

AMK

AMKASYN

THREE-PHASE SERVO DRIVES

AMKASYN

Programmable Controller PS

Programming Software APROS

Software 0205

Subject to technical change without prior notice

0595.E

Teile-Nr.: 25797

AMK

Arnold Müller, Antriebs- und Regeltechnik GmbH & Co.KG, D-73230 Kirchheim/Teck,
Tel.: 07021/50 05-0, Telefax: 07021/50 05-176

Content

1	INTRODUCTION	3
2	PROGRAMMING WITH APROS.....	4
2.1	Opening of a project	5
2.2	Input/Modification of the STL blocks and the assignment list.....	5
2.3	Compiling the STL blocks and generating an executable PS program	6
2.4	PS program transfer to the AZ-PSx.....	7
2.5	PS program test.....	7
3	DIALOG DESCRIPTION	8
3.1	Functions dialog	8
3.2	Choice dialog.....	8
4	FORMAT AND OPERATION OF MENU.....	10
4.1	Main menu item "Project"	11
4.2	Main menu item „Programming“	13
4.2.1	Functions in the main menu item „Programming“	13
4.2.2	Editor-Menu	15
4.3	Main menu item "Test"	18
4.3.1	Description of the window contents and the related function keys	20
4.3.2	Description of the generally valid function keys	22
4.3.3	Description of the test title menu.....	23
4.4	Main menu item "Startup"	26
4.5	Main menu item "options"	27
4.6	Help	28
5	ON-LINE HELP	29
6	PRINTOUT REPRESENTATION	30
7	ANNEX.....	31
7.1	Editor description.....	31
7.2	Compiler Description	33
8	IMPRESSUM.....	40

1 Introduction

The AMK PS Programming Software "**APROS**" renders possible programming, test and monitoring of the AMKASYN Programmable Control (PS) as well as a project oriented administration of the generated statement list (STL) programs.

The programming software APROS requires the following hard- and software conditions:

- User memory: 640 kB
- Available user memory for APROS: >590 kB
- PC-286, -386 or -486
- DOS 3.3 or higher

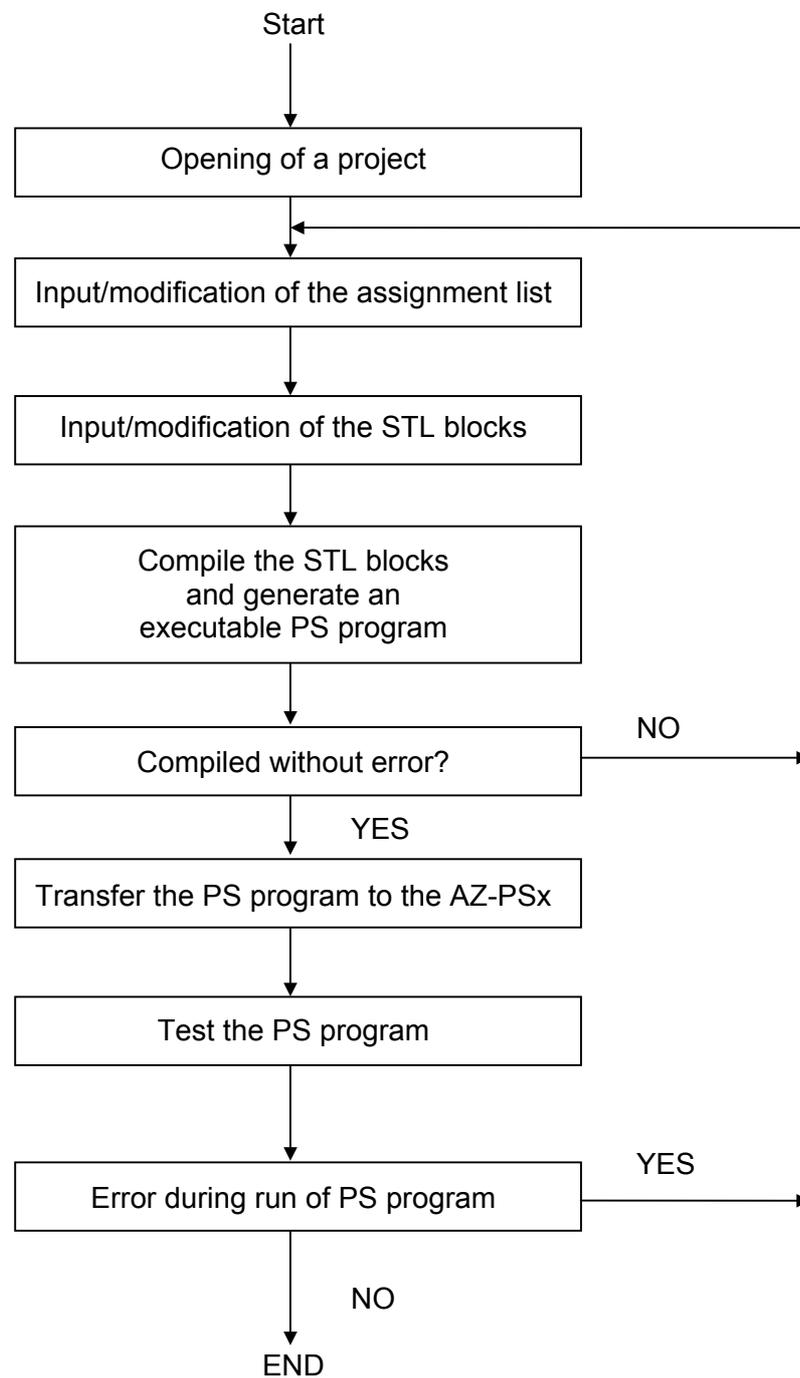
Detailed notices concerning hardware and software conditions as well as the installation of APROS on the PC are to be taken from the file "readme.txt". The installation diskette „APROS“ contents this file.

Characteristics of the PS programming software APROS:

- To operate APROS no knowledge of the MS-DOS operating system is needed.
- APROS is multilingual.
- All user operations are supported by a help function.
- Program parts (OB, PB, FB, DB blocks) are administered collectively by APROS in a "project".
- The user can access to its blocks and the assignment list via the linked project.
- The program development cycle "correction -> test -> correction" is supported by the APROS menu format.
- Programming in statement list STL (with reference to STEP 5).
- The data processing is supported in bit mode , byte mode , word mode and double-word mode.
- It is possible to use symbolic identifiers, symbolic values and symbolic jump labels.

2 Programming with APROS

For generation of an executable and faultless PS program the following steps are necessary:



APROS offers all necessary programming aids to carry out these steps.

2.1 Opening of a project

The programs for the PS are administered by APROS as so-called "projects". All STL blocks together with the assignment list as well as user specific options are summarized in a project. The individual elements can only be accessed within this project. For his work with APROS the user needn't have special knowledge in the PC and its operating system.

Each project consists of the STL blocks, the assignment list and user specific information. Symbolic identifiers can be assigned to the operands of the STL blocks in the assignment list. The user specific information consists of:

- the project name,
- the project title,
- the target PS and
- the notes.

The **Project name** is the name under which the PC stores this project on diskette or hard disk. If this project is to be worked with subsequently, APROS refers to this name. The project name has a maximum of 8 characters and must consist of letters and numbers but has to begin with a letter.

The **Project title** can be chosen freely. It can consist of special characters and blanks besides letters and numbers. The maximum length of the project title is 127 characters.

The choice of the **Target PS** determines by which PS of the AMKASYN range the PS program shall be processed (not yet used).

The **Project note** can also be described freely. All letters, numbers and special characters are permissible. The maximum length of the project note is 256 characters and can be used for comments on the project.

2.2 Input/Modification of the STL blocks and the assignment list

The programming and test program APROS contains an editor (compare chapter 4.2.2. and 7.1) by which the STL blocks and the assignment list can be input and corrected. The editor also supports the error correction while the cursor jumps to the faulty line after a program compilation.

