Modsoft AKF → Micro
Type: MICROAKF
Version 2.0 – 2.1
Configuration MICROAKF

User Instruction
DOK-704107.33-1197

Translation of the German Description DOK-703412

Belongs to software kit E-No. 424 704 703

Documents in the software package

Documentation	Area of application
Installation User Instruction DOK-704104	Explains the usage and installation of the diskettes included.
MICROAKF for Beginners User Instruction DOK-704105	Serves to introduce new customers to MICROAKF. The user learns how to use the software in small steps.
Short Form Guide User Instruction DOK-704106	Tables for validity ranges and system markers, SFB-Formal operands for quick use onsite.
SFB Block Library DOK-704572	Description of the Standard Function Blocks.
Configuration User Instruction DOK-704107	Contains the new features of the current version and explains the functions of the individual software menus for the configurer.
Masterindex User Instruction DOK-704108	Index of all documentation.

Notes

Application Note

Caution The relevant regulations must be observed for control applications involving safety requirements.
For reasons of safety and to ensure compliance with documented system data, repairs to components should be performed only by the manufacturer.

Training
Schneider Automation offers suitable training that provides further information concerning the system (see addresses).

Data, Illustrations, Alterations
Data and illustration are not binding. We reserve the right to alter our products in line with our policy of continuous product development. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us by using the form on the last page of this publication.

Addresses
The addresses of our Regional Sales Offices, Training Centers, Service and Engineering Sales Offices in Europe are given at the end of this publication.

Copyright
All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including copying, processing or any information storage, without permission in writing by the Schneider Automation. You are not authorized to translate this document into any other language.

Trademarks
All terms used in this user manual to denote Schneider Automation products are trademarks of the Schneider Automation.

Terminology

Note This symbol emphasizes very important facts.

Caution This symbol refers to frequently appearing error sources.

Warning This symbol points to sources of danger that may cause financial and health damages or may have other aggravating consequences.

Expert This symbol is used when a more detailed information is given, which is intended exclusively for experts (special training required). Skipping this information does not interfere with understanding the publication and does not restrict standard application of the product.

Path This symbol identifies the use of paths in software menus.

Figures are given in the spelling corresponding to international practice and approved by SI (Système International d‘ Unités).
I.e. a space between the thousands and the usage of a decimal point
(e.g.: 12 345.67).

Abbreviations

ABS	absolute addressing
Adr.	Address (signal address)
AKF	Instruction List, Ladder Diagram, Function Block Diagram
IL	Instruction List
AWP	User program
DB 0 ... 9	Data block(= SYM/COM block)
D-Word	Double word
I/O	Input / Output siganls (e.g. of a module)
FB	Function block
FBD	Function Block Diagram
G-Word	Floating point word
HW	Hardware (e.g. PLC)
IB	Initialization block
LD	Ladder diagram
OB	Organisation block
PB	Program block
PADT	Programming and debugging tool (= Programming panel)
React.	Reaction during descriptions of steps (on screen)
SFB	Standard function block
PLC	Programmable controller system (= programmable controller)
SSP	Signal memory
SW	Software
SYM	Symbolic addressing
SYM/COM	Symbols and comments
SA	Node
<Return>	actuate the Return key
<Esc>	actuate the Esc key
<Ctrl>+<Alt>+<Del>	actuate the keys Ctrl, Alt and Del simultaneously (begin with Ctrl and end with Del)

Objectives

The functions of the Micro configuration software are described. The documentation is built up as a reference document. Frequent consultation of the index is recommended.

Arrangement of This Guide

Chapter 1 contains new features and a rough breakdown of the software structure

Chapter 2 goes into the use of the software

Chapter 3 contains the actual function descriptions, divided into Edit, Load, Online, Print, Special and Setup.

Validity Note

This instruction manual is valid for the software MICROAKF, version 2.0 – 2.1

x

Table of Contents

Chapter 1
Introduction

This chapter includes general information on the present manual. In addition, an overview is given on its available functions of the software.

1.1 General

These programming instructions are intended as a reference for the Modsoft AKF software→ Micro version number 2.0

The software is used for the structured programming of PLC applications by means of modern window technology (windows, pulldown menu).

☞ **Note** After installation, you may call a help text menu by pressing the <F10> key twice. General information on the Modsoft AKF software and on the user interface can be obtained under ”Information on help” and ”Modsoft AKF Software Overview”.

The texts in the ”Programming” chapter are arranged to match the order in the Modsoft AKF main menu from left to right, although programming should start with the setup functions. Within the description, the alphabetic index explains where to find function descriptions in the text.

1.2 New Performance Features of version 2.1 compared to 1.0

Compiling and loading of the program into the PLC with the new version is required only if you want to utilize the additional functions of this version.

- **General**
  - New modules:
    CPU 522 for telecontrol
    CPU 622 for telecontrol
    TXT 201
    TXT 2C1
    TXT 2E1
    ZAE 201
    ZAE 204
    FRQ 204

- **MOT201 and MOT202**
  - The use of the expert modules is not possible.

  - New SFBs:
    BLA4
    BLA5
    BLA6
    DLA4
    DLA5
    DLA6
    FRQ204
    LA4
    LA5
    LA6
    LBS
    SB
    SHW
    TXT_RCV
    TXT_SEND
    TXT_WRTE1
    TXT_WRTE2
    VBIS

VBYS
VWS
WEG
ZAE
ZAE204

❒ **Parameters for Slave**
- The menue ”Load → Parameters for Slave” enables the configuration of a slave micro from a master micro loaded with Modsoft AKF.

❒ **Load Modsoft**
- In the menue ”Load → Load Modsoft” you can load the Modsoft-Executive to use an AKF-Micro as a slave.

❒ **Interface COM1 / COM2**
- The Selection of the ”Interface” at the PLC Station Pre-setting is be carry out at the menue ”SeTup” → ”PLC Station”.

1.3 New Performance Features of version 2.1 compared to 2.0

❒ **General**
- New module:
  CPU 722 for telecontrol

❒ **Read RTU Parameters**
- The telecontrol specific program parts will be loaded into the telecontrolstation with the menue ”Load” → ”Read RTU Parameters”.

❒ **Bus Modnet 1F / Modnet 1W**
- The selection of the dedicated line for the Geadat station is be carry out at the menue ”SeTup” → ”PLC–Station” → ”PLC Station Pre–setting”.

1.4 Validity Ranges

The validity ranges for MICROAKF version 2.1 can be found in the MICROAKF Helptext (Menu point ”Validity Ranges”).

1.5 Basic organization of the Software

The software is divided into the following basic structures:

Edit

- Overview
- Block
- Symbols and Comments
- EQuipment List
- TItle Block
- Replace Signals
- Compile (Blocks)
- Compile (Data Blocks)

Load

- Program Link
- Program to PLC
- Exchange Online
- Read EQuipment list
- Read-Out PLC
- Compare
- Set Date/Time
- Bootloading
- Parameters for Slave
- Load Modsoft
- Read RTU Parameters

Online

- StArt PLC
- StOp PLC
- Dynamic Status Display
- Online-List
- PLC Status

Print

- Overview
- Program Protocol
- Symbols and Comments
- Cross Reference List
- Signal Occupancy List
- EQuipment List
- Complete Documentation

Special

- Directory
- Archive Station
- Restore a Station
- Erase a STation
- Format Disks
- CoPy Disks
- EraSe Files
- Copy Files
- Operating System MS-DOS
- System Information
- End of Station Handling

SeTup

- PLant
- PLC station
- Printer
- Color
- Modem-Connection

Chapter 2
Operation

Programming with the Modsoft AKF software is carried out on standard IBM compatible computers.
The operator may make use of a mouse and/or keyboard.

2.1 Operation With a Mouse

In addition to the typewriter keyboard, the line editor and the cursor keys, you can also operate the software using the mouse. In this connection it should be noted that a parallel mouse (Bus mouse) requires its own connector slot in the PADT, which must be configured.
Please consult the manufacturer’s instructions regarding the installation of the mouse.

The functions of the mouse in the software
The left hand key serves as a means of calling and confirming the function. It thus corresponds to the <Return> key.
The right hand key is used to break off a function - this corresponds to the <Esc> key.

Expert The mouse can be operated at the COM1 port if the /NOSPS parameter is used when calling up.
For call up without /NOSPS parameter the mouse must be connected to the COM2 port.

2.2 Operation With a Keyboard

2.2.1 Keyboard Sections

The keyboard of the programming panels has been divided into three different sections:

- **Typewriter Keyboard and Control Keys**
  It is located in the middle (white keys) and is surrounded by grey control keys. Only the white keys cause a display of the imprinted symbol to appear on the screen when they are pressed.
  In Modsoft AKF, these keys are used to operate the line editor, the key macros and the reference letters (see ”Special keys”, chapter 2.2.7), as well as to enter text.
- **Function Keys**
  These are of secondary inportance as far as pulldown menu operation is concerned, since all the functions are selected with the mouse, using reference characters or cursor keys. These keys are located above or beside the typewriter keyboard, depending on the PADT model, and are labeled with <F1> ... <Fx>.
- **Numeric Keypad**
  The cursor keys and the numeric keypad are located on the right-hand side of the typewriter keyboard. The <NumLock> key is used to switch from the numeric keypad to the cursor block.
  In Modsoft AKF, principally the cursor keys are required on the numeric keypad. Amongst other uses, they serve to select menu lines (see below).
  The numeric keys are required for generating the graphic characters (e.g. in the title block): <Alt> + <digit>.

2.2.2 ”Line editor”

The line editor is used for entering texts and/or numerals. It involves the use of combinations of control keys and the typewriter keyboard.

The following editing functions are available:

<Home>	Cursor to the start of the entry line	
<End>	Cursor to the end of the entry line	
<→ >	Cursor to the right	
<← >	Cursor to the left	
<Return>	Effects the entry	
<backspace>, <⇐ >	Delete character at the left	
<Del>	Delete character under the cursor	
<Ctrl>+<T>	Delete entry	
<Ctrl>+<R>	Restore original text	
<Ins>	Toggle between	Insert (large cursor)/ Overwrite (small cursor)
<Esc>	Abort entry	

The <Esc> key is used to abort an entry and restore the previous text.

The key macros (see below) may be used to insert saved texts such as symbols, comments, hardware addresses, data names or parts of such texts.

2.2.3 Screensave

This function is used to copy the complete current screen contents into a file. No printer need be connected. The file is stored in IBM-ASCII format in the directory of the current station. The copy (file) thus generated can subsequently be edited using a text editor.

Step To generate a screen copy, simultaneously press the keys <Alt>+<F2>.

The first screen is stored under the name ”BILD0.BLD”. Screen 2 is called ”BILD1.BLD” etc.

☞ **Note** If Modsoft AKF is left and restarted, the screen counter starts with ”BILD0.BLD” again. Previously created copies with the same name are overwritten.

2.2.4 Keyboard buffer, ”Learning”

This function can combine up to 40 procedures (key inputs).

Step 1 Press the key combination <Alt>+<F8> to activate the function.

Step 2 Press the key combination to be assigned to input sequences. You may assign <Alt>+<0>, <Alt>+<1> etc. up to <Alt>+<9>.

The status line shows ”learning”.

Step 3 Select the desired menus by means of the reference characters (up to 40 entries per assigned key).

Step 4 Press the key combination <Alt>+<F8> to terminate learning.

”Learning” disappears from the status line.

Step 5 Leave Modsoft AKF to store.

Step 6 Call up ”MICROAKF”.

Step 7 Press the key combination <Alt>+<0>, <Alt>+<1>, etc. up to <Alt>+<9> simultaneously to have the stored procedures run.

☞ **Note** The menus should be selected by reference characters in order to make the learned functions independent of the current menu bar position.

2.2.5 Key text buffer, ”key macros”

This function facilitates editing of symbols, comments, hardware addresses, etc. Frequently used absolute or symbolic addresses can be stored as text. Proceed as follows.

Step 1 Press the key combination <Alt>+<F9> to activate the function.

Step 2 Press the key combination to which text is to be assigned. You may assign <Alt>+<A>, <Alt>+<B>, etc. to <Alt>+<Z>.

Step 3 Enter any desired text (up to 19 characters).

Step 4 Actuate the <Return> key to store.

Step 5 Press the key combination <Alt>+<A>, <Alt>+<B>, etc. up to <Alt>+<Z> to recall the stored text and cursor position.

You may abort the entry in step 4 by pressing <Esc>.

Example: the text **Valve** was stored. It is inserted during the symbol/command entry (**Valve** 1 open, **Valve** 2 closed, **Valve** 1 at mid position, ...).

2.2.6 Auto-Repeat-Function

Most keys in the PADT are equipped with the auto-repeat-function:
When a key is pressed down, the symbol which is represented on it is displayed on the screen, or a corresponding function is executed. If this key is pressed for longer than approx. 0.5 seconds, the function of this key is repeated up to ten times per second until the key is released. It is thus important to ensure that this key function is not unintentionally activated.

2.2.7 Special Keys

The following keys are especially important for driving the menu:

- **Reference Characters**
  Reference characters are used to directly select and execute a menu function, with the aid of the typewriter keyboard. A desired function is selected by entering the capital letter which is in inverse text. The screen marking colors may be adjusted in the ”SeTup” and ”Colors” sections.

- **<Ctrl>+Reference Letters**
  Within the editor individual functions can be called up with <Crtl>+Reference Letter without having to open the menu.

- **Cursor Keys**
  Individual lines of a menu may be selected using the cursor movement keys (also known as movement keys, cursor keys, direction keys) < ↑ >, < ↓ >, < → >, <← >.
  In this way, for example, the key-words are selected in the helptext.
  Each time one of these keys is pressed while editing, the following takes place:

< ↓ >	one line down (in the same column)
< ↑ >	one line up (in the same column)
< → >	one column to the right (in the same line)
< ← >	one column to the left (in the same line)

  Please bear in mind that the <NumLock> key must be inactive for the the keys to respond in this way. When editing a block, the cursor may be moved at will using the mouse.

- **<Return> Key**
  The <Return> key (also known as <Enter>) is used to select specific functions or to accept (end). It can also be used to ”toggle” (see below).

Call

- within the help function in order to call the help texts for the selected keyword
- in the pulldown menus in order to call the selected menu line (menu function)
- in ”Edit” (blocks, equipment list etc.): <Ctrl.>+<Return> creates the windows for the editing functions

Terminate, Accept

- when editing the network and making entries using the line editor, the <Return> key serves to accept the parameter, the hardware address and/or the comments just entered.

- **”Toggling”**

  A large number of menu functions offer direct selection of pre-set modes. Thus, for example, you can switch between ”off” and ”on” or ”IL”, ”LD”, ”FBD” directly (”toggling”) without having to enter the text.
  Toggling can be carried out in the following manner:
  1 Type in the specified reference character
  2 Select the line with the cursor keys and toggle using the <Return> key.
  The <Return> key corresponds to the left-hand key of the mouse.

- **<Esc> Key**

  Every function and/or input within Modsoft AKF software can be aborted using the <Esc> key.
  The <Esc> key corresponds to the right-hand mouse key.

Caution If you are at the operating system level you can only abort with <Ctrl>+<C>. This also applies to the functions in the ”Special” and in the ”Print” menu.

- **<Ins>-Key**

  In the line editor, for comments and headlines, the <Ins> Key is used to toggle between the text mode ”Insert” and ”Overwrite”. Using the <Ins> Key in the Block Editor the network is vertical spread at the cursor position.

Table of Important Keys

No single comprehensive listing of important special keys and key combinations can be provided here. The same key sequence can have different functions at different times. If questions arise regarding a particular key combination, please look it up in the key table in the index. The available key combinations and their significance are also given in the explanations of the corresponding menus.

Chapter 3
Programming

This chapter describes programming with the Modsoft AKF software for Micro.
The chapter is intended as a reference for the person configuring the system. Its structure follows the menu structure. Menu points that occur more than once are only discussed once (the first time). The index contains the page numbers for the description of menu points that occur more than once.

3.1 General Information

The individual menu points are described in the following order:

Firstly you will be shown how to select the menu line required (the ”path” from the main menu). Then, the functions which you can execute in this position (programming, documentation, archiving etc.) are described.

This symbol tells you how to select the function described. The steps are always given starting from the main menu. For example:
- ”Edit”, ”Equipment List”, ”Parametrize Central Controller”, ”Marker Bits”
means:
Select ”Edit” menu,
confirm ”Equipment List” line using Return,
select ”Parametrize Central Controller” and confirm using Return,
the ”Marker bits” is the function which has just been described.

Note Some characters are written as capitals in the following description. They identify the reference characters:
e.g. Se**T**up

3.2 Modsoft AKF Main Menu

Status line, here path of the current Plant/ Station; according to installation at
A:
B:
C:
D: or E:

Serial no. of the Diskettes as part of the software kit

Version no. of the present software

pulldown menus selectable with reference characters, cursor keys and Return or mouse

3.2.1 Status Line

The so-called status line is the lowermost screen line.

The 78 characters of the status line are subdivided as follows:

```
12345678901234567890123456789012345678901234567890123456789012345678901234567
8
```

- 1 - 30 complete path of systems and station name
- 32 - 48 Message: LEARNING
- 50 - 64 Modsoft AKF status: FORCE active
- 65 - 78 PLC cycle status: PLC stopped/PLC active

3.2.2 Pulldown Menus

The individual menu lines can be selected using reference character, cursor keys and <Return> or using the mouse (see chapter ”Operation”).

3.2.3 Help Function

Help texts are descriptive texts on different functions. They give information on the possibilities offered by the current menu line or window. Help texts are also available for the more general menu points.

The <F10> key displays a help text for the selected line of the pulldown menu. This text can contain so-called keywords. These can be used to get additional help texts. The keywords are selected using cursor keys and the <Return> key. This helps to keep the help texts simple. The texts are mainly memory aids for Modsoft AKF beginners. The keywords are written in color. You can return to the previous help text using <Alt>+<F10>.
If you actuate <F10> once again, you arrive at the help text overview. The selectable keywords are marked here without their own menu line.
You may call references for any keyword in the index (index for help texts).

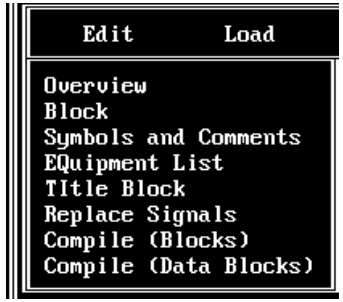
On the lower edge of every help window, all available functions are listed:

<↑>, <↓>, <←>, <→>	Selects keyword
<Return>	Calls help text for the keyword
<Esc>	Aborts and exits the help window
<F10>	Help text overview
<Alt>+<F10>	Calls previous help text
<PgUp>, <PgDn>	Pages backwards/forwards within the help text

3.3 Edit

The functions in the Pulldown menu ”Edit” are used to configure the user-program logic.

The following menu lines are provided:



Caution The menu will carry out processing on the plant/station selected under ”SeTup”, ”Plant”, ”PLC Station”. The equipment list must be entered before the blocks. The data block must be entered under ”Symbols and Comments” if you wish to program symbolically.

3.3.1 Overview

- ”Edit”, ”Overview”

This function provides a tree-structured graphic overview of the program run. This overview shows how the complete program is structured, where and how often which block is called. This function provides for shifting of the image, selection using the cursor (with the possibility of branching to the editing mode) and block search. The overview is generated starting from the OB1. The display is sorted by networks. If several blocks are called in a network, it is possible that they will not be given in the correct sequence in this display.

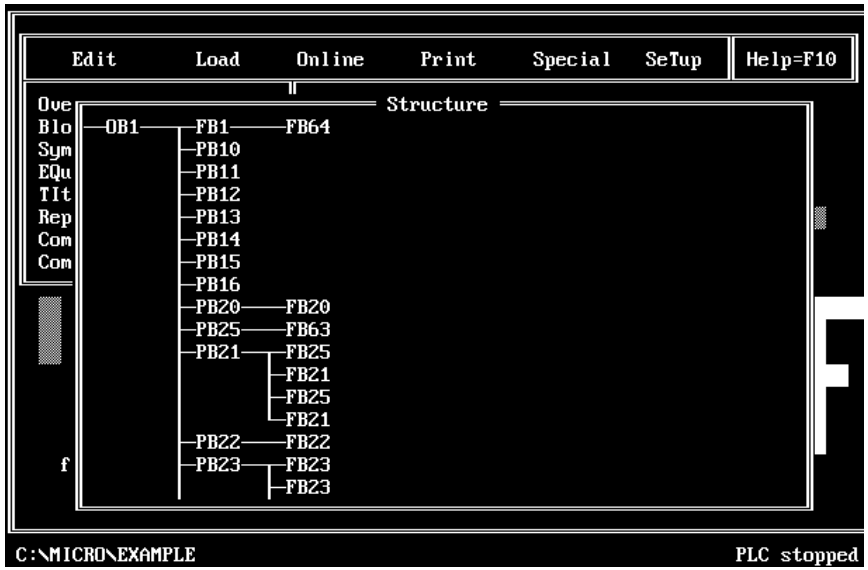
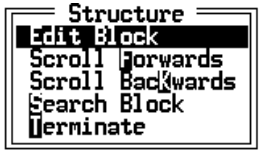


Figure 1 Screen Display of a Program Overview

After pressing <Ctrl>+<Return>, you can select other special functions via a pull-down window:



The functions can be called up outside of the Pulldown Menu with <Ctrl>+<Reference Letter>. Furthermore the following special keys are available:

<Esc>	Break off
<↓>	Move the cursor down one element
<↑>	Move the cursor up one element
< → >	Move the cursor to the right
< ← >	Move the cursor to the left
<PgDn>	Next page
<PgUp>	Previous page

Edit Block

- ”Edit, ”Overview”, ”Edit Block”

First, the block to be modified is selected using the cursor keys or the search run. Next, branch into the editing mode by using the <Ctrl>+<Return> key. The block on which the cursor was situated can now be processed.

Caution After leaving the editing mode, no new program overview will be generated. If you have erased or inserted block calls during editing, they will not appear on the image.

Page forwards/backwards

- ”Edit”, ”Overview”, ”Scroll Forwards”
- ”Edit”, ”Overview”, ”Scroll Backwards”

This is a means of paging to the previous or next page in the overview. <PgUp> corresponds to paging forwards, <PgDn> corresponds to paging backwards.

Search Block

- ”Edit”, ”Overview”, ”Search Block”

This is a means of searching for blocks within the overview.
The searching direction can be specified by prefixing a “+” or “–”.
The name of the block must be completely specified.

Terminate Overview

- ”Edit”, ”Overview”, ”Terminate”

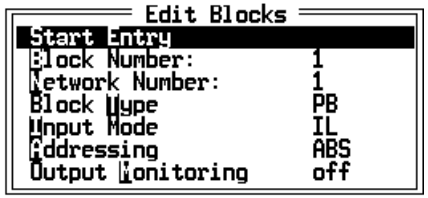
The overview function is broken off with this command sequence. You can also leave the overview using the <Esc> key.

3.3.2 Block

- ”Edit”, ”Block”

This pulldown menu is used to generate the actual program. Using the editors, the individual blocks are programmed in the type of presentation previously selected. In the following, the block editors are first described, then the individual menu lines are explained.

The following functions are available in this pulldown menu:



Block editors
The IL, LD and FBD editors (depending on ”Input mode”) are the available block editors.

The OB and PB blocks can be entered and represented in the dedicated Modsoft AKF language in instruction list (IL), ladder diagram (LD) or function block diagram (FBD).

Network 1 of an FB can only be entered and represented in IL. The succeeding networks (2 ... 999) can be entered and represented in IL, LD, FBD.

While editing the user program (after “Start Entry”), you can toggle between the different presentation modes under ”Presetting” (”toggling”). Networks are displayed on the screen in the type of presentation selected, regardless of which type of presentation they were programmed in.

Exceptions are possible. For example, if a network is programmed in IL and is subsequently to be presented in FBD, which cannot be done since it contains commands (LD,T, jumps) which cannot be presented in FBD. Another exception would be if the width of the screen were not sufficient, due to the number of ele-

ments to be presented. (An ”AND” with more than 7 inputs, an ”OR” with more than 16 inputs, or several FBD elements in a parallel path cannot be represented in LD).

In such exceptional cases, the presentation is always done in IL, regardless of the type of presentation selected.

The addressing mode can be changed during editing and representation.
After input, the program is automatically checked for correct syntax.

Start Entry

- ”Edit”, ”Block”, ”Start Entry”
- ”Edit”, ”Overview”, ”Select block”, <Ctrl>+<Return>, ”Edit block”

Note Editing is started here with <Return> or the reference characters only after laying down the other settings of the pulldown menu.

If the block entered is not available, it will be generated.
To call the pulldown edit menu (”Edit network”), press the <Ctrl>+<Return> key again.
In addition, you may call the appropriate help text for any step with <F10>.
You may abort the editing process with the <Esc> key at any time.
However, if you do this, then all previously entered data except for comments and symbols of signals will be lost.

Editing in various types of presentation
It is possible to program in three types of presentation. It is possible to switch between the individual dedicated languages at any time.

The various editors are now explained.

Certain menu lines or functions are present at several locations in the editor, or are the same in several editors. An example of this is the “edit network” menu. Such positions will only be described once.
The index register is a useful reference for further information.

IL Instruction list

IL editor

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

☞ **Note** When you select an already existing block in the desired type of presentation, you can then process it. After pressing <Return>, you will gain access to the ”edit network” in the menu, the structure of which is the same for all the types of presentation.
The common functions are described here in IL.

You are now in the IL editor, which will be described below.

Editing in IL can be carried out
with ”edit network” in the pulldown menu, using the reference characters, by seleting the menu line with the cursor key and choosing it with <return>, or with the aid of the mouse.

When pushing the <Return>- key, you are able to do the function mentioned above with the menu lines of the pulldown menu ”Edit Network” The menu is displayed on the right bottom of the screen.

During the programming of the network in IL, editing is carried out line by line in a tabular format. That is to say specific tabular positions are reserved for the IL commands (operations), the operands and the IL line comments (in FBs). These positions are reached by pressing the <Tab> key or the cursor keys.

As far as OBs, PBs and FBs are concerned, every IL line can be provided with its own comments (a maximum of 32 characters). The line editor (typewriter keyboard) with all keys (also e.g. <Backspace>) is used within the comment lines.

Editing IL, LD and FBD / Insert network

- ”Edit”, ”Block”, ”Start entry”, existing block, <Ctrl>+<Return>, ”Insert”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

By means of the ”Insert” function in the ”Edit network” pulldown menu, you may insert a new, empty network in front of the network displayed.

Exception: no insertion is permitted in front of the first network of an FB.

After having inserted a new, initially empty network, you are in the correction mode of the editor.

Editing the IL, LD, FBD / Erase network

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Erase”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can erase the indicated network using the “Erase” function in the “Edit network” pulldown menu. The last remaining network of a block and the block-end network cannot be erased using this function.
This can only be done by erasing the entire block in “Special”, “Erase Files”.

