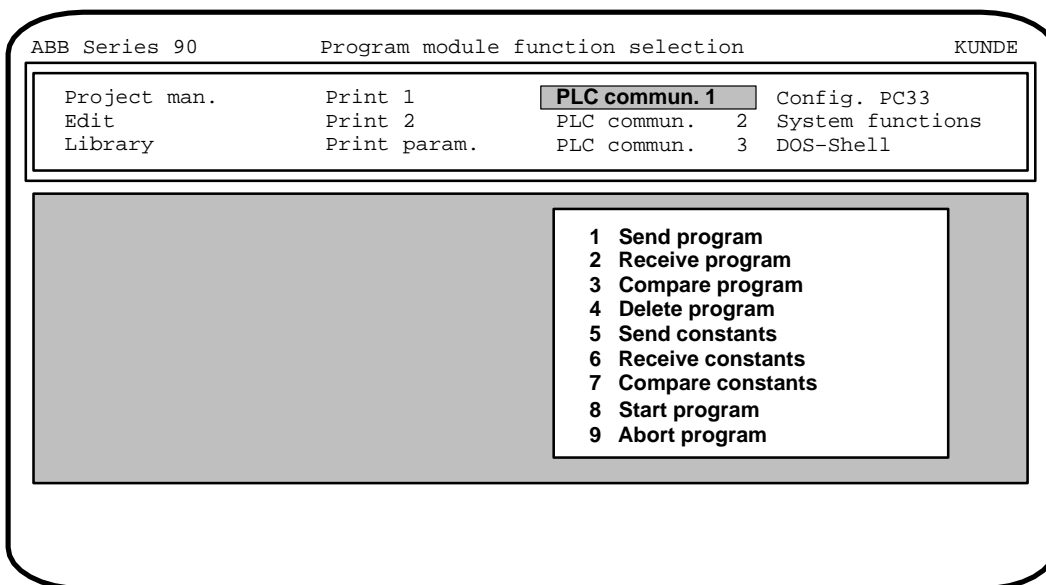
The user does not notice this check of the access path if the settings in the PLC are the same as the settings in the programming system. If differences are established during the check, a message is displayed, informing the operator what setting exists in the PLC and what setting exists in the programming system. The user must then acknowledge whether the setting in the programming system is to be transferred to the PLC or not.

Message if the names do not correspond:  
 ID in PLC:XXX / im System:YYY. Change? (Y/N)  
 XXX = project name in the PLC  
 YYY = project name in the programming system

Note on communication:  
 Communication between the PLC and the programming and test system is possible only if operating mode DEUTSCH is set on the PLC (default setting when delivered). This also applies if working with the English programming software.

### 4.1.1 PLC communication 1

The programming system displays the following after calling menu option *PLC commun. 1*:



## 1 Send program

The PLC program which exists in the form of an instruction list is transferred if menu option *Send program* is selected.

After transfer, you are prompted to decide whether the values of the constants (K, KW, KD) are also to be transferred. If you answer “Yes”, the constants are transferred.

We must distinguish between two cases when transferring programs to the PLC. We suggest the following procedure:

### Case 1:

The program in the PLC is in status “ABORTED”:

- Transfer program to the PLC
- Transfer constants to the PLC
- Save program in flash EPROM
- Start program

### Note:

Before starting the PLC program the system constants KW00,01 ... KW00,05 have to be set to 0 (= default) if the program has been modified in order to be on the safe side, so that all operand areas are initialized with 0 when starting the program. This makes sure that stored values of operands or historical values do not result in malfunctions. After program start the system constants KW00,01 ... KW00,05 can again be set to the values required for the application.

### Case 2:

The program in the PLC is in status “RUNNING”: – Transfer program to the PLC

- Transfer constants to the PLC
- Enable changes in the PLC for processing
- Save program in flash EPROM

### Notes:

When the alterations are taken over, there is *no* PLC standstill time.

The user must exercise *extreme care* when making changes to a *running* PLC program, as the program changes immediately influence the running process without the possibility of tests. Programming errors can have grave results.

## 2 Receive program

The program is read from the PLC and stored in the programming system if you select menu option *Receive program*.

The start and end address of the program or program part to be read out must be specified. If the address of the end of the program PE is known, the program can be read out specifically up to this address. It is advisable to determine the address of the end of the program from the translated IL since entering the end address substantially speeds up the process of reading back from the PLC. If the PLC program is not read out as of word address 0, the range between word address 0 and the entered start address is padded automatically with NOPs in the programming system.

## 3 Compare program

The program is read out of the PLC and compared to the current program stored in the programming system if you select menu option *Compare program*.

### Modularized projects:

#### Calling up from the overall project level

In the case of modularized projects, the program comparison is conducted module–by–module. The modules are compared in the order in which they are listed in the modularization editor.

### Modularized projects:

#### Calling up from the program module level

In the case of modularized projects, program comparison takes place on a module–related basis if it is called up from the program module level. The PLC program is read out and compared starting at the address where the module begins up to the last address of the module.

After the program has been read out of the controller, all instructions are checked for differences. Stating the word number, it is shown whether the command, operand or operand number differ.

The program comparison can be aborted after every detected difference. The number of differences found is displayed at the end.

## 4 Delete program

The program is deleted from the PLC if you select menu option *Delete program*.

The programming system suggests “0” as an initial word number and “30583” as an end word number. Change the end word number to the maximum number of addresses supported by the PLC.

## 5 Send constants

The indirect constants (K, KW, KD) are transferred to the PLC if you select menu option *Send constants*.

If only values of constants are changed in the programming system and the program remains unchanged, it suffices to transfer these to the PLC with function “*Send constants*”. In this case, it is not necessary to transfer the program itself again to the PLC.

Function *Send constants* can be performed if the PLC program is in

- status “RUNNING” or in
- status “ABORTED”.

## 6 Receive constants

The indirect constants (K, KW, KD) of the PLC program are read from the PLC and stored in the programming system in the variable list if you select menu option *Receive constants*.

The two binary constants (K 00,00 and K 00,01) are read out automatically. In the case of the word constants (KW) and the double word constants (KD), the required range must be specified. For example:

*Word constants from: KW 00,00 to KW 01,00.*

You can quit the function with <ESC>. When reading out, it is of no importance whether the constants to be read out are actually used in the program or not.

## 7 Compare constants

The constants (K, KW, KD) of the PLC program are read from the PLC and compared with those in the current project of the programming system if you select menu option *Compare constants*. Differences are displayed.

## 8 Start program

The PLC program is started if you select menu option *Start program*. If there is a program in the Flash EPROM, this program is copied to the user program RAM and started. The user must thus ensure that

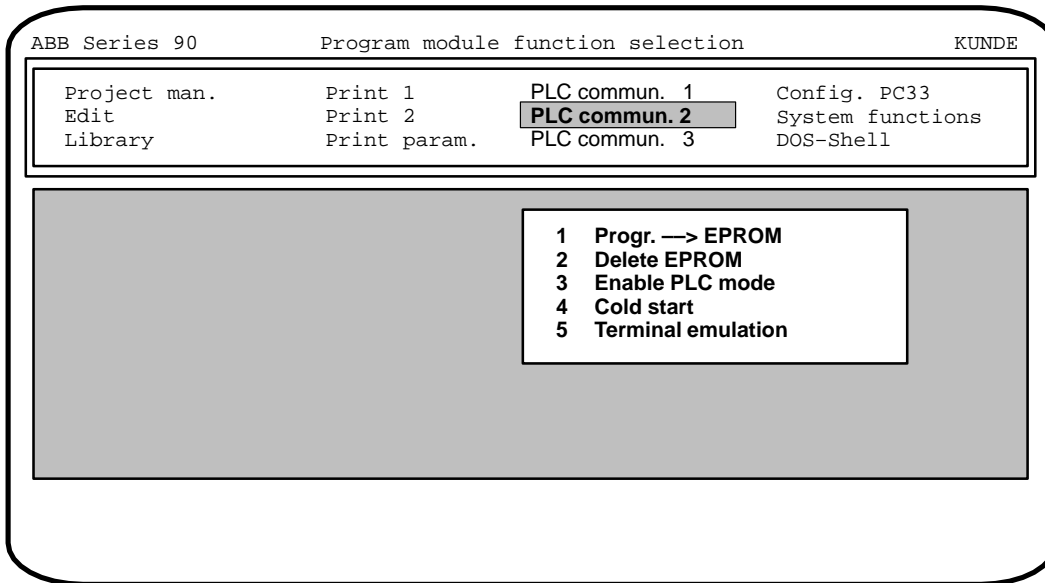
- the most recent program version is always saved in the Flash EPROM or
- he works with an erased Flash EPROM during the test phase in which many program changes occur.

## 9 Abort program

The PLC program is aborted if you select menu option *Abort program*. The related process outputs are set to the value 0.

## 4.1.2 PLC communication 2

The programming system displays the following after calling menu option *PLC commun. 2*:



### 1 Program —> EPROM

When the menu point *Progr. —> EPROM* is selected, the user program is transferred from the RAM memory of the PLC into the Flash EPROM. This safeguards the user program against power failure.

Whenever the PLC program is started up, the PLC program is first automatically transferred from the Flash EPROM to the RAM memory. The PLC program is then started from the RAM memory. Automatic transfer of the PLC program from the Flash EPROM to the RAM memory naturally only occurs when a PLC program is actually present in the Flash EPROM.

### 2 Delete EPROM

Selecting the menu point *Delete EPROM* deletes the user program in the Flash EPROM of the PLC. The Flash EPROM should be deleted, for example, when you wish to prevent the program being overwritten in the RAM by the program version from the Flash EPROM at the start of the program (commissioning phase).

### 3 Enable PLC mode

The PLC mode (e.g. Stand-alone or Master) is defined by the value of the system constant KW 00,00. The default setting for the mode is 'Stand-alone mode' (KW 00,00 = -2).

A change of the PLC mode is performed in three steps:

1. Change system constant KW 00,00 and transfer to the PLC
2. Save the PLC program in the Flash EPROM (this saves the changed system constant KW 00,00 in the Flash EPROM)

3. Activate the new PLC mode by:

- calling menu option *Enable PLC mode* or
- switching the power off and back on again or
- calling menu option *Cold start*

If the RUN/STOP switch is set to RUN, the PLC program is also started with *Enable PLC mode*.

### 4 Cold start

A cold start is performed on the PLC if you call menu option *Cold start*.