Before calling the editor, all available blocks are presented for choice. The block which is to be edited has to be selected in the list.

If a new block is to be created APROS organizes the name input, the choice of the block type as well as the determination of the block number, also called "Index".

2.3 Compiling the STL blocks and generating an executable PS program

After input or correction of the STL blocks and/or the assignment list the PS program has to be compiled. The integrated compiler converts the input STL instructions into a PS comprehensible form.

The compiler opens a window as follows:

```
AMKASYN STL Compiler Vx.x, (C) AMK
Arnold Müller GmbH & Co. KG
Antriebs- und Regeltechnik
73230 Kirchheim/Teck
```

It then processes all blocks belonging to the project plus the assignment list and debugs all lines. While compiling the name of the block being processed is displayed in the window.

The compiler subdivides into three groups:

- Fatal errors,
- STL errors,
- Warnings,

Fatal errors are directly displayed in the compiler window. They refer to a given block. Fatal errors are reported, for example, if:

- an existing block can't be read from the hard disk.

STL errors refer to a program line of a block. After compilation these errors are displayed by the editor for correction. STL errors, for example, are:

- A false or unknown instruction,
- Violating the parameter value range,
- Jump on a non-existing label,
- Call of a block which doesn't exist in the project.

Warnings are messages from the compiler to the programmer. If the compiler displays a warning the PS program can be executed correctly by the PS but there can be a hidden programming error. Warnings are displayed, for example, if:

- there is a block in the project that isn't called.

After finishing the compilation the following message is displayed in case of correct compilation:

```
Compilation finished without error
File with STL code created, length xxxx byte
```

After this message the PS program can be transferred to the target PS, the target PS can be started for the PS program test.

If the compilation wasn't correct, the following message is displayed:

- "Compilation finished with errors"

The errors recognized by the compiler then have to be corrected before the program can be processed by the PS. If there are STL errors the editor jumps to the faulty lines for error correction.

2.4 PS program transfer to the AZ-PSx

The PS program has to be transferred from the PC via the AMK-S-Bus to the AZ-PSx. There it is stored in the remanent memory (EEPROM). After resetting (by the instruction „reset PS“, compare chapter 4.3.3. or after „restart“ of PS) the PS loads the new program in its main memory and processes it cyclically.

2.5 PS program test

APROS supports the program test by means of different aids:

- Processing in different modes such as run, cycle, instruction and step.
- Indication of STL instructions with comments.
- Display and influence of internal PS data such as ACCU1, ACCU2, RLO,
- Display and influence of input-, output and flag values.
- Display and influence of data blocks.
- Display and influence of counters and timers in all functions.
- Display and influence of the PS information in different data type and data format representations.
- Setting of breakpoints.
- Display representation by variable window technics.
- Configurable representation of different PS data in a common monitor window.

3 Dialog description

APROS uses two types of dialogs:

- Functions dialog
- Choice dialog

3.1 Functions dialog

The functions dialog is used when certain statements must be interrogated to carry out a function. The dialog is constrained, i.e. the user is led from one dialog item to the next. Functions that are at the user's disposal in the individual dialog items are represented by function keys. In the functional dialog the keys <ENTER> and <ESC> have the following meaning:

- <ENTER>: Confirmation of the input or accept of the displayed setting and change over to the next dialog item.
- <ESC>: Abort of dialog and rejection of input.

3.2 Choice dialog

The choice dialog is used when certain settings can be selected from windows or when the project or the block for which the function is to be used can be selected. The choice dialog works with the dialog keys "OK" and "abort". With the "OK" key the input/choice are accepted, with the "abort" key the choice dialog is stopped. In the choice dialog jumps can be made between the dialog items. In the choice dialog the following keys have a meaning:

- <ENTER>: Take over of the input and change over to the next dialog item or execution of the dialog key function.
- <ESC>: Abort of dialog and rejection of input.
- <TAB>: Cursor to next dialog item.
- <SHIFT><TAB>: Cursor to last dialog item.

The functions and choice dialog contain three types of input variables:

- numerical input,
- name input,
- choice input.

For **Numerical input** APROS expects a number. Depending on dialog item this can be a decimal number, a hexadecimal number or a binary number. The figures valid as inputs depend on this.

- Decimal number: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -
- Hexadecimal number: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f
- Binary number: 0, 1

With <ENTER> is checked whether the input number is conform to the value range valid for the dialog variable. In this case the number is accepted and the dialog item left. If the input number is not conform to the valid range or if it contains invalid characters it is returned to the dialog item's beginning.

With <ESC> the numerical input can always be left, the last inputs, however, will be ignored.

For **Name input** APROS expects a name or a whole line depending on dialog item Chapter 4 („Structure and operation of the menu“) describes the individual cases.

For name input the positions reserved for the name are displayed inversally. At the first input place the cursor blinks. With <ENTER> the input is accepted and the dialog item left.

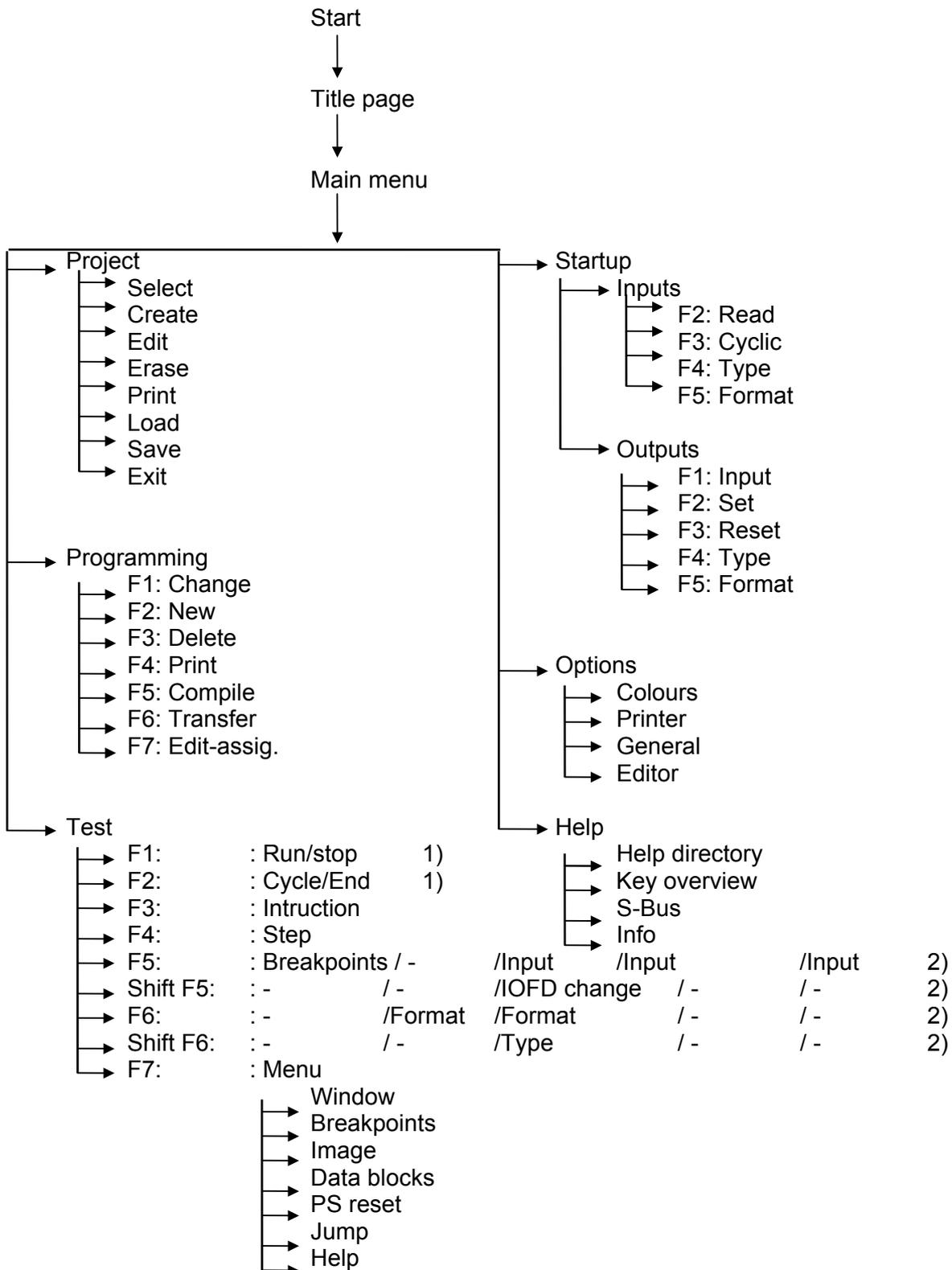
The name input can be left with <ESC>. The last input, however, is ignored.