Edit IL, LD, FBD / Copy network

- ”Edit”, ”Block”, ”Start Entry”, Existing block, <Ctrl>+<Return>, ”Copy”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

By means of the ”Copy” function in the ”edit network” pulldown menu, you can insert (copy) an already created network in front of the network displayed.

Exception: no insertion is permitted in front of the first network of an FB.

After having selected the function, a window appears, in which you have to enter the network to be copied. It is then copied in front of the current network.

After having copied the network, you are in the correction mode of the editor.

Editing IL, LD, FBD / Modify network

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The correction mode of the Block editor is selected with the “modify” function in the “Edit network” pulldown menu. You can then modify the network which is presented on the screen.
It is not possible to modify the network with ”block end”.

IL editor / correction mode

- ”Edit”, ”Block”,
  ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can reach the correction mode of the IL editor from the “Edit network” menu (existing block) and/or from the “Edit IL” menu (new block).

a) The pulldown menu ”Edit IL” consists of the following functions:

b) The following special key functions are available:

< ← >, < → >, <↓>, <↑>	Move cursor
<Ctrl>+<PgUp>	Scroll screen upwards
<Ctrl>+<PgDn>	Scroll screen downwards
<tab>	Next input field (tab position)
<backtab>	Previous input field (tab position)
<Esc>	Abort input without saving

By pressing the <Return> key, you can also execute the listed functions via the menu lines in the “Edit IL” pulldown menu.

Editing IL / Insert line in correction mode

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Insert Line”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Using this function you can insert IL lines in the network wherever they are required.
The inserted line is always placed in front of the line on which the cursor is positioned.

Example:

: A M 2.4	: A M 2.4	: A M 2.4
: O I 6.12	:	: A M 4.15
: = Q 7.3	: O I 6.12	: O I 6.12
: ***	: = Q 7.3	: = Q 7.3
	: ***	: ***
insert in front of the 2nd line	line is inserted	enter instruction

Edit IL / Erase Line in correction mode

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Erase line”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can delete IL lines wherever required in the network using this function.
The line on which the cursor is situated, is erased.

Edit IL / Terminate in the correction mode

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Terminate”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The correction mode of the editor and thus the correction of the network is terminated using this function and the inserted, copied or changed network is taken over.

Caution If the ”Edit network” menu is aborted by ”Break” or <Esc>, the change is lost. To compile and transfer the network to the user drive, ”Edit network” and ”Terminate” must be selected.

Edit IL, LD, FBD / Break in correction mode

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Break”
- ”Edit”, ”Overview”, Select Block, <Ctrl>+<Return>, ”Edit Block”,..

The correction mode of the network is aborted using this function or the <Esc>key. The changes are ignored and the old status remains.

Edit IL, LD, FBD / Network Title in correction mode

- ”Edit”, ”Block”, ”Start entry”, <Ctrl>+<Return>, existing block, <Return>, ”Modify”, <Ctrl>+<Return>, ”NW Title”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

The network title, with up to 32 characters, can be entered in the status line.

Edit IL, LD, FBD / Network comment in correction mode

- ”Edit”, ”Block”, ”Start entry”, <Ctrl>+<Return>, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”NW comments”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

In a window, up to 15 x 60 characters may be entered as comment on the network. The lines can be set up in any fashion desired.
If <Ctrl>+<Return> is pressed, the following window is called:

End editor without saving
Save comment and end editor
Position cursor at start of text
Positioning cursor at end of text
Insert a blank line before current line
Erase cursor line, following text moves up
Erase entire text, without ending the editor

The following functions are available through special keys:

< → >, < ← >, <↑>, <↓>	Move cursor
<Ins>,	Toggle between insert/overwrite modes
<Del>	Delete character under the cursor
<backspace	Delete character in front of the cursor. At the beginning of the line: append current line to the previous one
<Ctrl>+<Y>	Erase current line, following text moves up
<Home>	Move cursor to the beginning of the line
<End>	Move cursor to the end of the line
<Esc>	Leave the editor without saving
<Return>	Cursor at beginning of line: call menu, Cursor at end of line: Set cursor into next line, Cursor in line: Insert all text after the cursor into next line

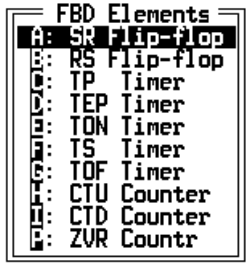
Edit IL / FBD elements in correction mode

- ”Edit”, ”Block”, ”Start entry”, program element, <Ctrl>+<Return>, ”FBD elements”
- ”Edit”, ”Block”, ”Start entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”FBD elements”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”,..

In the pulldown menu, the elements can be called by the reference character or by selecting the menu line and <Return>. Outside the pulldown menus, they can be called by <Ctrl>+<reference character>.
After selection of an FBD element, the editor specifies the appropriate sequence of instructions without operand.

Note The called FBD element begins without an AND-instruction in the first line.

The following FBD elements can be inserted in IL:



Edit IL, LD, FBD / Presetting in the correction mode

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Pre-setting”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

By means of this function, you can change the basic settings of the editor.

The functions which are available to you in the presetting menu in the correction mode of the editors:

Editing IL, LD, FBD / Enter Symbol in DB

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”, “Enter symbol in the DB”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can toggle between ”on” and ”off”.
When the switch position is ”on” you can enter an absolute address, a symbol and a comment in the preset data Block (DB) in the IL, LD or FBD editors.

A symbol may be entered in DB (an input window appears):
as soon as the input in the editor is terminated using the <Return> key, the input is error-free, and no symbol exists yet for the hardware address.

Your input signal shows up in the first line. Under this line, a symbol with a maximum length of 8 characters can now be entered and under this a comment up to 40 characters in length. All inputs should be terminated using the <Return> key. After the input has been terminated, the symbol or the comment can only be changed under the “Symbols and Comments” menu. “Break” or <Esc> causes the entries in the data block to remain unchanged.

Edit IL, LD FBD / Scroll network Forwards

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Scroll forwards”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The next network is called after selecting this function or with <PgDn>.
If networks cannot be presented in the selected presentation mode (LD/FBD), IL is automatically used.

Edit IL, LD, FBD / Scroll network Backwards

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Scroll backwards”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The previous network is called after selecting this function or using <PgUp>.
If networks cannot be presented in the selected presentation mode (LD/FBD), IL is used automatically.

Edit IL, LD, FBD / Terminate (save)

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Terminate (save)”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Editing is terminated by selecting this function or with the <End> key.
The input block is checked for syntax, compiled and then saved on the user drive.

Edit IL, LD, FBD / Break

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Break”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

By selecting this function or pressing <Esc>, and confirming, editing is aborted. After you have acknowledged, all program changes will be lost.
Comments and symbol entries into the data block remain unchanged.

Edit IL, LD, FBD / Signal search

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Signal search”
- ”Edit”, ”Block”, ”Start Entry”, <Return>, ”dyn. State Display”, <Ctrl>+<Return>, ”Signal search”
- ”Online”, ”Dynamic Status Display”, ”Start display”, <Ctrl>+<Return>, ”Signal search”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can search for signals inside the block using the search function.
The signal names can be entered absolutely or symbolically. The searching direction (forwards, backwards) can be specified by prefixing the character with “+” or “-”. No prefix causes forward searching.
The search begins before or after the network which is being set at the present.
You can access this function in the window by pressing the <Return> key.
You can close the window again by using the <Esc> key.

Absolute or symbolic addresses can be input. The searching direction is specified by a prefixed < - > or a < + >.

Edit IL, LD, FBD / Network Search

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Network Search”
- ”Edit”, ”Block”, ”Start Entry”, <Return>, ”Dynamic Status Display”, <Ctrl>+<Return>, ”Network Search”
- ”Online”, ”Dynamic Status Display”, ”Start display”, <Ctrl>+<Return>, ”Network Search”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can search for networks within the block using this searching function. In order to do this you have to input the network number.

By pressing the <Return> key, you can access the function in the window. You can close the window again by using the <Esc> key.

Edit IL, LD, FBD / Exchange Online

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Exchange Online”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

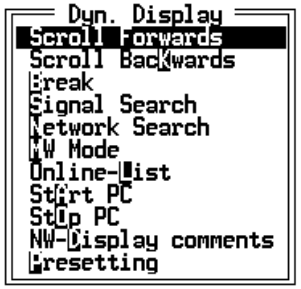
☞ **Note** The description can be found under ”Load”, ”Exchange Online”.

Edit IL, LD, FBD / Network Dynamic State Display

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Dynamic State Display”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The signal states are dynamically overlaid in the current network after <Return> is pressed. You return to the editing mode by using the <Esc> key.

If <Ctrl>+<Return> is pressed, the following pulldown menu is called.



The functions which are available in the dyn. status display are shown here.

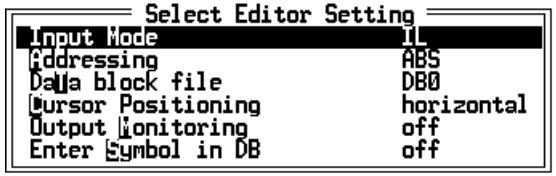
After <Return> is pressed, the signal states are dynamically overlaid into the current network.

You return to the editing mode by using the <Esc> key.

Edit IL, LD, FBD / Network Presetting

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can change the basic settings of the editor using this function. The following functions are available:



(LD/FBD/IL)
(absolute/symbolic)
(DB0 to DB9)
(horizontal/vertical/off)
(on/off)
(on/off)

”Output Monitoring” and “Enter Symbol in DB” are available in the correction mode of the IL editor.

Edit IL, LD, FBD / Presetting / Data Block File

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”, “Data Block File”
  “SeTup”, “Station”, “Data Block File”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Here, you must specify a data block that has already been edited in the ”Edit”, ”Symbols and Comments” menu.

Edit IL, LD, FBD / Presetting / Cursor positioning

- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”, “Cursor positioning”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

In automatic cursor positioning you can adjust (by toggling) the direction in which the cursor should move between the question mark fields when entering in LD/ FBD.

Horizontal position:	When entering addresses the cursor moves within a network line horizontally from one element to the next.
Vertical position:	When entering addresses the cursor moves vertically in a network column from one element to the next.
Off-position:	The cursor is moved one character at a time on the screen.

Ladder diagram LD

LD editor

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return> (when selecting an existing block)
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Note An existing block can always be processed subsequently. Pressing <Return> brings you into the “Edit Network” menu, whose structure is identical for all types of presentation. The common functions are handled in IL.

You are now in the LD editor, which is described below.

You can edit in LD

a) in the ”Edit Network” line of the pulldown menu, using the reference characters, by selecting the menu lines with cursor keys and <Return>, or with the mouse, or
b) with the special keys.

a) The following functions are available in the pulldown menu ”Edit network”.

b) Special keys for the present block; Edit network

<PgDn>	page to next network
<PgUp>	page to previous network
<Esc>	Break off input without saving

Pressing the <Ctrl>+<Return> key selects this function in the window. By pressing <Return>, you can also select the above functions via the menu line of the pulldown menu. The menu appears on the bottom right hand side of the screen. The window is closed again by using the <Esc> key.

When programming the network in LD, it must be noted that elements can only be entered at specific cursor positions. These positions must be moved to using the cursor keys.
For editing, the cursor must be situated on the line.

Insert:	to the right of the cursor
Insert FBD element:	Cursor has to be situated on the first character of the contact (spread the space first or toggle to insert mode, if required)
Erase:	to the right of the cursor
Erase FBD element:	Elements of the same type can be replaced, if the cursor is situated on the first entry (first rung).

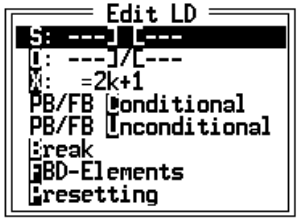
It is possible to position the cursor automatically using the “Presetting” menu. The address input begins with the first question mark.

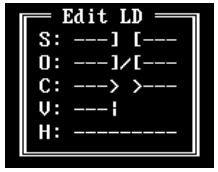
LD editor / correction mode

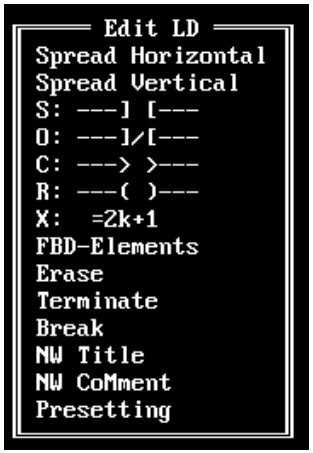
- “Edit”, “Blocks”, “start entry”, existing block, <Ctrl>+<Return>,
  ”Modify”, <Ctrl>+<Return>,
- ”Edit”, ”Overview”, ”Select Block”, <Return>, ”Edit Block”,..

You can get to the correction mode of the LD editor from the “Edit network” menu (existing block) or from the “Edit LD” menu (new block, after entering the first element).

a) There are 3 “Edit LD” pulldown menus, which appear during the editing:







b) The following functions are available via the special keys:

<←>, <→>, <↓>, <↑>	Move cursor
<Ins>	vertical spreading
<Del>	Delete element
<PgUp>	Scroll screen upwards
<PgDn>	Scroll screen downwards
<Ctrl>+<PgDn>	Page screen upwards (page forwards)
<Ctrl>+<PgUp>	Page screen downwards (page backwards)
<character>	Change contact address
<Esc>	Abort without saving
<Home>	Terminate correction

By pressing <Ctrl>+<Return>, you can also execute the above functions via the menu lines in the various “LD” pulldown menus.

Edit LD / Horizontal spreading in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Spread horizontal”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Spread Horizontal”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The network is spread horizontally to the right of the cursor position using this function. The network is optimized when inserting FBD elements.

Spreading is not required in the insert mode, since Modsoft AKF provides the required space.

Edit LD / Spread vertical in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Spread Vertical”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Spread Vertical”
- ”Edit”, ”Overview”, ”Select Block”, <Return>, ”Edit Block”,..

It is necessary to spread vertically before the insertion of a new parallel path. The cursor must be situated below the horizontal series rung. Spreading takes place above the cursor position. Spreading vertically corresponds to the <Ins> key.

Spreading is not required in the insert mode, since Modsoft AKF provides the required space.

Edit LD / Normally open contact in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Normally open contact”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Normally open contact”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol ⊣⊢ reference character <S>
This function serves to add a normally open contact to the right of the cursor position:

a) in a series rung:
   When a new network is selected, the output is automatically inserted.
b) in a parallel rung:
   A normally open contact can be edited with this function as the first element of a parallel rung. In order to accomplish this, the cursor must be situated at least 3 lines below an element or the insert mode must be active.

Edit LD / Normally closed contact in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Normally closed contact”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Normally closed contact”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol ⊣/⊢ reference character: <O>
This function serves to insert a normally closed contact to the right of the cursor position:

a) in the serial ladder diagram line:
   When a new network is selected, an output is inserted at the same time.
b) in the parallel ladder diagram line:
   A normally closed contact can be edited as the first element of a parallel rung by using this function. In order to accomplish this, the cursor must be situated at least 3 lines below an element or the insert mode must be active.

Edit LD / Connector

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, ”Connector”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Connector”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”,..

Symbol -> >- Reference character <C>

By means of the connector, a signal is applied to an intermediate marker or output without register initialization.

Example

NW1

In NW1, the marker M4.5 is assigned to a connector.

Edit LD / Output in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Output”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Output”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol -0- reference character: <R>
This function is used for inserting an output:
Using this function, an output signal is multiplied in an existing network. The cursor must be situated at least 3 lines below the previous output or the insert mode must be on.

Edit LD / exclusive OR block in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, ”Exclusive OR”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Exclusive-OR”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”edit block”,..

Symbol =2k+1 Reference character <X>
This function is used to edit an exclusive OR block.

The output is activated, if a ”1” signal is applied to an odd number of inputs (”odd element”).

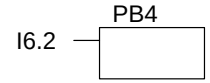
If inserted, the network design is optimized.

Edit LD, FBD / PB/FB Conditional

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, ”Insert”, <Ctrl>+<Return>, ”PB/FB Conditional”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

This function is used for a conditional call of a block. Thus, the call of a block depends on the result of a signal address.

e.g.



PB4 is only called if I6.2 = 1

You have to enter the address of the condition and of the block to be called. The call condition can be negated.

FB: The FB user standard has to be declared before the call.
PB: may be called (before the programming as well)
OB: may not be called

Edit LD, FBD / PB/FB Unconditional

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, ”Insert”, <Ctrl>+<Return>, ”PB/FB Unconditional”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

This function is used for an unconditional call of a block, i.e. the block is called in every scan.

You have to enter the address of the block to be called yourself.

FB: The FB user standard must be declared before the call.
PB: are callable (before programming as well)
OB: not callable

Edit LD / FBD Element in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”FBD Element”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”FBD Element”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

In the pulldown menu, the elements can be called by the reference character or by selecting the menu line and <Return>.
For the input of FBD elements, the same conditions apply as for the LD/FBD editor and for ”Edit”, ”Block”.
In addition:
On insertion, the network is optimized.
FBD elements of the same type can be replaced (at the same position). For this purpose, the cursor must be positioned on the rung of the first input and the overwrite mode must be active.
A list of available FBD elements is to be found in the section ”Entry mode”.

Edit LD / Erase in correction mode

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Erase”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Erase”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

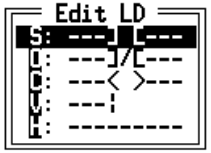
This function is used for erasing elements from a network. All LD elements to the right of the cursor are erased in the LD editor. When dealing with FBD elements, the cursor must be situated on the the first input rung in order to erase.

Edit LD / Parallel path

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, ”Parallel path”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, Parallel path
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The pulldown menu which is called with <Ctrl>+<Return> serves as a means of editing or correcting a parallel path.

The menu is at your disposal until the termination of the parallel path.
You can select among the following functions:



Edit LD / Normally open contact in the parallel path

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, parallel path, Normally open contact
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”edit Block”,..

Symbol ⊣⊢ reference character: <S>
This function is used for inserting a series normally open contact into a parallel path.

Edit LD / Normally closed contact in the parallel path

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, parallel path, Normally closed contact
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”edit Block”,..

Symbol ⊣/⊢ reference character: <O>
This function is used for inserting a series normally closed contact into a parallel path.

Edit LD / Connector in parallel path

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, parallel path, connector
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”,..

Symbol -> >- Reference character: <C>
This function is used for inserting a series connector into a parallel path.

Edit LD / Cross-connection of a parallel path

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, parallel path, ”Cross-connection”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol ⊣ reference character: <V>
The still open parallel rung is closed at the cursor position using this function. It is only possible to insert the symbol if the parallel path has been programmed correctly, otherwise the wrong path is erased.

Edit LD / Continuation of a parallel path (conductor in series)

- ”Edit”, ”Block”, ”Start Entry”, program element, <Ctrl>+<Return>, parallel path, continuation
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol — reference character: <H>
The rung is continued with no contacts.

Function block diagram FBD

FBD editor

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>
  (when selecting an existing block)
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

☞ **Note** An existing block can subsequently be processed. By pressing <Return>, you gain access to the “Edit network” menu, the structure of which is the same for all types of presentation. The common functions are dealt with in IL.

You are now in FBD editor, which is described below.

You can edit in FBD by
a) using the “Edit network” line in the pulldown menu, with the aid of the reference characters, by selecting the menu line using arrow keys and <Ctrl>+<Return> or with the mouse, or
b) with the special keys.

a) The following functions are available in the ”Edit network” pulldown menu.

b) Special keys outside the pulldown menu

<PgDn>	page to the next network
<PgUp>	page to the previous network
<End>	terminate input, compile and save
<Esc>	terminate input without saving

By pressing <Return>, you can also select the above functions by using the menu line of the pulldown menu, which appears on the bottom right hand side of the screen.

When programming a network in FBD it should be noted that elements can only be input at specific cursor positions.
These positions are selected using the arrow keys (or mouse):

Insert:	at the right of the cursor; the cursor must be located on the input rung.
Erase:	at the right of the cursor; the cursor must be located on the input rung.
Exchange:	elements of the same type can be exchanged, if the cursor is located on the first input (rung) and the overwrite mode is active.

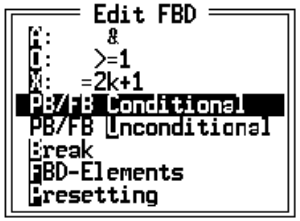
It is possible to position the cursor automatically under the “Presetting” menu for the address input.
The address input begins on the first question mark.

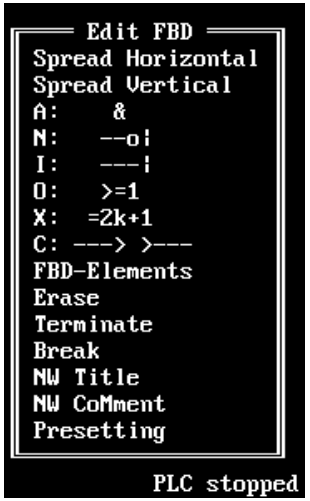
FBD editor / Correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, ”Insert”, <Ctrl>+<Return>
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

You can get to the correction mode via the FBD editor from the “Edit network” menu (existing block) or from the “Edit FBD” (new block, after entering the first element).

a) There are 2 “edit FBD” pulldown menus which appear during editing:





b) The following special key functions are available:

<←>, <→>, <↓>, <↑>	Move cursor
<Ctrl>+<→>	spread horizontally
<Del>	delete element
<PgUp>	scroll image upwards
<PgDn>	scroll image downwards
<Ctrl>+<PgDn>	page up
<Ctrl>+<PgUp>	page down
<Character>	change actual Operand
<Esc>	abort without saving

By pressing the <Ctrl>+<Return> key, you can also execute the above functions via the menu lines in the various “Edit FBD” pulldown menus.

Edit FBD / AND block in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, “AND”,
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “AND”-
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol & reference character <A>
This function is used for editing an AND block (with a maximum of 46 inputs).
The network is optimized when this element is entered.

Edit FBD / Input negated in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, “program element”, <Ctrl>+<Return>, “Input negated”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “Input negated”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol ⊣o⊢ reference character <N>
This function is used for negating an input. Outputs cannot be negated. The negation of an existing input is possible only if the cursor is positioned on the current path of the input. A negated input is inserted, when the cursor is located to the left of the block edge. The network is optimized when this element is entered.

Edit FBD / Input in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, “program element”, <Ctrl>+<Return>, “Input”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “Input”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol ⊣⊢ reference character <I>
This function is used for editing an input. An additional input can be inserted if the cursor is situated on the left next to the available symbol.
One can overwrite negated inputs with a ⊣⊢. The network is optimized when this element is inserted.

Edit FBD / OR block in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, “program element”, <Ctrl>+<Return>, “OR”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “OR”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Symbol >=1 reference character <O>
This function is used for editing an OR block (with a maximum of 46 inputs). The network is optimized when this element is inserted.

Edit FBD / exclusive OR block in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, ”Exclusive OR”
- ”Edit”, ”Block”, ”Start Entry”, Existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Exclusive-OR”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

Symbol =2k+1 Reference character <X>
This function is used for editing an exclusive OR block.

The output is activated if an odd number of inputs has ”1” signals applied (”odd element”).

The network is optimized when this element is inserted.

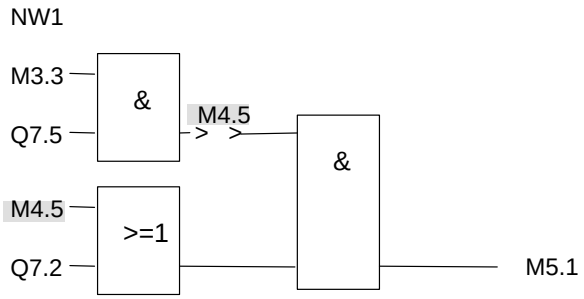
Edit FBD / connector

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, ”Connector”
- ”Edit”, ”Block”, ”Start Entry”, Existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, ”Connector”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

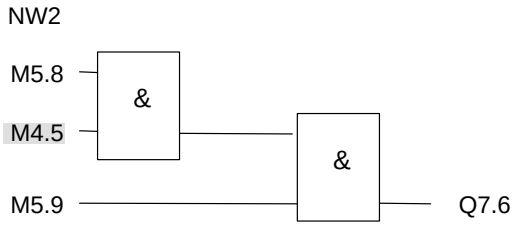
Symbol -> >- Reference character <C>

The connector applies a signal to an intermediate marker or output without initializing the register.

Example



In NW1, a connector is assigned the marker M4.5. The content of the marker is used in the same scan as input of the OR block.
In addition, the marker M4.5 is used as input of the first AND block in the next network.



Edit FBD / Spread Horizontal in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, “Spread Horizontal”,
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “Spread Horizontal”-
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

The network is spread horizontally to the right of the cursor position by using this function or the <Ctrl>+<→ >. The network is optimized when FBD elements are inserted.

In the insert mode, no spreading is required, since Modsoft AKF provides the space.

Edit FBD / Spread Vertical in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, “Spread Vertical”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “Spread Vertical”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

Vertical spreading is necessary in order to insert further inputs into the FBD symbol. The cursor must be situated below the first input ladder diagram line of the FBD system. The network is optimized when the element is inserted.
This function corresponds to the <Ins> key.

In the insert mode, no spreading is required, since Modsoft AKF provides the space.

Edit FBD / Erase in correction mode

- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, program element, <Ctrl>+<Return>, “Erase”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Modify”, <Ctrl>+<Return>, “Erase”
- ”Edit”, ”Overview”, ”Select Block”, <Return>, ”Edit Block”,..

This function or the <Del> key is used for erasing elements from the network.

Erasing an element:	Cursor must be situated by the top left corner of the block.
Erasing several connected elements:	Cursor must be situated at the beginning of the row of elements to be erased
Erasing of inputs:	Cursor must be positioned on the signal assigned an input.

The network is optimized when this function is executed.

Block Number

- ”Edit”, ”Block”, ”Block Number”
- ”Load”, “Exchange Online”, ”Block Number”
- ”Online”, ”Dynamic Status Display”, ”Current display”, ”Block Number”

You can input the number of the block here (1-999) which you wish to process. After hitting <Return>, can enter or change the number using the line editor (typewriter keyboard).

Note For the OB, only block No. 1 is possible!

Block Type

- ”Edit”, ”Block”, ”Block Type”
- ”Load”, ”Exchange Online”, ”Block type”
- ”Online”, ”Dynamic Status Display”, ”Current display”, ”Block Type”

The type of the block to be processed is entered here (organization block OB, program block PB, function block FB).

Note Exchange online: for the organization block, only block No. 1!

Organization Block OB1

- The organization block contains the information contents of the entire user-programmed logic.
- The OB is processed cyclically. Every scan begins with the processing of OB1, network 001 and ends with the processing of the last network contained in the OB.
- PB calls, FB calls and any user-programmed logic (if required) are all present in the networks and are consecutively numbered, beginning with network 001.
- Each network can contain only one PB or FB call (with the exception of the IL).
- It makes conditional or unconditional calls to PBs, FBs and SFBs.
- It can also contain program parts in IL, LD or FBD.