A cold start sets the PLC to its initial state (condition as delivered). All RAMs are tested and erased. If there is a program in the Flash EPROM, this program is transferred to the program RAM. The modes configured in this program are set on the PLC.

### 5 Terminal emulation

The programming and test system emulates a terminal if you select menu option *Terminal emulation*.

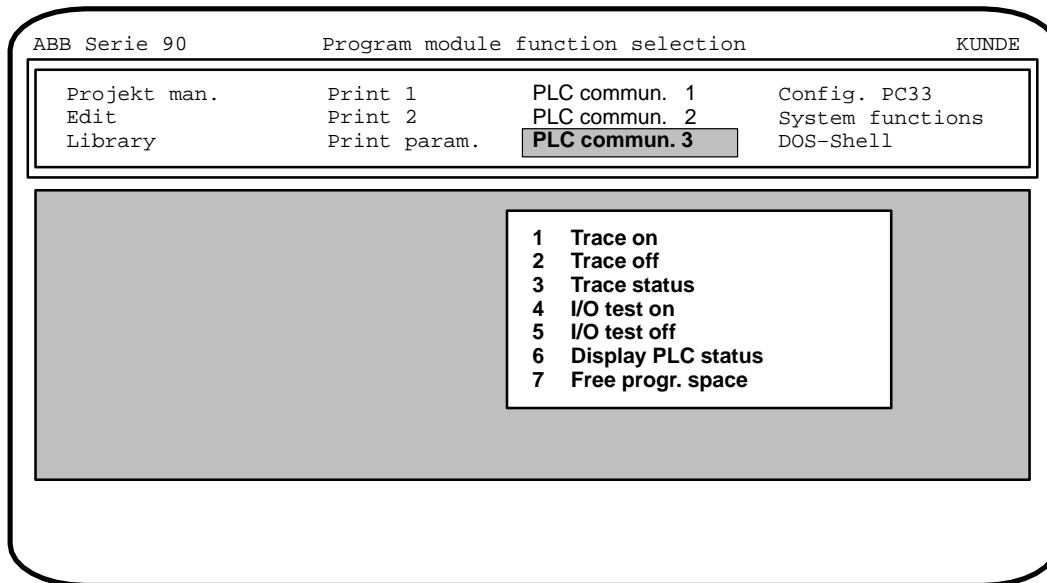
The PLC has integrated programming, test and monitor functions which provide the user with access to all PLC capabilities via a terminal. Certain PLC functions which cannot be addressed directly by the programming system can be used with the aid of *Terminal emulation*. The possible functions within terminal emulation are described in the System description Advant Controller 31, Section 7.

Quitting *Terminal-Emulation*:

You can quit terminal emulation again by pressing key <ESC>.

### 4.1.3 PLC communication 3

The programming system displays the following after calling menu option *PLC commun. 3*:



#### Trace mode

In Trace mode, the PLC notes in a trace register the user program address last executed. After a system crash, this tells you at what point in the program the crash occurred. Trace mode increases the execution time of the user program. The contents of the trace register are retained even in the following cases:

- in the case of a warm start (*Enable PLC mode*)
- after power-up, if the PLC is equipped with a back-up battery.

Not available in Series 30, 40, 50.

#### 1 Trace on

Trace mode is activated on the PLC if you select menu option *Trace on*.

#### 2 Trace off

Trace mode is deactivated on the PLC if you select menu option *Trace off*.

#### 3 Trace status

The contents of the trace register are displayed if you select menu option *Trace status*. The trace register contains the last program address executed.

#### 4 I/O test on

Operating mode I/O test is activated if you select menu option *I/O test on*.

In operating mode I/O test, the user can check the correctness of the wiring of his I/O signals from the PLC program through to the process.

The PLC program is *not* executed after start in operating mode I/O test. Only the I/O signals planned in the program are operated, i.e. the input signals are read in and the output signals are output. However, the output signals are output only if they have been set by the user, e.g. by forcing.

By actuating limit switches etc., it is possible to check whether these signals are arriving in the PLC with the declared I/O designations. By specifically *overwriting* or *forcing* outputs, you can check whether the signals issued are arriving at the correct point in the process.

The values of the I/O operands of interest in the PLC can be displayed with function *Status display* in online mode.

Command *I/O test on* can be executed even with the program running. The operating mode is then activated at the start of the next program cycle.

Not available in Series 30, 40, 50.

#### 5 I/O test off

Operating mode I/O test is deactivated by selecting menu option *I/O test off*, i.e. the user program is executed normally again as of this point. It is practical to abort the program before activating the I/O test.

## 6 Display PLC status

The entire status of the PLC is displayed as follows if you select menu option *Display PLC status*:

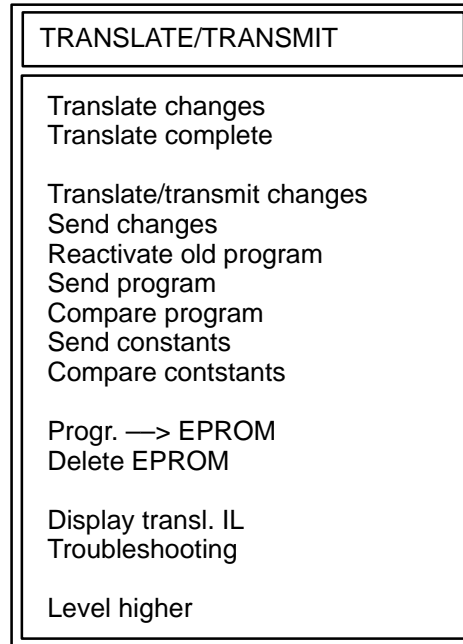
- Program identification (project name)
- Cycle time in ms
- Program status
- Active test functions
- Contents of the TRACE register
- Error messages
- Capacity utilization

## 7 Free program space

The number of program memory words still free in the PLC is calculated if you select menu option *Free progr. space*. The program memory in the PLC is checked from the end for NOPs. The number of NOPs found is then displayed.

## 4.1.4 PLC communication from FBD/LD and extended IL

Besides PLC communication directly from the menu interface, it is also possible to communicate with the PLC from the FBD/LD or the extended IL. When you press the space bar or the right mouse button, a window is pulled down containing an overview of the functions available in the FBD/LD or in the extended IL. You will see the following window after you select function *TRANSLATE/TRANSFER*:



*Note on how to carry out changes in a program:*

If changes are made in a program, they have to be translated and send before leaving the editor.

### Fast translation and transmission of the program

A fast procedure can be used for translating the FBD/LD or extended IL, namely "Translate changes". However, a precondition is that the entire program has already been translated once.

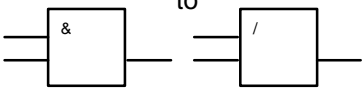
Sending the translated IL to the PLC can also be accelerated by "Send changes". However, a precondition is that the entire program has already been sent once.

With "Send changes", whole partial plans (those in which changes have been made) are sent.

In addition, it is also possible to transfer program changes or a complete program online, i.e. with the PLC running.

In order to be able to use all possibilities, a few instructions must be observed. These are listed in the table below.

## Instructions for 'Translate changes' and 'Send changes'

Type of change in FBD/LD or extended IL	Guidelines
<ul style="list-style-type: none"> <li>● Change variable e.g. E0,0 to M0,0</li> <li>● Change text constant e.g. #”Text1 to #”Text2</li> <li>● Swap CE e.g.</li> </ul>  <ul style="list-style-type: none"> <li>● Double CE connection</li> <li>● Set or delete inversion</li> </ul>	<p><i>Translate changes</i> is possible, even with changes in several segment plans. <i>Send changes</i> is possible.</p>
<ul style="list-style-type: none"> <li>● Add CE</li> <li>● Delete CE (no CEs with historical values; see CE description in Volume 7)</li> <li>● Copy block</li> <li>● Read in block with &lt;CTRL–KR&gt; (block fits in the current SP)</li> </ul>	<p><i>Translate changes</i> is possible, even with changes in several segment plans. <i>Send changes</i> is possible, as long as no segment plans were added.</p>
<ul style="list-style-type: none"> <li>● Split, move, insert or append segment plan</li> <li>● Modify sequence of segment plans</li> <li>● Copy segment plan</li> <li>● Insert block into already existing segment plan (with block function &lt;Ctrl&gt;–K–R), block does not fit in the current SP</li> <li>● Read in block in the segment plan management</li> <li>● Change a variable with &lt;CTRL–Z&gt;</li> </ul>	<p><i>Translate changes</i> is possible, even with changes in several segment plans. <i>Send changes</i> is not possible. The entire program must be transferred with <i>Send program</i>.</p> <p>Note on using &lt;CTRL&gt;–Z If you just want to look at a variable with symbol and long text but not change it, please use the key combination &lt;CTRL&gt;–P. This tells 907 PC 331 that no changes were made and also does not request the translation. The function can also be called up in online mode.</p>
<ul style="list-style-type: none"> <li>● Shift segment plan</li> <li>● Delete whole segment plan</li> </ul>	<p><i>Translate changes</i> is not necessary (the programming software notes the deleted segment plan and does not send it to the PLC). However, the entire program must be transferred with <i>Send program</i>.</p>
<ul style="list-style-type: none"> <li>● Change one or more variables in the variable editor</li> <li>● Change a text constant with &lt;CTRL&gt;–Z or in the variable editor</li> </ul>	<p><i>Translation is not necessary</i>. The entire program has to be sent.</p> <p><i>Translate changes</i> is not possible. Everything must be translated. The entire program must be transferred with <i>Send program</i>.</p>
<p><b>Note:</b> It is possible to send changes both in the 'Program aborted' state and in the 'Program running' state.</p>	

## Further notes

- *Function blocks with historical values* may only be added or omitted at the *end of the program*.
- Any changes to indirect constants are permitted. This includes also changing the cycle time. If system constants are modified, take care of Chap. 2.1, 'System constants'.

*Send changes* is not possible if

- a new program module was inserted or deleted in the case of modularized projects
- the order of the program modules was changed in the case of modularized projects.

## Translate changes

The program changes made in the FBD/LD or in the extended IL are translated if you select menu option *Translate changes*. The time taken for translation is reduced since the entire program does not need to be translated. *Translate changes* can be selected almost always. In the case of changes which require translation of the entire program, the programming system issues a corresponding message.

## Translate complete

The entire program is translated if you select menu option *Translate complete*.

## Translate and transmit changes <Alt>-1

The program changes performed in the FBD/LD or in the extended IL are translated and then transferred to the PLC if you select menu option *Translate/transmit changes*. The time taken for translation and transfer is reduced since the entire program does not need to be translated and transferred. Please refer to *Send changes* for further information.