For **Choice input** APROS displays a list of choice for possible inputs. With the cursor keys the cursor can be positioned on the desired element. With <ENTER> the input is accepted.

The choice can always be left with <ESC>. The last input, however, is ignored.

4 Format and operation of menu

The menu is tree-structured and consists of three planes:



1) Depending on PS status: „Stop“/„Run“
 2) Depending on active window: „Program“/„Status“/„Image“/„Count Tim.“/ „Observe“

The desired menu item can be selected with the cursor keys and called with <ENTER>. With <ESC> the current submenu can be left and the next higher menu plane selected.

Before existing files are overwritten, texts deleted or files created APROS asks for the user's confirmation.

After call of APROS the title page appears. In the title page the address of AMK, the version number of APROS, the delivery date and the customer's name are displayed. By using any key the main menu is selected.

APROS can be left from any place of the menu with <ALT><X>.

In special files APROS stores certain settings of the user such as colour settings, monitor design in the menu item "test" as well as printer settings etc.

Each menu item is explained by a help text. The help function is called through function key <F10>. A help window with the corresponding help text is hereby displayed.

The **Main menu** represents the supreme menu plane. After leaving the title page the main menu is displayed. The main menu contains the following menu items:

- Project
- Programming
- Test
- Startup
- Options
- Help

These menu items consist of all functions for startup, programming and testing of a PS.

4.1 Main menu item "Project"

In the menu item **"Project"** the user disposes of the following functions for the project administration:

- Select
- Create
- Edit
- Erase
- Print
- Load
- Save
- Exit

In the menu item >**"Select"** the project which is to be processed next can be selected. After call of the menu item all existing projects are displayed for choice. With the cursor keys the desired project is selected and confirmed with <ENTER>.

In the menu item project->**"Create"** a new project can be created. After call of the menu item it is asked for input of project name, project title, the target PS and a comment.

Project titles and notes are input directly. The target PS is determined by choice of the displayed possible target PS. With the cursor keys the desired target PS is selected (at present only AZ-PSx is possible) and confirmed with <ENTER>. The project settings are remanently stored with the "OK" key of APROS under the project name. The process can be stopped at any place with <ESC>.

In the menu item ->"**Edit**" the project settings such as project title, comment and target PS can be changed. After call of the menu item the projects already existing are displayed for choice. After confirming the choice of the desired project with <ENTER>, the user is asked to change the following project settings:

- Project title,
- Target PS,
- Notes

The input is done as described under menu item project->"**Create**".

In the menu item ->"**Delete**" a whole project can be deleted from the PC. When a project is deleted all blocks, assignment list and all user specific information are erased from the PC. After call of the menu item "**Erase**" all projects are displayed for choice. By means of the cursor the desired project is selected and confirmed with <ENTER>. APROS now asks whether the selected project is really to be deleted. After confirming with "YES", the project is deleted. When answering with „NO“ or <ESC> the menu item "Delete" is left and the selected project remains.

In the menu item project->"**Print**" the assignment list as well as all STL programs of a project are printed.

In the menu item project->"**Load**" a complete project can be loaded from a backup diskette into the PC. The standard backup unit is the 3;5" diskette drive. The diskette with the projects is inserted in the 3,5" diskette drive. After call of the menu item all projects are displayed for choice. The desired projects are selected with the cursor keys and confirmed with <ENTER>. The query follows:

- "Shall the global options also be copied?".

If answered with „NO“ the project settings, the assignment list and all blocks are loaded onto the PC's hard disk. If answered with "YES" the settings are also loaded in the main menu "options". During the load procedure the names of the blocks which will be loaded are displayed in a report window. With the display:

- "Project loaded without errors",

the diskette can be erased from the drive. An error message is displayed when errors occur during the load procedure.

In the menu item ->"**Save**" a complete project can be transferred from the PC to a diskette. The standard backup unit is the 3,5" diskette drive. The diskette with enough capacity is inserted. After call of the function "project -> save" all projects available on the PC are displayed for choice. The desired project can be selected with the cursor keys and confirmed with <ENTER>. The query follows:

- "Shall the global options also be copied?".

If answered with „NO“ the project settings, the assignment list and all blocks are copied onto the backup unit. If answered with "YES" the settings are also saved. During the backup the names of the blocks which are saved are displayed in a report window. If the message follows:

- "Project saved without errors",

the diskette can be erased from the drive. An error message is displayed when errors occur during the backup.

When the menu item->"**Exit**" is called APROS is left after a further confirmation with „YES“. If answering with „NO“ it is returned to the menu item. When APROS is ended the user settings are stored in the PC's file system and automatically restored when APROS is restarted.

4.2 Main menu item „Programming“

4.2.1 Functions in the main menu item „Programming“

In the menu item „**Programming**“ the assignment list and the STL blocks of the project are processed. This menu item consists of the following functions:

- Change,
- New,
- Delete,
- Print,
- Compile,
- Transfer,
- Edit assign.

If no project was specified before call of the menu item „Programming“ the dialog for the project choice is inserted. In this dialog the desired project has to be specified, then the menu item „Programming“ is continued.

After call of the menu item „Programming“ a block window and the function keys <F1> to <F7> appear. In the block window all blocks of the project are displayed. The function keys contain a grammalogue of the functions that are activated by using them.

Functions:

With <F1> programming->"**Change**" a block's content can be changed.

When pressing <F1> the cursor appears in the block window. With the cursor keys the desired block is selected. After confirmation with <ENTER> the editor (compare chapter 4.2.2) is called, the block content can be changed. When finishing with <ESC> it is returned to the menu item „Programming“.

With <F2> programming->"**New**" a block can be created and input.

By pressing <F2> three new windows are displayed:

- Name window,
- Type window,
- Index window.

At first the cursor items at the name window. Here the block name is input. A false input can be corrected with the cursor keys and with the key. The name input is confirmed with <ENTER> and the cursor then marks the type window.

In the type window the block type has to be selected with the cursor keys out of the different block types. After confirmation with the <ENTER> key the cursor moves to the index window.

In the index window a new block number is displayed for the block. If the new block is to get another block number the number can be overwritten in the index window. After confirmation with <ENTER> it is checked whether a block with this number already exists. In this case a message is displayed that the input in the index window has to be changed. If the block is still free the new block is created and the editor called. Now the statements for the program must be entered. After the end of the input it is returned to the menu item „Programming“. The input dialog can be aborted at any place with <ESC>.

With <F3> programming ->“**Delete** ” a block can be deleted.

After pressing the function key <F3> the cursor appears in the block window. With the cursor keys the desired block is selected. After confirmation with <ENTER>, it is again asked whether the block is to be deleted. When answering with 'YES' the block is deleted and the dialog returns to the menu item „Programming“. When answering with „NO“ it is returned at once to the menu „Programming“ without deleting the block.

With <F4> programming->“**Print**” the content of the block can be output to the printer.