Program Block PB

- A PB consists of the interfacing of networks with consecutive numbers, beginning with network 001.
- It can make conditional or unconditional calls to PBs, FBs and SFBs.
- The same PB can be called repeatedly.
- It contains program parts in IL, LD or FBD.
- A block to be called is represented in the OB (or other PB) as a rectangular box in the network (in the LD/FBD). The PB number is situated above the rectangle. In the event of conditional PB calls, the signal address of the calling condition is located to the left of the rectangle.
- A PB which is never called from any position is not processed.

Conversion of a program block into a function block:
Program blocks which have been tested and found to be suitable by the user can be easily converted into function blocks.

To do so, the following procedure should be followed:

Step 1 Set the ”Input Mode” to ”IL” (for instruction list) under “Edit”, “Blocks”.

Step 2 Also select the program block required and the first network.

Step 3 Insert another network (dummy network) in front of the first network and save the program block on the hard disk.

Step 4 Copy this program block within your station with the command sequence ”Special”, ”Copy Files” into a new function block (the source block is the PB, the target block is the new FB).
If necessary, erase the old program block.

Step 5 Call the new function block with the commands ”Edit”, ”Block”, ”Start Entry” and select network 1.
This network is still blank.
Simply input the new function block name (overwrite NW end).
If necessary, finish processing the new FB by entering the required formal parameters.

☞ **Note** The ”key macros” function is useful for entering formal parameters (refer to chapter 2.2.5).

Step 6 Save the new function block.

Upon concluding, do not forget to erase the corresponding program block calls within your user-programmed logic and to program the new function block calls.

Function Block (FB)

- Up to 999 user-specific function blocks can be parameterized (designation FB1 ... FB999).
- The FB includes a user program part in instruction list, ladder diagram, function block diagram.
- When the FB call is reached in the user-programmed logic, this is the point at which the FB is processed in run time. Thus, you can optimize the program yourself and by doing so achieve shorter scan times.
- Prior to processing a block, the formal operands of the FB will be replaced by the actual operands specified. Formal operands include up to 6 characters. The first character is a letter, the rest can be arbitrarily selected.
- An FB which is not called from any position, is never processed.

Structure

The FB consists of the declaration part (network 001) and the instruction part (succeeding networks).

Declaration part

The name of the FB and the list of the formal operands, with type specifications, is contained in the declaration part.

Table 1 Example of the declaration part of an FB

	Formal-operand	Type-possibilities	Type
NAME	: BEISP1		
IDT	: OP1	(I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R)	I L
IDT	: OP2	(I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R)	I L
IDT	: OP3	(I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R)	I L
IDT	: AUS	(I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R)	Q R
	: ***		

The line with (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) indicates the possible type specifications.

The meanings are as follows:

I	:	Input
Ix	:	Input (x = B, W, D: Byte, word,m double word)
Q	:	Output
Qx	:	Output (x = B, W, D: Byte, word, double word)
M	:	Marker (as bit)
Mx	:	Marker (x = B, W, D: Byte, word, double word)
SM	:	System marker
SMx	:	System marker (x = B, W, D: Byte, word, double word)
T	:	Timer
C	:	Counter
PAR	:	Node
Bx	:	Bit string
CNT	:	Number

L	:	The signal is an input parameter of the block
R	:	The signal is an output parameter of the block.

An appropriate element must be selected from each of these two groups. The L/R group determines whether that particular parameter should appear in the graphical representation as an input on the left side of the block, or as an output on the right side of the block. The declaration part closes with “***”, the end of the network. The formal operand name can be changed at any time, however, the block calls must be reprogrammed every time such a change is carried out.

Instruction part

The instruction part implements the algorithmic connections between the formal operands of the declaration part (in IL, LD, FBD). The formal operands must be marked with a “=” character in the instruction part. A number may be situated on the right hand side next to the instruction list; this number specifies the bracket nesting depth of the line concerned. The instruction list always ends with ”***”; the instruction part with ”:BE” (block end).

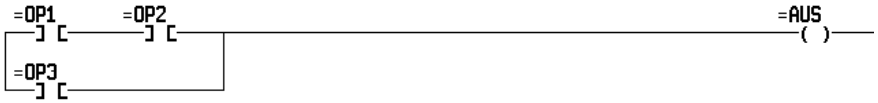
The following figures show the instruction part of the above mentioned Funktion Block ”EXP1” in IL, LD and FBD.

IL

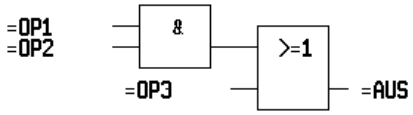
```
:O(                    1
:A     =OP1            1
:A     =OP2            1
:O     =OP3            1
:)
:=     =AUS
:***
```

Specification of the nesting depth

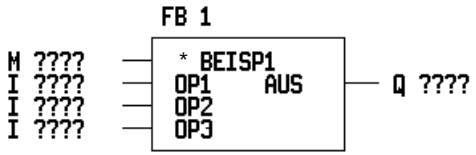
LD



FBD



Call



Conditional call of a function block in ladder diagram or function block diagram. * = conditional instruction Actual operands (M1.1, Q7.4 ...) must be entered where question marks are shown.

An FB can be called from another FB or from a PB.

The FB is represented as a rectangle in the network (in LD/FBD). Its identification (abbreviation) shows up again inside the rectangle. The input formal operands are displayed on the left side of the rectangle, while the actual operands are displayed outside on the left; a condition, if applicable, is displayed just above this. The output formal operands are situated on the right hand side of the rectangle and the actual operands are situated outside on the right hand side.

A block call in IL is effected with the operands BC or BCC.

```
      :A    M
      :BCC  FB 1
NAME  :BEISP1
OP1   :     I ????
OP2   :     I ????
OP3   :     I ????
AUS   :     Q ????
      :***
```

Conditional call of a function block in the instruction list.
Actual operands (M1.1, Q7.4 ...) are to be entered in place of the question marks.
The first line (condition) must be manually created and logically linked.

If formal operands are changed, erased or edited while editing the declaration part of the function block, all the function block calls have to be parameterized again. The global cross-reference list can be used to find out the locations of all the function blocks in the user programmed logic.

It is also possible to call a function block from the instruction part of another function block (nesting). The called function block can have as its actual operands the formal operands of the calling function block in the instruction part. In this way, actual operands can be passed on to the lowest possible nesting depth (10) during the run time of the user-programmed logic.

Program blocks which have been tested by the user and found to be suitable can easily be converted into function blocks FB1...FB999 (see PB).

Network Number

- ”Edit”, ”Block”, ”Network Number”
- ”Online”, ”Dynamic Status Display”, ”Current display”, ”Network Number”

Enter here the network (1 to 999) of the block which wish to process after “Start Entry”.
After hitting the <Return> key, you can enter or change the number using the line editor (typewriter keyboard).

Input Mode (types of presentation IL, LD, FBD)

- ”Edit”, ”Block”, ”Input Mode”
- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, “Dynamic Status Display”, <Ctrl>+<Return>, ”Presetting”, ”Input Mode”
- ”Online”, ”Dynamic Status Display”, ”Start Entry”, <Ctrl>+<Return>, ”Presetting”, ”Output Mode”

Enter here (by toggling) the type of presentation (in dedicated language) which is to be used during editing or representation.

Types of presentation

Instruction list IL

The following applies to Modsoft AKF→ Micro:
OBs, PBs and FBs are programmable in IL.
A maximum of 45 input operands and/or output operands can be defined as formal operands in the declaration part of an FB, in total not more than 64 parameters.
Only formal operands previously defined in the declaration part can be accepted in the IL of a user function block; every formal operand must be preceded by a "=" character. This clearly differentiates it from a universal symbol name.
A maximum of 24 jump markers per network are possible within a user FB. Furthermore, PB and FB calls (conditional or unconditional) are also possible in IL.
Software syntax checks are carried out in Modsoft AKF during the programming of networks in IL, so that non-permissible inputs are rejected.

Rules for structuring an instruction list

Network beginning / end

Example:

:	A..., O... or LD...	Every network must begin with the operations A, O (for logic) or LD (for arithmetic)
:	***	and end with " *** " .

Initializing Pulse

The system marker SM2 (start-up marker) serves as an initializing pulse. The value of SM2 is "1" for the first scan only.

Use: For standardizing the actual values of timers and counters and for effecting a hot start when switching on supply in hot restart and/or program start.
For information on programming memory preferred states after powering-up, see "flipflop".

Marker Bit

Example

```
Network 001
:   A   I6.1        Markers (binary intermediate results) store
:   A   I6.2        results of logic operations, for use at other positions in the
                    program.
:   O   I6.3        They must be defined before being used.
                    If nesting is required, markers or bracket operations
:   =   M5.1        must be used.
:   ***
Network 002
:   A   I6.4
:   A   M5.1
:   =   Q7.1
```

Bracket operations (only in IL)

Example

```
:   A   (           Bracket operations can also be used instead of markers.
:   A   I6.1        After a “bracket open” instruction, the next instruction
:   A   I6.2        must begin with a A or O, as for program start.
:   O   I6.3        The maximum nesting depth in bracket operations is 13.
:   )               The number of the “bracket open” operations must equal
:   A   I6.4        the number of the “brackets closed” operations.
:   =   Q7.1
:   ***
```

Block end

The last network of every block contains only the ”Block end”:

```
:   BE
```

Logic operations

Sequence

Example

```
:   A   I6.1
:   A   I6.2
:   O   I6.3
:   O   I6.4
:   =   Q7.1
:   ***
```

During the calculation of logical operation sequences, the “AND before OR” rule (as in Boolean algebra) applies. This reduces the need for brackets. The IL at left implements the operation $(I6.1 \wedge I6.2) \vee I6.3 \vee I6.4 = Q7.1$

Standardizing Operations

Instruction with standardization

Example

```
:   A   I6.1
:   S   M5.19
:   A   I6.2
:   ***
```

Operations such as S... and R... standardize the result of an operation. The result may not be used directly in the next instruction; that instruction must always begin with an A or O instruction. This also applies for the instructions A(, O(, X(, AN(, ON(, XN(.

Instruction without standardization

Example

```
:   A   M5.4
:   A   M5.5
:   =   M5.6
:   =   M5.7
:   =   M5.8
:   A   I6.7
:   ***
```

The result of logic operation A M5.4, A M5.5 is not standardized if there is a “=” sign assignment; it remains available for other assignments (repeat assignments are possible). Before executing the subsequent “AND” operation, the register is first initialized, that is to say erased, so the result of the previous logic operation is no longer available.

Connector

The connector =C is used to apply a signal to an intermediate marker or register. The register is not standardized. This reduces the number of networks. The marker type must match the operands of the other instructions within the command sequence.

Examples

Logic

```
:   A   M3.2
:   A   M4.5
:   =C  M3.5
:   O   M4.8
:   =   Q7.7
:   ***
```

Arithmetic

```
Network 001
:   LD      V12
:   ADD     MW1
:   =C      MW30
:   SUB     MW2
:   =       MW44
:   ***

Network 002
:   LD      MW30    Intermediate result from NW001.
:   ADD     MW15
:   =       MW45
:   ***
```

Memory Operations

It is imperative that the sequence of IL commands be observed. The SM2 start-up markers, giving the preferred state at power-on, are optional, as are any other instructions before the ”=” assignment. If there are no other instructions before the “=” assignment, the markers M* in the examples below can be replaced by Q7.*.

Example: SR Flip-flop, normally reset, preferred state after power-on 1

```
:   A   (
:   O   I6.1
:   O   SM2      The system marker SM2 has a “1” signal in the 1st
:   )            program scan, thus the preferred state after power-on is 1.
:   S   M7.1
:   A   I6.2     The resetting condition is processed after setting,
:   R   M7.1     for this reason this is a normally reset flip-flop.
:   =   Q7.1
:   ***
```

Example: RS Flip-flop, normally set, preferred state after power-on 1

```
:   A   I6.2
:   R   M7.2
:   A   (
:   O   I6.1     The system marker SM2 has a “1” signal during the 1st
:   O   SM2      program scan, therefore the preferred state after power-on is 1.
:   )
:   S   M7.2     The setting condition follows the resetting condition,
:   =   Q7.2     therefore this is a normally preset flip-flop.
:   ***
```

☞ **Note** Examples for normal state, preferred state after power-on 0 and preferred state after power-on 1, see page 98.

Counter

It is imperative that the sequence of the C, S, L and R commands be observed so as to avoid syntax errors. Additional instructions may be located between the commands.

Example: CTU, up-counter 0 ... 32 767

```
:   A     I6.1     Count pulse input (counts on 0→ 1 edges on I6.1).
                   The counter output is set at “1” with the first edge.
:   CTU   C10      C10, counter up.
:   A     I6.2     The setpoint is loaded with V30 upon a “1” signal occurring
:   S     C10      at I6.2.
:   LD    V30
:   A     I6.3     Resetting input (actual value and counter output are set
:   R     C10      to “0” upon a “1” signal occurring on I6.3). If actual
                   value = setpoint,
:   =     Q7.1     Q7.1 is set to ”0”.
```

Example: CTD, down-counter 0 ... 32 767

```
:   A     I6.1     Count pulse input (counts on 0→ 1 edges on I6.1).
                   The first edge sets the counter output to “1”.
:   CTD   C12      C12, counter down.
:   A     I6.2     CSW12 is loaded into the setpoint upon a “1” signal
:   S     C12      at I6.2.
:   LD    CSW12
:   A     I6.3     Resetting input (actual value and counter output are set
:   R     C12      to “0” by a “1” signal on I6.3). If actual value = setpoint
:   =     Q7.2     value, Q7.2 is set to ”0”
```

C+, C-, up-counter, down-counter -32 768 ... +32 767
This counter counts up and down.
Example:

```
:   A     I6.1     Counting pulse input (count up for 0→ 1 edges
                   at input I6.1).
:   C+    C12      C12, count up
:   A     I6.2     Counting pulse input (count down for 0→ 1 edges
                   at input I6.2).
:   C-    C12      C12, count down
:   A     I6.3     The setpoint value CSW12 is loaded by a ”1” signal
:   S     C12      occurring at I6.2
:   LD    CSW12
:   A     I6.4     Reset input (actual value and counter output are reset
:   R     C12      to ”0” by a ”1” signal occurring at I6.4).
:   =     Q7.2     When actual value = setpoint, a ”1” signal is output at Q7.2.
:   ***
```

Timers

For IL input, the operation and run-time behaviour of the timers correspond exactly to the FBD elements of the FBD or LD editor. Five time functions are available. The timing diagrams are found under “FBD elements”. The specified sequence of commands must be observed in order to avoid syntax errors. Additional instructions may be located between the commands.

Example: TP, pulse (Monoflop)

```
:   A   I6.1        Starting (input signal) of the T30 timer
:   TP  T30
:   DTB 100MS       Defines the time base in ms: 10, 100 or 1000, or
                    input 10MS, 100MS, 1000MS
:   LD  V30         Loading of the V30 constants for the setpoint value
:   A   I6.2        Resetting the timer block
:   R   T30
:   =   Q7.1        A ”1” signal is set at Q7.1 for a 0 → 1 edge at I6.1.
```

TEP, Extended pulse

```
:   A     I6.1
:   TEP   T31
:   DTB   100MS
:   LD    TSW31
:   A     I6.2
:   R     T31
:   =     Q7.1
:   ***
```

TON, ON-delay

```
:   A     I6.1
:   TON   T32
:   DTB   100MS
:   LD    TSW32
:   A     I6.2
:   R     T32
:   =     Q7.2
:   ***
```

TS, Storing ON-delay

```
:   A     I6.1
:   TS    T33
:   DTB   100MS
:   LD    MW32
:   A     I6.2
:   R     T33
:   =     Q7.3
:   ***
```

TOF, OFF-delay

```
:   A     I6.1
:   TOF   T34
:   DTB   100MS
:   LD    MW33
:   A     I6.2
:   R     T34
:   =     Q7.4
:   ***
```

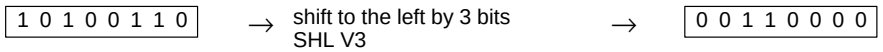
Edge detection
The following operations are able to detect pulse edges.

EDP Detects a transition 0→ 1
EDN Detects a transition 1→ 0
ED Detects transitions 0→ 1 and 1→ 0

Examples

```
EDN                 EDP                 ED
:  A     I6.1       :  A     I6.1       :  A     I6.1
:  EDN   M1.10      :  EDP   M1.10      :  ED    M1.10
:  =     M1.11      :  =     M1.11      :  =     M1.11
:  ***               :  ***               :  ***
```

Shifting (only in IL)
These operations shift the register contents by number of bits specified by the user. The bits shifted out are lost, as would any sign information shifted out. Hence, shifting is recommended for logic operations only.



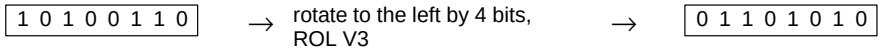
Example:
SHL, Shift to the left

```
:  A     MB1     The contents of MB1 are shifted to the left by 3 bits.
:  SHL   V3      SHL is possible only with constants.
:  =     MB1     The shifted value is located in MB1.
:  ***
```

Example:
SHR, Shift to the right

```
:  A     MB1     The contents of MB1 are shifted to the right by 5 bits.
:  SHR   V5      SHR is possible only with constants.
:  =     MB1     The shifted value is located in MB1.
:  ***
```

Rotate (only in IL)
These operations rotate the register contents by a number of bits specified by the user. The rotated bits are reinserted at the opposite side.



Example:
ROL, Rotate to the left

```
:   A     MB1     The contents of MB1 is rotated to the left by 4 bits.
:   ROL   V4      ROL is possible only with constants.
:   =     MB1     The shifted value is located in MB1.
:   ***
```

Example:
ROR, Rotate to the right

```
:   A     MB1     The contents of MB1 is rotated to the right by 5 bits.
:   ROR   V5      ROR is possible only with constants.
:   =     MB1     The shifted value is located in MB1.
:   ***
```

Increment/decrement (only in IL)
The following functions increase or decrease the contents of a byte, word, or double word by 1. This is used to change setpoint values, for example.

INC Increase value by 1
DEC Decrease value by 1

Example:

```
INC                     DEC
:   A     MW1           :   A     MB2
:   INC                 :   DEC
:   =     MW1           :   =     MB2
```

Load, transfer

LBB, LBW: bit → byte,word
The LBB and LBW operations are used to load (read into the register) a defined bit sequence (byte register, word register).
The operand of the load operation is the least significant binary signal of the bit sequence. During input, the bit sequence is checked for syntax, depending on the equipment list.

TBB, TBW: byte,word → bit
The operations TBB and TBW assign the register contents (binary) to a bit sequence.
The transfer starts with the least significant bit of the register contents. The start of the bit sequence is defined by the operand.

To avoid memory area violation and undesired storage, a function is introduced where the user specifies the width of the bit string.
The width of the bit string is defined by the command

DBB	CNT	xx	define bit sequence breadth

The operand CNT (number) can accept the following values:

LBB, TBB with bit operand

DBB	CNT	1...8	for a maximum of 8 bits in the byte register
Formal parameters in the FB declaration part			B8

LBW, TBW with bit operand

DBB	CNT	1...16	for a maximum of 16 bits in the word register
Formal parameters in the FB declaration part:			B16

LBW, TBW with byte operand

DBB	CNT	1...2	for a maximum of 2 bytes in word
Formal parameters in the FB declaration part:			B2

For IL programming, the following rule applies:
It is imperative that the DBB instruction be positioned directly after the bit string width allocation instruction associated with it. In the formal parameter declaration, the declaration for the number must follow directly after the declaration of the address, since otherwise no checking or assignment would be possible. The declaration of B2, B8 and B16 setpoint types must be followed by the declaration of a CNT setpoint type.

☞ **Note** In a user function block the maximum number of instructions and moved bits (DBB CNTxxx) $\leq$ 1.000. Σ instructions + Σ xxx $\leq$ 1.000

Example:

```
Network 1      FB3
Name: FBTEST
IDT:    BYE1  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B2 L
IDT:    EX1   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT L

IDT:    BYA1  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B16 R
IDT:    EX2   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT R

IDT:    BIT2  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B8 L
IDT:    EX3   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT L

IDT:    BIA2  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B8 R
IDT:    EX4   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT R

IDT:    BIT3  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B16 L
IDT:    EX5   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT L

IDT:    BIA3  (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) B16 R
IDT:    EX6   (I/Ix/Q/Qx/M/Mx/SM/SMx/T/C/PAR/B2/B8/B16/CNT) (L/R) CNT R
***
```

```
Network 2
:   LBW   =BYE1          Load 1...2 bytes in word register
:   DBB   =EX1           Define bit string width 1...2
:   A     VH000F
:   TBW   =BYA1          Transfer word register contents to bit string
:   DBB   =EX2           Define bit string width 1...16
:
:   =     MW12
:   ***

Network 3
:   LBB   =BIT2          Load bit string in byte register
:   DBB   =EX3           Define bit string width 1...8
:   A     MB10
:   TBB   =BIA2          Transfer byte register contents to bit string
:   DBB   =EX4           Define bit string width 1...8
:
:   =     MB20
:   ***

Network 4
:   LBW   =BIT3          Load bit string into word register
:   DBB   =EX5           Define bit string width 1...16
:   A     MW10
:   TBW   =BIA3          Transfer word register contents to bit string
:   DBB   =EX6           Define bit string width 1...16
:
:   =     MW13
:   ***

Network 5
:   LBW   I8.12          Absolute addressing
:   DBB   =EX5           Define bit string width 1...5, as I8.12...I8.16;
:   =     MW22           if LBW, LBB are absolute, CNT xx must be
                         specified.
:   BE
```

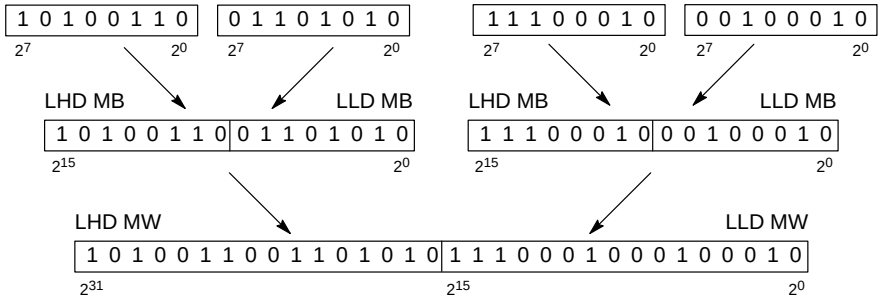
LLD, LHD: Load Low Data/ Load High Data→ word, double word

The operations LLD and LHD load:
bytes from the signal memory into the word register or
words from the signal memory into the double word register

A word consists of a high byte and a low byte. The low byte contains the least significant word part, while the high byte contains the most significant word part. Similarly, a double word consists of a high word and a low word.

The LLD command enables the user to:
define an arbitrary byte as low byte of a word or
define an arbitrary word as low word of a double word

The LHD command enables the user:
to define an arbitrary byte as high byte of a word or
to define an arbitrary word as high word of a double word



Examples:

Using LLD, the contents of MB112 (decimal 100) are loaded into the low word of marker word MW223.

:LD	V0		
:=	MB113		00000000
:LLD	MB112		01100100
:LHD	MB113		00000000
:=	MW223	00000000	01100100

Using LLD, the contents of MW212 (dezimal 32 612) are loaded into the low word of the marker double word MD200.

```
:LD    V0
:=     MW213                              00000000  00000000
:LLD   MW212                               01111111  01100100
:LHD   MW213                              00000000  00000000
:=     MD200        00000000  00000000     01111111  01100100
```

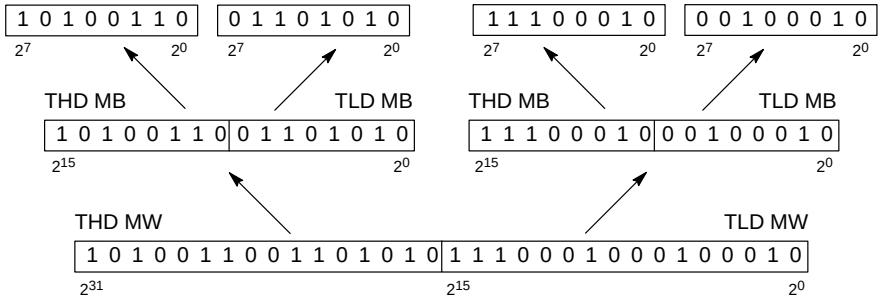
TLD, THD: Transfer Low Data/ Transfer High Data→ word, double word

The TLD and THD operations transfer:
bytes from the word register into the signal memory or
words from the double word register into the signal memory

A word consists of a high byte and a low byte. The low byte contains the least significant word part, while the high byte contains the most significant word part. Similarly, a double word consists of a high word and a low word.

The TLD command enables the user to:
transfer the low byte of a word to an arbitrary byte, transfer the low word of a double word to an arbitrary word

The THD command enables the user to:
transfer the high byte of a word to an arbitrary byte, transfer the high word of a double word to an arbitrary word.



Examples

Using the TLD and THD operations, the contents of marker word MW219 (decimal 1 024) are transfered into the low byte MB117 (0) and MB118 (4).

```
:LD    MW219      00000100  00000000
:TLD   MB217                00000000
:THD   MB118                00000100
```

Using the TLD and THD operations, the contents of marker double word MD17 (decimal 2 097 168) is transferred into the MW220 (low word 16) and MW221 (high word 32).

```
:LD    MD17          00000000 00100000 00000000 00010000
:TLD   MW220 (low word)                 00000000 00010000
:THD   MW221 (high word)                00000000 00100000
```

The analog value of IW5.1 (15 928) is loaded into the marker double word MD18. Then the contents are loaded into low word MW69 and high word MW70.

```
:LD    V0
:=     MW68                             00000000 00000000
:LLD   IW5.1                             00111110 00111000
:LHD   MW68                             00000000 00000000
:=     MD18          00000000 00000000  00111110 00111000
:LD    MD18          00000000 00000000  00111110 00111000
:TLD   MW69                              00111110 00111000
:THD   MW70                             00000000 00000000
```

Jumps (only in IL)

Entering jump marks

When programming a network in IL, conditional and unconditional jumps may be used. The target of the jump instruction must have the same mark that is entered at the jump call. Jump marks are introduced by means of an “=” character, just like formal parameters.