After transfer, you are prompted to decide whether the values of the constants (K, KW, KD) are also to be transferred. If you answer Yes, this is then done.

## Send changes <Alt>-2

The program changes performed in the programming system are transferred to the PLC if you select menu option *Send changes*. The time taken for transfer is reduced because the entire program does not need to be transferred.

The Flash EPROM is automatically deleted beforehand. This prevents the program in the RAM being overwritten by the old program version from the Flash EPROM during the next program start (commissioning phase). After commissioning is completed, the PLC program should always be stored in the Flash EPROM.

This menu point can only be used if the entire program has already been transferred to the PLC once and if program changes were subsequently made in the programming system. Program changes exist when changes have been made and translated with *Translate changes* in the FBD/LD or in the extended IL.

You can use function *Send changes* if the PLC program is in

- status "RUNNING" or in
- status "ABORTED".

A distinction must be made between two cases when transferring programs to the PLC. We suggest the following procedure for this:

### Case 1:

The program in the PLC is in status "ABORTED":

- Transfer program changes to the PLC
- Transfer constants to the PLC
- Save program in Flash EPROM
- Start program

### Note:

Before starting the PLC program the system constants KW00,01 ... KW00,05 have to be set to 0 (= default) if the program has been modified in order to be on the safe side, so that all operand areas are initialized with 0 when starting the program. This makes sure that stored values of operands or historical values do not result in malfunctions. After program start the system constants KW00,01 ... KW00,05 can again be set to the values required for the application.

### Case 2:

The program in the PLC is in status "RUNNING":

- Transfer program changes to the PLC
- Transfer constants to the PLC
- Enable changes in the PLC for processing
- Save program in Flash EPROM

### Notes:

When the alterations are taken over, there is *no* PLC standstill time.

The user must exercise *extreme care* when making changes to a *running* PLC program, as the program changes immediately influence the running process without the possibility of tests. Programming errors can have grave results.

## Reactivate old program <Alt>-8

### Not available in Series 30, 40, 50

The changes made *and* enabled on a running PLC program are cancelled again by the PLC if you select menu option *Reactivate old program*. The PLC then reestablishes the program status as it existed *before* transfer of the changes.

The PLC requires approximately 1 ms to reactivate the old program status.

You can use the command if you recognize that the program changes made on the *running* PLC program are not having the required effect.

*Note:*

The above-described functionality runs only in the PLC, i.e. the program changes made and stored in the programming system are not affected by this. The user himself must ensure that the changes are cancelled again in the programming system so that the program in the PLC is the same as the program in the programming system again.

### Send programm <Alt>-3

The PLC program which exists in the form of an instruction list is transferred completely if you select menu option *Send program*.

After transfer, the system prompts you to decide whether the values of the constants (K, KW, KD) are also to be transferred. If you answer Yes, this is then done.

We must distinguish between two cases when transferring programs to the PLC. We suggest the following procedure for this:

Case 1:

The program in the PLC is in status "ABORTED":

- Transfer program to the PLC
- Transfer constants to the PLC
- Save program in Flash EPROM
- Start program

*Note:*

Before starting the PLC program the system constants KW00,01 ... KW00,05 have to be set to 0 (= default) if the program has been modified in order to be on the safe side, so that all operand areas are initialized with 0 when starting the program. This makes sure that stored values of operands or historical values do not result in malfunctions. After program start the system constants KW00,01 ... KW00,05 can again be set to the values required for the application.

Case 2:

The program in the PLC is in status "RUNNING": - Transfer program to the PLC

- Transfer constants to the PLC
- Enable changes in the PLC for processing
- Save program in flash EPROM

*Notes:*

When the alterations are taken over, there is no PLC standstill time.

The user must exercise *extreme care* when making changes to a *running* PLC program, as the program changes immediately influence the running process without the possibility of tests. Programming errors can have grave results.

### Compare program <Alt>-4

The program is read out of the PLC and compared with the current program stored in the programming system if you select menu option *Compare program*.

In the case of modularized projects, program comparisons are made on a module-related basis. The PLC program is read out as from the address at which the module starts and is compared up to the address at which the module ends.

After the program has been read from the PLC, all instructions are checked for differences. The system displays whether command, operand or operand number differ, specifying the word number.

The program comparison can be aborted after every detected difference. The number of differences found is displayed at the end.

The help program "Compare" offers a further possibility for comparing PLC projects. It is described in chapter 5.

### Send constants <Alt>-6

The indirect constants (K, KW, KD) are transferred to the PLC if you select menu option *Send constants*.

If only values of constants have changed in the programming system and the program remains unchanged, it suffices to transfer these to the PLC with *Send constants*. The program itself does not need to be transferred to the PLC again.

You can perform function *Send constants* if the program is in

- status "RUNNING"
- or in
- status "ABORTED".

### Compare constants <Alt>-7

The constants (K, KW, KD) of the PLC program are read from the PLC and compared with those in the current project of the programming system if you select menu option *Compare constants*. Differences are displayed.

### Program —> EPROM

The user program is transferred from the RAM of the PLC to the Flash EPROM if you select menu option *Progr. —> EPROM*. This thus safeguards the user program against power failure.

Each time the PLC program is started, the PLC program is first transferred automatically from the Flash EPROM to the RAM. The PLC program in the RAM is then started. Automatic transfer of the PLC program from the Flash EPROM to the RAM is performed, of course, only if there is actually a PLC program in the Flash EPROM.

### Delete EPROM

The user program in the Flash EPROM of the PLC is deleted if you select menu option *Delete EPROM*. The Flash EPROM should be erased for instance if you wish to prevent the program in the RAM being overwritten by the program version from the Flash EPROM when starting the program (commissioning phase).

## 4.2 Test

Extensive operator control and test functions are available for testing the PLC program in online mode. These operator control and test functions can be used

- in the FBD/LD
- in the extended IL
- in the variable list
- in the online list.

In addition, online information such as program status, variable status and breakpoint addresses is displayed on the monitor.

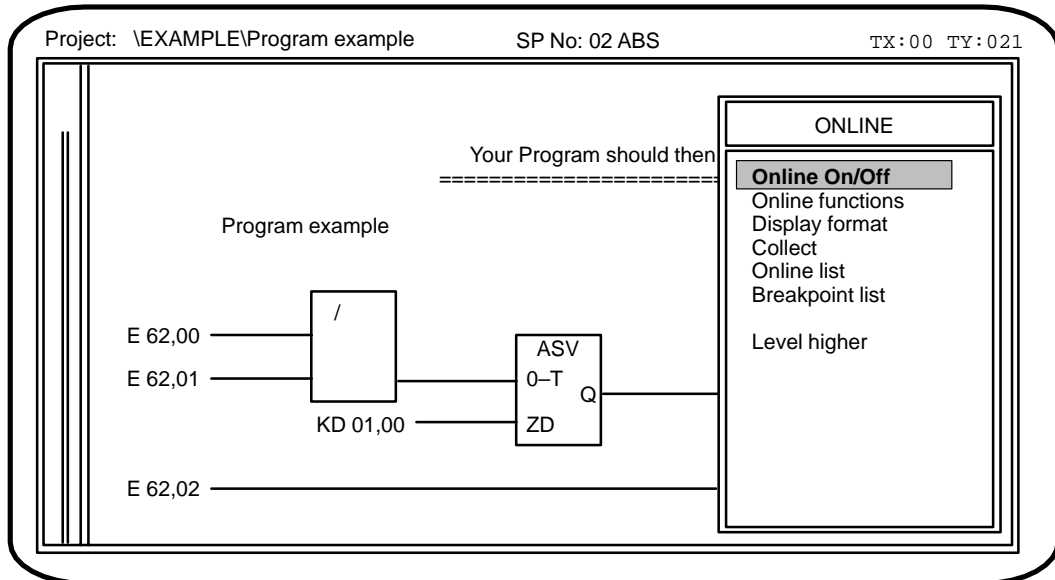
Status information is designated or abbreviated as follows:

Running	Program execution is running.
Aborted	Program execution is aborted; the outputs are set to "0".
Stopped	Program execution is stopped; the outputs retain their last value.
On breakpoint	Program execution has stopped since Single step (ES) or Single cycle (EZ) is active or since a breakpoint has been reached.
EZ	Single cycle active
ES	Single step active
On breakpoint	Breakpoint set. The breakpoint address, consisting of module number and word number, is displayed when a breakpoint is reached.
BV	Breakpoint pursuit on
FO	Forcing active
Variable sta outp	Status of selected variables on

**The following functions are not available in the Series 30, 40, 50:**

- Stop
- On breakpoint
- EZ
- ES

## Online



The online test is activated and deactivated by selecting menu option *Online On/Off*. The menu is displayed after you press the space bar or the right mouse button. You can select the required menu option with the Enter key or the left mouse button.

The online mode is possible:

**in the function block diagram/ladder diagr. FBD/LD**  
**in the extended instruction list (extended IL)**  
**in the variable list**  
**in the online list**