After pressing the function key <F4> the cursor appears in the block window. The desired block is selected with the cursor keys. After confirmation with <ENTER> the block is output to the printer. Then it is returned to the menu item „Programming“.

With <F5> programming->“**Compile**” the blocks are compiled. Hereby the program that can be processed by the PS is generated.

With <F5> the compiler is called. A window is opened in which the compiler displays its version number. During the compilation the name of the block is displayed in the window that is being compiled.

After compilation of all blocks the compiler outputs the information whether an error was detected or whether the project could be compiled without errors. If an error was detected in a block the editor is automatically called after pressing any key. In the concerned block the incorrect STL line is displayed. After the correct compilation the PS program length is displayed. By pressing any key it will be selected whether to transfer the compiled program to the PS or whether to return directly to the menu item „programming“. If answered with 'YES' the function <F6> programming->“**Transfer**” is called and executed.

<F6> programming->“**Transfer**” transfers a compiled PS program via the serial interface to the AZ-PSx.

With <F6> once again all blocks of the compiled PS program are displayed and then sent to the PS. The regular transfer of the PS program is displayed as well as whether the transfer had to be stopped due to transfer errors. In this case it is returned to the menu item „Programming“ by pressing any key. If the transfer to the PS was successful a query is started whether the test menu is to be called or whether to return to the menu item „Programming“.

With <F7> programming->“**Edit-assig.**” the assignment lists are created or changed.

With the function key <F7> the editor is called by means of the assignment list. Now the content of the assignment list can be changed. When editing is finished the dialog returns to the menu item „Programming“.

4.2.2 Editor-Menu

With the function keys <F1>, <F2> (and choice or input of the block to be edited, compare chapter 4.2.1) or <F7> in the main menu item „Programming“, the editor is called. The editor functions can be called via the function keys or the title menu.

After call of the editor, the editor window with heading and footer is displayed as well as the function keys with editor functions at the bottom of the screen edge.

The editor window's header contains the name of the block to be changed or created. From left to right the footer contains the following information:

Block on/ - marker mode on/off
<Field empty>
Gk/GK - Pay attention to capital letters/or not (compare. chapter 4.5)
G/<empty> - block was changed/not changed
E/Ü - Insert/overwrite mode
Line:xxxx - Number of the cursor line
Column:xxx - Number of the cursor column

The following function keys are available:

- <F1> Save
- <F2> Error
- <F3> Block
- <F4> Copy/Shift
- <F5> Insert
- <F6> Search
- <F7> Menu

The function key <F1> "**Save**" stores the current text in the file. With the function key <SHIFT>-<F1> or <ESC> the editor is left.

With <F2> "**Error**" or. <SHIFT>-F2 the editor jumps to the next or the former error the compiler detected during the last compilation.

With **<F3> "Block"** the marker mode is switched on or off. The status „On“ is displayed in the footer's left corner with „Block on“. When the marker mode is switched on a text area can be marked with the cursor keys.

With **<F4> "Copy"** the text area formerly marked with <F3> and the cursor keys is copied into the intermediate memory. By pressing the function key <SHIFT>-F4 "Shift" the text area marked is deleted after the copying. If the function key <F4> is pressed without a text area being marked the current line is copied into the intermediate memory.

With **<F5> "Insert"** the content of the intermediate memory is inserted at the current cursor's place. The content of the intermediate memory remains when the block or even the project is changed.

With **<F6> "Search"** a character string is searched. Function key <F6> searches forward, function key <SHIFT>-<F6> searches backward. It is either searched for the character string that was latest input as a search string or if a character string is marked in the text for this character string.

With **<F7> "Menu"** the title menu is called.

The title menu of the editor contains the following menu items:

- Block
- Process
- Search
- Substitute
- Help

All functions of the function keys can also be called via the title menu. In the menu items of the title menu the editor functions are fully named. If the function can also be called via a function key the function key is displayed behind the function name. The inputs for the functions „Search“ and "Substitute" are made in the title menu by dialog.

The menu item **"Block"** contains the following functions:

- Insert block
- Save block F1
- End editor SHIFT-F1

In the menu item **"Insert block"** the content of a block out of the block library can be inserted in the current program text. The needed block is chosen via a choice menu. When confirming the choice with the <ENTER> key the content is inserted at the current cursor position. The menu item can be left with <ESC> without any change of the program text.

By calling the menu item **"Save block"** the content of the block is saved on the PC.

By calling the menu item **"End editor"** it is asked after a block modification whether the new content is to be saved. By answering the query with „YES“ the new block content is saved on the PC and the editor left. By answering the query with „NO“ the editor is left without overwriting the old block content. With the function key "Abort" the editor isn't left.

If there is no change in the current block, the editor is left without query.

The menu item "**Process**" contains the following functions:

- Insert/Overwrite INS
- Block on/off F3
- Copy block F4
- Shift block SHIFT-F4
- Insert block F5
- Insert line ALT-I
- Delete block/characters DEL
- Delete line ALT-Y

The function "**Insert/Overwrite**" changes from the insert mode to the overwrite mode and vice versa.

With the function "**Block on/off**" the marker mode is switched on or off. The status „On“ is displayed in the footer's left corner with „Block on“. With the marker mode switched on a text area can be marked by means of the cursor keys. The status is displayed in the footer's left corner.

By calling the function "**Copy block**" the text area formerly marked with <F3> and the cursor keys is copied into the intermediate memory. The text remains at its original place. By calling the function "Copy block", without a text area being marked, the current line is copied into the intermediate memory.

By calling the function "**Shift block**" the marked text area is shifted to the intermediate memory and deleted in the program text.

By calling the function "**Insert block**" the content is inserted in the intermediate memory at the current cursor position. The function "Insert block" doesn't change the content of the intermediate memory.

The function "**Insert line**" inserts a space line before the line in which the cursor is positioned.

The function "**Delete block/character**" deletes the character on the cursor position or the marked text area.

With the function "**Delete line**" the line in which the cursor is positioned is deleted.

The menu item **"Search"** contains the following functions:

- Search F6
- Next error F2
- Previous error SHIFT-F2

The menu item **"Search"** calls the input dialog for the search function. In the search dialog window the following parameters can be input:

- | | |
|---|---|
| Forward/Backward | - Search forward or backward from the current cursor position, |
| Attend to capital letters/
small letters | - Differentiation between capital letters/small letters during search |
| Search for: | - Character string to be searched for in the program text |

After confirmation of the inputs with „OK“ the search for the character string starts. If the character string is found, the search is stopped and the corresponding marked part of the program text is shown in the editor window. With the functional keys <F6> or <SHIFT> F6 the search can be continued.

The menu item **"Substitute"** contents the input dialog to substitute one character string by another. After call of a menu item the substitute dialog window is displayed with the following content:

- | | |
|-------------------------------------|---|
| Forward/Backward | - Substitute forward or backward from the current cursor position, |
| Attend to capital/
small letters | -Differentiation between capital and small letters small during search |
| Search for: | - Character string to be searched for in the program text |
| Substituted by: | - Character string which is to substitute the searched character string in the program text |

If the inputs are confirmed with „OK“, the substitution begins. With the <ESC> key the dialog is stopped. During the substitution each character string that was found is successively displayed and presented for substitution. With the function key <F1> the found character string is substituted , with <F2> the position in the program text is skipped and the next character string is searched. With <F3> the substitution is finished. If no further character string is found or if the substitution was stopped, the number of the executed substitutions is shown in a window.

With the menu item **"Help"** the help index is displayed. By means of the cursor keys a catchword is selected from the list. With <ENTER> the help text belonging to the catchword appears; with <ESC>the „help“ menu is left.

4.3 Main menu item "Test"

After the STL program has been transferred to the PS it can be tested under the menu item "Test" (Remark: The cycle time of the PS program is extended during the test mode by the factor 2 to 3).