Jump targets should consists of a loading instruction, for arithmetic, or the beginning of an operation, as only this initializes the register. Since the register is initialized after assignment, there are effects on the jump target (see example JT, conditional jump)

Example: JI, unconditional jump

```
          :   A   I6.1          Jump marking targets are entered in
          :   A   I6.2          front of the colon. To do this, move the
          :   =   Q7.1          cursor there (using the cursor keys or the
          :   JI  =AM0012       mouse). Jump markers consist of up to six
          :   ...               characters, with the first character being a capital
AM0012    :   A   Q7.1          letter. The other characters may be numerals,
                                special characters or letters, but no
          :   ...               blank spaces.
```

Example: JT, conditional jump with initialization

```
          :   A   I6.3
          :   A   I6.4
          :   JT  =AM0021       Jump is executed on ”1” signal
          :   ...
          :   =   M4.1
AM0021    :   A   I6.5          with initialization
          :   ...
AM0021    :   =   Q7.2          without initialization
```

☞ **Note** It is not possible to represent jumps in LD or FBD.

NOP Instruction (only in IL)
The ”NOP” instruction is a dummy operation. (No operation)

☞ **Note** IL’s with ”NOP” instructions cannot be represented in LD or FBD.

FREE instruction (only in IL)
FREE is a dummy variable in the instruction list. In contrast to NOP, which can stand for any instruction, FREE only applies to inputs of memories, timers, and counters.

Example:

without dummy			with dummy			
:	A	I6.1	:	A	I6.1	In this network, the timer
:	TEP	T31	:	TEP	T31	block cannot be reset.
:	DTB	100MS	:	DTB	100MS	
:	LD	V10	:	LD	V10	
:	A	I6.2	:	FREE		
:	R	T31	:	=	Q7.1	
:	=	Q7.1	:	***		
:	***					

Arithmetic Functions

Loading

Arithmetic functions are always loaded with ”LD”.

Constants

The constants V, VO, and VH are loaded with the operation ”LD”.

Arithmetic

In general, all arithmetic operations must start with a load operation.

Example

```
:   LD    V12
:   ADD   MW1      Several successive arithmetic commands are possible
:   SUB   MW12
:   =     MW20
:   ***

:   LD    MW22     There is no ”multiply-before-dividing” rule in arithmetic.
:   MUL   MW33     The expression <MW11> + <MW22> x <MW33> =
:   ADD   MW11     <MW44> must be programmed as shown at left.
:   =     MW44
:   ***
```

Conversion takes place between different data types, see ”Conversion”.

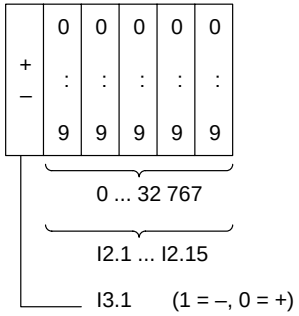
Parentheses may not be used in arithmetic commands. If intermediate results are to be used later, intermediate markers should be introduced.

Overflow markers are set up for arithmetic operations. These are listed in the User Instructions ”Short Form Guide”. Each type of operand has its own overflow marker.

If system markers are to be used, they must be scanned immediately after the arithmetic operation.

Calculation of the two's complement

For BCN switches with separate sign, the two's complement must be provided for negative values.



corresponding IL:

```
      :   LBW   I2.1
      :   DBB   CNT15
      :   =     MW1
      :   A     I3.1
      :   JF    =END
      :   LD    V0
      :   SUB   MW1
      :   =     MW1
END   :   LD    MW1
      :   =     MD1
```

Comparators

The comparators can only be used with operands of the same data type, in accordance with the conversion rules.

If representation in LD/FBD format is dispensed with, then arithmetical instructions are permissible between the loading operations and the comparison operations. Several instructions for bit operations may be placed before the “=” assignment.

= equal			> greater than			< smaller than		
:	LD	MW30	:	LD	MB30	:	LD	V20
:	==	MW31	:	>	MB31	:	<	MW51
:	=	Q7.1	:	=	M5.22	:	=	Q7.3
:	***		:	***		:	***	

<> not equal			≥ greater than/equal			≤ smaller than/equal		
:	LD	MW32	:	LD	MD40	:	LD	V21
:	<>	MW33	:	>=	MD41	:	<=	MW52
:	=	Q7.4	:	=	M5.23	:	=	Q7.6
:	***		:	***		:	***	

Conversion

Note There are two types of processing and various types of operands. The possible conversions are executed automatically. The user may implement different types of operands and processing in IL. The following functional description is used to explain the way values are generated in the dynamic status display.

Conversion into other types of operands

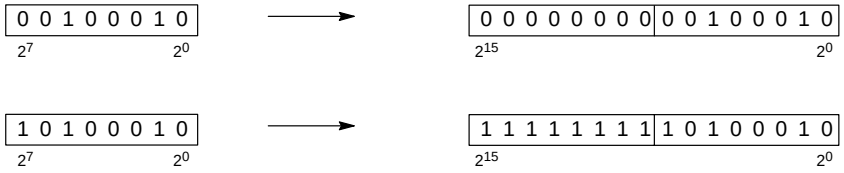
- Logic: Operand types, unsigned
  bit, byte, word, and double word

- Arithmetic: Operand types, signed
  byte, word, and double word

Conversion always proceeds from the smaller type of operand to the larger type.

- Logic: unsigned operand types; the unused more significant bits are padded with ”0” during conversion.
  Example: Conversion from MB to MW

- Arithmetic: Signed operand types; the unused more significant bits are padded during conversion, depending on the sign.
  In case of positive values, the more significant bits are padded with ”0”.
  In case of negative values, the more significant bits are padded with ”1”.
  Examples: Conversion from MB to MW.



Conversion between two types of processing (logic→ arithmetic, or arithmetic→ logic)

- The conversion from logic to arithmetic is always made without sign.

- The conversion from arithmetic to logic is always made with sign.

Ladder Diagram LD

OBs,PBs and FBs are programmable in LD.
A maximum of 16 signals per network are interlockable in parallel as inputs in ladder diagram and 7 signals plus output are interlockable serially. There is only one output per LD network, and may not be negated. The valence of the output signal can be applied to a maximum of 16 contacts.
Jumps are not possible in LD, only block calls. After processing the called blocks, program execution is continued behind the calling position. In LD, only one program block call or function block call may be in a network.
When programming a network in LD one should note that the elements can only be entered at specific cursor positions.

The following symbols are available in programming a network in LD:

	Normally opened contact (queries 1-Signal)
	Normally closed contact (queries 0-Signal)
	Connector
=2k+1	Exclusive OR block
	Connection of parallel conductors
	Continuation in parallel without contact
	Output (no storage)
FBD elements	further elements available in LD and FBD (memories, timers, counters, and comparators) may be selected and called up.

Note regarding FBD elements: When programming in LD, it is normally only permitted to program one FBD element per network. The exceptions to this, are several timers, counters and flipflops opposite each other in the first serial path.

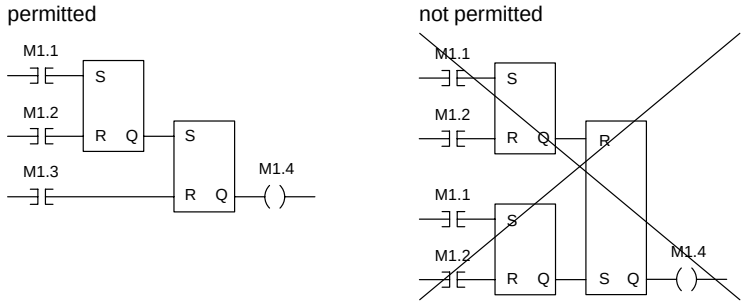


Figure 2 Input of FBD elements in a ladder diagram

Function Block Diagram FBD
OBs and PBs are programmable in FBD.
The ladder diagram permits a maximum of 6 FBD elements horizontally and 46 input signals per network, as well as up to 12 FBD elements, each with two vertical inputs. There is only one output per FBD network, and it may not be negated.
The valence of the output signal can be applied to a maximum of 16 contacts.

There are no jumps in FBD, only block calls.
After processing the called blocks, the program returns to the calling position.
In FBD, only one program block call or function block call is permitted per network.

When programming a network in FBD one should note that the elements can only be entered at specific cursor positions.

The following symbols are available for programming the network in FBD:

&	AND block
>=1	OR block
=2k+1	Exclusive OR block
——|	Input (queries 1-Signal)
——o|	Negated input (queries 0-Signal)
—> >—	Connector
FBD elements:	other elements, which may be called in LD and FBD (memories timers, counters, and comparators) can be selected and called.

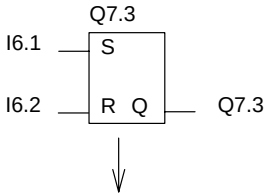
FBD elements

The FBD elements are standard blocks which you can select in the FBD editor or LD editor.

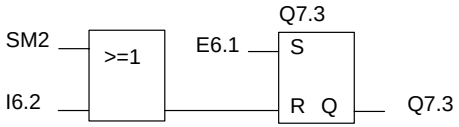
All AKF functions (FBD elements) are non-volatile, that is to say all signals and words remain unchanged after power loss and recovery. To select some particular initial state characteristics (initialization upon power being restored), the SM2 system marker must be applied to the initialization input of the function.

The FBD elements are illustrated below, using examples and timing diagrams where necessary.

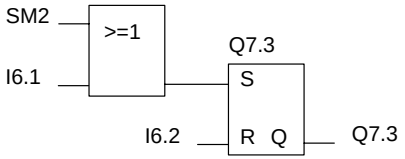
SR Flipflop normally reset



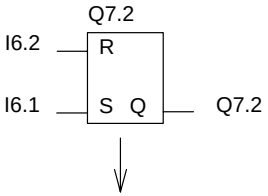
Initial state 0 (normal position)



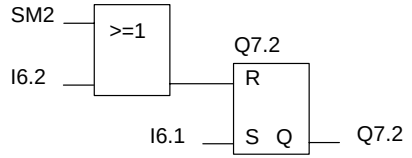
Initial state 1



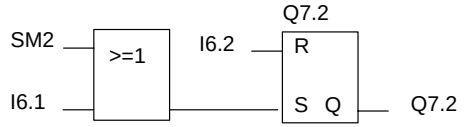
RS Flipflop normally set



Initial state 0



Initial state 1 (normal position)



Counter

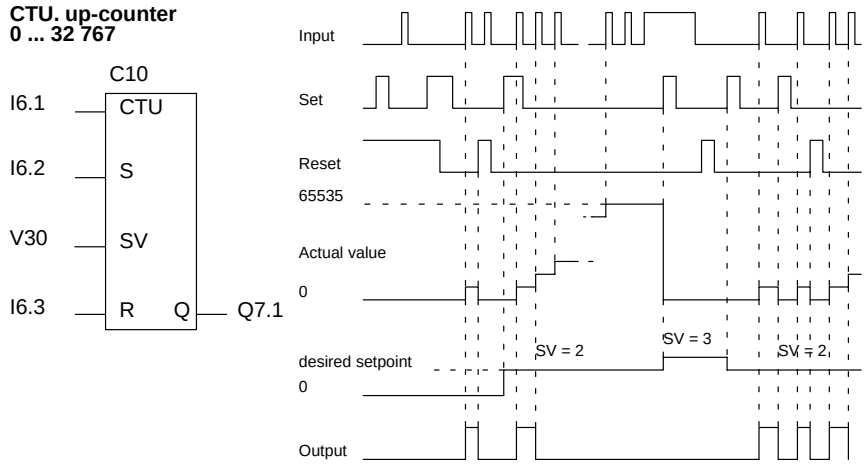


Figure 3 FBD element and timing diagram of up-counter

The actual value and the output ”Q” are set to ”0” by a ”1” signal at the reset input, ”R”.

A 0 → 1 transition at the set input ”S” causes the setpoint ”SV” to be taken over into the setpoint register. The actual value becomes ”0”. The setpoint/actual value comparison is only done when a setpoint has been taken over at least once.

The following setpoints value (”SV”) are permitted: V, MW, CSW

The output ”Q” is set to ”1” while the actual value lies between 0 and the setpoint value.

A 0 → 1 transition at the ”CTU” input increment the actual value by 1 (up to max. 65 535) and compares it with the setpoint value.

The actual and setpoint values in the counter (CAW, CSW) can be controlled via the online list and/or the initial values. Counter actual and setpoint values may be determined by means of a load instruction, and transferred into other operand areas.

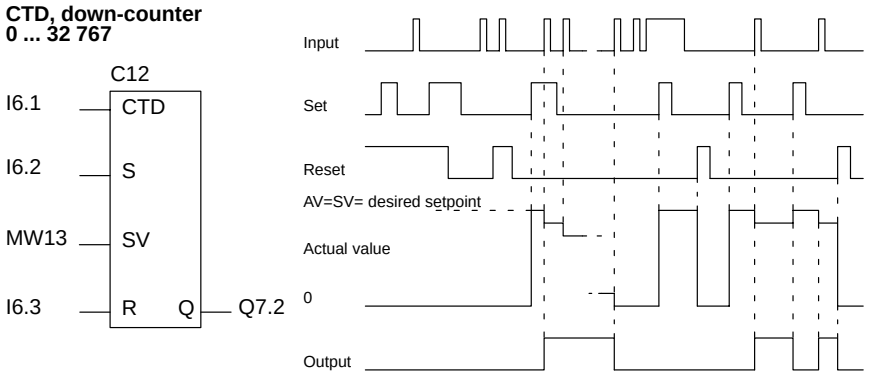


Figure 4 FBD element and timing diagram for down-counter

When a ”1” appears at the reset input ”R”, the actual value and the output ”Q” are reset to ”0”.

A 0 → 1 transition at the set input ”S” causes the setpoint value ”SV” to be taken over into the setpoint register. The actual value is made equal to the setpoint value. A setpoint / actual value comparison is only carried out if a setpoint value has been taken over into the setpoint register at least once.

Permissible values for the setpoint ”SV” are: V, MW, CSW

The output ”Q” is set to ”1” while the actual value lies between 0 and the setpoint value.

A 0 → 1 edge transition at the ”CTD” input decrement the actual value by 1 (down to 0) and compares it to the setpoint value.

The counter actual and setpoint values (CAW, CSW) can be controlled by means of the online list and the initial values. Their values may be determined by a load instruction, and they may be transferred into another operand area.

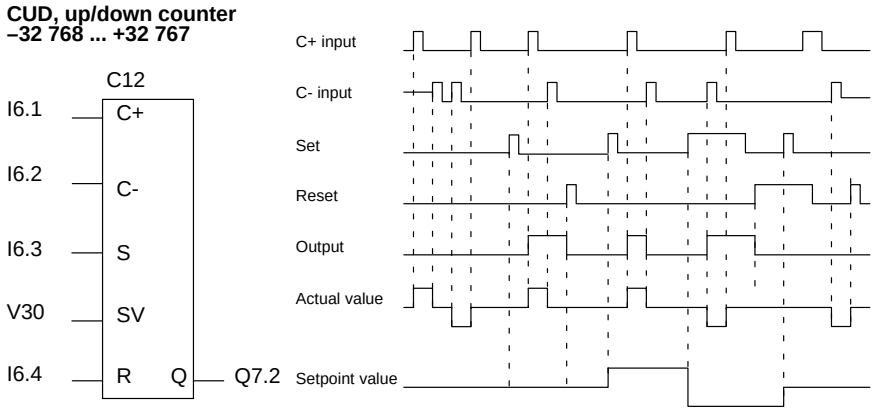


Figure 5 FBD element and timing diagram for up-counter, down-counter

A ”1” signal at the ”R” reset input sets the actual value, the ”Q” output and the SM 26 system marker to ”0”.

A 0 → 1 edge transition at the ”S” set input causes the ”SV” setpoint value to be taken into the setpoint value register. The actual value remains unchanged. A comparison of setpoint value and actual value is not made until at least one setpoint value was taken over.

The following setpoints value (”SV”) are permitted: V, MW, CSW

The ”Q” output is set to ”1” as soon as the actual value is greater than or equal to the setpoint value.

A 0 → 1 edge at the C+ input increases the actual value by 1.
A 0 → 1 edge at the C– input decreases the actual value by 1.

If in the next counting operation of the counter the actual value is greater than 32 767 or less than -32 768, then the system marker SM26 (actual value overflow) is set to ”1”. The system marker SM26 is initialized with another counter in the user program.

The actual values and setpoints in the counter (CAW, CSW) can be controlled by means of the online list. The actual value and setpoint counters can be addressed and their contents transferred to another operand area by means of a load instruction.

Timers

TP, Timer pulse

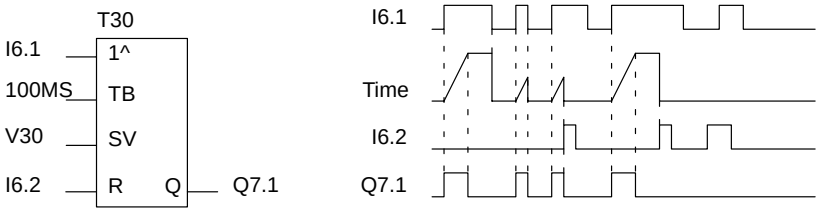


Figure 6 FBD element and timing diagram pulse

TEP, extended pulse

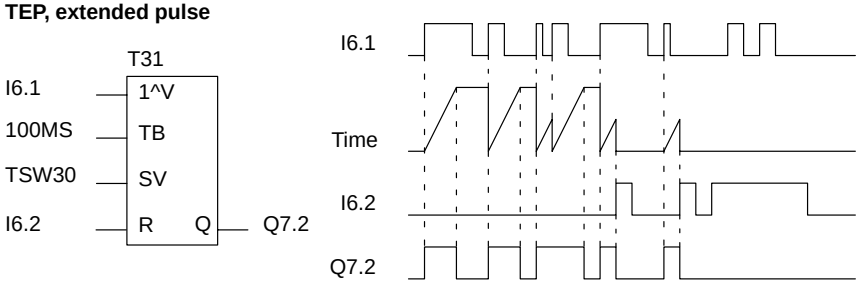


Figure 7 FBD element and timing diagram extended pulse

TON, ON-delay

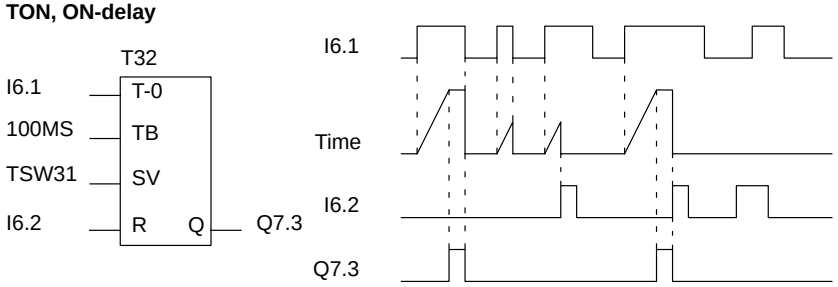
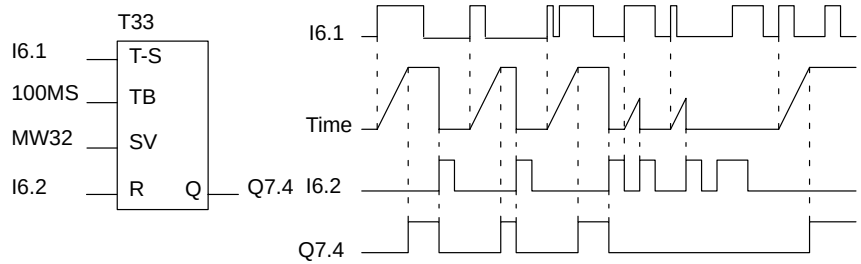


Figure 8 FBD element and timing diagram ON-delay

TS, storing ON-delay



Reset only possible with ”R”

Figure 9 FBD element and timing diagram storing ON-delay

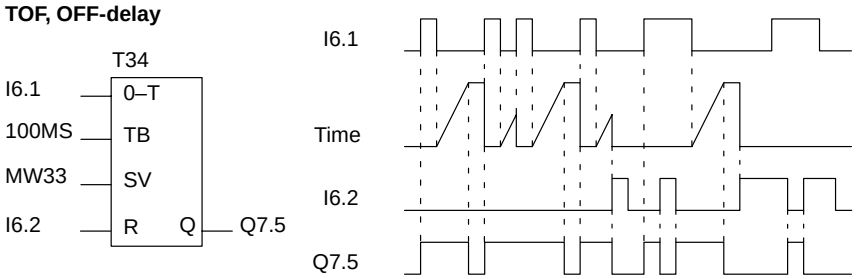
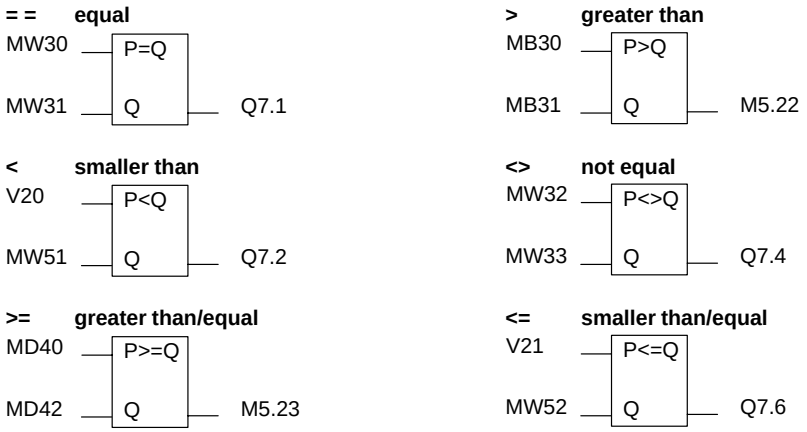


Figure 10 FBD element and timing diagram OFF-delay

Comparators

The contents of two bytes or words are compared. Six different comparators can be called.



If the condition described in FBD element applies, the output ”Q” = 1.

Edge detection
3 FBD elements are available for pulse edge detection.

EDP, edge detection 0 –> 1

EDN, edge detection 1 –> 0

ED, edge detection 0 –>1 and 1 –>0

When the condition described in the FBD element is met the output for a scan ”Q” = 1.

The intermediate marker (here M1.10) and the output marker (here M11.1) may not be identical.

During network programming in FBD, the Modsoft AKF software carries out syntax checks to ensure that illegal entries are rejected.

Further information regarding the programming of networks in function block diagram can be found in the FBD editor and in the ”Correction mode” of the FBD editor.

ADU 205

☞ **Note** Programming for ADU 205:
On the diskette, in the system called ”Example”, there is a function block, FB2, available as an application.

In order to copy FB2 from the example system into another, proceed as follows:

Step 1 Under ”SeTup”, ”Plant” enter the system in which you want to implement the ADU 205

Step 2 Select ”SeTup”, ”PLC Station”, ”PLC Station Name” and specify a station

Step 3 Select the menu point ”Special”, ”Copy Files”

Step 4 Enter the ”source” as follows: A:\Example\FB2 (Note that A:\ is the disk drive in which your MICROAKF diskette is found; use B:\ if applicable)

Step 5 Leave the ”target” empty; the files will be copied into the station selected under Setup

Addressing

- ”Edit”, ”Block”, ”Addressing”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”, ”Addressing”
- ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, ”Dynamic Status Display”, <Ctrl>+<Return>, ”Presetting”, ”Addressing”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..
- ”Online”, ”Dynamic Status Display”, ”Start display”, <Ctrl>+<Return>, ”Presetting”, ”Addressing”
- ”SeTup”, ”PLC Station”, ”Addressing”

All inputs, outputs, markers and marker words can be addressed absolutely (e.g. I6.15, M 1.17). They can also be addressed symbolically, if a data block has previously been generated.
By toggling, one moves between addressing modes.

Output Monitoring

- ”Edit”, ”Block”, ”Ouput monitoring”
- ”Edit”, ”Block”, ”Start Entry”, existing block, <Ctrl>+<Return>, ”Presetting”, ”Output Monitoring”
  ”Edit”, ”Block”, ”Start Entry”, <Ctrl>+<Return>, ”Dynamic Status Display”, <Ctrl>+<Return>, ”Presetting”, ”Output Monitoring”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..
- ”Online”, ”Dynamic Status Display”, ”Start display”, <Ctrl>+<Return>, ”Presetting”, ”Output Monitoring”
- ”SeTup”, ”Station”, ”Output Monitoring”

You can toggle between ”on” and ”off”.

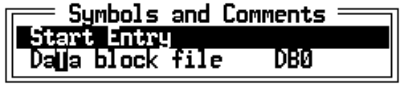
Address allocation is monitored during the editing process if the output monitoring option is ”on”. If an output address already used is entered a second time, a warning message is given.

The programmer is thus alerted to any multiple allocations. The message must be acknowledged.

3.3.3 Symbols and Comments

-”Edit”, ”Symbols and Comments”

This software function is used for programming the data blocks. The data block contains symbolic names and comments for absolute addresses. You can select and call the following menu lines in the ”symbols and comments” pulldown menu:



Start Entry

- ”Edit”, ”Symbols and Comments”, ”Start Entry”

First, enter the required number under “Data Block File”.

The DB is automatically generated if it does not yet exist. Editing of the data block begins by actuating the <Return> key.

Data block DB
The general description of the data block is contained in the User Instruction “MICROAKF for Beginners”.

- absolute addresses are provided with symbolic names and comments in order to clearly illustrate the connection between the address (input/output, marker, etc.) and its technical functions.
- The text of the symbolic names and comments is filed in the data block under the current station name.
- Activation of the DB by setting the addressing to ”SYM” (from ”ABS”).
- Documentation of the DB is under ”Print”, ”Symbols and Comments”
- Standard data block DB0 for the system markers (bit, byte and word)
- The system markers are automatically copied into every new DB.

Data block editor

The data block DB contains the hardware address assignments of symbolic names and comments. This assignment is effected with the aid of the data block editor.
These assignments determine how inputs, outputs, markers, marker words etc. are used within your program. The editor has various tools (menus, key macros and the line editor) that assist in making the data block input as simple and user-friendly as possible.

Input Symbol, Comment

 - ”Edit”, ”Symbols and comments”, ”Start Entry”

The data block is entered using the data block editor:
The signal table, symbol table and comment table of the first inputs appear when “Start Entry” is selected.

You can assign symbols (a maximum of 8 characters) and comments (a maximum of 40 characters) to the displayed hardware addresses by using the following special keys, the line editor and the key macros.

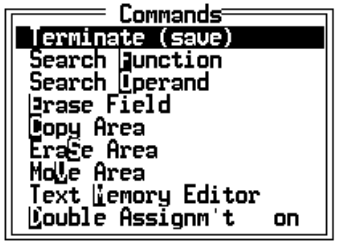
Special keys:

< ← >, < → >, <↓>, <↑>	move cursor
<Del>	erase character under cursor
<backspace>, <⇐ >>	erase character to the left
<Return>	termination of the input of a line or further menu
→| (tab)	next input field
shift |← (tab)	previous input field
<PgUp>	previous page
<PgDn>	next page
<Crtl>+<PgUp>	scroll line upwards
<Crlt>+<PgDn>	scroll line downwards
<Home>	jump to the upper edge of the screen
<End>	jump to the bottom edge of the screen
<Ins>	Insert mode on/off
<F1> to <F8>	text memory
<Esc>	abort without saving

Under certain conditions you may see an asterix (*) in front of a signal.
This signifies that this hardware address has not been entered into the equipment list. You can assign symbols and comments to these signals for later purposes, however you cannot program them in the block editor.

