

1 Basic Dialog

The basic dialog incorporates the “Framework” program for working with STRUC G.

The **task** of the basic dialog is primarily to create, select, process and document projects, master programs, macros and function packets.

The user files and libraries, required by the STRUC editors, are saved in a data bin.

1.1 Start and End of STRUC G

STRUC G is started by a double click of the STRUC icon using the left-hand mouse key.

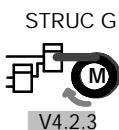
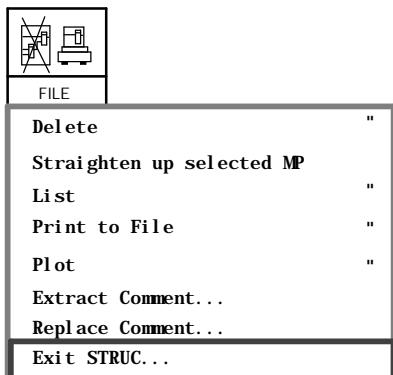


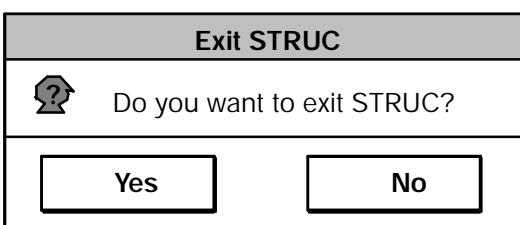
Diagram 1/1: STRUC Icon

After invoking STRUC, the basic dialog screen window displays the same state as it had when the previous STRUC session was terminated: the previously processed project is in the project line, the previously processed master program or the macro directory is in the master program line and the previously processed function packet or macro is selected in the function packet area.

The closing of the basic dialog is started by invoking ‘FILE / Close STRUC’.



Subsequently a question window is popped onto screen:



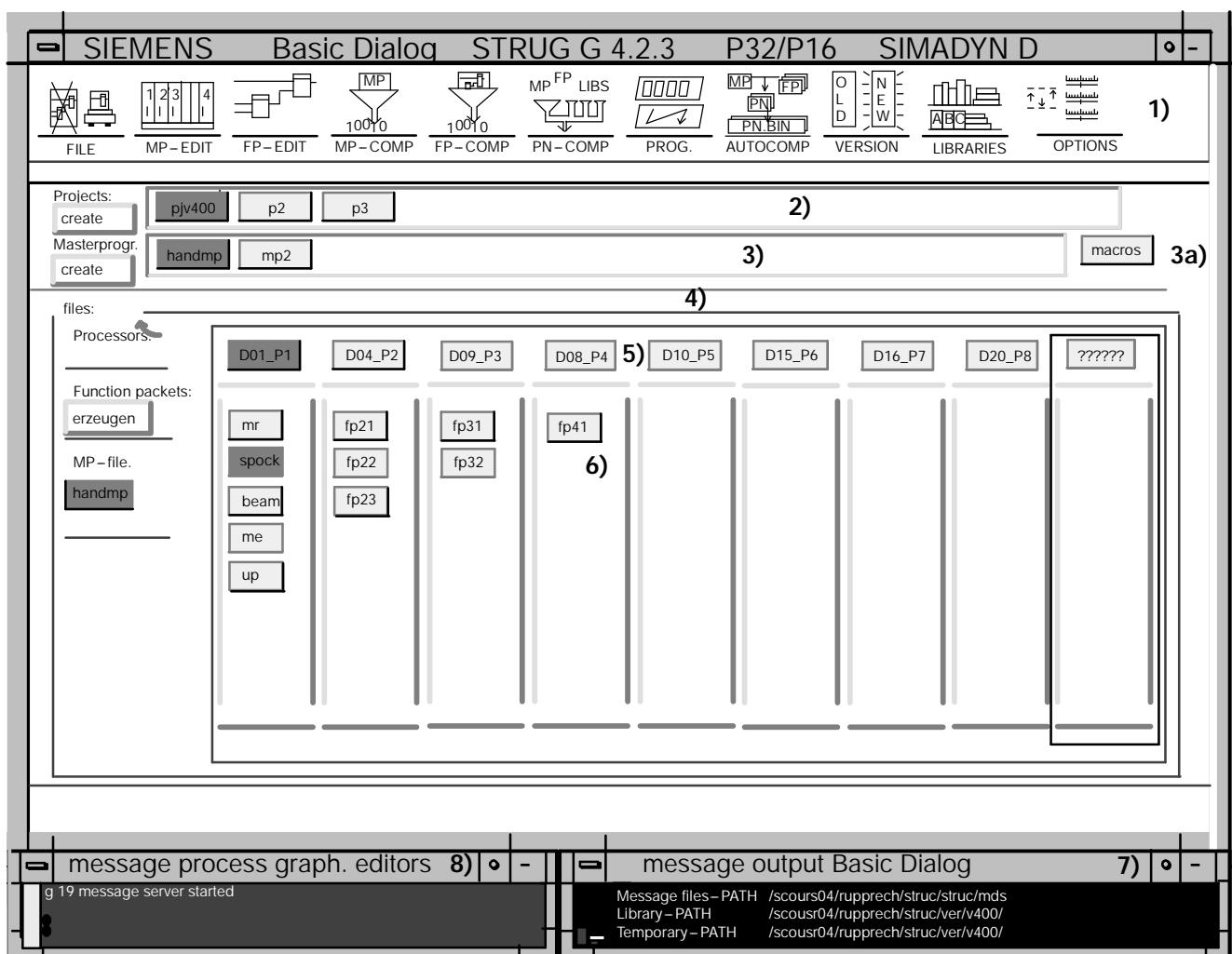
If you receipt with 'Yes', then the basic dialog is quitted.