With the menu item **"Test"** several windows with data from the PS, a title menu and the function keys <F1> to <F7> appear. With the <TAB> key or the <SHIFT><TAB> key the active window is changed. The active window is marked with a double frame and the cursor appears in the window. Inputs and changes of values can only be done in the active window.

The following windows can be displayed:

- Program window,
- Status window,
- Image window,
- Counter/timer window,
- Observation window

The functions keys activate the following functions:

- F1: Run/Stop 1),
- F2: Cycle/End 1),
- F3: Instruction,
- F4: Step,
- F5: Function depends on the active window,
- Shift F5: Function depends on the active window,
- F6: Function depends on the active window,
- Shift F6: Function depends on the active window,
- F7: Menu.

1) Depending on the PS-status "Stop" / "Run".

The title menu <F7> contains the following items:

- Window,
- Observe,
- Image,
- Data blocks,
- Jump,
- Options

If the user leaves the menu the PS remains in its current status ("Run" or "Stop").

4.3.1 Description of the window contents and the related function keys

In the **Program window** the source program of a block is displayed. By means of the cursor keys the cursor can freely be positioned in the program lines. After stopping the PS with the functions step, instruction or when a breakpoint is reached the block being processed is displayed in the program window. The program line to be processed next is displayed inversally.

Breakpoints are displayed in the program window by yellow writing of the corresponding STL line. Breakpoints can be set in the program window by positioning the cursor through the cursor keys in the desired STL line and by confirming it with the **<F5> key** (Breakpoints). The breakpoint is set and marked through yellow writing. By pressing the **<F5> key** once again the characters are again displayed in grey.

In the **Status window** the contents of the PS' internal registers as well as the status and the possible errors of the PS are displayed. With the function key **<F6>** (format) the data format of the accu contents can be changed. The following representations are possible:

- Decimal without signs,
- Decimal with signs,
- Hexadecimal and
- Binary

The selected data format appears in the title line of the window. Furtheron the contents of the following flags are displayed:

Name of the flag	Abbreviation
Result of logical operation	RLO,
First input bit scan	Erab,
Status	Stat,
Or stack	O stack, Os,
Display flag 0	Anz0,
Display flag 1	Anz1,
Overflow	OV.

With ENTER the cursor can be moved to the next display value; the previously input value is accepted.

In the window footer the current PS status, in case of disturbance the error code is displayed. Error code '00' displays the faultless operation of the PS.

In the **Image window** inputs, outputs, flags and data blocks of the PS are displayed in one APROS block. With the function key **<SHIFT><F5>** (IOFD-change) it can be switched between these areas.

With the function key <F6> (format) the data format can be changed. The following data formats can be selected:

- Binary,
- Decimal without signs
- Decimal with signs
- Hexadecimal

With the function key <SHIFT><F6> (type) the data type can be set:

- Byte (8 bit),
- Word (16 bit) or
- Double-word (32 bits)

With the cursor keys an element in the window can be selected. This element appears once again with designation, address and value in the footer of the window. With <F5> (input) a cursor appears in the window's footer. By inputting a new value in the window's footer this element can be changed. With <ENTER> the new value is accepted and the next element is displayed for change. The input can be left with <ESC>.

With <INS> the element selected by the cursor can be taken over to the observation window.

In the **Counter/Timer window** the counters and timers of the PS are displayed.

The data of the counters and timers are displayed as described below: (Capital letters are maintained in the display, small letters are substituted by the corresponding values):

Timers:	Txxx bb zzzz Ee Aa hh:mm:ss.sss
- T	Timer identification
- xxx	Timer number
- bb	Mode in which the timer was started SI, SV, SE, SA, SS
- zzzz	Actual status of the timer Passive - Timer not started, Active - Timer runs, End - Timer was started and has finished
- e	Actual status at timer input
- a	Actual status at timer output
- hh	Timer's rest time - Hours
- mm	Timer's rest time - Minutes
- ss.sss	Timer's rest-time - Seconds

Counter	Zxxx Ss Rn Zvv Zrr Aa zzz
-z	Counter identification
-xxx	Counter number
-s	Actual status of set input
-n	Actual status of reset input
-v	Actual status of forward counting input
-r	Actual status of backward counting input
-a	Actual status of output
-zzz	Counter content

With function key <F5> (input) the values of the counters and timers can be modified in the window's footer. With <ENTER> the new values are accepted and the next counter or timer is displayed for modification. When the cursor is displayed as an underline the input of a numerical value is expected. If the element to be input is displayed inversally a new assignment has to be selected with the cursor keys <arrow upward> and <arrow downward>. The input status can be left with any function key or with <ESC>.

With the <INS> key the counter or timer selected by the cursor can be taken over to the observation window.

In the **Observation window** status and contents of the selected inputs (I), outputs (O), flags (F), counters (C) and timers (T) are displayed.

For choice of the I, O, F, C and T the element concerned is marked in the image or counter/timer window with the cursor keys and the <INS> key. In each line of the observation window an element and its value is displayed in the selected format. Elements in the observation window are deleted by selecting them with the cursor keys <arrow upward> and <arrow downward> and by using the key.

An element can be selected in the window with the cursor keys. This element appears once again in the window's footer with designation, address and value. With <F5> (input) a cursor appears in the window's footer. By input of a new value in the window's footer this element can be modified. With <ENTER> the new value is accepted and the next element is displayed for modification. The input can be left by pressing any key.

4.3.2 Description of the generally valid function keys

With the menu item **"Test"** the function keys <F1> to <F7> are displayed with the following assignment:

- F1: Run/Stop 1),
- F2: Cycle/End 1),
- F3: Instruction,
- F4: Step,
- F5: Function depends on the active window,
- Shift F5: Function depends on the active window,
- F6: Function depends on the active window,
- Shift F6: Function depends on the active window,
- F7: Menu.

1) Depends on PS status "Stop" / "Run".

With the functional key <F1> test->"**Run**" (in the PS status "Stop") the user program is processed in the test mode. The values of the selected elements are displayed in the observation window. The remaining window contents are updated with a „Stop“ of the PS .

With the function key <F1> test->"**Stop**" (in the PS status "Run") the PS is offset to the "Stop" status. After the „Stop“ of the PS all window contents are updated.

With the functional key **<F2>** test->**"Cycle"** (in the PS status "Stop") the PS executes one complete cycle of the user program. All window contents are then updated.

With the function key **<F2>** test->**"End"** (in the PS status "Run") the test menu is left and the PS remains in the status „Run“.

With the function key **<F3>** test->**"Instruction"** the PS executes the user program until it reaches the next instruction that is displayed in the program window. With this function block calls can be executed in the „Run“. Window contents are updated when the PS stops.

With the function key **<F4>** test->**"Step"** the PS executes the next instruction of the user program. All window contents are then updated.

With the function key **<F7>** test->**"Menu"** it is switched to the test title menu.

4.3.3 Description of the test title menu

In the menu item **Test->"window"** it is selected which windows are to be displayed during the PS test. The program window is always displayed.

Choice of windows: By means of the cursor keys the cursor is positioned in the desired „window“ line:

- [] Status window,
- [] Observation window,
- [] Image window,
- [] Counter/timer window

With the „space line“ the window is marked with [x]. The windows marked with [x] are displayed during the test.

In the menu item **Test->"breakpoints"** all breakpoints of the breakpoint list are displayed in one window. By input of the block name as well as the line number new breakpoints can be set in the window's footer.

In the menu item "test->breakpoints" the following functions can be executed:

- F1: New,
- F2: End,
- F3: Delete,
- F4: Delete all.

With the function key **<F1>** test->breakpoints->**"New"** a new breakpoint can be added to breakpoint list. In the window's footer two input fields are presented. In the left input window at first the name of the block in which the breakpoint is to be set has to be input. After **<ENTER>** the STL line number has to be input. By confirming again with **<ENTER>** the breakpoint is stored. If the input block does not exist or if the input line is no instruction line an error message is displayed.