☞ **Note** All changes which have been carried out in this editor using its functions only apply to the data block and not to the OB, PB, FB.

With pressing <Ctrl>+<Return> you can call the following pulldown menu:



Terminate (save)

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Terminate (save)”

The newly entered or modified symbols and comments of the hardware addresses are stored in the preset data block under the preset station. The editor is terminated at the same time.

Search Function

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Search Function”

If you are looking for a certail signal or an already entered symbol within a data block, then this function should be used.
A window appears on the screen; the search word must be entered here. The input is checked for errors. It is possible to abort using the <Esc> key or by inputting a blank space. Search words which are found are displayed on the first line of the editor.
If a search symbol is not found, the cursor remains in its previous position and the search is broken off after an appropriate message has been displayed.

Symbols and comments can be entered for the following operands:

I	Inputs bit	SM	System marker bit
Q	Outputs bit	SMB	System marker byte
IB	Inputs byte	SMW	System marker word
QB	Outputs byte	SMD	System marker double word
IW	Inputs word	T	Timer
QW	Outputs word	TAW	Timer actual value
ID	Inputs double word	TSW	Timer setpoint value
QD	Outputs double word	C	Counter
M	Marker bit (center)	CAW	Counter actual value
MB	Marker byte	CSW	Counter setpoint value
MW	Marker word		
MD	Marker double word		

Searching Operands

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Search Operand”

An operand can be selected using:
Reference characters, by selecting the menu line of the cursor keys and pressing <Return> or by using the mouse.
The first available hardware address of the chosen operands is shown in the first screen line.
See also search function for further information.

Erase Field

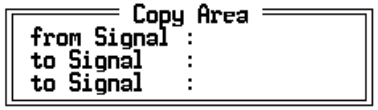
- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Erase Field”

The input field on which the cursor was positioned before the menu was called is erased.
For further information, see Erase area.

Copy Area

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Copy Area”

It is possible to assign symbols and comments to other signals.
After selecting this function, a window appears on the screen.



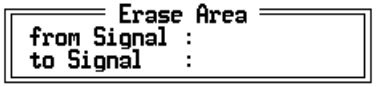
the first signal to be copied
the last signal to be copied
target address

The inputs can be specified absolutely as well as symbolically.
It is better to specify the hardware addresses during the copying procedure when programming symbolically.
Having been copied to a different position, the symbols now exist in two places; this must corrected.
Double symbols should be avoided. Therefore all the double assignments should be removed.

Erase Area

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Erase Area”

It is possible to erase symbols and comments in one block.
After selecting this function a window appears on the screen.



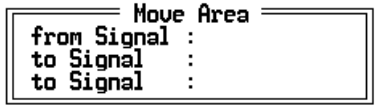
the first signal to be erased
the last signal to be erased

The inputs can be entered absolutely as well as symbolically.

Move Area

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Move Area”

It is possible to shift symbols and comments as a block.
After selecting this function a window appears on the screen.



the first signal to be shifted
the last signal to be shifted
target address

The inputs can be effected absolutely as well as symbolically.
It is better to specify the hardware addresses during moving when programming symbolically.

Text Memory Editor

- ”Edit”, ”Symbols and Comments”, ”Start Entry, <Ctrl>+<Return>, ”Text Memory Editor”

Frequently used comments need not be entered in full every time. The line editor can be used to assign any texts desired to the keys <F1> to <F8>.
The entry can have a maximum of 40 characters. Positioning in the data block editor is effected at the current cursor position by pressing the appropriate function key.

Note Key macros (permanent memory) can also be used instead of the function keys.

Double Assignment on/off

- ”Edit”, ”Symbols and Comments”, ”Start Entry”, <Ctrl>+<Return>, ”Double Assignment on/off”

If the ”on” option has been selected, a check is carried out for every symbol entered to determine whether it has already been assigned. If it has already been assigned, you receive an error message with the double symbols. Double assignments should be avoided. However, if you wish to describe hardware addresses which have not yet been entered into the equipment list, you can switch off the message on the screen using this option. The state (on/off) can be seen from the status line.

Data Block File

- ”Edit”, ”Symbols and Comments”, ”Datablockfile”
- ”Edit”, ”Compile (Data Blocks)”
- ”SeTup”, ”PLC-Station”, ”Datablock Number”

Enter the number of the data block which you would like to process here. The DB is automatically generated if it does not already exist.

3.3.4 Equipment List

 - ”Edit”, ”Equipment List”

This editor is used for setting up the hardware configuration and presettings for the operation of your station.

The equipment list editor is used to enter or change I/O module types plugged into the subracks.

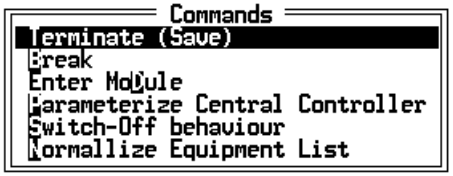
After selecting this function a window appears in which you can edit the equipment list.

Edit Equipment List

 - ”Edit”, ”Equipment List”, <Ctrl>+<Return>

In this menu the I/O modules are allocated to the corresponding slots in the subrack.
In addition, a comment with up to 40 characters for every entry can be entered using the line editor.
Available are
a) a pulldown menu
b) special keys.

a) Pressing <Ctrl>+<Return>in the module column brings up the following menu:



b) Special keys

<Home>	Jump to the upper edge of the screen
<End>	Jump to the lower edge of the screen
<Esc>	Exit the editor
<Del>	Delete the character under the cursor
<backspace>, <⇐ >	Delete the character to the left of the cursor
< ← >, < → >, <↓>, <↑>	Move cursor
→| (tab)	Next input field
shift |← (tab)	Last input field
<Return>	Edit comment

Equipment List / Terminate

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Terminate (save)”

The editing is terminated by selecting this function. The input equipment list is checked to ensure that it is correct and saved on the user drive.

Equipment List / Break

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Break”

If this function is selected and acknowledged, editing is broken off. All changes are lost after the user has acknowledged. (corresponds to <Esc> key)

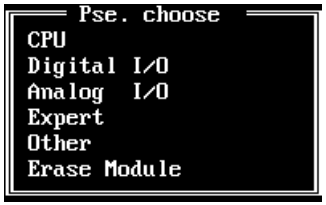
Enter Module

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Module”

This function is used for entering modules in the equipment list.

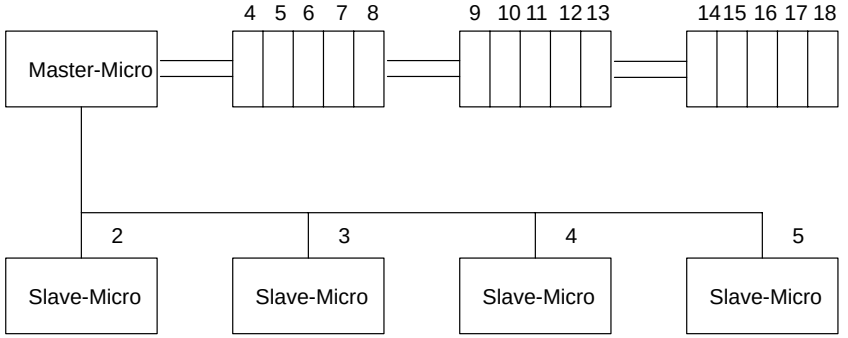
Up to 18 slots may be entered into the equipment list.

The following window will be initially displayed:



Caution Please note,

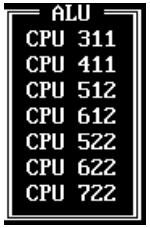
- the Micro Central Processing Unit occupies slots 1-3 in the Equipment List and hence on these no I/O modules can be inserted.
- the Micro Central Processing Unit itself offers I/O points
- that 4 Slave-Micros (serial extension) can be configured and those have to be entered into the EQL List in the slots 2-5,
- that the Slave-Micros must be programmed with Modsoft Lite,
- that in case the Slave-Micros 4 and 5 have been configured, on those slots no I/O module can be plugged and
- that the system marker 49/50 (INT1/INT2) and the System Marker Word 7 (counter CNT) can only be used with the Master-Micro.



Modules CPU

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Modules”, ”CPU”

The following CPUs can be entered:



Digital I/O modules

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Module”, ”Digital I/O”

The following digital input/output modules may be entered:

Analog I/O module

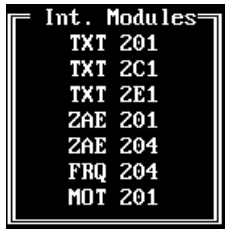
- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Module”, ”Analog I/O”

The following analog input/output modules may be entered:

Expert

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Module”, ”Expert”

The following intelligent functional modules may be entered:



Other modules

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Module”, ”Other”

The following modules may be entered:

Erase module

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Enter Modules”, ”Erase Module”

This function is used for erasing an item in the equipment list.

Parameterize Central Controller

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Parameterize Central Controller”

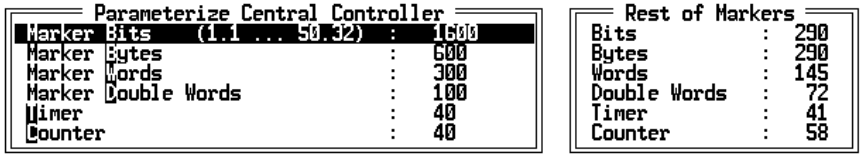
The ALU is the control system of the programmable controller. It executes the individual instructions of a program according to the rules of the basic software.

This includes:

- organizing the reading in of external data and signals into the signal memory
- processing this data and executing calculations
- continually saving processing results in the signal memory
- organizing the output of the results

When you parameterize the ALU, you adjust the number of markers, timers and counters. Return by pressing the <Esc> key.

The following entries are required:



Note The sum total of all the markers, I/O modules, software timers and software counters should not exceed 3970 bytes. Check this by looking at the right window where the memory remaining for allocation is shown. In this way, you can decide for yourself where and how to allocate the markers.

Setting the bit markers
The number of markers can range from 0 to 3970 and is entered in complete groups of 32. If the remaining memory permits, your input is rounded up into the next group of 32.
The factory presetting is 1600 (1.1 ... 50.32).

Setting the byte markers
The number of byte markers can range from 0 to 3970.
The factory presetting is 600.

Setting the word markers
The number of word markers can range from 0 to 1985.
The factory presetting is 300.

Setting the double word markers
The number of double word markers can range from 0 to 992.
The factory presetting is 100.

Setting the timers
The number of timers can range from 0 to 567. The factory presetting is 40.

Setting the counters
The number of counters can range from 0 to 794.
The factory presetting is 40.

Switch-off behaviour

- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Switch-off behaviour”

In the event of a power failure on the I/O bus, the switch-off behaviour determines the reaction of programmable controller outputs.

The switch-off behaviour can be separately defined for the individual slots.

A menu appears in which you can toggle between ”Sig. Mem. contents” and ”dominant 0” for any of the 18 slots.

Normalize Equipment List

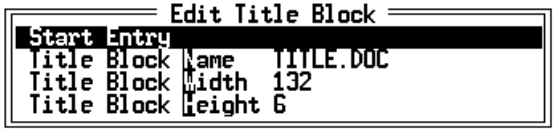
- ”Edit”, ”Equipment List”, <Ctrl>+<Return>, ”Normalize Equipment List”

All the values in the equipment list are set to the values given upon delivery. All I/O peripheral entries are erased.

3.3.5 Title Block

- ”Edit”, ”Title Block”

The following functions are available in the “Title Block” pulldown menu:



The entry is carried out via a Title Block Editor.

Start Entry

- ”Edit”, ”TItle Block”, ”Start Entry”

After the title block name, width and height have been specified and <Return> has been pressed, editing of the of the title block file is commenced. This title block may be used in all printing operations. If the title block file does not yet exist, it is automatically generated using the functions which are contained in the pulldown menu.

Title Block Editor

- ”Edit”, ”TItle Block”, ”Start Entry”

Use this editor to enter the title block for all station documentation lists for outputting to the printer or screen.

The title block editor uses special keys and graphic characters. Instead of designing your own title block file, you can also use the standard title block file, into which you will only have to enter the current information. This standard title block file is automatically copied onto the user drive under your station.

You can edit any text in the displayed framework using the following special keys.

Special keys

<Del>	erase character under cursor
<backspace>, <⇐ >	erase character on the left
Cursor keys	move cursor
<Return>	concludes a line input
→| (tab)	next input field
shift |← (tab)	previous input field
<PgUp>	shift image to the left
<PgDn>	shift image to the right
<Ins>	insert on/off

Here is the printout of the example title block:

Graphic characters:
You can also use graphic characters when programming a new title block. The latter are generated via the keyboard using the <Alt> key and the numeric key block.
Press the <Alt> key and hold it down. Input a three digit number using the keys from the numeric key block and then release the <Alt> key.
The character appears on the screen at the current cursor position.

The following grahic characters are available (the 3-digit numbers are on the left and the character on the right)

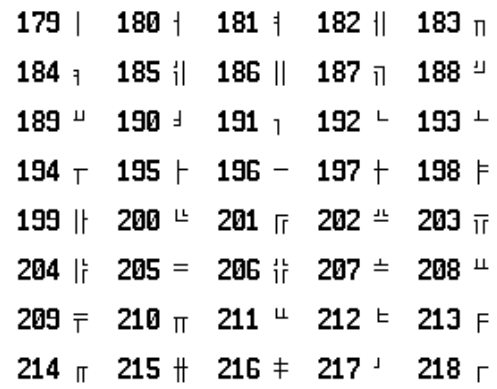


Figure 11 Available Graphic Characters in the Title Block Editor

Title Block Name

- ”Edit”, ”TItle Block”, ”Title Block Name”

Enter the name of the title block file which you wish to process here.

The standard title block file is called the ”TITLE.DOC”. The extension ”.DOC” is normally specified for title block names.

Title Block Width

- ”Edit”, ”TItle Block”, ”Title Block Width”

Enter the required width of the title block here.
64-132 characters are permissible.

The width cannot be subsequently changed.

Title Block Height

 - ”Edit”, ”TItle Block”, ”Title Block Height”

Enter the required number of lines of your title block here (1 to 12 lines are permissible).

The height cannot be subsequently changed.

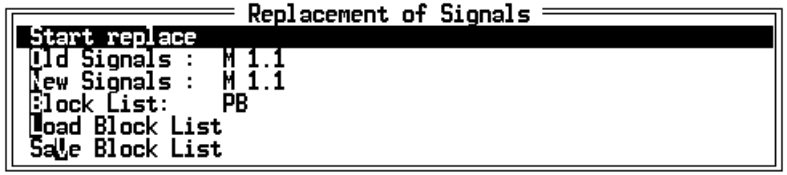
3.3.6 Replace Signals

- ”Edit”, ”Replace Signals”

By means of this function, you can in your user program several signal addresses

- replace by other signals (e.g. I6.1 to I6.6) or
- exchange signals (e.g. I6.1 to I8.1 and I8.1 to I6.1)

In this pulldown menu you can select and call the following functions:



You can display signal lists. In this way, you can carry out replacing can be executed by modules, for example. Furthermore, blocks can be replaced by means of the block list. A ”*” in the block list causes the replacement in the complete station.

Note Data block
Every edited block has one assigned data block. All assigned data blocks are automatically changed, i.e.
a) symbols and comments of addresses to be replaced are assigned to the new addresses
b) the entry of the addresses to be replaced is erased
If further data blocks are to be replaced, they must be individually specified in the block list.

☞ **Note** Signal exchange
To can exchange signal A with signal B, proceed as follows:

Step 1 ”Old signal”: A, ”New signal”: Auxiliary contact, Start replacing

Step 2 ”Old signal”: B, ”New signal”: A, Start replacing

Step 3 ”Old signal”: Auxiliary contact, ”New signal”: B, Start replacing

Start Replace

- ”Edit”, ”Replace Signals”, ”Start Replace”

When all required data have been entered (refer to pulldown menu lines below), the software first performs a safety check. If the new signals are unknown in the data block, replacing is started. If signals and comments do already exist in the data module, the replacement must first be acknowledged.

☞ **Note** Repeated replacing with the same signals erases symbols and comments.

Old and/or new signal

- ”Edit”, ”Replace Signals”, ”Old signal”
- ”Edit”, ”Replace Signals”, ”New signal”

Specify the desired address ranges (I, Q, M, TIW, SM ...) in which replacing is to be performed.

The following rules apply to replacing:

- The signal lists must be complete, they must not contain any wildcards (e.g. * or ?).
- Signal lists must be of the same length
- ”Old” and ”new” signal lists must be different (otherwise replacing would not make any sense).
- ”Old” and ”new” signal lists may overlap.

Examples for signal list inputs (in accordance with editor)

correct	incorrect
I6.1-8.16	I6.8-6.1
M2.2-2.7	I6.1-I8.4
Q8.3	Q6.1-I7.1

- Old and new signals must have the same validity status in the equipment list.

Example for signal list entries (in accordance with equipment list)

Equipment list:	Signal list entry
Slot6: Input module	correct
Slot7: unequipped	”Old” I6.1-8.32
Slot8: Input module	”New” I16.1-18.32
:	
Slot11: Input module	incorrect,since Slot7 unused and Slot12 input module
Slot12: Input module	”Old” I6.1-8.32
Slot13: Input module	”New” I11.1-13.21
:	
Slot16: Input module	
Slot17: unequipped	
Slot18: Input module	

Block List

- ”Edit”, ”Replace Signals”, ”Block List”
- ”Load”, ”Read out PLC”, ”Block List”
- ”Load”, ”Compare”, ”Blocks with PLC”, ”Block List”
- ”Print”: ”Program Protocol” / ”Cross Reference List” / ”Signal Assignment List”, ”Block List”

Enter here a list of the blocks to be processed. The list can consist of a maximum of 200 characters.

Permissible inputs:

*	”*” Is the presetting for the processing of all the blocks which exist under the current station.
OB, PB, FB	e.g. OB1, PB1, FB55, ...
PBxxx-yyy	e.g. PB8-19
FB10	only FB10

If several lists needed, a comma must separate the lists from each other, as follows: PB1-19, PB23-24, FB5-8, FB13-28
You can overwrite the available entries or erase them using the <Blank> key and then replace them.
This list can be stored under some name using the ”Save block list” and then be loaded again as often as required, in any position.

Load Block List

- ”Edit”, ”Replace Signals”, ”Load Block List”
- ”Load”, ”Read out PLC”, ”Load Block List”
- ”Load”, ”Compare”, ”Blocks with PLC”, ”Load Block List”
- ”Print”: ”Program Protocol” / ”Cross Reference List” / ”Signal Assignment List”, ”Load Block List”

The list which is programmed under ”Block list” and saved under ”Save Block List” can be loaded using this function, if required.
The block list name should not contain a period.

Save Block List

- ”Edit”, ”Replace Signals”, ”Save Block List”
- ”Load”, ”Read out PLC”, ”Save Block List”
- ”Load”, ”Compare”, ”Blocks with PLC”, ”Save Block List”
- ”Print”: ”Program Protocol” / ”Cross Reference List” / ”Signal Occupancy List”, ”Save Block List”

The list which is input under ”Block List” can be stored under a name chosen by the user.
If required, it can be called again using ”Load Block List”.
The block list name should not contain any periods.

3.3.7 Compile (Blocks)

- ”Edit”, ”Compile (Blocks)”

This function compiles all blocks of the modified station (with the exception of the data block).

When a new version of Modsoft AKF has been installed, the old stations must be readjusted to the current software version using the function ”Compile blocks”.

Transfer a Station from MICROAKF

Step 1 Generate a station with MICROAKF

Step 2 Copy MICROAKF stations into this newly defined station

Step 3 Delete file ”BEST.DAT”.

Step 4 Generate a valid Equipment List.
(If you have not carried out step 3, you will receive various error messages here.)

Step 5 Translate the blocks with ”Edit”, ”Compile (Blocks)”.

Step 6 Translate the SYMKOM blocks with ”Edit”, ”Compile (Data blocks)”

3.3.8 Compile (Data Blocks)

- ”Edit”, ”Compile (Data Blocks)”

This function translates all data blocks entered and adjusts them to the current software version.

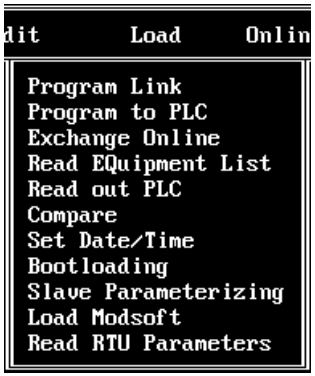
The following entries are required:

3.4 Load

The loading function serves as a means of communicating with the programmable controller and processing the user program in the programmable controller.

☞ **Note** Please note that, with the exception of ”Program Link”, all other functions are only possible if a programmable controller is correctly connected up.
The programmable controller ”Micro” is connected to the PADT’s COM1 serial interface.

The following loading functions are available in this pulldown menu:



3.4.1 Program Link

- ”Load”, ”Program Link”

When this function is selected, the programmable controller program is generated.
In so doing, the OB and all the blocks which are called from the OB are linked to form a program which is runnable in the programmable controller.
This complete program is filed in the ”CODE.SPS” file and can then be transferred to the programmable controller using the ”Program to PLC” function.
Linking is effected using the linking mode and ALU set up in ”SeTup” and ”PLC Station”.

The scope of additional information to be transmitted into the programmable controller can be specified in **Linking mode**.

- **Complete Retranslation**: This setting enables an unlimited representation of IL.
  The line comments,network comments, parameter symbols, and labels are represented, but not the symbols and comments from the data block.
  Comments, symbols, and labels are identical to the original.

- **Without comments**: Just like the full representation, but without the comments.

- **Without Retranslation**: With this setting, no representation is possible.

- **Automatic adjustment**: If the program is too large with a full representation, the link mode is switched into a form in which the program fits into the programmable controller.

3.4.2 Program to PLC

- ”Load”, ”Program to PLC”

After selecting this function, the file ”CODE.SPS”, which was generated from the OB with ”Link Program”, is transmitted to the PLC. This procedure can only be carried out with the PLC stopped. Before each transmission the whole user program area of the PLC is normalized.

On each transfer of the file CODE.SPS the EQL List is also transfered. Before transmission it is checked whether the HW equipment corresponds with the EQL List.

If the check of the file CODE.SPS shows that after the linking an online change has already been made, then the safety acknowledgment is requested, whether the transfer should definitely take place.

If under ”Setup”, ”PLC-Station”, ”Memory Variant: RAM Operation” is set up, then a check will be made whether the setup under ”Setup” - ”PLC-Station” complies with those at the time of the ”Linking”.

If upto here everything corresponds then a check is made whether all the values are also possible for the connected PLC.

If under ”Setup”, ”PLC-Station”, ”Memory Variant: FLASH-Operation” is setup, then only local checks will be made.

3.4.3 Exchange Online

- ”Load”, ”Exchange Online”
- ”Edit”, ”Block”, ”Start Entry”, existing Block, <Ctrl>+<Return>, ”Exchange Online”
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

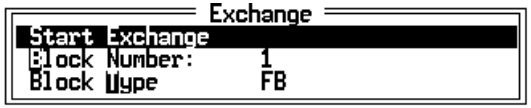
In this function, a new block is loaded to the running or stopped programmable controller.

It is also possible to replace a changed block in the running or stopped programmable controller.

Note New SFBs cannot be loaded with this function into the PLC. If new SFBs must be loaded into the PLC then the program has to be linked and then loaded into the PLC.

When approaching the limit of available memory, online replacement may no longer be possible. In this case, try to transmit the whole program into the programmable controller after relinking.

The following inputs are available in this menu:



Caution With online changing no automatic saving in the Flash-EPROM (passive program memory) is carried out. The new or changed program blocks are loaded into RAM (user memory) which can be lost when switching off (when no battery is used).

In order to save changes constantly into the Flash-EPROM in any case a ”Link Program” and ”Load Program into Flash-EPROM” has to be carried out.

Start Exchange

- ”Load”, ”Exchange Online”, “Start Exchange”,
- ”Edit”, ”Block”, ”Start Entry”, existing Block, <Ctrl>+<Return>, ”Exchange Online”, “Start Exchange”,
- ”Edit”, ”Overview”, ”Select Block”, <Ctrl>+<Return>, ”Edit Block”,..

After determining the block to be replaced (type and number) the replacement is commenced.
Small windows show you where you are in the processing.

3.4.4 Read Equipment List

- ”Load”, ”Read Equipment List”

The actual equipment list of the programmable controller is read into the programming panel using this function, and saved in the ”BES.DAT” file (of the current station).

Any existing ”BES.DAT” file in the station will be overwritten following acknowledgment of the operation!

This function is only carried out when the actual existing CPU is already entered in the Equipment List. Otherwise there will be an error message.

3.4.5 Read out PLC

- ”Load”, ”Read out PLC”

By means of this function, you can read from the PLC:
individual blocks
the complete program including equipment list

Choose the required block list with the aid of the following functions and select the required system and station:

Start PLC Read-Out

- ”Load”, ”Read out PLC”, ”Start PLC Read-Out”

After specifying the blocks to be read out (any available block lists, i.e. block lists to be loaded), reading out is commenced from the PLC into the specified system and station.
If this system and station already exist, the station is overwritten (after a query for confirmation is acknowledged).
If the system and/or station do not already exit, they are generated after confirmation.

Reading to plant

 - ”Load”, ”Read out PLC”, ”To Plant”

Here you can specify a plant into which the blocks read out are to be written. The default plant directory is the plant selected in the programmable controller.

Reading / Station Name

 - ”Load”, ”Read out PLC”, ”Station Name”

You can specify the station here, into which the read out blocks are to be written. The default station name is the station name selected in the programmable controller.

3.4.6 Compare

- ”Load”, ”Compare”

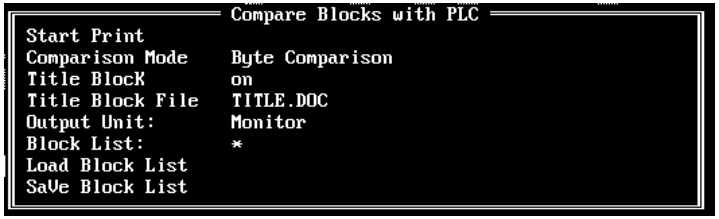
These functions compare data from the AKF station in the PADT with those in the programmable controller.