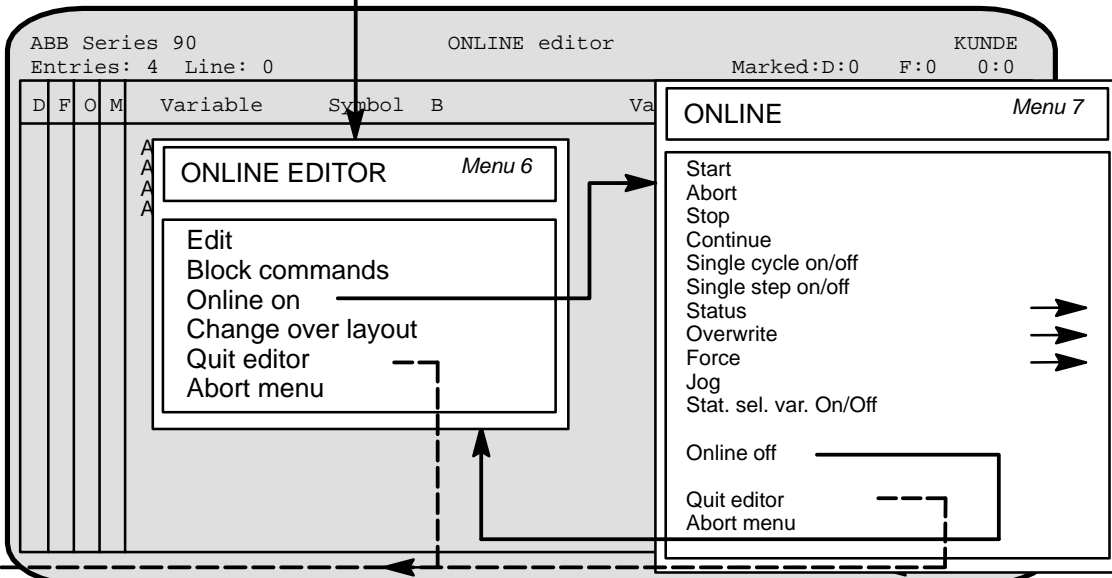
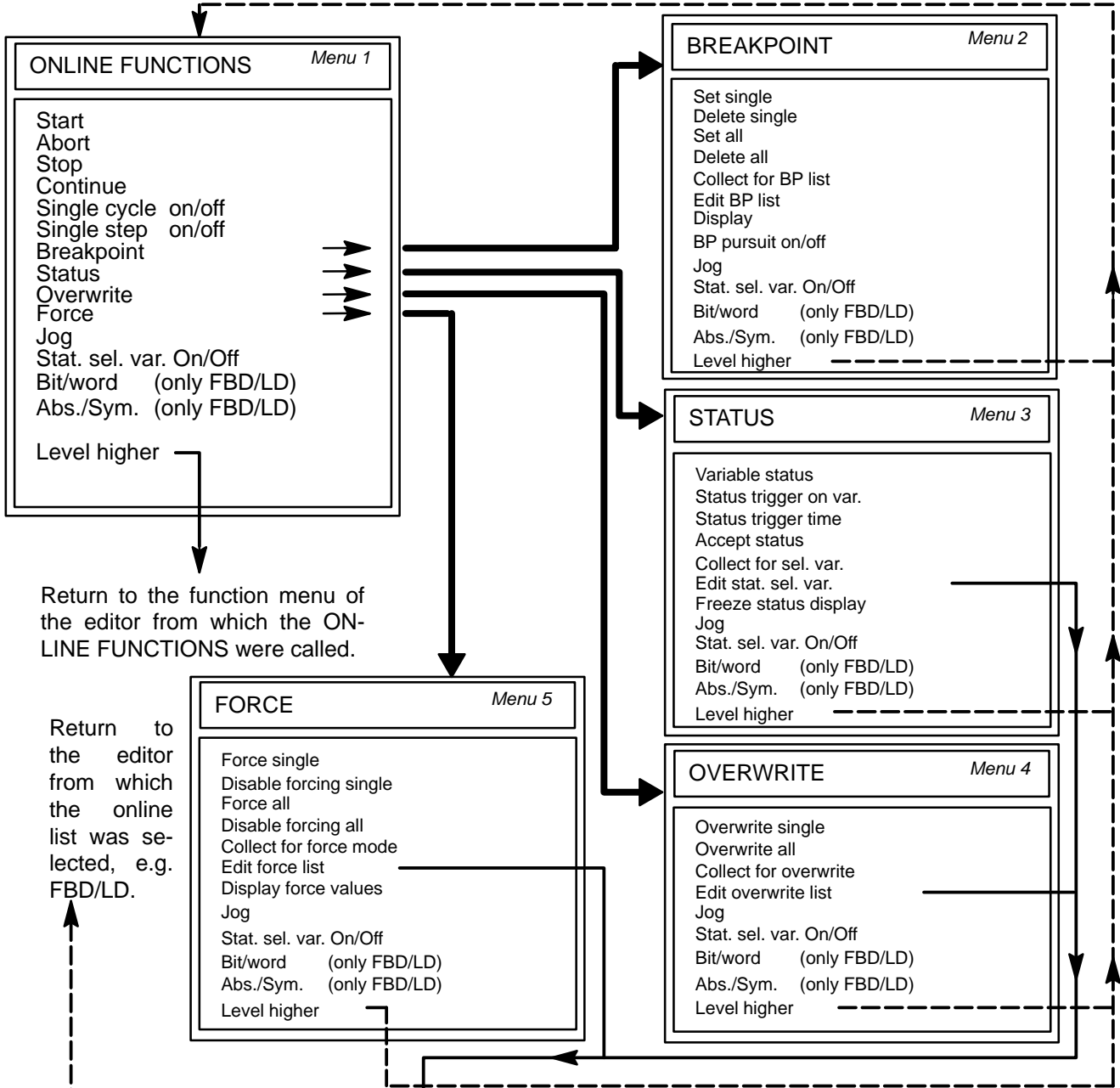
### Notes:

Display of variable states when the program is aborted:

**Inputs:** As the central units use the process image, the last states of the inputs retain in the input image. If therefore the program is aborted and after that, the states of the inputs change, then these states cannot be displayed as the process image is not updated.

**Outputs:** When the program is aborted, the *real*/outputs are set to 0. However the outputs in the process image still have their old values. That means that the 'old' states of the outputs are displayed in 907 PC 331. Only after a new start of the PLC program the process image will be updated.

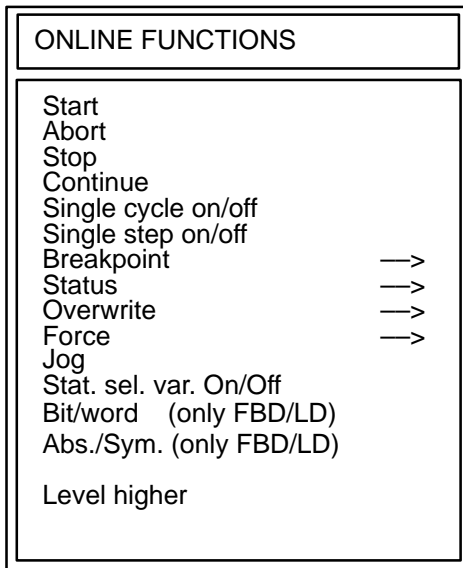
# Menu overview Online functions



## ONLINE functions

It is possible to call the required online functions by selecting the corresponding menu option during online test.

### Menu *ONLINE FUNCTIONS* (Menu 1)



#### Start program, <Alt>–S

The PLC program is started if you select menu option *Start* or if you press hotkey <Alt>–S.

You can start the program only if the program is in status *Aborted* and the RUN/STOP switch is set to position RUN.

Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

#### Abort program, <Alt>–A

The PLC program is aborted if you select menu option *Abort* or if you press hotkey <Alt>–A. Execution of the user program is aborted at the end of the program. The related process outputs are set to the value 0.

You can abort the program only if the program is in status *Running* or *Stopped*.

Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

#### Stop program, <Alt>–H

The PLC program is stopped if you select menu option *Stop* or if you press hotkey <Alt>–H. Execution of the user program is stopped at the end of the program. The flags and the process outputs retain their values.

You can stop the program only if the program is in status *Running*.

Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

**Not available in Series 30, 40, 50.**

#### Continue program, <Alt>–C

The PLC program is continued if you select menu option *Continue* or if you enter command <Alt>–C.

You can continue the program only if the program is in status *Stopped*.

Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

**Not available in Series 30, 40, 50.**

#### Single cycle On/Off, Z

Operating mode Single cycle is activated or deactivated if you select menu option *Single cycle* or if you press hotkey Z.

Execution of the PLC program is interrupted at the end of the program in operating mode Single cycle. The flags and process outputs retain their values.

Advancing by one program cycle: “Jog”.

A further program cycle is executed in each case with function “Jog”. Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

**Not available in Series 30, 40, 50.**

#### Single step On/Off, S

Operating mode Single step is activated or deactivated in the PLC program if you select menu option *Single step* or if you press hotkey S.

Execution of the PLC program is interrupted after each program step in operating mode Single step.

A program step extends in each case:

- through to the end of the next function block,
- through to the next assignment,
- through to the next “right bracket”.

The flags and process outputs retain the values assigned to them in the PLC program.

Advancing by one program step: “Jog”.

A further program step is executed in each case with function “Jog”.

**Not available in Series 30, 40, 50.**

Timers which have been started continue to run independently of the status of the user program in the operating system of the PLC. They are aborted only by a warm start or a cold start.

Not available in 07 KR 31.

## Breakpoint

Execution of the user program can be interrupted specifically at the program points at which breakpoints have been set in operating mode Breakpoint. The following sub-menu is called if you select menu option *Breakpoint*:

**Not available in Series 30, 40, 50.**

### Menu *BREAKPOINT* (Menu 2)

BREAKPOINT
Set single
Delete single
Set all
Delete all
Collect for BP list
Edit BP list
Display
BP pursuit on/off
Jog
Stat. sel. var. On/Off
Bit/word (only FBD/LD)
Abs./Sym. (only FBD/LD)
Level higher

### Set breakpoint, single, B

The cursor must first be on the appropriate line in the extended IL or must be positioned on the corresponding variable field in the FUPKOP where the breakpoint should be set.

Selecting the *Set single* menu point or the hot key B allows the breakpoint to be set.

An input window from the breakpoint list appears:

B	M	Mod	Addr	Comment
*		001	0005	

When you press <Return>, this confirms the display, the breakpoint is activated immediately in the PLC and is adopted as marked in the breakpoint list.

The entry is rejected if you press <ESC>. No breakpoint is activated.

Breakpoints can be set to the word numbers of the following elements:

- to the word No. of the last operand of a function block
- to the word No. of the operand after an assignment symbol
- to the word No. of a "right bracket"
- to the word No. of the end-of-program identifier.

Breakpoints can also be entered with the PLC program running.

In the case of *modularized* projects, the word number displayed in the window always relates to the *start* of the relevant *module*.

A maximum of 15 breakpoints are permitted simultaneously.

### Delete breakpoint single, <Alt>-B

The set breakpoint can be deleted singly, i.e. deactivated in the PLC if you select menu option *Delete single* or if you press hotkey <Alt>-B. You must move the cursor to the corresponding variable or to the corresponding line before selecting *Delete single* or before pressing <Alt>-B. You will then see the input window displayed beforehand. Confirm the display with <Return>, after which the breakpoint is deactivated immediately in the PLC. The marking of the breakpoint in the breakpoint list is cancelled.

The entry is rejected if you press <ESC>. The breakpoint remains set.

### Set breakpoints all, W

The breakpoints entered in the breakpoint list with *Edit BP list* are activated in the PLC if you select menu option *Set all* or if you press hotkey W.

Breakpoints can be set to the word numbers of the following elements:

- to the word No. of the last operand of a function block
- to the word No. of the operand after an assignment symbol
- to the word No. of a "right bracket"
- to the word No. of the end-of-program identifier.