With **<F2>** test->breakpoints->**"End"** the menu item test->„breakpoints" is finished and it is returned to the menu item "test".

With **<F3>** test->breakpoints->„**Delete**“ the breakpoint marked with the cursor is deleted.

With **<F4>** test->breakpoints->„**Delete all**“ all breakpoints are deleted.

In the menu items "**Test->image**" and "**Test->data blocks**" the content of the image window can be set up.

After call of the menu item "**Test->image**" three choice windows, an "OK" key and an „abort“ key are displayed. The content of the image window can be:

- PS inputs,
- PS outputs and
- PS flags

In the choice window "**Format**" the format of the displayed elements is selected. The following numerical formats are presented for choice:

- Hex: Values displayed in hexadecimal form
- Dec :Values displayed in decimal form as a positive number
- +/-dec :Values displayed in decimal form with signs
- Binary: Values displayed in binary form

In the choice window "**Type**" the display of the numerical values can be selected:

- BYTE: Display as 8 bit values (0...255 bzw. -128...+127)
- WORD: Display as 16 bit values (0...65 535 bzw.-32 768...-32 767)
- LONG: Display as 32 bit values (0...4 294 967 295 bzw.
-2 147 483 648...+2 147 483 647)

The desired display is selected with the cursor keys and confirmed with <ENTER>. With<ESC> the choice can be aborted. With "OK" the new setting is accepted and it is returned to the menu "test".

After call of the menu item **Test->"data blocks"** three choice windows, an „OK“ key and an "abort" key are displayed.

In the window "display choice" the display of the active block or the display of a certain block in the image window can be selected. If a certain block is to be selected its number has to be input.

In the choice window "**Format**" the numerical format of the displayed values can be selected:

- Hex : Values displayed in hexadecimal form
- Dec : Values displayed in decimal form as a positive number
- +/-dec : Values displayed in decimal form with signs
- Binary: Values displayed in binary form

|

In the choice window "**Type**" a display of the numerical values can be selected:

- BYTE: Display as 8 bit values (0...255 bzw. -128...+127)
- WORD: Display as 16 bit values (0...65 535 bzw. -32 768...+32 767)
- LONG: Display as 32 bit values (0...4 294 967 295 bzw.
-2 147 483 648...+2 147 483 647)

With the cursor keys the desired display is selected and confirmed with <ENTER>. With the <ESC> key the choice can be aborted. With the "OK" key the new setting is accepted and it is returned to the menu "test".

In the menu item **Test->"PS reset"** the PS can be reset. The PS has to be reset if:

- An error was detected during the program's run
- A new PS program is to be started. After the PS is reset the program is loaded and is then ready for execution in the PS.

There are two reset functions which are treated different in the PS:

- PS reset
- STL start.

With the function "**PS reset**" the PS starts to process the program just as it does after a 'restart' (power on)

- Reset inputs, outputs, flags, counters and timers,
- Load PS program,
- Start processing OB1 (or. OB22, if it exists), execute instruction 1.

After call of the function "**STL start**" :

- Reset inputs, outputs, flags, counters and timers,
- Start processing OB1 (or. OB21, if it exists), execute instruction 1.

With the menu item **Test->"jump"** any block of the project can be displayed in the program window.

With choice of the block by means of the cursor keys and confirmation with <ENTER> the number of the line which is to be displayed in the program window has to be input. This input is also confirmed by <ENTER>. In the program window the block is then displayed in the input line.

If the selected block or the program line doesn't exist an error message is displayed.

After processing one of the instructions 'run', 'cycle', 'instruction' or 'step' the STL instruction to be processed next is displayed in the program window.

4.4 Main menu item "Startup"

In the menu item "**Startup**" the inputs and outputs of the PS can be influenced without a PS program. Thus it is possible to test the I/O wiring in a system.

With call of this menu item the PS is stopped and all outputs are switched off. When leaving the menu item the user program (if it exists) has to be restarted.

The menu item "startup" contains two functions:

- Inputs and
- Outputs

After call of the menu item **Startup->"inputs"** a startup window is opened. All inputs of the PS are displayed in it. The following functions are provided to test the PS inputs:

- F2: Read
- F3: Cyclic
- F4: Type
- F5: Format

With the function key **<F2>** startup->inputs ->**"Read"** the inputs are read and their status is displayed in the startup window.

With the function key **<F3>** startup->inputs ->**"Cyclic"** the inputs are read cyclically (about all 0.5 s) and their status is displayed in the startup window. This functions can be left with **<ESC>**.

With the function key **<F4>** startup->inputs->**"Type"** the display type of the values in the startup window is changed. By pressing **<F4>** the next data type is presented:

- BYTE: Display as 8 bit values (0...255 bzw. -128...+127),
- WORD: Display as 16 bit values (0...65 535 bzw.-32 768...+32 767),
- LONG: Display as 32 bit values (0...4 294 967 295 bzw.
-2 147 483 648...+2 147 483 647).

With the function key **<F5>** startup->inputs ->**"Format"** the format of the numerical values can be selected in the startup window:

- Hex : Values displayed in hexadecimal form,
- Dec : Values displayed in decimal form as a positive number,
- \pm dec : Values displayed in decimal form with signs,
- Binary: Values displayed in binary form.

After call of the menu item **"Startup->outputs"** a startup window is opened. All outputs of the PS are displayed in it. The following functions are provided to test the PS outputs:

- F1: Input
- F2: Set
- F3: Reset
- F4: Type
- F5: Format

With the function key <F1> startup->outputs ->„**Input**“ the internal process image of the PS outputs can be changed without influencing the hardware outputs. "1" means PS output is set, "0" means PS output is reset in the process image „outputs“.

With the function key <F2> startup->outputs ->“**Set**“ the actual process image of the outputs is transferred directly to the hardware outputs (AZ-EAx).

Attention: With assignment of status „1“ the output is set and can activate an uncontrolled function in the connected system. In order to avoid damage the user has to take all necessary security precautions!

With the function key <F3> startup ->outputs ->“**Reset**“ the PS outputs are again set to "0". The current process image of the outputs is maintained.

With the function key <F4> startup ->outputs ->“**Type**“ the display type of the values in the startup window is changed. By pressing <F4> the next data type is presented:

- BYTE: Display as 8 bit values (0...255 bzw. -128...+127),
- WORD: Display as 16 bit values (0...65 535 bzw.-32 768...+32 767),
- LONG: Display as 32 bit values (0...4 294 967 295 bzw.-2 147 483 648...+2 147 483 647).

With the function key <F5> startup->outputs ->“**Format**“ the format of the numerical values in the startup window can be selected.

- Hex : Values displayed in hexadecimal form
- Dez : Values displayed in decimal form as a positive number
- +/-dez : Values displayed in decimal form with signs
- Binary: Values displayed in binary form

4.5 Main menu item "options"

In the menu item "**Options**" global parameters can be modified by APROS. The new settings are remanently stored in a file. The parameters are divided into three areas:

- Colours,
- Printer,
- General.

Menu item **Options->“colours”**:

The user can set up all colours of menus, windows and footers. The settings for display background and main menu only become effective after a repeated call of APROS.

Menu item **"Options->printer"**:

The user can preset the number of lines per page, characters per line as well as the distance from the left margin. In addition it can be selected by means of the <arrow upward> key or <arrow downward> key whether a title page is to be printed or not (title page not yet implemented). Furthermore the output aid (e.g...PRN , A:, C:\PRINT, ..) can be defined.

Menu item "Options->general":**Time-Out-Time:**

The user can select by means of the <arrow upward>- or <arrow downward> keys whether the warning-tone is switched on or off. Furthermore the Time-Out-Time (Tout) for the S-bus-communication has to be set up between the programming equipment (PC/APROS) and the PS.

- Tout > 1: Online; Time = Tout * 1/20 sec.
- Tout = 1: Offline; no time out

Language:

As far as APROS is concerned the languages „German“ or „English“ can be selected.