The following comparison functions are available:

Compare Blocks with PLC

- ”Load”, ”Compare”, ”Blocks with PLC”

The following functions are available:

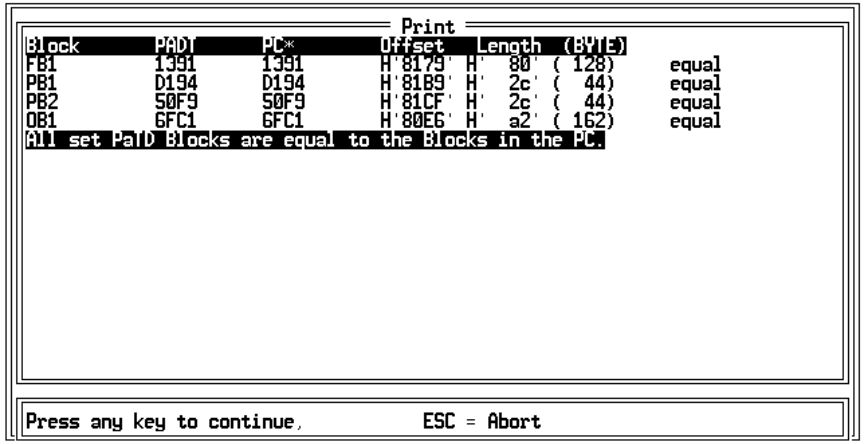


Start Print

 - ”Load”, ”Compare”, ”Blocks with PLC”, ”Start Print”

The blocks named in the block list are compared in the PADT and programmable controller according to the specified mode.

The following example shows a successful block comparison.



Comparison Mode

 ”Load”, ”Compare”, ”Blocks with PLC”, ”Comparison Mode”

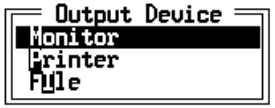
You can choose between the following by toggling:

Byte comparison:	The blocks are compared byte by byte (highest level of certainty)
Fast comparison:	Only one test polynomial per block is compared.

Output Unit

- ”Load”, ”Compare”, ”Blocks with PLC”, ”Output Unit”
- ”Load”, ”Compare”, ”Program with PLC”, ”Output Unit”
- ”Load”, ”Compare”, ”Equipment Lists”, ”Output Unit”
- ”Online”,”PLC Status”, ”Output Unit”
- ”Print”, ”Overview”, ”Output Unit”
- ”Print”, ”Program Protocol”, ”Output Unit”
- ”Print”, ”Symbols and Comments”, ”Output Unit”
- ”Print”, ”Cross Reference List”, ”Output Unit”
- ”Print”, ”Signal Assignment List”, ”Output Unit”
- ”Print”, ”Equipment List”, ”Output Unit”
- ”Print”, ”Complete Documentation”, ”Output Unit
- ”Special”, ”System Information”, ”Output Unit”

A menu appears in which you can choose:



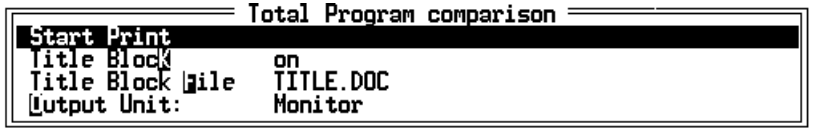
- Output page by page to the screen
- Output to the printer chosen under ”Setup”
- Output into the MS-DOS file, is specified with the complete path. If a file is found under the specified name, the file is overwritten after a query is achnowledged.

Program with PLC

- ”Load”, ”Compare”, ”Program with PLC”

This comparison compares various parameters and all linked blocks of the complete program within the PLC with those in PADT.

The following functions are available:



The following example shows a successful program comparison (first page).

M A N A G E M E N T T A B L E

		PLC	PaTD
Code length UP	**:**	**00544**	**00544 Bytes**
Plant	**:**	**C:\Test**	**C:\Test**
Station name	**:**	**MICRO**	**MICRO**
Identifier	**:**	**micro**	**micro**
Link number	**:**	**000703206**	**000703206**
Index	**:**	**007**	**007**
PLC-Type	**:**	**CPU 512**	**CPU 512**
Link Mode			
PLC	**:**	**Complete Retranslation**	
PaTD	**:**		**Complete Retranslation**
Date	**:**	**18.11.1994**	**18.11.1994**
Time	**:**	**06:34**	**06:34**
Max. No. of Blocks	**:**	**020**	**020**

Block insertions :

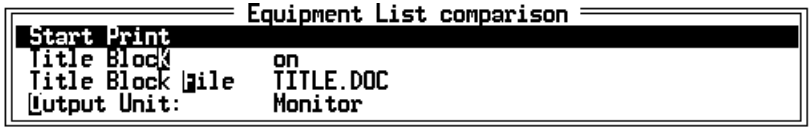
		PLC		PaTD			
No.	**Name**	**Length bytes**	**CRC**	**Name**	**Length bytes**	**CRC**	**Block is**
3	**OB1**	**49**	**A169h**	**OB1**	**49**	**A169h**	**equal**
7	**PB1**	**232**	**60AEh**	**PB1**	**232**	**60AEh**	**equal**

Equipment List

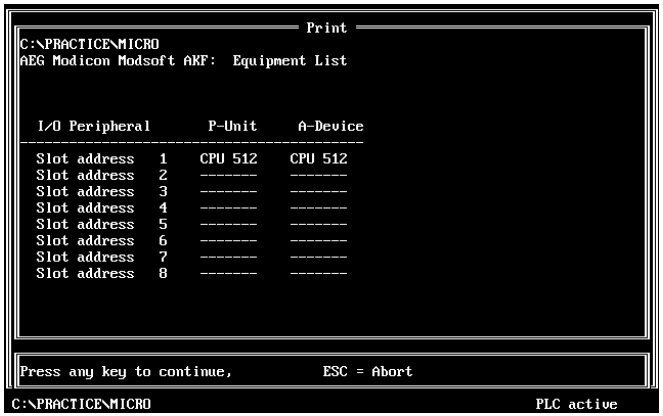
- ”Load”, ”Compare”, ”Equipment List”

After selecting this function, you will see on the screen the equipment list in PADT and the actual equipment in the PLC.
Prior to being transferred to the programmable controller, a program must be linked to the programmable controller, and it must match the equipment list in the programmable controller.

The following functions are available:



The following example shows the printout of a comparison of equipment lists.



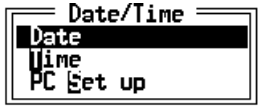
3.4.7 Set Date/Time

- ”Load”, ”Set Date/Time”

This menu point is used for setting date and time for the PADT as well as for the programmble controller.

When the function ”Set SPS” is selected, date and time are sent to the programmable controller. This brings the system marker bytes SMB6 to SMB12 up to date.

The following functions are available:



Date

- ”Load”, ”Set Date/Time”, ”Date”

The current date of your PADT is visible in the first line. This date should correspond to today’s date if you have adjusted the date and time during system start-up or your PADT has an built-in multi-function card. If this is the case, leave the editor using the <Esc> key.
A new date must be entered in the second line. The input must be repeated until a valid date has been input or the editor has been left.

Note Your PADT’s date will be overwritten with the new date.
Here is an example of a valid input: (day, month, year)
12.3.1989 12 3 1989 12/3/1989 12,3,1989 12-3-1989 12:3:1989

Time

- ”Load”, ”Set Date/Time”, ”Time”

Your PADT’s current time is visible in the first line.
This time should correspond to the actualtime if you adjusted the date and time during system start-up or your PADT has a built-in multi-function card.
In this case, leave the editor using the <Esc> key.
The new time is entered in the second line. The input must be repeated until a valid time has been input or the editor is left using the <Esc>.

Note Your PADT’s time will be written with the new time.
Here are examples of a valid input: (hour, minute, second)
12.13.19 12 13 19 12/13/19 12,13,19 12-13-19 12:13:19

PLC Set up

- ”Load”, ”Set Date/Time”, ”PLC Set up”

If you have just entered the new time and date, these will be sent to the PLC. Otherwise, the system date and time will be sent to the programmable controller. Then, the system markers SMB6 to SMB12 in the programmable controller are brought up to date.

The date and the time of the system can also be adjusted on the PADT using the ’date’ and ’time’ operating system functions after powering up (or carrying out a warm restart).

3.4.8 Bootloading

- ”Load”, ”Bootloading”

With this function the basic software of the CPU311/411/512/612 from the PADT can be loaded into the PLC. The AKF-executive is loaded. If under ”Setup”, ”PLC-Station”, ”Memory Variant: FLASH Operation” is setup, then an existing linked program will be loaded into the PLC. A comparison of the EQL List with the PLC will not be carried out.

3.4.9 Parameters for Slave

- ”Load”, ”Parameters for Slave”

Here a Micro loaded with MODSOFT BSW can be configured as a slave Micro.

Note The Micro to be configured must be directly connected to the PC.

After selecting this function the user must enter the slot address that the PLC occupies in the Equipment List. Subsequently, the assignment can be started.

The entries in the following menu are then available:

Parameters for Slave

Start parametrizing!	
Slot–Number	**2**

3.4.10 Load Modsoft

- ”Load”, ”Load Modsoft”

Here you can load the Modsoft-Executive to use an AKF-Micro as a slave.

Note The Micro to be configured must be directly connected to the PC.

3.4.11 Read RTU Parameters

- ”Load”, ”Read RTU Parameters”

For operating a substation the telecontrol software will be loaded into the selected telecontrolstation (CPU 522, 622 or 722) with this menue.

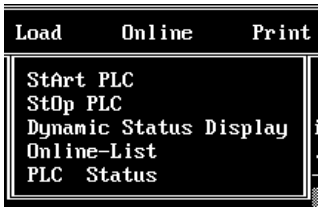
Further informations about assembly, parameterization and operation are in the ”Benutzerhandbuch Geadat, Unterstation U060, A91M.12-704 512”

3.5 Online

The online functions serve as a means of testing the program in the programmable controller.

☞ **Note** Please note that all functions except for ”Program Link” are only possible with a properly connected programmable controller. The programmable controller ”Micro” is connected to the serial interface COM1 of PADT.

The following functions are available:



3.5.1 Start PLC

- ”Online”, ”Start PLC”

This function is used to start the program in the programmable controller.

Before the PLC program is started, the start must be acknowledged.

After the start, the status line is updated (programmable controller active)

3.5.2 Stop PLC

- ”Online”, ”Stop PLC”

This function is used to stop the program in the programmable controller and to update the status display of the programmable controller.

Prior to stopping the programmable controller, an acknowledgement is required.

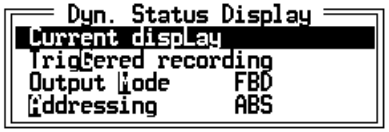
After stopping, the status line is updated (programmable controller in stop).

3.5.3 Dynamic Status Display

- ”Online”, ”Dynamic Status Display”

The dynamic condition of a program in the programmable controller can be displayed using this function.

The following functions are first displayed when this menu item is selected:

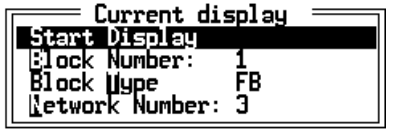


IL, LD, FBD
SYM/ABS

Current Display

- ”Online”, ”Dyn. Status Display”, ”Current Display”

In this form of display, updated information on changing signals is taken into account. The following functions are available:



1 ... 999
OB/PB/FB
1 ... 999

Start Display

- ”Online”, ”Dynamic Status Display”, ”Current Display”, ”Start Display”

After you have selected all the other functions as desired, use this line to start the display.

Dynamic displays

- ”Online”, ”Dynamic Status Display”, ”Current Display”, ”Start Display”, <Ctrl>+<Return>
- ”Edit”, ”Blocks”, ”Start Input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>

A pulldown menu appears with which you can execute various operations during the dynamic display.

You are now in the dynamic display mode.

a) You can select the following functions in the pulldown menu:

The window is closed again by pressing the <Esc> key.

b) Special keys (corresponding to the pulldown menu):

<PgDn>	Page to the next network
<PgUp>	Page to the previous network
<↑>	Scroll screen up
<↓>	Scroll screen down
<Esc>	Abort the dynamic display

Scroll forward / Scroll backward

- ”Online”, ”Dyn. Status Display”, ”Current Display”, or ”Triggered recording”, ”Start Display”, <Ctrl>+<Return>, ”Page forward”
- ”Edit”, ”Block”, ”Start Input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Page forwards”
- ”Online”, ”Dyn. Status Display”, ”Current Display”, or ”Triggered recording”, ”Start Display”, <Ctrl>+<Return>, ”Page backward”
- ”Edit”, ”Block”, ”Start Input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Page backward”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

This function is used for paging back and forth, network by network, within the selected block.

Break

- ”Online”, ”Dyn. Status Display”, ”Current Display,” or ”Triggered recording”, ”Start Display”, <Ctrl>+<Return>, ”Abort”
- ”Edit”, ”Block”, ”Start Input”, <Ctrl>+<Return>, ”Dyn.Status Display”, <Ctrl>+<Return>, ”Abort”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”..

After selecting this function the dyn. display is terminated.

By pressing the <Return> key this function is activated in the window.

The window is closed again by pressing the <Esc> key.

Search for Signal

- ”Online”, ”Dyn. Status Display”, ”Current Display”, or ”Triggered recording”, ”Start Display”, <Ctrl>+<Return>, ”Search for Signal”
- ”Edit”, ”Block”, ”Start Input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Search for Signal”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”..

This function is used for finding a signal in the selected block. To do this, enter the desired signal at the prompt ”Identifier” and confirm with <Return>. The searching direction is specified by putting a <+> (forwards) or <-> (backwards) in front.

Search for Network

- ”Online”, ”Dyn. Status Display”, ”Current Display”, or ”Triggered recording”, ”Start Display”, <Ctrl>+<Return>, ”Search for Network”
- ”Edit”, ”Block”, ”Start Input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Search for Network”
- ”Edit”, ”Overview”, select block, <Ctrl>+<Return>, ”Edit block”..

This function is used for searching for a network in the selected block. To do this, enter the network desired and confirm with <Return>.

MW Mode

- ”Online”, ”Dyn. Status Display”, ”Current Display”, ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”MW-Modus”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”MW-Mode”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

IL
In the status line, the symbol and the comment of the signal selected with the cursor are always displayed.
The following is displayed in the IL line:

MW mode = off	the decimal signal status and the line comment
MW mode = on	decimal and binary signal status

LD/FBD
The status line shows:

MW mode = off	Symbol and comment of the signal selected
MW mode = on	Signal and decimal, binary, hexadecimal and ASCII signal status

The window can be closed again by using the <Esc> key.

Online List

- ”Online”, ”Dyn. Status Display”, ”Current Display” or ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”Online List”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Online List”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”.

This function is used for processing the lists within the Dynamic Status Display. Description under ”Online List”.

Start PLC

- ”Online”, ”Dyn. Status Display”, ”Current Display” or ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”Start PLC”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Start PLC”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

This function makes it possible to start the user program in the programmable controller during the dynamic status display.

Stop PLC

- ”Online”, ”Dyn. Status Display”, ”Current Display” or ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”Start PLC”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Start PLC”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

This function is used for stopping the user program in the programmable controller during the dynamic status display.

Display NW-Comments

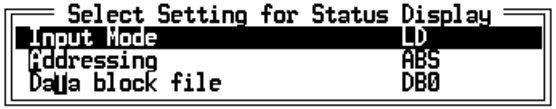
- ”Online”, ”Dyn. Status Display”, ”Current Display” or ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”Display NW-comments”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Display NW-comments”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

The network comments generated in a window under ”Edit” can be called up at this point.

PresettinG

- ”Online”, ”Dyn. Status Display”, ”Current Display” or ”Triggered recording”, ”Start display”, <Ctrl>+<Return>, ”Presetting”
- ”Edit”, ”Block”, ”Start input”, <Ctrl>+<Return>, ”Dyn. Status Display”, <Ctrl>+<Return>, ”Presetting”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”..

The following functions are available at this point:



IL/LD/FBD
ABS/SYM
DB0 ... DB9

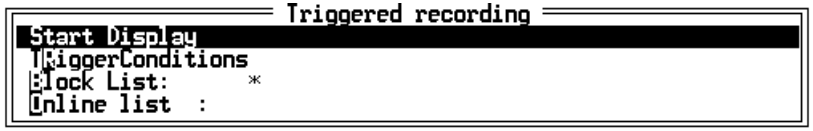
Triggered Record

- ”Online”, ”Dynamic Status Display”, ”Triggered recording”

The conditions of selected signals are recorded for several scans.

Caution For this function, the programmable controller must be in scan.

The following functions are available in the pulldown-Menu:



Start display

- ”Online”, ”Dynamic Status Display”, ”Triggered recording”, ”Start display”

After the signals to be recorded have been determined and the trigger conditions laid down, the function can be started with this line.

Initially, two display windows will appear. The first one can be used for aborting the function during the recording (using <Esc>-key). The second window displays the report when the job is completed, along with the number of records generated.
Finally, network 1 of the first block (first record) is brought onto the screen.

Triggered Display

- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Start display”, <Ctrl>+<Return>

a) At the right edge of the screen, a window will appear with the following functions:

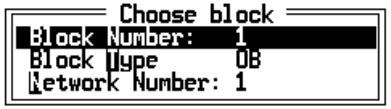
b) In addition to the pulldown menu, the following special function keys are available:

<PgDn>	page to the next network
<PgUp>	page to the previous network
<↑>	Scroll screen upwards
<↓>	Scroll screen downwards
<Esc>	Abort the dynamic display

Choose Block

- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Start display”, <Ctrl>+<Return>, ”Choose Block”

Use this function to select any block you like. The following menu will appear:



1 ... 999
OB/PB/FB
1 ... 999

Next Record

- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Start display”, <Ctrl>+<Return>, ”Next Record”

Use this function for selecting the next record. The record number is in the head line in the top left hand corner of the screen.

Last Record

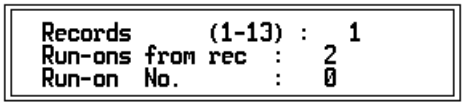
- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Start display”, <Ctrl>+<Return>, ”Last Record”

Use this function to select last records. The record number is in the head line, in the top left hand corner of the screen.

Choose Record

- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Start display”, <Ctrl>+<Return>, ”Choose Rec.”

This function is used for displaying any desired record. The record number is in the head line in the top left hand corner of the screen. The following menu is used for steering the selection:



Number of records, incl. run-on reco
Starting number of run-on records
Number of selected run-on record

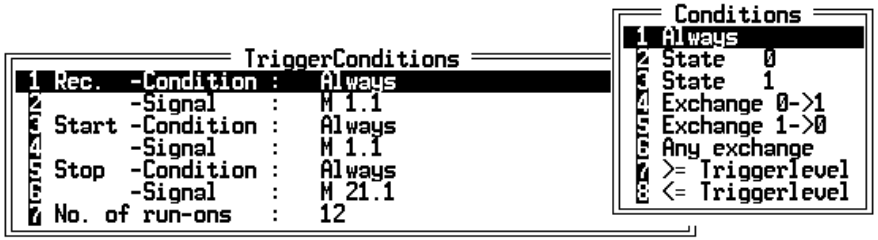
Trigger conditions

- ”Online”, ”Dynamic Status Display”, ”Triggered Record”, ”Trigger Conditions”

To carry out triggering, various conditions must be laid down under the menu item ”Trigger Conditions”:

- the condition for carrying out a recording
- the condition for starting the recording (Start-condition)
- the condition for stopping the recording (Stop-condition)
- the number of records after the stop-condition (run-on)

The following pulldown menu is available:



For each of the recording condition, start condition and stop condition, one of the following functions is selected from the right window and taken over into the left window with the <Return> key.

Always	Condition is always fulfilled
State 0	The condition is fulfilled when the signal is in state 0- (low)
State 1	The condition is fulfilled when the signal is in state 1- (high)
Exchange 0 → 1	The condition is fulfilled for a 0 to 1 state transition. (positive edge)
Exchange 1 → 0	The condition is fulfilled for a 1 to 0 state transition (negative edge)
Any Exchange	The condition is fulfilled for any state transition
<= Triggerlevel	The condition is fulfilled when the signal (byte, word or double word) becomes less than or equal to the specified trigger level
>= Triggerlevel	The condition is fulfilled when the signal (byte, word or double word) becomes greater than or equal to the specified trigger level

Then, the pertinent signal address (symbolic or absolute) is entered. This input is not required for the ”unconditional” condition. Any signal information already entered will be ignored in this case.

The number of run-on records is also to be specified (0 ... N).

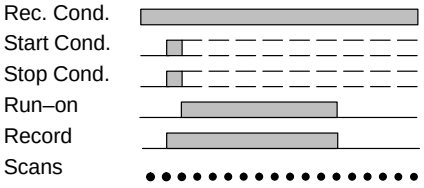
When all the information is correct, the entire menu is taken over using the <Esc>-key.

Examples of a triggered status display

☞ **Note** In the time-scale depictions of the various conditions, it is not the signal value (0 or 1) which is depicted, but rather the ”truth” of the given condition (i.e. fulfilled or not fulfilled).

a) Triggering on ”1”-State, with run-on records

Timing diagram

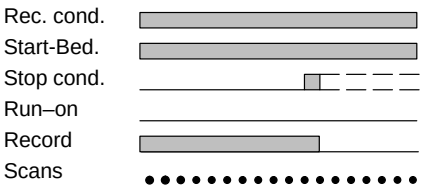


Trigger conditions

Recording condition	Always
Record signal	
Start condition	State 1
Start signal	M4.4
Stop condition	State 1
Stop signal	M4.4
Number of run-on records	10

b) Pre-triggering on positive edge

Timing diagram

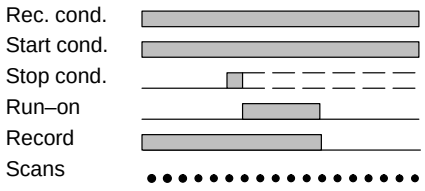


Trigger conditions

Recording condition	Always
Record signal	
Start condition	Always
Start signal	
Stop condition	Transition 0-1
Stop signal	M4.5
Number of run-ons	0

c) Mid-triggering on positive edge, run-ons

Timing diagram

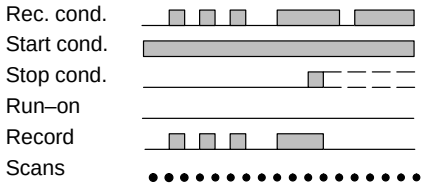


Trigger conditions

Recording condition	Always
Record signal	
Start-Condition	Always
Start signal	
Stop condition	Transition 0-1
Stop signal	M4.6
Number of run-ons	5

d) Records all changes in Marker M4.7 until Marker M4.8 enters state ”1”. If the available memory does not suffice for saving all the records, the events at any rate can be counted.

Timing diagram



Trigger conditions

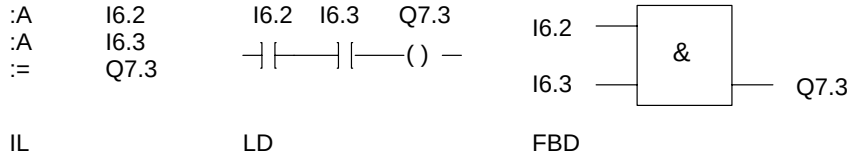
Recording condition	any exchange
Record signal	M4.7
Start-Condition	Always
Start signal	
Stop condition	State 1
Stop signal	M4.8
Number of run-ons	0

Output Mode

- ”Online”, ”Dyn. Status Display”, ”Output Mode”
- ”Print”, ”Program listing”, ”Output Mode”

You can choose the mode of output of your program listing in this menu line. Available modes are instruction list (IL), ladder diagram (LD) and function block diagram (FBD).
Select by toggling between the modes.

The following example shows the same program in the three output modes:



Online list

- ”Online”, ”Dynamic Status Display”, ”Triggered recording”, ”Online list”

If you enter a name in the online list under ”Triggered recording”, the signals in the list will also be recorded. This makes an extended program investigation possible.

3.5.4 Online list

- ”Online”, ”Online List”
- ”Online”, ”Dynamic Status Display”, ”Start Display”, <Ctrl>+<Return>, ”Online List”,
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit Block”,..

This list is used for testing, checking, and monitoring the user program in the programmable controller. Direct access to the programmable controller’s signal memory is used.

You may edit several lists per station. Each list may include status, control and forcing instructions.

The following window appears when this function is selected:

Load online list

- ”Online”, ”On-line list”, ”Load On-line List”
- ”Online”, ”Dynamic Status Display”, ”Start Display”, <Return>, ”On-line List”, ”Load On-line List”
- ”Edit”, ”Overview”, Select block, <Return>, ”Edit block”,..

This function is used to load and call the online lists. Several lists can be created for each station. The lists are station-specific.

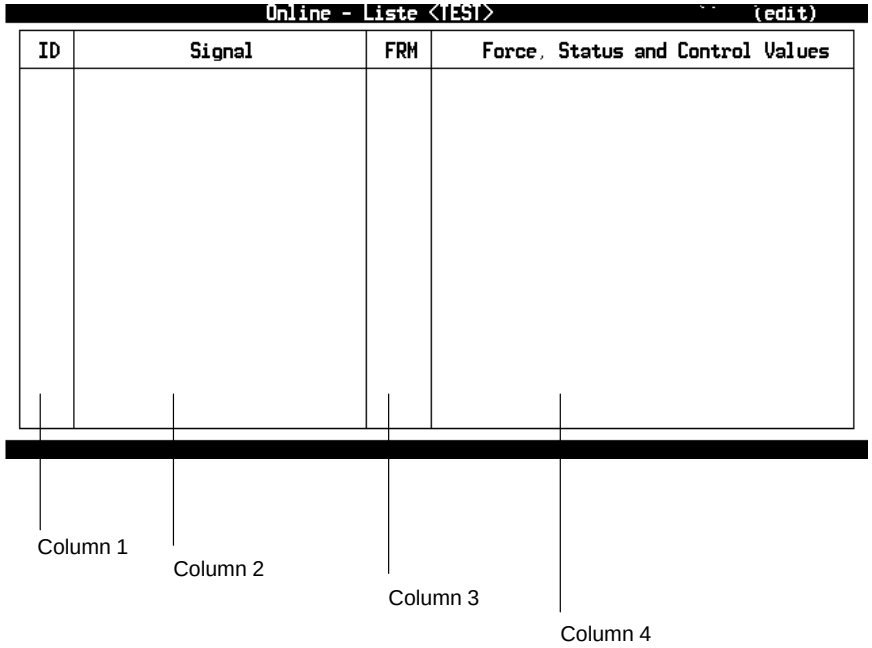
When the function is selected, a prompt line is displayed for entering a name. If lists already exist, the latest list edited is shown. This name must then either be confirmed by pressing <Return>, or a different one entered with the line editor. By entering a blank and pressing <Return>, you obtain a window in which you can select an already existing list by using the cursor keys.

Now a window will appear in which you can edit and run the online list.

Edit Online List

- ”Online”, ”On-line List”, ”Load On-line List”, Name, <Return>
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

A window with four columns appears.



Signals can be displayed (status), controlled, and forced.

By pressing <Ctrl>+<Return>, the list can be processed (run) by means of a pulldown menu.

Column 1 (ID)
To set individual signals without having to discard the entire list, the following identifiers (ID) are possible.