Breakpoints can also be entered with the PLC program running.

Advancing to the next breakpoint: "Jog".

Program execution is enabled until the next breakpoint is reached with function "Jog".

A maximum of 15 breakpoints are permitted simultaneously.

### Delete breakpoints all, <Alt>-W

All set breakpoints can be cancelled at once, i.e. deactivated in the PLC, if you select menu option *Delete all* or if you press hotkey <Alt>-W. All markings are removed from the breakpoint list.

### Collect for breakpoint list, 2

Program addresses can be entered in the breakpoint list if you select menu option *Collect for BP list* or if you press

hotkey 2. You must move the cursor to the corresponding variable in the FBD or to the corresponding line of the extended IL before execution of the function. The same rules apply as when setting individual breakpoints. Collection for the breakpoint list is not possible in the ladder diagram.

### Edit breakpoint list, L

You call the breakpoint list by selecting menu option *Edit BP list*.

B	M	Mod	Addr	Comment
*		001	0001	

You can then enter a list of breakpoint addresses, specifying the relevant module number, and these must then be marked in column B with <F5>. The marking can be cancelled again with <F6>.

In the case of *modularized* projects, the word number always refers to the *start* of the relevant *module*.

### Display breakpoint, Q

Selecting the menu point Display or the hot key Q reads the set breakpoints into the breakpoint list. They can be displayed with the function Enter BP list (enter breakpoint list, hot key L). The read-in breakpoints are marked.

In the case of modularized projects, the word number always refers to the start of the relevant module.

### Breakpoint pursuit On/Off, P

The relevant, active breakpoint can be detected in the PLC and displayed in the FBD/LD (orientation column or variable field), in the extended IL (corresponding line) or in the translated IL (corresponding line) if you select menu option *BP pursuit on/off* or if you press hotkey P.

If the active breakpoint is not in the current module in the case of modularized projects, the current module is quit and the corresponding module is automatically called up. Any entered changes are taken into account by an enquiry.

### Jog, G

This command is operable only in modes "Single step" or "Single cycle" or if breakpoints have been set.

Selecting menu option *Jog* or pressing hotkey G means that

- the program executes the next program step in Single step mode,
- the program is run through once in Single cycle mode,
- the program is executed through to the next breakpoint if breakpoints are set.

### Status of selected variables On/Off, A

Only the status values of the variables *selected* by the user are updated on the monitor if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A.

The variables are selected by entering them in the online list and marking them in column D. The statuses of all other variables are no longer updated.

This operating mode is deactivated again if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A again, and the statuses of *all* variables are updated on the monitor.

### Bit/word (in the FBD/LD only), X

The status display for word variables is switched over between the two display modes Binary and Word value if you select menu option *Bit/word (only FBD/LD)* or if you press hotkey X.

Binary display mode:

- The variable name is highlighted in the FBD/LD if the value is not equal to 0

Word value display mode:

- The value of the word variable is displayed in place of the variable name in the FBD/LD

### Absolute/symbolic (in the FBD/LD only), <F9>

The status display is switched over between the two display modes Absolute and Symbolic if you select menu option *Abs./sym. (only FBD/LD)* or if you press key <F9>. This means that *either* the *absolute* or the *symbolic* designations are shown for the variables.

### Level higher

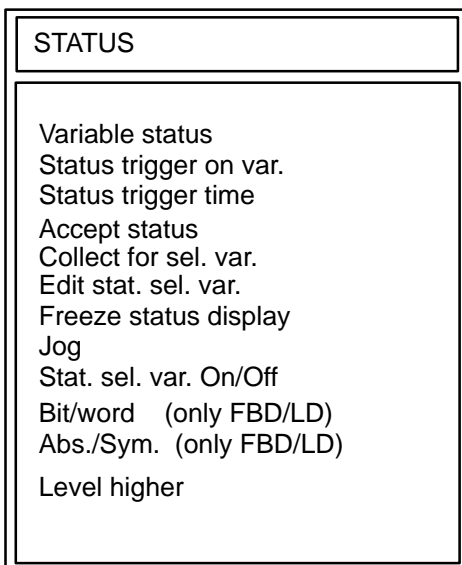
You return to menu ONLINE FUNCTIONS if you select menu option *Level higher*.

### Status

The programming system makes available various options for status display of variables. The variables are always displayed cycle-consistently, i.e. the variable values supplied by the PLC originate from the same program cycle.

The following sub-menu is called if you select menu option *Status*:

### Menu STATUS (Menu 3)



### Variable status, V

The status of the variables is displayed cyclically if you select menu option *Variable status* or if you press hotkey V. Display modes set beforehand such as Status trigger time for example are thus deactivated again.

The PLC makes available the variable statuses as follows:

- Always at the end of the program
- At the relevant breakpoint if operating mode Breakpoint or Single step is active

### Status trigger to variable, T

The programming system awaits entry of a *trigger variable* and the required *trigger condition* if you select menu option *Status trigger on var.* or press hotkey T. The trigger variable must be entered as an absolute variable.

The variable statuses are only ever updated on the monitor if the trigger condition is fulfilled. The following trigger conditions are possible:

In the case of bit variables:

status: " 0 "  
 status: " 1 "  
 rising edge: " / "  
 falling edge: " \ "

In the case of word variables:

greater than: " > value "  
 less than: " < value "  
 equal to: " = value "

### Status trigger time, <Alt>-M

The status of the variables is polled by the PLC at given time intervals and the display is updated on the monitor if you select menu option *Status trigger time* or if you press hotkey <Alt>-M.

The user must enter the polling interval in *milliseconds*.

### Accept status, 6

A "snapshot" of the variable statuses displayed in ONLINE mode can be transferred to the online list if you select menu option *Accept status* or if you press hotkey 6. This information can then be further-processed subsequently, e.g. stored on hard disk.

The variable statuses can also be stored with the block functions.

### Collect for selected variables, 5

Variables can be collected for function “Display selected variables” (*Stat. sel. var. On/Off*) if you select menu option *Collect for sel. var.* or if you press hotkey 5. You must move the cursor to the corresponding variable in the FBD or to the corresponding line in the extended IL before execution of the function. The corresponding variables are entered in the online list and marked in column D. Only the selected variables are then updated with hotkey A (*Stat. sel. var. On/Off*).

### Edit status of selected variables, O (Edit online list)

The editor for processing the online list is called if you select menu option *Edit stat. sel. var.* or if you press hotkey O.

The programming system displays the following:

ABB Series 90				ONLINE editor			KUNDE		
Entries: 2				Line: 0			Marked: D:1 F:0 O:0		
D	F	O	M	Variable	Symbol	B	Value	Long text	
*				A 62,00		B	0		
				A 62,01		B	0		

Any selection of variables can be edited in the online list.

Menu *ONLINE-EDITOR* can be called within the editor for processing the online list.

The online list is required in order to:

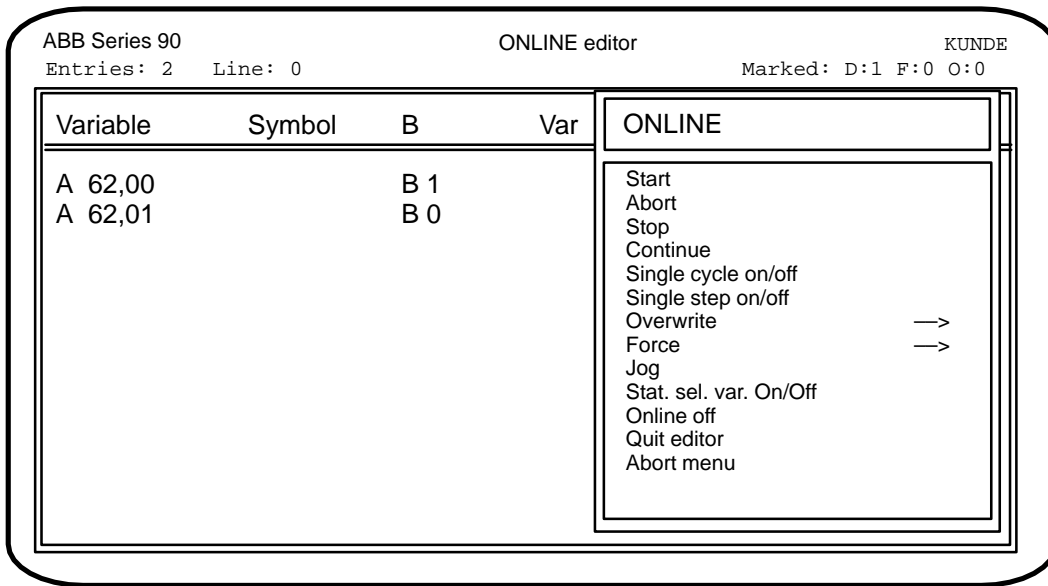
- display status values of variables in the online list;
- display status values of selected variables in the program (FBD/LD, extended IL; column D).  
You must first move the cursor to column D. Mark the required variable with <F5>. Only the status values of the *marked* variables are then displayed in the program. You can cancel the marking again with <F6>;
- overwrite status values of variables in the PLC (column O);
- force status values of inputs and outputs in the PLC (column F);
- store status values of variables in the online list (column M)  
Set the marking with <F5>. Delete the marking with <F6>; please refer to Block processing for further operations.

### Menu *ONLINE-EDITOR* (Menu 6)

ONLINE EDITOR
Edit Block commands  Online On  Change over layout  Quit editor Abort menu

## Online On

The programming system displays the following if you select menu option *Online On* and then press the space bar:



The programming system now communicates with the PLC. The status values of the variables are displayed and the online functions are available.

## Menu STATUS, continued (Menu 3)

### Freeze status display, 0 (Zero)

The current variable statuses are frozen on the monitor when this menu is called, and the user can decide whether he wishes to also transfer this status to the online list with menu *Accept status*.

### Jog, G

This command is operative only in modes "Single step", "Single cycle" or if breakpoints have been set. Selecting menu option *Jog* or pressing hotkey G means that

- the program executes the next program step in Single-step mode,
- the program is run through once in Single-cycle mode,
- the program is executed as far as the next breakpoint if breakpoints have been set.

### Status of selected variables On/Off, A

Only the status values of the variables *selected* by the user are updated on the monitor if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A. The variables are selected by entering them in the online list and marking them in column D. The statuses of all other variables are no longer updated.