Menu item "Options->editor":

The user can input the following settings for editor parameters:

- Tabulator length,
- Insert YES/NO,
- Pay attention to big/small letters.

4.6 Help

The menu item "help" is divided into the following subitems:

- Help index,
- Overview of keys,
- S-Bus,
- Info.

In the menu item "**Help index**" the key word index of the on-line help can be called. By means of the cursor keys a key word is selected and the related help text is displayed by pressing the <ENTER> key.

In the menu item "**Overview of keys**" the functions of the <ESC>, <ENTER>, <TAB>, <SHIFT>-<TAB> and arrow keys <- , -> are described.

In the menu item "**S-Bus**" the functions of the serial connection to the PS can be tested.

In the menu item "**Info**" the APROS version number is displayed.

5 On-line help

The on-line help (short „help“) is called with the function key <F10>. The help window is opened and the help text for the current menu item is displayed. The marked words are key words. By positioning the cursor on a key word and by pressing <ENTER> a new text explaining this key word is displayed.

By once again pressing the function key <F10> the key word index is displayed in the help window. A new key word can be selected with the cursor key. With <ENTER> the explanation for the key word is displayed in the help window.

6 Printout representation

Only simple ASCII characters are used for the printout of the STL blocks. This ensures a printout on any printer.

The printout of a block (menu item "programming-> print") is structured as follows:

On page 1 the block head is printed before the STL text:

APROS Version x.xx	AMK, Kirchheim/Teck
Name of the project: <project name>	
Title of the project: <project title>	
Notes: <project note>	
Block name: <block name> <block type> <block index>	
Last change: <date of last change>	

<Line number> STL text

Every further page starts with the following header:

Block name: <block name> <Date of last change>

Each page ends with the following footer:

AMKASYN PS program current date> Page:<No>

7 Annex

7.1 Editor description

Control commands	
Key	Meaning
→	Cursor one position to the right
←	Cursor one position to the left
↑	Cursor one position up
↓	Cursor one position down
←	Cursor to next left TAB stop
→	Cursor to next right TAB stop
Page ↑	Cursor one page up
Page ↓	Cursor one page down
Pos1	Cursor to first character in the line != ' '
End	Cursor to last character in the line
CTRL →	Cursor to next word
CTRL ←	Cursor to previous word
CTRL Pos1	Cursor to first line in the file
CTRL End	Cursor to last line in the file
F2	Cursor to next error
SHIFT F2	Cursor to previous error

Note: The cursor control commands have the same function in the mode 'block on', but in this mode the skipped text is marked.

Input commands	
Key	Meaning
Each key	E: Character is inserted into the text
0...254	Ü: Character at the cursor position is overwritten with the new character (exception: Backspace, Return, Tabulator)
↵	E: Insert line feed Cursor to first position next line

Delete commands	
Key	Meaning
DEL	The character at the cursor position / the marked block is deleted
BS	Ü: The character at the cursor position is overwritten with ' ' E: The character at the cursor position is deleted
ALT Y	The cursor line is deleted and copied into the intermediate memory

Copy commands	
Key	Meaning
F3	Block marking on/off
F4	Block on: Marked block is copied into the intermediate memory
	Block out Current line is copied into the intermediate memory
SHIFT F4	Marked block is copied into the intermediate memory and deleted in the editor text
F5	The intermediate memory's content is inserted at the cursor position into the editor text
SHIFT F5	A function block is inserted from the function block library.

Control commands	
Key	Meaning
F1	Save editor text
SHIFT F1/ ESC	Leave editor
SHIFT F3	Search/Substitute: Pay attention to capital/small letters
INS	Change insert mode/Overwrite mode

Meaning of the characters in the footer	
Abbr.	Meaning
Block on	Marking on
Gk	Capital/small letters not ignored during search/substitute
GK	Capital/small letters ignored during search/substitute
G	File has been changed
Ü	Overwrite mode on
E	Insert mode on

7.2 Compiler Description

Structure of lines in the STL program

Each instruction line, even pure comment lines must necessarily be concluded with "CR","LF" (0x0DH,0x0AH). The maximum length of a line is **128** characters. The program line consists of a maximum of 6 fields which are:

Field ZN	Line number	Max. three characters, isn't examined and tested by the compiler. This field can be dropped.
Field MA	Label	Max. 6 characters long, used as a jump label. No character which is declared as a separator is permitted. This field can be dropped.
Field OT	Operation-part	Must be available. Consists of one to four characters (only capital letters are permitted).
Field KN	Identifier	Consists of one or two characters (only capital letters); must be available if required by operation part (see also description of instructions).
Field PA	Parameter field	It can consist of one or two parameters (depending on instruction). The used characters have to correspond to the instruction. This is examined during the compilation.
Field KM	Comment	Can be used for explanation of the instruction line. All ASCII-characters are permitted. The content is not evaluated.

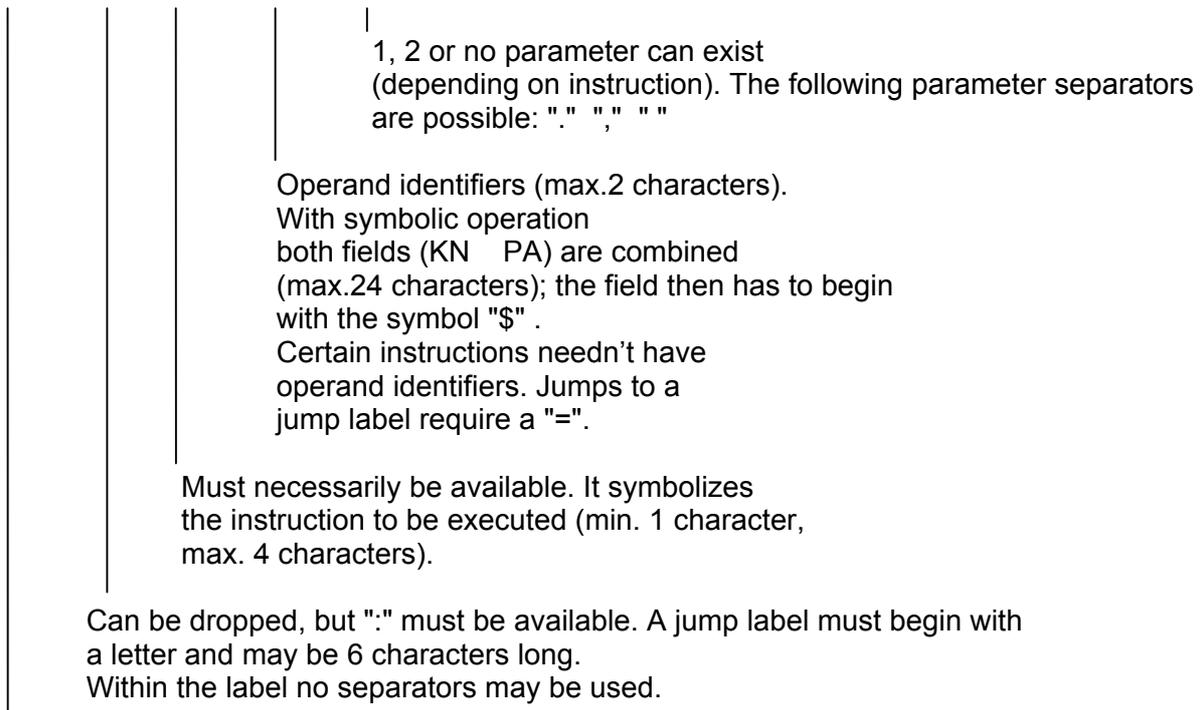
At least one „space line“ (0x20H) or one "TAB" (0x09H) is to be set as a separator (in the following description marked with the symbol " ").

None of the above separators except within the comment field may be used within the single fields. The comment field has to begin with a semicolon (;' 0x3BH).