ID	effects
F	Markers set to forcing, not active
FO	Forcing on, current signal becomes active on start-up
C	Marker set to control, not active
CO	Control on, current signal becomes active on start-up
”non entry”	Status display active

Column 2 (signal)
The following signal types are possible for the various parts of the list (refer to chapter 3.1).

Operand	Status	Control	Forcing
I	possible	possible	possible
IB	possible	possible	possible
IW	possible	possible	possible
ID	possible	possible	possible
Q	possible	possible	possible
QB	possible	possible	possible
QW	possible	possible	possible
QD	possible	possible	possible
M	possible	possible	possible
MB	possible	possible	possible
MW	possible	possible	possible
MD	possible	possible	possible
SM (Bit)	possible	not possible	not possible
SMB	possible	not possible	not possible
SMW	possible	not possible	not possible
SMD	possible	not possible	not possible
T	possible	not possible	not possible
TAW	possible	not possible	not possible
TSW	possible	possible	possible
C	possible	not possible	not possible
CAW	possible	possible	possible
CSW	possible	possible	possible

Column 3 (FRM)
In this column, the format of the signal value of column 4 is specified.
The following formats are possible:

FRM	Format	Example of signal value
ASC	ASCII	John
BIN	Binary	11011001
DEZ	Decimal	1234567890
FP	Floating point	12.234
HEX	Hexadecimal	9AFF

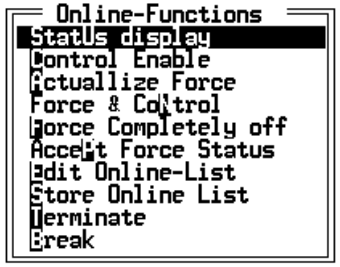
Column 4 (force/status/control value)
In this column, the signal contents are output in the format selected under ”FRM” (refer to column 3).

Edit
A new list can be edited at once (refer to header of list).

The following special keys can be used for editing:

<←>, <→>, <↓>, <↑>	Move cursor
<Esc>	in ”edit” mode: break in ”active” mode: switch to ”edit” mode
<tab>	Set cursor to next input field of a line
<Shift>+<tab>	Set cursor to previous input field of a line
<PgUp>	One page back in the list
<PgDn>	One page forward in the list
<backspace>	Erase character at the left of the cursor
<Del>	Erase character under the cursor

After the entry, the list can be processed (run) by the <Ctrl>+<Return> in an on-line menu.



Status display

- ”Online”, ”Online List”, ”Load on-line List”, Name, <Return>, <Ctrl>+<Return>, ”STatus display”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online list”, ”Load on-line List”, Name, <Return>, <Ctrl>+<Return>, ”STatus display”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

If this function is selected, the complete online list changes into ”active” status (displayed in the header) and cannot be edited. In this mode, the current actual values of all signals entered in the list are cyclically scanned by the programmable controller and displayed in the column for the signal value. The number system for the display is defined in the column ”FRM”.

The keys < ← >, < → >, < ↑ > and < ↓ > retain their function even in case of an active list. Furthermore, the functions of the online menu remain available except for ”STatus display” (already active) and ”Accept force status”.

From the active list, you can return to the ”edit” mode by means of the <Esc> key or the ”Edit online List” function (online menu). At the same time, the status display becomes inactive and any overwritten signal values are automatically restored.

Control enable

- ”Online”, ”Online List”, ”Load on-line List”, Name, <Return>, <Ctrl>+<Return>, ”Control enable”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online list”, ”Load on_line List”, Name, <Return>, <Ctrl>+<Return>, ”Control enable”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

All signals available in the list with the identifier ”CO” (Control On) are controlled to the corresponding signal value, when this function has been selected. Controlling is effective only for the period of a programmable controller program scan. The status of the controlled signals is maintained until it is overwritten with new values by the program in the programmable controller or by functions of the online list (forcing, controlling).

Signals having only the identifier ”C” are not controlled! The identifier ”C” is merely a classification of the signal required for editing.

Update forcing

- ”Online”, ”Online List”, ”Load on-line List”, Name, <Return>, <Ctrl>+<Return>, ”Update Forcing”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online list”, ”Load on_line List”, Name, <Return>, <Ctrl>+<Return>, ”Update Forcing”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

Signals of the online list having the identifier ”FO” (Forcing On) are forced to the corresponding signal value, when this function is selected. Forcing remains active until explicitly disabled. Disabling is done with the function ”Forcing completely off” (online menu). It can also be disabled by changing the identifier of the applicable signal from ”FO” to ”F” and subsequently selecting the function ”Actuallize forcing”. A force signal can be recognized by the color change.

Force & control

- ”Online”, ”Online liste”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Force & Control”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Force & Control”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

This function combines the two functions ”Control enable” and ”Update Force”. The signals are simultaneously forced and controlled.

Forcing completely off

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Forcing completely off”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Forcing completely off”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

Forcing is inactive for signals curently being forced. The applicable identifiers remain unchanged.

Accept force status

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Accept force status”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Accept force status”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

If some signals (not entered in the online list selected) are forced due to prior operations in the programmable controller, the list can be adjusted to the status of the programmable controller by means of this function.
Missing entries are always inserted behind the last list entry. If an F signal exists in the list, but with a different signal value than the one forced in the programmable controller, then signal will be assigned the value from the programmable controller. Excess force signals in the online list are not effected by this function.

Edit online list

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Edit online List”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online list”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Edit online List”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

An active online list (recognizable by the entry ”active” in the 1st line) is switched to ”edit” status by this function. Changes in the list can be made in ”edit” status only.

The <Esc> key has the same function.

Store online list

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Store online list”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Store online List”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

This function is used to store the list under the current station. The new file name may be selected as desired. If you enter a blank and <Return>, a window appears containing all available lists. The list is stored under the name selected. You stay in the list editor.

Terminate (online list)

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Terminate”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Terminate”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

This function terminates and stores the list.

Break (online list)

- ”Online”, ”Online List”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Break”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Onlinelist”, ”Load On-line List”, Name, <Return>, <Ctrl>+<Return>, ”Break”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

This function causes an exit without storing.

Erase online list

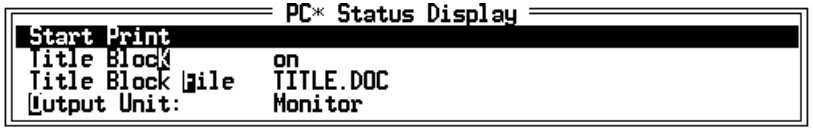
- ”Online”, ”Online List”, ”Erase On-line List”
- ”Online”, ”Dynamic status display”, ”Start display”, <Ctrl>+<Return>, ”Onlinelist”, ”Load On-line List”, ”Erase Online List”
- ”Edit”, ”Overview”, Select block, <Ctrl>+<Return>, ”Edit block”,..

This functions erases the list selected from the station directory.

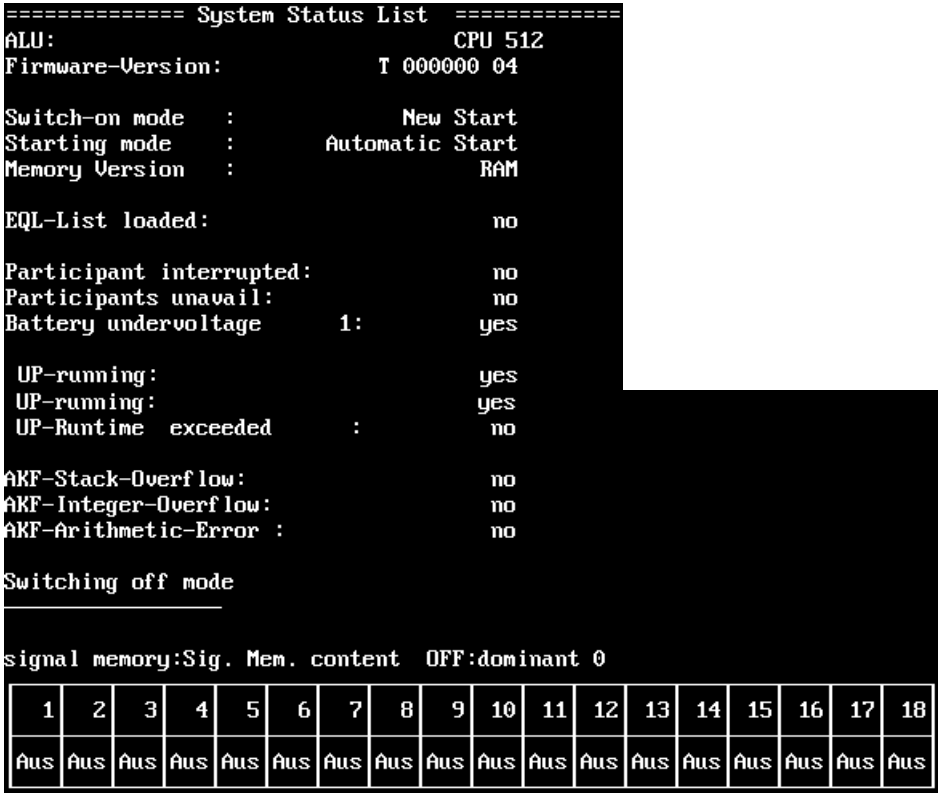
3.5.5 PLC Status

- ”Online”, ”PLC Status”

You can output the following list on the screen, to the ”SeTup” adjusted printer or in a file.



This online function displays different parameters and system markers of the programmable controller. The following figure shows a part of the print.

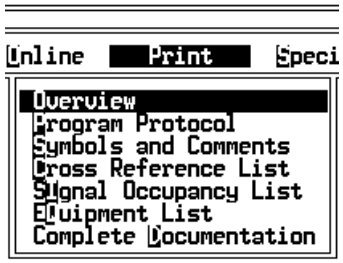


3.6 Print

It is possible to output various presentations and lists to the printer, the screen or in a file.

With selecting the reference characters, the corresponding menu line can be directly selected.
There is another possibility of gaining access to the submenu this is achieved by moving the cursor onto the corresponding menu line and actuating the <Return> key.

The following functions are at your disposal in the pulldown menu:

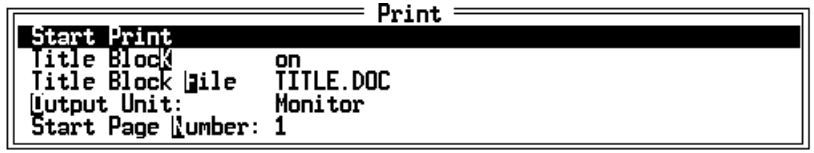


3.6.1 Overview

- ”Print”, ”Overview”

You can print out the program overview which was programmed under ‘‘Edit”, ‘‘Overview” at this position.

The following functions are at your disposal:



Start Print

- ”Print”, ”Overview”, ”Start Print”
- ”Print”, ”Program Protocol”, ”Start Print”
- ”Print”, ” Symbols and Comments”, ”Start Print”
- ”Print”, ”Cross Reference list”, ”Start Print”
- ”Print”, ”Signal Occupancy List”, ”Start Print”
- ”Print”, ”Equipment List”, ”Start Print”
- ”Print”, ”Complete Documentation”, ”Start Print”
- ”Load”, ”Compare”, ”Blocks with PLC”, ”Start Print”
- ”Load”, ”Compare”, ”Program with PLC”, ”Start Print”
- ”Load”, ”Compare”, ”Equipment lists”, ”Start Print”
- ”Online”, ”PLC Status”, ”Start Print”
- ”Special”, ”System Information”, ”Start Print”

You start the output using this function according to the remaining entries in the pulldown menu.

Title Block

- ”Load”, ”Compare”, ”Blocks with PLC”, ”Title Block”
- ”Load”, ”Compare”, ”Program with PLC”, ”Title Block”
- ”Load”, ”Compare”, ”Equipment lists”, ”Title Block”
- ”Print”, ”Overview”, ”Title Block”
- ”Print”, ”Program Protocol”, ”Title Block”
- ”Print”, ” Symbols and Comments”, ”Title Block”
- ”Print”, ”Cross Reference list”, ”Title Block”
- ”Print”, ”Signal Occupancy List”, ”Title Block”
- ”Print”, ”Equipment List”, ”Title Block”
- ”Print”, ”Complete Documentation”, ”Title Block”
- ”Special”, ”System information”, ”Title Block”

It is possible to output the title block which you programmed during ‘Edit”, ‘‘Title Block” or which is available under the name ‘‘TITLE.DOC” as a standard feature.

You can toggle between ”on” and ”off”.

The ‘‘on” entry means that the title block, whose data name is entered under the ‘‘title block file” menu line appears on the printout.
If the title block required is not under the station when entered, the output is rejected by sending an error message.

Title Block File

- ”Load”, ”Compare”, ”Blocks with PLC”, ”Title Block File”
- ”Load”, ”Compare”, ”Program with PLC”, ”Title Block File”
- ”Load”, ”Compare”, ”Equipment lists”, ”Title Block File”
- ”Print”, ”Overview”, ”Title Block File”
- ”Print”, ” Symbols and Comments”, ”Title Block File”
- ”Print”, ”Cross Reference list”, ”Title Block File”
- ”Print”, ”Signal Occupancy List”, ”Title Block File”
- ”Print”, ”Equipment List”, ”Title Block File”
- ”Print”, ”Complete documentation”, ”Title Block File”

You should enter the name of the title block file which exists under the station here, if you have actuated the ‘‘on” entry in the ‘‘title block” menu line. If this is not so, the message ‘‘title block file not found. Press any key” appears at the start of the printing.
You should then either enter the name of another title block file which exists under the station, or copy the corresponding title block file which has already been generated under this station, in the ‘‘special”, ‘‘copy” menu line.

You can enter the data names using the line editor.

Start Page Number

- ”Print”, ”Overview”, ”Start Page Number”
- ”Print”, ”Program Protocol”, ”Start Page Number”
- ”Print”, ” Symbols and Comments”, ”Start Page Number”
- ”Print”, ”Cross Reference list”, ”Start Page Number”
- ”Print”, ”Signal Occupancy List”, ”Start Page Number”
- ”Print”, ”Equipment List”, ”Start Page Number”
- ”Print”, ”Complete documentation”, ”Start Page Number”

You can enter the page number (1 to 999) for the first page of the printer output if this function is used.

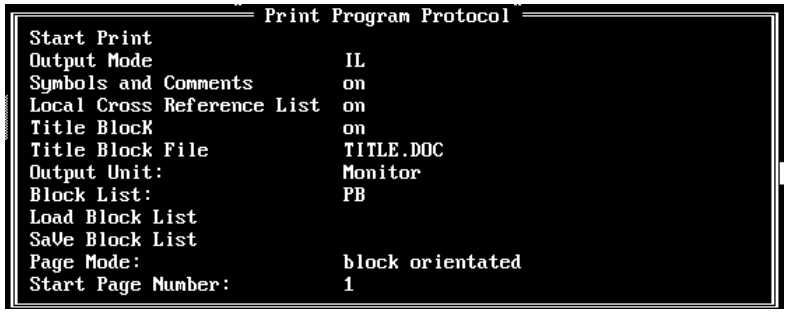
If you use a block-type page mode, every list and/or every block is started anew. The output is continuously numerated if the page mode is continuous.

3.6.2 Program Protocol

- ”Print”, ”Program Protocol”

Using this function you can output your user programmed logic (OB, PB, FB), which you programmed under ‘‘Edit”, ‘‘Blocks” in IL, LD or FBD to the printer, screen or file.

You can select and call the following functions in this pulldown menu:



Symbols and Comments

- ”Print”, ”Program Protocol”, ”Symbols and Comments”
- ”Print”, ”Cross Reference list”, ”Symbols/Comments”

You can toggle between ‘‘on” and ‘‘off”.

The entry ”ON” means that symbolic names and comments assigned by you to the individual signals (inputs, outputs, markers, ...) appear as additional information next to the signal.
The entry ”OFF” prevents the appearance of this additional information (symbolic names and comments) in the program log or cross reference list.

Local Cross Reference List

- ”Print”, ”Program Protocol”, ”Local Cross Reference List”

All the inputs, outputs and markers which you used in the user-programmed logic are listed in the local cross reference lists (in the blocks according to the block list). In addition to this, the networks in which the addresses occur in the respective blocks are specified.

You can toggle between ‘‘on” and ‘‘off”.

The ‘‘on” entry means that the local cross-reference list appears on the program protocol.
After this, other networks in which the signals occur, are specified for every block on the printout:

IL/LD/FBD	Entry ”is used in NW: 1(I) 4(O)” means: that the signal asked for, appears in network 1 as an input (I) and in network 4 as an output (O)

Page Mode

- ”Print”, ”Program Protocol”, ”Page Mode”
- ”Print”, ”Cross Reference list”, ”Page Mode”
- ”Print”, ”Signal Occupancy List”, ”Page Mode”

Using this function, you can determine how the page numeration should be carried out by toggling:

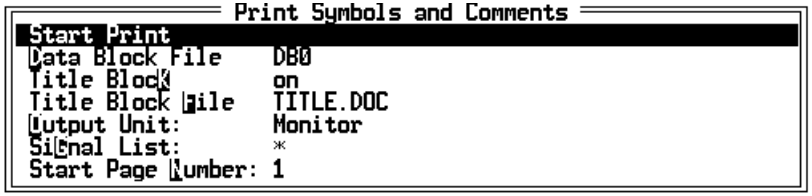
Continuous	the start page number is only determined at the beginning. The various blocks of the list are counted up.
Block orientated	a new page 1 is begun (for start page numbers) for every new block / every new list.

3.6.3 Symbols and Comments

- ”Print”, ”Symbols and Comments”

It is possible to print out the signals (inputs, outputs, markers, ...) to which you have assigned symbolic names and comments under ‘‘Edit”, ‘‘Symbols and comments”.

The following functions are at your disposal under this pulldown menu:



Data block file (DB number)

- ”Print”, ”Symbols and Comments”, ”Data block File”
- ”Print”, ”Cross reference list”, ”Data block File”

A data block which has already been edited in the ‘‘Edit”, ‘‘Symbols and comments” menu can be specified here (DB0 to DB9).

Signal List

- ”Print”, ”Symbols and Comments”, ”SiGnal list”
- ”Print”, ”Cross reference list”, ”SiGnal list”
- ”Print”, ”Signal occupancy list”, ”SiGnal list”

You have to enter the signal ranges here, those which you wish to generate cross reference lists about, and/or allocation lists.

For example, the following entries are at your disposal (with line editor):

”*”	:	all the inputs, outputs, markers, marker words, ..., which exist in the station
”Operand” (I,Q,M etc.)	:	all the signals of an operand operand
”Operand x.y-a.b”	:	Signals of the operand from x.y to a.b
”Operand x.y”	:	Only a specific signal

Several entries in this menu line have to be separated using commas (without Space).

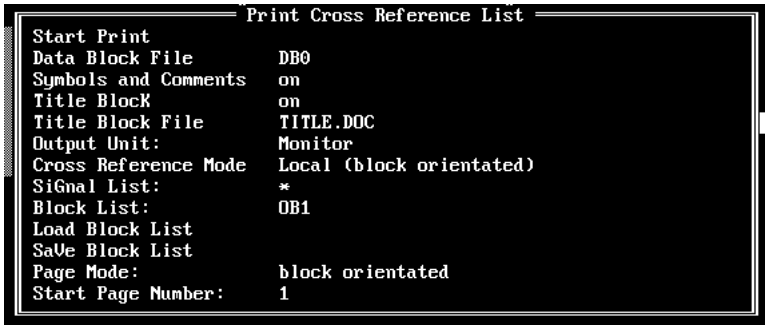
3.6.4 Cross Reference list

- ”Print”, ”Cross Reference list”

All the inputs, outputs, markers, ... which have been used by you in the cross reference list are listed and are specified the parts of the program in which they appear.

It is possible either to output the global, that is to say block-overlapping, or the local, that is to say blockwise cross reference list.

The following functions are at your disposal under this pulldown menu:



Cross Reference mode

 - ”Print”, ”Cross Reference list”, ”Cross Reference mode”

You have to decide whether you wish to have the global or the block orientated cross reference list output, that is to say block-overlapping or local.

You can toggle between ‘‘global” and ‘‘local”.

In addition to the local cross reference list, in the global cross reference list the blocks are output, in which the signals which where entered by you in the ‘‘Signal List” occur (inputs, outputs, markers, ...).

In the ‘‘Block List” you have to decide which blocks the cross reference lists should be generated from.

3.6.5 Signal Occupancy List

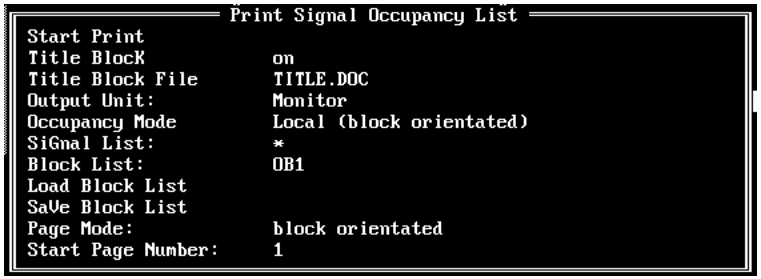
- ”Print”, ”Signal Occupancy List”

In the signal occupancy list all inputs, outputs, markers ... are output, which you used in the blocks of the block list.

An ”X” in the corresponding position of the signal table means that this signal occurs in the blocks which are entered in the ‘‘Block List”.

In so doing, it is possible to either output the global, that is to say block-overlapping, or the local, that is to say block orientated signal occupancy list.

The following functions are at your disposal in this pulldown menu:



Occupancy mode

- ”Print”, ”Signal Occupancy List”, ”Occupancy mode”

You have to decide here, whether you wish to output the global, that is to say block-overlapping or the local, block orientated signal occupancy list.

You can toggle between ‘‘global” and ‘‘local”.

local: the occupancy list merely concerns the blocks specified in the block list

global: the occupancy list concerns blocks specified in the block list and all blocks called by those (direct or indirect call across all subordinate levels)

Example: OB1/global = all Blocks(*)/local

You have to decide in the ‘‘Block List” which blocks the signal occupancy list should be output from.

The following image is an example of the signal occupancy list on the screen.

C:\MIST\MICRO\PB1
AEG Modicon Modsoft AKF: Signal Assignment List Cross Reference Mode

Signal										1						
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6
Q1.x	X	X	X	X	X	X	X	X								

										1																		2														3		
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
M1.x	X	X	X																												
M2.x	X	X	X	X	X	X	X	X																							
M15.x	X	X	X	X																											

	1	2	3	4	5	6	7	8	9
MW7x		X							

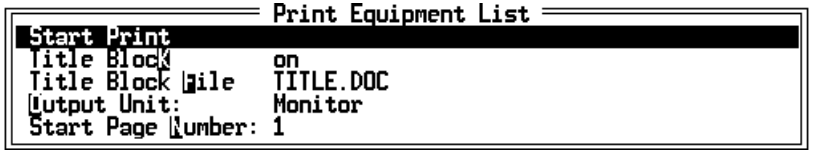
	1	2	3	4	5	6	7	8	9
SM1x					X				

3.6.6 Equipment List

- ”Print”, ”Equipment List”

The Equipment List contains the types of I/O module, which are placed into the subracks. A list can be output using the presettings.

The following functions are at your disposal under this pulldown menu:

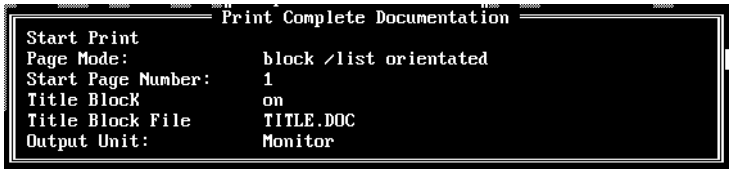


3.6.7 Complete Documentation

- ”Print”, ”Complete Documentation”

A complete output of all the lists is output using this function.
Order: equipment list, signal occupancy list, cross reference list, symbol/comment list, program protocol with subsequent contents directory. The presetting of the respective menus (block list, signal lists, etc.) is taken into consideration.

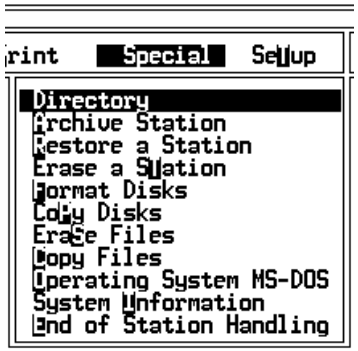
The following functions are at your disposal under this pulldown menu:



3.7 Special

The programming panel related functions are contained in this menu.

The following system functions are at your disposal:



3.7.1 Directory

- ”Special”, ”Directory”

You can display the contents directory of your current station using this function. In so doing, you have an overview with regard to the blocks and files which are located there. Specify a mask for the files which have to be displayed with the aid of the line editor:

e.g.	*.*	Display all of the files
	*.txt	Display of all the text files
	FB*	Display of all the function blocks

After displaying the last contents directory part, press any key and you return to the pulldown menu.

3.7.2 Archive Station

- ”Special”, ”Archive Station”

Whole stations can be backed up on diskettes using this function.

Caution The function cannot be carried out with MS-DOS version 6.0.

Caution all ”old” files are erased on the diskette during archiving. Archiving and restoring should be performed by the same MS-DOS version (refer to MS-DOS manual under ”Backup” and ”Restore”).

The following functions are at your disposal:

Start Archiving

- ”Special”, ”Archive Station”, ”Start Archiving”

After inputting the station and the target station, archiving is started.
Follow the instructions on the screen.

Caution all ”old” files on the diskette are erased during backing up.

It is only possible to interrupt during the archiving, by actuating the <Ctrl>+<C> keys simultaneously.

After the archiving, you regain access to the menu immediately.

Archive From Station

- ”Special”, ”Archive Station”, ”From station”

Enter the station (parts) which have/has to be backed up with the aid of the typewriter keyboard (line editor):

e.g.: C:\MICRO\EXAMPLE*.*

Input:
a) <Return> (introduce editing)
b) Input station name
c) Terminate using <Return>

Archive To Station

- ”Special”, ”Archive station”, ”To station”

The target user drive is entered here with the aid of the typewriter keyboard (line editor).

For example in: A:

Input: a) <Return> (introduce editing)
b) Input target user drive
c) Terminate using <Return>

3.7.3 Restore a Station

- ”Special”, ”Restore a Station”

Previously backed up stations are restored with the same name from backup by using this function.

Caution The function cannot be carried out with MS-DOS version 6.0.

Warning Existing files of the same name are overwritten without acknowledgement. Archiving and restoring should be performed by the same MS-DOS version (refer to MS-DOS manual under ”Backup” and ”Restore”).

The following functions are at your disposal:

Start Restoring

- ”Special”, ”Restore a Station”, ”Start Restoring”

After entering the source drive and the name of the station to be restored, the restoration function is started.
Follow the instructions on the screen.

Caution the name of the target station has to be the same one which was used to back up the station.