This operating mode is deactivated again and the statuses of all variables are updated on the monitor by selecting menu option *Stat. sel. var. On/Off* again or by pressing hotkey A again.

### Bit/word (in the FBD/LD only), X

The status display for word variables is switched over between the display modes Binary and Word value if you select menu option *Bit/word (only FBD/LD)* or if you press hotkey X.

Binary display mode:

The variable name is highlighted in the FBD/LD if the value is *not equal to 0*

Word value display mode:

The value of the word variable is displayed in place of the variable name in the FBD/LD

### Absolute/symbolic (in the FBD/LD only), <F9>

The status display is switched over between the two display modes Absolute and Symbolic, i.e. either the absolute or the symbolic designations are displayed for the variables, if you select menu option *Abs./sym. (only FBD/LD)* or if you press key <F9>.

### Level higher

You return to menu ONLINE FUNCTIONS if you select menu option *Level higher*.

## Overwrite

The Overwrite function permits you to overwrite variables with assignable values in the PLC. These values are retained until new values are assigned again to these variables by the PLC program, the process or the user. It is possible to overwrite both with the program running and with the program aborted.

Overwrite is possible for the following operands:

- Flags: M, MW, MD
- Inputs: E, EW
- Outputs: A, AW
- Indirect constants: K, KW, KD
- Steps: S  
(only Set possible; Reset is possible, by definition only by setting another step)

Overwrite procedure:

Several variables simultaneously:

- Enter the name of the variable to be overwritten and the required value(s) in the online list with menu option *Edit overwrite list*;
- Variables are overwritten in the PLC with menu option *Overwrite all* or by pressing hotkey R. This applies to all variables identified accordingly in the online list.

Overwrite single variables:

- A single variable is overwritten in the PLC with menu option *Overwrite single* or by pressing hotkey U.

When does overwriting occur?

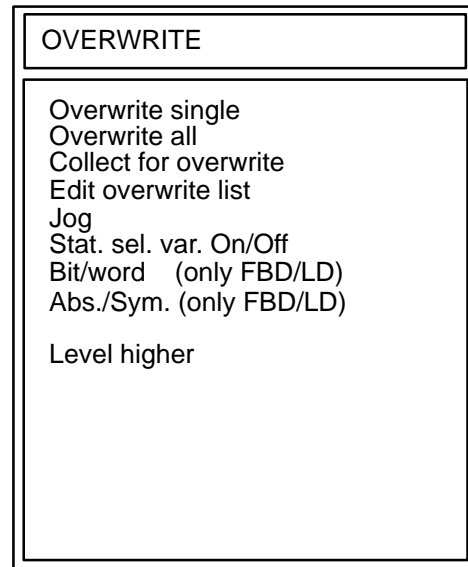
Program execution running::

- Overwriting occurs at the end of the program; it is active in the next program cycle.

Program execution stopped:

- e.g. in the case of breakpoint, Single step, Single cycle; overwriting occurs immediately and is active immediately.

## Menu *OVERWRITE* (Menu 4)



### Overwrite single, U

Selecting the menu point *Overwrite single* or hot key U overwrites an individual variable in the PLC.

The cursor must first be at the appropriate line in the extended IL or must be positioned on the appropriate variable in the FBD/LD.

The following input window appears when the menu point *Overwrite single* is selected or when hot key U is pressed:

D	F	O	M	Variable	Symbol	B	Value	Long
		*		A 62,00			B 0	

You confirm the display with <Return> and the overwrite value is then activated immediately in the PLC. Use <ESC> to reject the setting. No overwrite value is activated.

Note:

If indirect constants are to be overwritten, the constant value must be entered in column "Value" (not "Long text").

### Overwrite all, R

It is possible to overwrite variables with assignable values in the PLC if you select menu option *Overwrite all* or if you press hotkey R. These values are retained until other values are once again assigned to these variables by the PLC program.

The precondition for calling menu option *Overwrite all* or for pressing hotkey R is that the variables to be overwritten and the related values have been entered in the online list and marked in column O.

### Edit overwrite list, O (Edit online list)

You can call the editor for processing the online list by selecting menu option *Edit overwrite list* or by pressing hotkey O.

The programming system displays the following:

ABB Series 90				ONLINE editor				KUNDE	
Entries: 2		Line: 1		Marked: D:0 F:0 O:1					
D	F	O	M	Variable	Symbol	B	Value	Long text	
		*		A 62,00		B	0		
				A 62,01		B	0		

Any selection of variables can be entered in the online list.

The online list is required for:

- displaying status values of variables in the online list;
- displaying status values of selected variables in the program (FBD/LD, extended IL) (column D);
- overwriting status values of variables in the PLC (column D).  
You must first move the cursor to column O for this purpose. Mark the required variable with <F5>. Only the status values of the marked variables are then overwritten in the program. You can cancel the marking again with <F6>;
- forcing status values of inputs and outputs in the PLC (column F);
- storing status values of variables in the online list (column M);  
Set the marking with <F5>. Delete the marking with <F6>; please refer to Block processing for further operations.

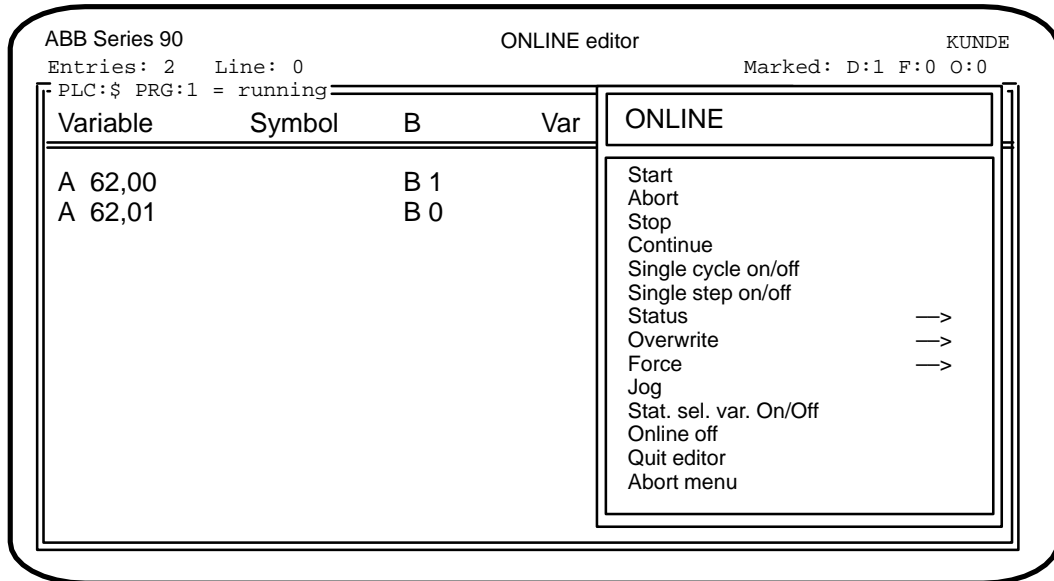
The menu ONLINE EDITOR can be called up via the <Space bar> inside the editor in order to process the online list.

### Menu *ONLINE EDITOR* (Menu 6)

ONLINE EDITOR
Edit
Block commands
Online On
Change over layout
Quit editor
Abort menu

## Online On

When the menu point *Online On* is selected, the programming system starts communicating with the PLC. The status values of the variables are displayed and the online functions are available. If the *<Space bar>* is also pressed, the following image appears.



### Menu *OVERWRITE*, continued (*Menu 4*)

#### Collect for overwrite, 4

Selecting the menu point *Collect for overwrite* or pressing hot key 4 enables variables to be collected for overwriting.

The cursor must be positioned on the appropriate variable field in the FBD or must be positioned on the appropriate line in the extended IL before the function is carried out.

The appropriate variables are entered in the online list and are marked in column O. The variables entered in the online list and marked in column O are overwritten with the hot key R (*Overwrite all*).

#### Jog, G

This command is operative only in modes "Single step", "Single cycle" or if breakpoints have been set.

Selecting menu option *Jog* or pressing hotkey G means that

- the program executes the next program step in Single-step mode,
- the program is run through once in Single-cycle mode,
- the program is executed through to the next breakpoint if breakpoints have been set.

#### Status of selected variables On/Off, A

Only the status values of the variables *selected* by the user are updated on the monitor if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A. The variables are selected by entering them in the online list and marking them in column D. The statuses of all other variables are no longer updated.

This operating mode is deactivated again and the statuses of all variables are updated on the monitor by selecting menu option *Stat. sel. var. On/Off* again or if you press hotkey A again.

#### Bit/word (in the FBD/LD only), X

The status display for word variables is switched over between the display modes Binary and Word value if you select menu option *Bit/word (only FBD/LD)* or if you press hotkey X.

Binary display mode:

- The variable name is highlighted in the FBD/LD if the value is *not equal to 0*

Word value display mode:

- The value of the word variable is displayed in place of the variable name in the FBD/LD.

### Absolute/symbolic (in the FBD/LD only), <F9>

The status display is switched over between the two display modes Absolute and Symbolic, i.e. either the *absolute* or the *symbolic* designations are displayed for the variables if you select menu option *Abs./sym.* (only FBD/LD) or if you press key <F9>.

### Level higher

You return to menu ONLINE FUNCTIONS if you select menu option *Level higher*.

### Force

Forcing means that the user can preset fixed values for inputs and outputs. These values remain operative until they are changed again by the user or until forcing is cancelled again.

This capability of the PLC may be helpful when commissioning if all real process inputs and outputs are not yet available.

The FORCE values are stored power-fail-safe in the PLC if a battery is fitted and can be entered either with the program running or in any other program status.

Maximum number of variables which can be forced:

- 64 binary inputs (E)
- 16 word inputs (EW)
- 64 binary outputs (A)
- 16 word outputs (AW)

### for Series 30, 40, 50:

- 31 binary inputs/outputs
- 7 word inputs/outputs

Force procedure:

- Enter the inputs and outputs to be forced and the required values in the online list and mark them in column F with <F5>.
- The inputs and outputs can then be forced in the PLC with menu option *Force all* or by pressing hotkey E for all variables identified accordingly in the online list.
- Individual inputs or outputs can be forced in the PLC with menu option *Force single* or by pressing hotkey F for single inputs/outputs.

When does forcing occur?

Program execution running:

- Forcing occurs at the end of the program. It is active in the next program cycle.

Program execution stopped:

- e.g. in the case of breakpoint, Single-step, Single-cycle; forcing occurs immediately and is activated immediately.