If symbolic addresses are used this is to be marked by preceding the special character "\$" before the symbolic identifier.

This results in the following standard line structure:

ZN □ MA:OT □KN □PA ; KM



The line number runs from 1 - 255. It is either generated by the compiler or taken over (if it exists). No test is made for ascending numerical order.

Example:

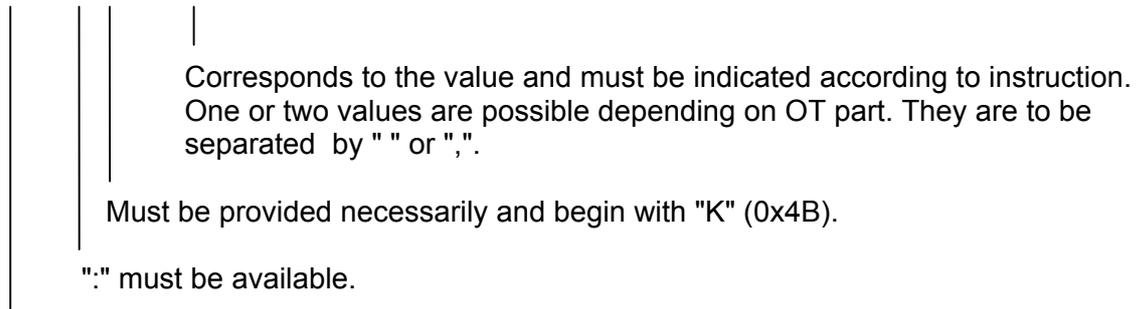
```
001 anf:U E 1.0;
002 :UN E 10.7
:= A 0.0
004 :SPB = anf
:BE
```

Line structure in the STL program for data blocks

The structure of an instruction line for creation of a data block differs from the previous structure. This structure has only 4 fields (ZN, MA, OT, PA).

This results in the following standard instruction line:

ZN □ : OT □ PA



Same statement as for the STL line.

Example:

```
001 :KD 23456789
:KB 123
:KC aA
002 :KY 12,45
```

Line structure of assignment list:

The line structure of this file must have the following form:

Field SY	Symbol name	The name can have max. 23 characters. (It must be taken into consideration that the same character string is used in the STL program and in the file.)
Field KN	Identifier	Same condition as for STL.
Field PA	Parameter	Same condition as for STL, with restriction that only 16 characters are permitted.
Field KM	Comment	Begins with a ";" and can fill the line up to a length of 128 characters.

Each line must be concluded with "CR", "LF" otherwise an error message is displayed.

The line structure is as follows:

□ SY□KN PA ; KM

Extended forms of the line structure:

In order to be able to create and compile "formatted" programs any number of separators (space lines or tabulators) are permitted between the single fields (restriction: max. 128 characters per line). Within the single fields no separators may be used (exception: see below).

Example:

```
001      anf:U      E      1.0
002      :UN      E      10.7
          :=      A      0.0
004      :SPB =    anf
          :BE
```

This display is compiled without errors.

The following style, however, is not compiled:

```
004      : S P B = anf
```

Hereby the maximum length of the OT field is exceeded. The compiler issues the message:

- Instruction not found: S P B

An **exception** to the rule that no separators are to be used represents the following instruction:

```
:L   KM   0000 0000 0000 0000
```

For clarity reasons separators are permitted in the **field PA**. It must be taken into consideration that not more than 24 characters (with separators) are provided for the PA field.

Overview of the permitted character set per instruction:

The table provides a summarized overview of the permitted character set in the parameter field.

Instruction	permitted characters in the PA field
U E	all digits and "."
U A	(0x30 - 0x39; 0x2E)
U M	"
UN E	all digits and "."
UN A	(0x30 - 0x39; 0x2E)
UN M	"
O E	all digits and "."
O A	(0x30 - 0x39; 0x2E)
O M	"
ON E	all digits and "."
ON A	(0x30 - 0x39; 0x2E)
ON M	"
S E	all digits and "."
S A	(0x30 - 0x39; 0x2E)
S M	"
R E	all digits and "."
R A	(0x30 - 0x39; 0x2E)
R M	"
= E	all digits and "."
= A	(0x30 - 0x39; 0x2E)
= M	"
U T	all digits
U Z	"
UN T	all digits
UN Z	"
O T	all digits
O Z	"
ON T	all digits
ON Z	"
SI T	all digits
SV T	"
SE T	"
SS T	"
SA T	"
R T	"
ZV Z	all digits
ZR Z	"
S Z	"
R Z	"
L EB	all digits
L AB	"
L MB	"

Instruction	permitted characters in the PA field
L EW	"
L AW	"
L MW	"
L ED	all digits
L AD	"
L MD	"
L DL	all digits
L DR	"
L DW	"
L DD	"
L PY	all digits
L PW	"
L PD	"
T EB	all digits
T AB	"
T MB	"
T EW	"
T AW	"
T MW	"
T ED	"
T AD	"
T MD	"
T DL	all digits
T DR	"
T DW	"
T DD	"
T PB	all digits
T PW	"
T PD	"
L T	all digits
LC T	"
L Z	all digits
LC Z	"
SPA OB	all digits
SPA PB	"
SPA FB	"
SPB OB	"
SPB PB	"
SPB FB	"
A DB	all digits
E DB	"
SLD	all digits
SRD	"
SPA =	all digits and letters, if labels are used
SPB =	"
SPZ =	"
SPN =	"
SPP =	"
SPM =	"

Instruction		permitted characters in the PA field
SPO =		"
This instruction-group permits different characters for each instruction.		
L	KB	only digits
L	KC	all capital and small letters, digits as well as those characters that are in the range 0x20H - 0x7EH and 0xB0H - 0xE4H It should be taken into consideration that for the second character 0x20 and 0x3B are not permitted.
L	KF	all digits, as well as "+" and "-" (0x30 - 0x39; 0x2B ; 0x2D)
L	KH	the capital letters A - F and the small letters a - f. (0x41 -0x46 ; 0x61 - 0x66)
L	KM	the digits 0 and. 1, as well as the space line (0x30; 0x31; 0x20)
L	KY	all digits, as well as ","(0x30 - 0x39; 0x2C)
L	KT	all digits
L	KZ	all digits
L	KD	all digits, as well as "+" and "-"(0x30 - 0x39; 0x2B ; 0x2D)

The following instructions have no entry in the PA field:

```

O          U(          O(          )
+FD        -FD        *FD        /FD
I=FD      ><FD        >FD        >=FD    <FD    <=FD
BE         BEA        BEB
NOP0      NOP1
STP       STS
***
KED       KZD
WDL       WZD
UD        OD         XOD
    
```

8 Impressum

Title AMKASYN Programming Software APROS

Objective Software Description for AMKASYN Programming Software APROS

Part-Number 25797

History

Date
1995/05

Copyright

© AMK GmbH & Co. KG

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose without the express written permission of AMK GmbH + Co. KG. Violations are subject to legal action. All rights in case of patent filings or user-sample registrations are reserved.

Disclaimer

We reserve the right to change the contents of the documentation and the availability of products at any time without prior notice.

Service

Tel.: +49(0)7021 / 5005-191, Fax –193

Business Hours:

Mo-Fr 7.30 - 16.30, On weekends and holidays calls are forwarded to an emergency response number by the automated answering system.

To assure a fast and accurate response to solve customer problems we ask for your cooperation in providing us with the following information:

- Nameplate data
- Software version
- System configuration and application
- Description of problem and presumed cause of failure
- Diagnostic message (error code)

Publisher

AMK Arnold Müller Antriebs- und Steuerungstechnik GmbH & Co. KG
Gaußstraße 37 – 39, 73230 Kirchheim/Teck

Tel.: 07021/5005-0, Fax: 07021/5005-176

E-Mail: info@amk-antriebe.de

For further information www.amk-antriebe.de