It is only possible to interrupt during the restoration by actuating the <Ctrl>+<C> keys simultaneously.

After the restoration, you regain access to the menu immediately.

Restore from Station

- ”Special”, ”Restore a Station”, ”From Station”

The source drive is entered here with the aid of the typewriter keyboard (lineeditor).

For example in the drive: A:

Inputs:
a) <Return> (introduce editing)
b) Input source drive
c) Terminate using <Return>

Restore to station

- ”Special”, ”Restore a station”, ”To Station”

Input the target user drive with the aid of the typewriter keyboard (lineeditor):

e.g.: C:\PLANT\STATION*.*

Warning Existing files with the same name are overwritten!

Input: a) <Return> (introduce editing)
b) Input station name
c) Terminate using <Return>

3.7.4 Erase a STation

- ”Special”, ”Erase a STation”

You can erase a complete contents directory including all its subdirectories using this function.
You are consulted before the subdirectory is erased whether this directory should indeed be erased.
Therefore, one can also erase a number of directories selective.

If you enter <Return>, a window appears containing a list of all available stations of the plant. You can select a station by means of the cursor keys and <Return>. The current station cannot be erased.

3.7.5 Format Disks

- ”Special”, ”Format Disks”

Commercially available new diskettes have to be formatted before they can be used as data carriers. Formating prepares the diskette in such a way that MS-DOS data can be stored and read from the operating system.

Caution all data on the data carrier is destroyed during formating (it can also be used to erase a diskette completely).

The following functions are at your disposal under this pulldown menu:

Start Formatting

- ”Special”, ”Format of Disks”, ”Start Formatting”

After specifying the user drive and if necessary the parameters, formatting is started here.

After the function has selected, follow the instructions on the screen.

Caution Formatting cannot be aborted.

User Drive during the Formatting

 - ”Special”, ”Format Disks”, ”Drive”

The data carriers which have to be formated are entered here with the aid of the line editor.

B: Diskette in user drive B: is formatted
A: Diskette in user drive A: is formatted

Parameters during Formatting

 - ”Special”, ”Format Disks”, ”Parameter”

You can specify additional parameters with the aid of the line editor here when formatting (additional information is contained in the MS-DOS manual).

☞ **Note** To make later identification easier, we recommend to use parameter ”/V” here to insert the station name.

Examples for parameters (valid for portable AEG programming panels, the parameters are partially manufacturer specific, please see your MS-DOS Handbook):

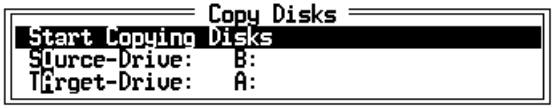
/1	Only the first side of a diskette is formatted. The diskette is formatted with 9 Sectors/track, except when you enter /8
/3	A 720 kByte diskette is formatted in a 1.44 MByte drive
/4	The diskette is formatted with 40 tracks/side. You use this parameter to format a 5 $^1/_4$ inch diskette with 360 kByte in a 1.2 MByte drive. Thus you can use 360 kByte diskettes in a highdensity drive. However, it can not be guaranteed that you can also read these diskette in a 2D- or DD-drive.
/8	8 Sectors per track are formatted. Without this parameter the diskette, dependent on the type of drive, is formatted with 9 or 15 sectors/track.
/N:nn	formats a 3 $^1/_2$ inch diskette with number of sectors given under nn. Enter Sie 9 for a 720 kByte diskette.
/T:nn	Formats a 3 $^1/_2$ inch diskette with the number of tracks given under nn. Enter 80 for a 720 kByte- or 1.44 MByte diskette.
/V	After formatting the disk the entry of a data carrier characteristic is automatically requested.
/S	MS-DOS system files are stored on the diskette in the set drive on the newly formatted diskette.
/*	Generates a status line during formatting.

3.7.6 CoPy Disks

The whole contents of a diskette is copied onto a another diskette (same capacity) using this function.

Caution Data which are possibly on the target drive are erased with this function, because the target drive is formatted while copying.

The following functions are at your disposal in this pulldown menu:



Start Copying Disks

- ”Special”, ”CoPy Disks”, ”Start Copying Disks”

The copying is started here after the source user drive and target user drive have been specified.
After the function has been selected, please follow the instructions on the screen.
It is possible to break off using <Ctrl>+<C> after starting this function.

Copy Disks / Source Drive, TArget Drive

- ”Special”, ”CoPy Disks”, ”Source Drive”
- ”Special”, ”CoPy Disks”, ”TArget Drive”

With the line editor, you can enter one drive twice or two different drives.

Example 1: same drives	Example 2: different drives
Source drive: A:	Source drive: A:
Target drive: A:	Target drive: B:

or B: to A:

3.7.7 EraSe Files

- ”Special”, ”EraSe Files”

You can erase any number of files using this function.
In order to do this, a mask for the file(s) which has to be erased is specified with the aid of the line editor:

e.g.	*.*	Erase all of the files of the plant / station enterd in ”Setup”
	C:\MICRO\EXAMPLE*.*	Erase all of the files of the named plant /station
	DB*	Erase all of the data blocks of the plant / station entered in ”Setup”
	C:\MICRO\EXAMPLE\FB10	erases on the hard disk in the station ”EXAMPLE”, the function block FB10

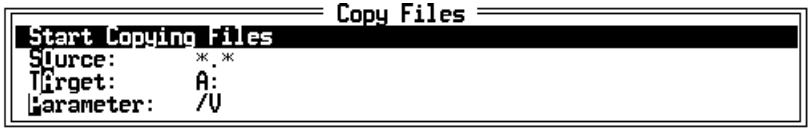
It is only possible to interrupt during erasing by actuating the <Ctrl>+<C> keys simultaneously.

3.7.8 Copy Files

 - ”Special”, ”Copy Files”

You can load station parts (files, blocks etc.) onto or from diskettes using this command. Files on the hard disk can also be copied from one station on to another.

The following functions are at your disposal under this pulldown menu:



You can use the wildcards ‘‘?” and ‘‘*” in the names of the source and the target. When using the replacement characters, please always ensure that you do not overwrite important files unintentionally.

Start Copying Files

 - ”Special”, ”Copy Files”, ”Start Copying Files”

The copying procedure is started after the source, target and, if necessary, parameters have been specified.

Copy Files / Source

 - ”Special”, ”Copy Files”, ”Source”

The file to be copied is specified using the whole path of the source (using the line editor).

E.g.: C:\MICRO\EXAMPLE\PB7

The program block 7 from the ‘‘EXAMPLE” station is copied (to ‘‘Target”) from the hard disk in the MICROAKF directory.

If only a block name is input instead of a path, the block from the current station is copied.

Copy Files / Target

 - ”Special”, ”Copy Files”, ”Target”

The target directory for the file(s) which appear(s) under ‘‘source” is specified with the aid of the line editor.

Target e.g.:	A:	on diskette
	C:\TEST	on hard disk under the ”TEST” directory
	C:\TEST\FB10	the source is copied after ‘‘FB10” onto the hard disk under the ”TEST” directory.
	no specification	in the current station

Copy Files / Parameters

 - ”Special”, ”Copy Files”, ”Parameter”

You can specify additional parameters here when copying with the aid of the line editor (additional information is available in the MS-DOS manual).

Parameters:
/V: Check the copy
/A: The file is treated as a text file.
Source: It is copied up to the EOF character (exclusively).
Target: The file is terminated using the EOF character.
/B: Source: The whole file is copied.
Target: The file is not terminated using EOF characters.

3.7.9 Operating System MS-DOS

 - ”Special”, ”Operating System MS-DOS”

Extract from the MS-DOS handbook:

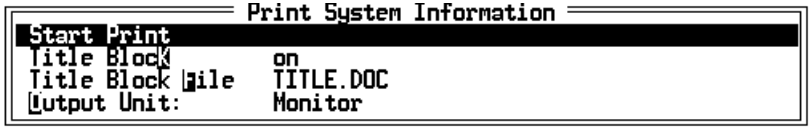
What is MS-DOS ?
MS-DOS stands for Microsoft Disk Operating System. It controls the basic functions of the computer and connects the individual parts of the equipment in a way that the user programs (word processing, calculation, data banks etc.) can be easily used. MS-DOS makes it possible to store and recall data from diskettes or the hard disk. It allows data to be entered using the keyboard and output via a printer. MS-DOS enables you to handle data on diskettes e.g. copying, deleting, comparing, renaming, saving and resaving. It creates directories for your storage mediums, and monitors all entries, logging them with time and date. Apart from this, it controls the security of your data while transferring it from the hard disk to a cassette tape drive.

You can undertake the functions detailed in the MS-DOS manual.
Using the entry ‘‘EXIT” you return to Modsoft AKF.

3.7.10 System Information

- ”Special”, ”System Information”

This function gives you an overview regarding the current hardware configuration of your program panel.
In addition to this, there is information regarding processes, DOS version, interfaces and main memories.
The overview can be output on the screen, file or printer.



3.7.11 End of Station Handling

- ”Special”, ”End of Station Handling”

You can terminate the processing of Modsoft AKF → Micro by using this function.

You can only restart the AKF software now by making a renewed call.

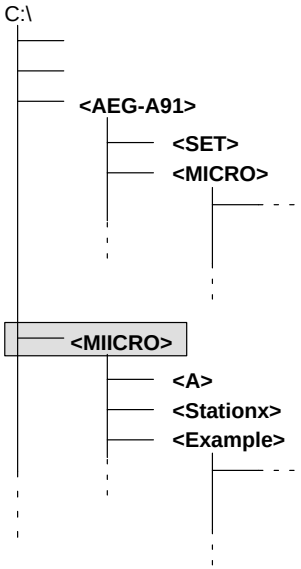
3.8 SeTup

The SeTup functions serve as a means of presetting fixed system data.
The following settings belong to this:

The menu lines ”Plant” and ”Station” have to be edit before the user-programmed logic in any case.

Directory structure of the PADT

The installation of the Modsoft AKF software is effected in two different directories on the hard disk.



- the program directory (Presetting ”C:\AEG-A91”)
  It has to be given during installation.

- the data directory (Presetting ”C:\MICRO”)
  It is devided in the plant and the station directory.

  Mark: This is the data directory of the example program.
  It has to be given during installation.

3.8.1 PLant

- ”SeTup”, ”PLant”

A system (plant) corresponds to the directory on the PADT.
It can contain several different stations.

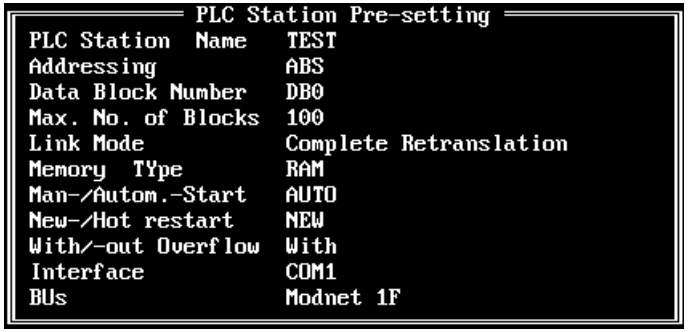
The systems are adjusted using the available function. If the entered plant does not exist yet, it is then generated after confirming.
As far as existing plant is concerned, the last station which was processed is automatically selected.

Example: C:\MICRO

3.8.2 PLC Station

- ”SeTup”, ”PLC Station”

Presettings for the station which is programmed by the user are carried out under this function:



Station Name

- ”SeTup”, ”PLC Station”, ”PLC Station Name”

You can select or program a station via the line editor (typewriter keyboard) using this function.
If you enter a blank, a window appears containing the available stations of the current plant. You can select the station by means of the cursor keys and <Return> in this window.
If the station entered does not exist, it is created after an acknowledgement by the <Return> key (corresponding to the directory in PADT).

e.g.: TEST

All the subsequent processing functions are then executed in the TEST station and the generated files are filed under this directory.

Note The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

Maximum Block Number

- ”SeTup”, ”PLC Station”, ”Max. No. of Blocks”

The number of blocks which you can enter in this menu ranges from 20 to 100 for CPUs 311 and 411 and from 20 to 200 for CPUs 512 and 612.

When linking the program, Modsoft AKF reserves the corresponding amount of space in the programmable controller.
If the number has been adjusted and is too small, it is not possible to link the program. You should enter the estimated number of blocks plus an additional 10 to 20 % at this point.
Block numbers can be set between 1 and 999, independently of the number given here. The sum of all the blocks must not exceed the number given here, however.

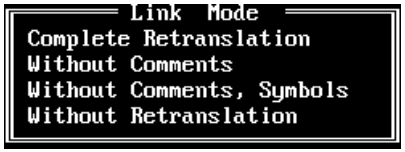
☞ **Note** The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

☞ **Note** If the list is full, no new block can be added online.

Link Mode

- ”SeTup”, ”Station”, ”Link Mode”

You can specify here the amount of additional information which is to be transmitted to the programmable controller. You can choose between:



☞ **Note** The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

Link Mode / Complete Retranslation

- ”SeTup”, ”Station”, ”Link Mode”, ”Complete Retranslation”

An unrestricted retranslation of the IL is possible using this adjustment.
Comments, symbols and labels are identical to the original.
Line comments, network comments, symbols (formal operands) and labels are represented.

Link Mode / Without Comments

- ”SeTup”, ”PLC Station”, ”Link Mode”, ”Without Comments”

The comments are not represented using this adjustment. Otherwise, the function corresponds to the ‘‘complete retranslation”.

Link Mode / Without Retranslation

- ”SeTup”, ”PLC Station”, ”Link Mode”, ”Without Retranslation”

Retranslation is no longer possible using this adjustment.

Link Mode / Automatic Adjustment

- ”SeTup”, ”PLC Station”, ”Link Mode”, ”Automatic Adjustment”

An adaption is effected with respect to of the free memory (located in the programmable controller) when using this adjustment.
If the program is too large, the additional information is progressively removed.

Memory Type

- ”SeTup”, ”PLC station”, ”Memory Type”

The destination of the IL is defined - Flash or RAM

Note The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

Note With OS/2 it is recommended to leave the DOS Box for loading the Flash memory.

Manual/Autom. Start

 - ”SeTup”, ”PLC station”, ”Manual-/Autom.-Start”

Here you setup whether the PLC runs in the Manual or Automatic start mode.

- Manual Start: The PLC remains stopped when the supply voltage energized and must be started manually via a programming unit.

- Automatic Start: The PLC starts automatically when the supply voltage is restored. The mode ”Automatic Start” is only useful on Flash-operation or battery buffered PLCs.

Please take care that no dangerous process states develop during the voltage failure or on voltage return. On power failure special care has to be taken as the voltage return is contrary to switching the plant on and off by the operator as it takes place at an undefined moment.

 **Note** The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

New/Hot Re-Start

 - ”SeTup”, ”PLC station”, ”New/Hot Start”

Here you set up whether the PLC runs in New or Hot Restart operation.

- New Start: The PLC generally starts with standardized signal memory at the program start.

- Hot Re-start: The PLC continues the program at the interrupted point with the saved signal memory data.

Please ensure that during a voltage failure or on voltage return no dangerous process states develop. On power failure special care has to be taken as the voltage return is contrary to switching the plant on and off by the operator as it takes place at an undefined moment.

☞ **Note** The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

Without/With Overflow

- ”SeTup”, ”PLC station”, ”Without/With Overflow”

Here you can setup whether the overflow System Marker is processed or not during arithmetic operations. Choose ”With Overflow” when an AKF12 program should be used without changes. The performance with the setup ”With Overflow” corresponds with the performance of the AKF12. The performance with the setup ”Without Overflow” corresponds with the performance of an A250 User Program. Without overflow you can achieve faster and smaller user programs.

☞ **Note** The setup and changes carried out here are only valid after linking and loading of the program into the PLC.

Interface

- ”SeTup”, ”PLC station”, ”Interface”

Here you can choose the serial interface to be used . You can choose between COM1 and COM2.

BUs

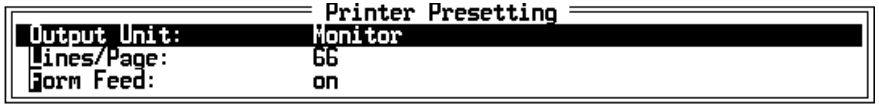
- ”SeTup”, ”PLC station”, ”BUs”

Here you can choose the Bitbus communications system. You can choose between Modnet 1/F and Modnet 1/W.

3.8.3 Printer

”SeTup”, ”Printer”

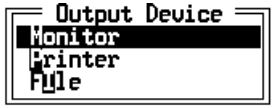
The following adjustments are possible in the printer menu:



Output Unit

”SeTup”, ”Printer”, ”Output Unit”

A menu appears from which you can choose the following:



- Display on screen, in pages
- Printout on printer that was selected under ”SeTup”
- Output to MS-DOS file (use whole path). If a file carrying this name exists already, the user is asked whether he wants to overwrite it or not.

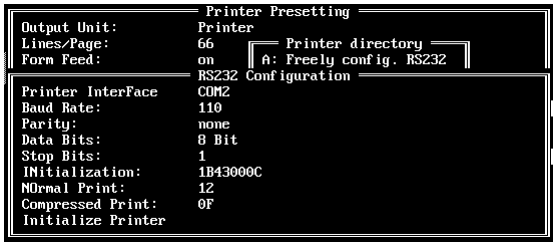
Another window opens under the “Printer” menu line in which you can choose the printer selection:

In case of DIN A4 printers the line length has to be followed: maximum length of signal comment: 32 characters.

Freely configurable RS232C port (serial)

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”Freely configurable RS232”

The following parameters have to be entered for a printer with freely configurable RS232 port:



Printer Interface

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Printer Interface”

You can toggle between ”COM1” and ”COM2”.

Please note the PLC is connected to COM1 as standard.

Caution When using a serial interface the following setup from the operating system level must be carried out before starting AKF:
MODE COM1:96,,,,P resp. MODE COM2:96,,,,P

Baud Rate

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Baudrate”

You can select one of the following baudrates:

110 Bd, 300 Bd, 600 Bd, 1200 Bd, 2400 Bd, 4800 Bd, 9600 Bd

The baudrate is setup by toggling.

Please note the setup on your printer.

Parity

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Parity”

Parity can be selected by toggling between ”even”, ”odd” and ”none”.

Data Bits

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Data Bits”

You can toggle between ”8 Bit” and ”7 Bit”.

Stop-Bits

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Stop-Bits”

You can toggle between ”1” and ”2” stop bits.

INitialization

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”INitialization”

ASCII sequence for initialization of the printer
e.g. character set, Auto LF etc.
or
FILE=dddd.eee
Name of the initialization file e.g. for laser printer.

NOrmal-Font

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”NOrmal Font”

ASCII sequence for initialization to 80 characters / line
corresponds with 10 cpi
e.g. for DRU096 ’15’
or
FILE=dddd.eee
Name of the initialization file e.g. for laser printer.

Compressed Font

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Compressed Font”

ASCII sequence for initialization to 143 characters / line
corresponds with 20 cpi
e.g. DRU096 ’0F’
or
FILE=dddd.eee
Name of the initialization file e.g. for laser printer

Initialize Printer

- ”SeTup”, ”Print”, ”Output Unit”, ”Printer”, ”Free Config.V24”, ”Initialize Printer”

By pressing the <Return> Key the setup parameters are accepted.

DRU 096

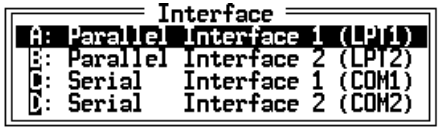
- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”PRT096”

The selection is effected here between:

DRU 120

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”PRT120”

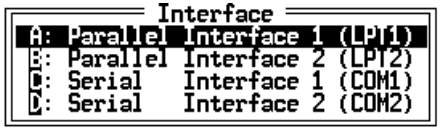
The selection is effected here between:



DRU 292

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”PRT292”

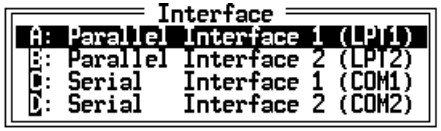
The selection is effected here between:



DRU 293

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”PRT293”

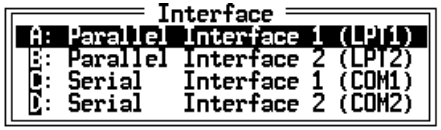
The selection is effected here between:



PRT 294

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”PRT294”

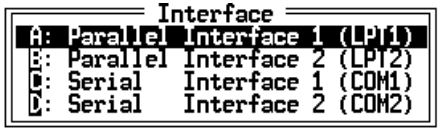
The selection is effected here between:



PRT 295

- ”SeTup”, ”Printer”, ”Outpu unit”, ”Printer”, ”PRT295”

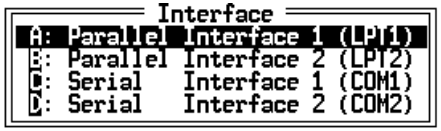
The selection is effected here between:



PRT 1200

- ”SeTup”, ”Printer”, ”Output unit”, ”Printer”, ”PRT1200”

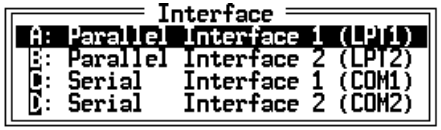
The selection is effected here between:



Printer_xyz

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”Printer_xyz”

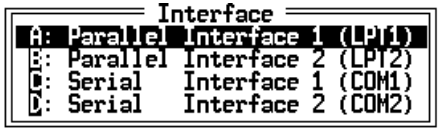
The selection is effected here between:



HP Laser-Jet II/III/IV

- ”SeTup”, ”Printer”, ”Output Unit”, ”Printer”, ”HP Laser-Jet II/III/IV”

The selection is effected here between:



Lines/Page

- ”SeTup”, ”Printer”, ”Lines/Page”

You can specify the number of lines per page corresponding to the paper format which is used in your printer here with the aid of the line editor.

☞ **Note** When using pure DIN A4 printers a maximum of 32 characters can be input as signal comments, otherwise a pagination causes a line scew after 132 characters.
The AEG printers (DRU 120, DRU 292 and PRT294) are effected.

Form Feed

The outputs can be carried out with or without form feed, depending on the type of printer used. If the form feed is switched off, the print program outputs a number of lines which corresponds to the number of blank lines.

You can toggle between ”on” and ”off”.

Extension of the Printer Driver

When using a new printer with the AKF system a simple editor is necessary with which you can edit the file ”DRUAKF40.DRV” in the sub–directory ”MICRO”.

In this file up to 15 printers can be named. A printer description is started by the name ”Printer=”. The then following name is displayed by the AKF software in a Pulldown Menu. Then 3 initialization sequences to be entered in hexadecimal follow. Please note the capital letters.

The initialization sequences can be taken from the handbook of your printer.

Examples:

Printer=DRU293
parallel/serial
1B3C Initialization
12 Switching to 80 characters
0F Switching to 132 characters

Printer=DRU096
parallel
1B3C Initialization
12 Switching to 80 characters
0F Switching to 132 characters

Definition of the Epson Fx-80
Printer=Epson FX-80
parallel
1C1B04 Initialization
040577 Switching to 80 characters
0F444488 Switching to 132 characters

Caution When using a serial interface the following setup from the operating system level must be carried out before starting AKF:
MODE COM1:96,,,,P resp. MODE COM2:96,,,,P

Restrictions exist concerning:

Character set: The IBM graphic character set is necessary.
Cable occupation: see Printer and PADT handbooks.

3.8.4 Color

”SeTup”, ”Color”

If your PADT is equipped with a color monitor and the corresponding board, you can set the colors by yourself. You may set the colors for pulldown windows, help windows, and message windows as you like it.

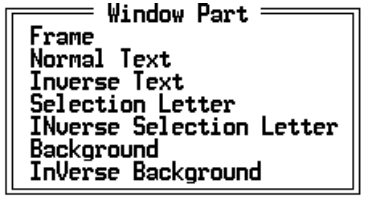
The software has to be called with the extension ”/COL”.

Furthermore, you may use the screen parameters ”/GR” for grey scales and ”/BW” for black and white.
If the software is called with these parameters, the system uses fixed settings.

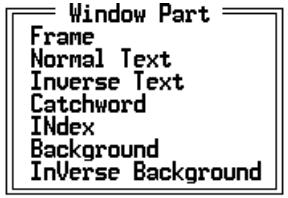
Recommendation: Color screen ”/COL”, liquid cristal display”/BW”, monochrome screen ”/GR”.

If the software is called with the parameter /COL, a further pulldown menu appears for each type of window. The colors can be selected for the following parts of the window:

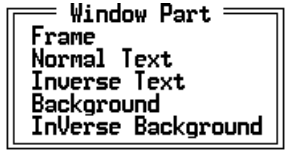
Pulldown Window



Help Window



Message Window



A window appears with a color pallette and you can select a color value from this window by selecting one of these menu lines.

The current adjustment is visible in the demonstration windows.
You can leave the ”color” menu by the <Esc> key.
The new color settings appear after you leave the SeTup pulldown menu.

3.8.5 Modem-Connection

”SeTup”,”Modem-Connection”

Note The use of the modems only works correctly when AKF is started with the parameter /Modem.
Only in this case the menufunction ”Modem-Connection” is available in the ”Setup” Menu.

The PLC can be controlled remotely via a modem connection. A Hayes compatible modem is necessary which must be connected to COM1 or COM2. The modem must be capable of 9600 baud and be setup as follows:

- 9600 Baud
- Odd Parity
- 8 data bits
- 1 stop bit

For the modem connected to the PLC the same requirements are valid. On the PLC side a zero-modem resp. a cable such as YDL 37 must be cennected in between.

Before connecting to the PLC the program M_SLAVE.EXE installed by AKF must be started. It configures the modem for auto-answer-operation. With self-adaptive modems the correct transmit parameters for the coupling to the PLC are also set up (9600 baud, Odd Parity, 8 data bits and 1 stop bit).

Following functions are offered for establishing the modem connection:

- Init-String: Hereby an Init-String for the modem initialisation is preset.

e1:	Output of modem messages
&c0:	Ignore status of carrier signal of remote modem
w1:	Output of connection messages
&d0:	DTR not evaluated
–k1:	NMP-Operation not evaluated
%c3:	Allow of MNP5- and V42.bis- Compression
f0:	Acknowledge capabilities of the remote modem (Type of modulation and speed)
\n3:	Establishment of a normal, MNP of V.42 connection

- Addition: Here it is optional to state additional parameters (max. 21 characters) which are sent after the Init-String to the modem.

  e.g.:

M1:	Loudspeaker on till carrier signal acknowledgement
L1:	Low volume

- Telephone number: Here you can enter a max. 28 digit telephone number, preceeded by P or T to change the pre-set dialling proceedure (pulse or tone dialling).

- Dialling: Sending of the Init-String and the additional string and dialling of the entered number.

- hang Up: Interruption of the telephone connection.

Index

G

H

I

J

K

L

M

N

O

P

T

U

W