### Menu FORCE (Menu 5)

FORCE
Force single
Disable forcing single
Force all
Disable forcing all
Collect for force mode
Edit force list
Display force values
Jog
Stat. sel. var. On/Off
Bit/word (only FBD/LD)
Abs./Sym. (only FBD/LD)
Level higher

### Force single, F

It is possible to force an input or output in the PLC if you select menu option *Force single* or if you press hotkey F. Forcing means that the user can set the inputs and outputs to selectable values. These values are retained until they are changed again by the user or until forcing is cancelled again with hotkey <Alt>-F.

First, the cursor must be positioned on the corresponding line in the extended IL or at the appropriate variable to be forced in the FBD/LD.

The following input window is displayed if you select menu option *Force single* or if you press hotkey F:

D	F	O	M	Variable	Symbol	B	Value	Long
	*			A 62,00		B	0	

You can confirm the display by pressing <Return>, and the force value is activated immediately in the PLC. The preset value is rejected with <ESC>. No force value is activated.

### Disable forcing single, <Alt>-F

It is possible to cancel forcing for single outputs/inputs if you select menu option *Disable forcing single* or if you press hotkey <Alt>-F.

You must have set the cursor to the corresponding variable before selecting the function.

### Force all, E

It is possible to force inputs and outputs in the PLC if you select menu option *Force all* or if you press hotkey E. Forcing means that the user can set the inputs and outputs to selectable values. These values are retained until they are changed again by the user or until forcing is cancelled again by pressing hotkey <Alt>-E.

The precondition for calling menu option *Force all* or for pressing hotkey E is that the inputs or outputs to be forced and the related force values have been entered in the online list and marked with in column F with <F5>.

Note:  
See menu "FORCE" for the precise procedure.

### Disable forcing all, <Alt>–E

Forcing is cancelled for all forced inputs and outputs if you select menu option *Disable forcing all* or if you press

hotkey <Alt>–E.  
The markings in the online list are cancelled.

### Collect for force mode, 3

Variables can be collected for forcing if you select menu option *Collect for force mode* or if you press hotkey 3. Before the function is carried out, the cursor must be positioned on the appropriate variable field in the FBD or on the appropriate line in the extended IL.

The corresponding variables are entered in the online list and marked in column F. The variables entered in the online list and marked in column F are transferred to the PLC and forced if you press hotkey E (*Force all*).

### Edit force list, O (Edit online list)

You can call the editor for processing the online list by selecting menu option *Edit force list* or if you press hotkey O.

The programming system displays the following:

ABB Series 90				ONLINE editor				KUNDE	
Entries: 2		Line: 1		Marked: D:0 F:1 O:0					
D	F	O	M	Variable	Symbol	B	Value	Long text	
	*			A 62,00		B 0			
				A 62,01		B 0			

Any selection of variables can be entered in the online list.

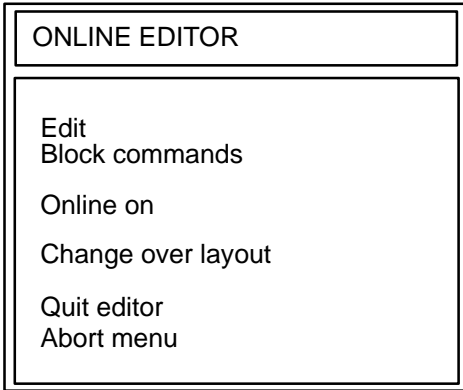
The online list is required for:

- displaying status values of variables in the online list;
- displaying status values of selected variables in the program (FBD/LD, extended IL; column D);
- overwriting status values of variables in the PLC (column O);

- forcing status values of inputs and outputs in the PLC (column F).  
To do this, the cursor is first brought into column F. The desired variable is marked with <F5>. Only the marked variables are then forced in the program. The marking can be deleted again with <F6>.
- storing status values of variables in the online list (column M).  
Set the marking with <F5>. Delete the marking with <F6>; please refer to Block processing for further operations.

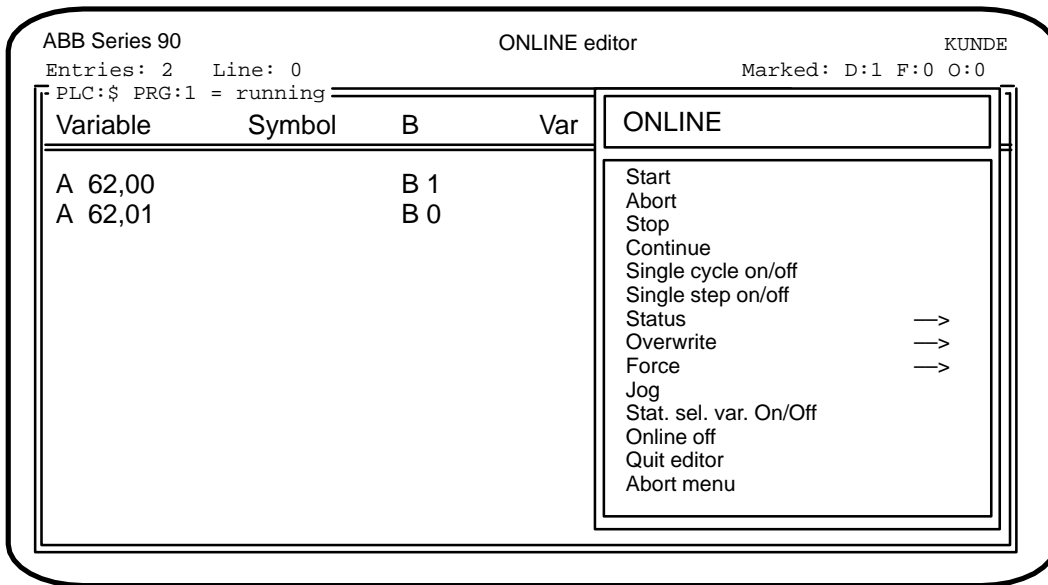
You can call menu ONLINE EDITOR within the editor for processing the online list.

### Menu *ONLINE EDITOR* (Menu 6)



### Online On

When the menu point *Online On* is selected, the programming system starts to communicate with the PLC. The status values of the variables are displayed and the online functions are available. The following image appears if the *<Space bar>* is also pressed.



### Display force values, <Alt>–D

The inputs and outputs forced in the PLC and the related force values are read out from the PLC and stored in the online list if you select menu option *Display force values*. The information on the forced inputs and outputs is available if you select menu option *Edit force list* or if you press hotkey O (in online mode). The forced inputs and outputs read in are marked as described above.

### Jog, G

This command is operative only in modes “Single–step”, “Single–cycle” or if breakpoints have been set.

Selecting menu option *Jog* or pressing hotkey G means that

- the program executes the next program step in Single–step mode,
- the program is run through once in Single–cycle mode,
- the program is executed through to the next breakpoint if breakpoints have been set.

### Status of selected variables On/Off, A

Only the status values of the variables *selected* by the user are updated on the monitor if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A. The variables are selected by entering them in the online list and marking them in column D. The statuses of all other variables are no longer updated. This operating mode can be deactivated again and the statuses of all variables are updated on the monitor if you select menu option *Stat. sel. var. On/Off* again or if you press hotkey A again.

### Bit/word (in the FBD/LD only), X

The status display for word variables is switched over between the two display modes Binary and Word value if you select menu option *Bit/word (only FBD/LD)* or if you press hotkey X.

Binary display mode:

- The variable name is highlighted in the FBD/LD if the value is *not equal to 0*.

Word value display mode:

- The value of the word variable is displayed in place of the variable name in the FBD/LD.

### Absolute/symbolic (in the FBD/LD only), <F9>

The status display is switched over between the two display modes Absolute and Symbolic, i.e. either the absolute or the symbolic designations are displayed for the variables, if you select menu option *Abs./sym. (only FBD/LD)* or if you press key <F9>.

### Level higher

You return to menu ONLINE FUNCTIONS if you select menu option *Level higher*.

### Menu ONLINE FUNCTIONS continued (Menu 1)

### Jog, G

This command is operative only in modes “Single–step”, “Single–cycle” or if breakpoints have been set.

Selecting menu option *Jog* or pressing hotkey G means that

- the program executes the next program step in Single–step mode,
- the program is run through once in Single–cycle mode,
- the program is executed through to the next breakpoint if breakpoints have been set.

### Status of selected variables On/Off, A

Only the status values of the variables *selected* by the user are updated on the monitor if you select menu option *Stat. sel. var. On/Off* or if you press hotkey A. The variables are selected by entering them in the online list and marking them in column D. The statuses of all other variables are no longer updated.

This operating mode can be deactivated again and the statuses of all variables are updated on the monitor if you select menu option *Stat. sel. var. On/Off* again or if you press hotkey A again.

### Bit/word (in the FBD/LD only), X

The status display for word variables is switched over between the two display modes Binary and Word value if you select menu option *Bit/word (only FBD/LD)* or if you press hotkey X.

Binary display mode:

- The variable name is highlighted in the FBD/LD if the value is *not equal to 0*.

Word value display mode:

- The value of the word variable is displayed in place of the variable name in the FBD/LD.

**Absolute/symbolic (in the FBD/LD only), <F9>**

The status display is switched over between the two display modes Absolute and Symbolic, i.e. either the absolute or the symbolic designations are displayed for the variables, if you select menu option *Abs./sym.* (*only FBD/LD*) or if you press key <F9>.

**Level higher**

You return to the function menu of the editor from which the ONLINE FUNCTIONS have been called if you select menu option *Level higher*.

# 5 "Compare" program

## Method of operation of COMPARE

COMPARE is a help program for comparing two IL files. It compares two print files record by record as from a particular address through to the end of the shortest file. It reads only the instructions of the instruction lists printed into the files. The header, footer, comments and symbols are ignored. As a comparison result, all records which do not correspond with each other are displayed.

If the two compared records are of a different length, both are displayed. The comparison is then continued in the next record. If the only difference between the two files is that records have been deleted or added, COMPARE displays the rest of the records as being different as from the point at which the records differ. In this case, COMPARE offers the possibility of starting the comparison of the files as from certain record numbers, in every file from a different record number. Different comparison types are possible by defining parameters.

## Before using COMPARE

If you wish to compare two PLC programs with each other, you must print out the IL list of every program. This is possible via the main menu point *Print 1* → submenu *IL*. For *Print target*, simply state a file name instead of *PRN*.

Note:

Comments must not be within records of the IL being printed.

The left edge of the print file must be at 10 (main menu point *Print Param.* → Sub-menu *Print parameter* → *Left edge: 10*).

## Using COMPARE

To use COMPARE, you must use the following syntax:

COMPARE *file1 file2 output /parameter*

*File1* and *File2* are the names of the two files to be compared.

You can display or file the comparison result in different ways with *Output*:

- If you omit *Output*, the comparison result is displayed on the screen.
- If you state *PRN*, the comparison result is printed out on the printer.
- If you give a *File name*, the comparison result is stored in this file.

By stating different *Parameters*, you can control the type of the comparison and the output. Several parameters are simply given together, e.g. */DN*.

The following parameters are available:

- /L* Ignore record length  
This parameter allows the comparison of records of different lengths from the start of the record through to the end of the shorter record or until the first difference.
- /S* Ignore record number  
This parameter allows the comparison of records if they have different record numbers.
- /W* Ignore word number  
With this parameter, the word numbers are ignored during the comparison.
- /C* Ignore commands  
With this parameter, the commands within the words are ignored during the comparison. In this way, for example, the difference between "&" and "&N" is ignored.
- /T* Ignore operand type  
With this parameter, the operand types are ignored during the comparison.
- /A* Ignore operand address  
With this parameter, the operand addresses are ignored during the comparison.
- /D* State only the number of different records  
With this parameter, only the number of different records is given.
- /N* Screen display without interruption  
Normally, the display of the comparison result is stopped after every full screen page and is not continued again until after a key is pressed. This parameter switches off this interruption.
- /Qn* Abort after n differences  
This parameter aborts the display after n differences.

`/1#xxxxx` Start comparison in file 1 as from record number xxxxx  
With this parameter, the comparison in file 1 is not started until as from record number xxxxx. The record number must always be given as 5 digits (e.g. `/1#01013`).

If this record number does not exist, an error message is issued. If different start addresses are given for both files, the parameters `/S` (ignore record number) and `/W` (ignore word number) are switched on automatically.

`/2#xxxxx` Start comparison in file 2 as from record number xxxxx  
With this parameter, the comparison in file 2 is not started until as from record number xxxxx. The record number must always be given as 5 digits (e.g. `/2#01013`).

If this record number does not exist, an error message is issued. If different start addresses are given for both files, the parameters `/S` (ignore record number) and `/W` (ignore word number) are switched on automatically.

After the comparison, the number of differences found is stated. If no differences were found, the files are identical (apart from differences in the case of ignored elements). The comparison can also be aborted by pressing `<Esc>`.

### Examples

```
COMPARE IL1.PRT IL2.PRT
```

These two files are compared and the comparison result is shown on the screen.

```
COMPARE IL1.PRT IL2.PRT COMPARE.DAT
```

These two files are compared and the comparison result is written into the file `COMPARE.DAT`.

```
COMPARE IL1.PRT IL2.PRT PRN /1#00002T
```

These two files are compared. The comparison does not start until as from record number 2 in `IL1.PRT`. Also, the operand types are ignored. The comparison result is output on the printer.

## Variable list

In the user program, 5 different variable types are used:

- ⇒ The physical, digital or analog input and output variables
- ⇒ The internal bit, word or double word flags are used for intermediate calculations in the user program.
- ⇒ The indirect bit, word or double word constants.
- ⇒ The step chains, which allow the simple programming of sequential operations.
- ⇒ The historical values: Certain functions require several program cycles for execution. For instance, the variable ZEIT is updated in each cycle during the processing of a time function, i.e. this variable is an example for a historical value.

It is not possible to directly access the historical values via the user program for basic units of the series 40 and 50.

### Notes on the historical values:

The maximum permissible number of historical values for a project is 256.

A list of the functions with their historical values can be found in the appendix.

The historical values of the timer functions are independent of the other functions historical value list. The number of authorized timer functions is unlimited, but the number of timer functions used at the same time is limited to 42.

Value ranges:

- Bit variables (status 0 or 1)
- Word variables (value range -32768 to 32767)
- Double word variables (value range -2147483648 to 2147483647)
- Text (ASCII characters)

Type	Variables from to		Description	Series			
				30	40	50	90
Digital inputs	E 00,00	E 61,15	Digital inputs of the CS31 system bus	x		x	
	E 62,00	E 62,07	Digital inputs of the basic unit		x	x	
	E 62,00	E 62,11	Digital inputs of the basic unit	x			
	E 63,00	E 68,15	Digital inputs of the basic unit expansions		x	x	
Analog inputs	EW 00,00	EW 61,15	Analog inputs of the CS31 system bus			x	
	EW 00,00	EW 05,15	Analog inputs of the CS31 system bus	x			
	EW 06,15		Fast counter, inquiring counter content	x			
	EW 07,00	EW 07,07	Reserved variables	x			
	EW 07,08	EW 07,14	Reading real-time clock	x			
	EW 07,15		Status for CS31 system bus	x			
	EW 62,00		First potentiometer of the basic units		x	x	
	EW 62,01		Second potentiometer of the basic units		x	x	
	EW 62,02	EW 62,07	Reserved variables		x	x	
	EW 62,08		Seconds (0...59)			x	
	EW 62,09		Minutes (0...59)			x	
	EW 62,10		Hours (0...23)			x	
	EW 62,11		Weekdays (1...7)			x	
	EW 62,12		Days of the month			x	
	EW 62,13		Months			x	
	EW 62,14		Years			x	
	EW 62,15		Status word: Bit 0: displaying class 2 errors Bit 1: for detection of the CS31 system bus in slave basic units Bit 8 to 15: are used in master basic units and define the number of remote modules which are detected on the CS31 system bus.		x	x	
	EW 63,00	EW 68,15	Analog inputs of the basic unit expansions			x	
	x: valid for the specified series				30 = 07 KR 31, 07 KT 31	50 = 07 KR 51, 07 KT 51	
				40 = 07 CR 41, 07 CT 41	90 = 07 KR 91, 07 KT 92, 07 KT 93, 07 KT 94		
				For further information refer to volume 4, pages 1-1 to 1-3			

Type	Variables		Description	Series			
	from	to		30	40	50	90
Digital outputs	A 00,00	A 61,15	Digital outputs of the CS31 system bus			x	
	A 62,00	A 62,05	Digital outputs of the basic unit		x	x	
	A 62,00	A 62,07	Digital outputs of the basic unit	x			
	A 63,00	A 68,15	Digital outputs of the basic units expansions		x	x	
Analog outputs	AW 00,00	AW 61,15	Analog outputs of the CS31 system bus			x	
	AW 00,00	AW 05,15	Analog outputs of the CS31 system bus	x			
	AW 06,15		Fast counter, adopting initial value	x			
	AW 62,00	AW 62,15	Reserved variables		x	x	
	AW 63,00	AW 68,15	Analog outputs of the basic units expansions		x	x	
Step chains	S 000,00	S 125,15	Steps		x	x	
	S 000,00	S 015,15	Steps	x			
Internal bits	M 000,00	M 021,15	Bit flag	x			
	M 230,00	M 239,15	Bit flag	x			
	M 000,00	M 099,15	Bit flag		x	x	
	M 230,00	M 254,15	Bit flag		x	x	
	M 255,00	M 255,15	Diagnosis flag	x			
	M 255,00		Special flag: 2 Hz Oscillator		x	x	
	M 255,01		Special flag: 1 Hz Oscillator		x	x	
	M 255,02		Special flag: 0,5 Hz Oscillator		x	x	
	M 255,03		Special flag: 0,01667 Hz Oscillator (Period duration 1 minute)		x	x	
	M 255,04	M 255,09	Reserved variables		x	x	
	M 255,10	M 255,14	Error flag		x	x	
	M 255,15		Variable, which is always set to "0" during the start process. It can be used for detection of the first program cycle.		x	x	
	Internal words	MW 00,00	MW 05,15	Word flags	x		
MW 230,00		MW 239,15	Word flags	x			
MW 000,00		MW 099,15	Word flags		x	x	
MW 230,00		MW 253,15	Word flags		x	x	
MW 254,00		MW 254,07	Detailed information error class 1	x	x	x	
MW 254,08		MW 254,15	Detailed information error class 2	x	x	x	
MW 255,00		MW 255,07	Detailed information error class 3	x	x	x	
MW 255,08		MW 255,15	Detailed information error class 4	x	x	x	
Internal double words	MD 00,00	MD 01,15	Double word flag	x	x	x	
Bit constants	K 00,00	K 00,01	Binary constants	x	x	x	
Word constants	KW 00,00	KW 00,15	System constants	x	x	x	
	KW 01,00	KW 07,15	Word constants	x			
	KW 01,00	KW 15,15	Word constants		x	x	
Double word const.	KD 00,00		Cycle time (system constant)	x	x	x	
	KD 00,01	KD 01,15	Double word constants	x	x	x	
x: valid for the specified series      30 = 07 KR 31, 07 KT 31      50 = 07 KR 51, 07 KT 51 40 = 07 CR 41, 07 CT 41      90 = 07 KR 91, 07 KT 92, 07 KT 93, 07 KT 94 For further information refer to volume 4, pages 1-1 to 1-3							

## Initialization of the basic units

The existing program in the PLC is automatically overwritten during the transfer. However, it is recommended to newly configure the basic unit with the factory settings, before transferring a new program:

- ⇒ Standalone basic unit
- ⇒ The standard basic unit system configuration (refer to the next chapter).
- ⇒ No program.

To **completely** initialize the basic unit with the factory settings, select the following menu item in the main menu of the programming software 907 PC 331:

- ⇒ "PLC communic. 2"

there select

- ⇒ "2 Delete EPROM"

and then

- ⇒ "4 Cold start"

The following initializations are also possible:

Initialization type	Definition
Switching on or PLC type enabling or "Warm start"	<ul style="list-style-type: none"><li>– Delete program in RAM</li><li>– Copy content of the Flash-EPROM into the RAM</li><li>– Delete data in RAM according to the configuration</li></ul>
STOP/RUN-switch of the basic unit	<ul style="list-style-type: none"><li>– Copy content of the Flash-EPROM into the RAM. If no program is available in the Flash EPROM, the RAM remains unchanged.</li><li>– Delete data in RAM according to the configuration</li></ul>
Software " <i>Cold start</i> "	<ul style="list-style-type: none"><li>– Delete program and data in RAM</li><li>– Copy content of the Flash-EPROM into the RAM</li></ul>





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](mailto:desst.help@de.abb.com)  
Internet <http://www.abb-sst.de>

Printed in the Federal Republic of Germany (08.99)