If you receipt with 'No', then the basic dialog is activ further on.

1.2 The Basic Dialog Screen Window

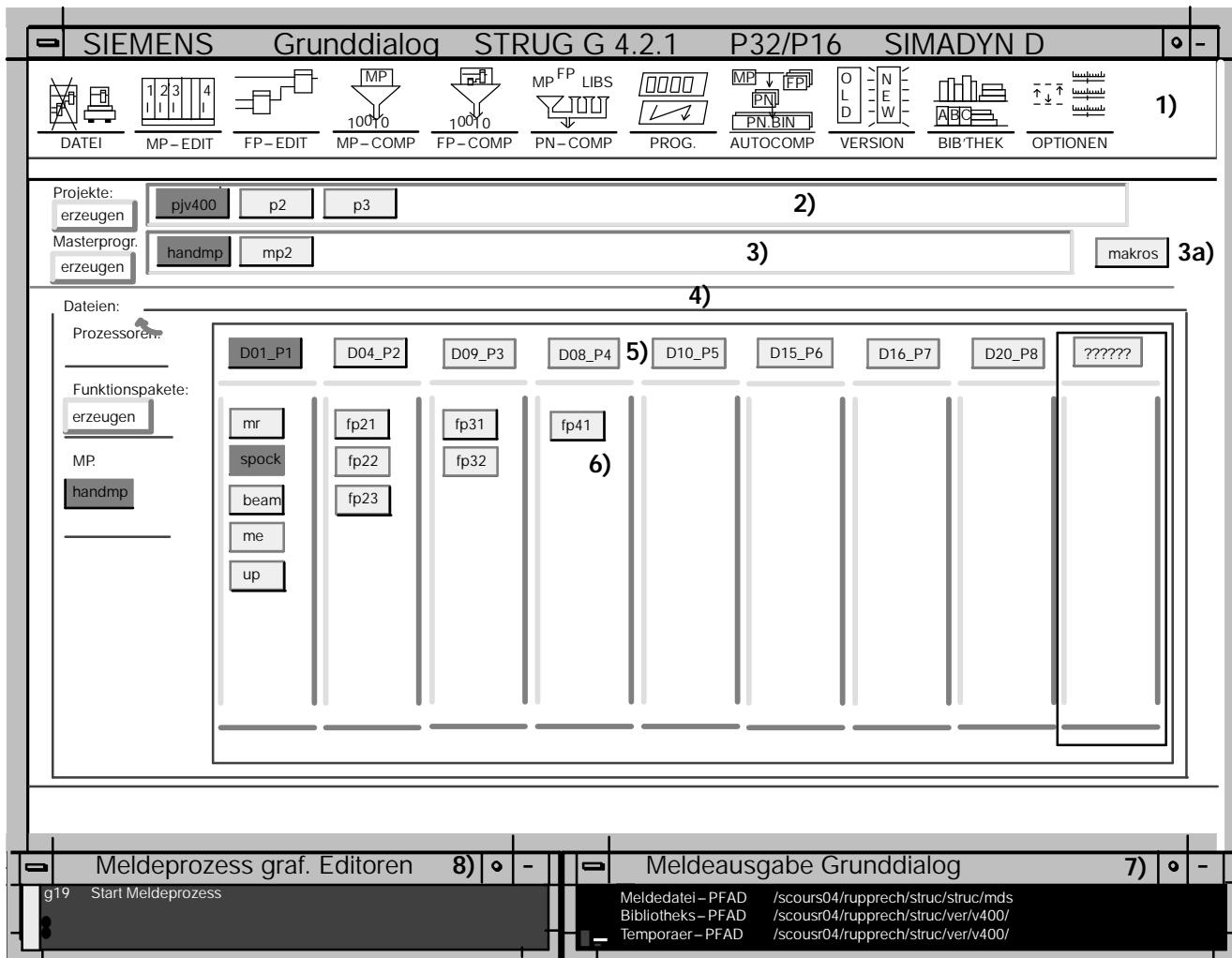
Screen Window Showing the Function Packets

2/2 indicates the basic dialog screen window that is popped onto screen, after invoking STRUC G, when a **master program** is selected.

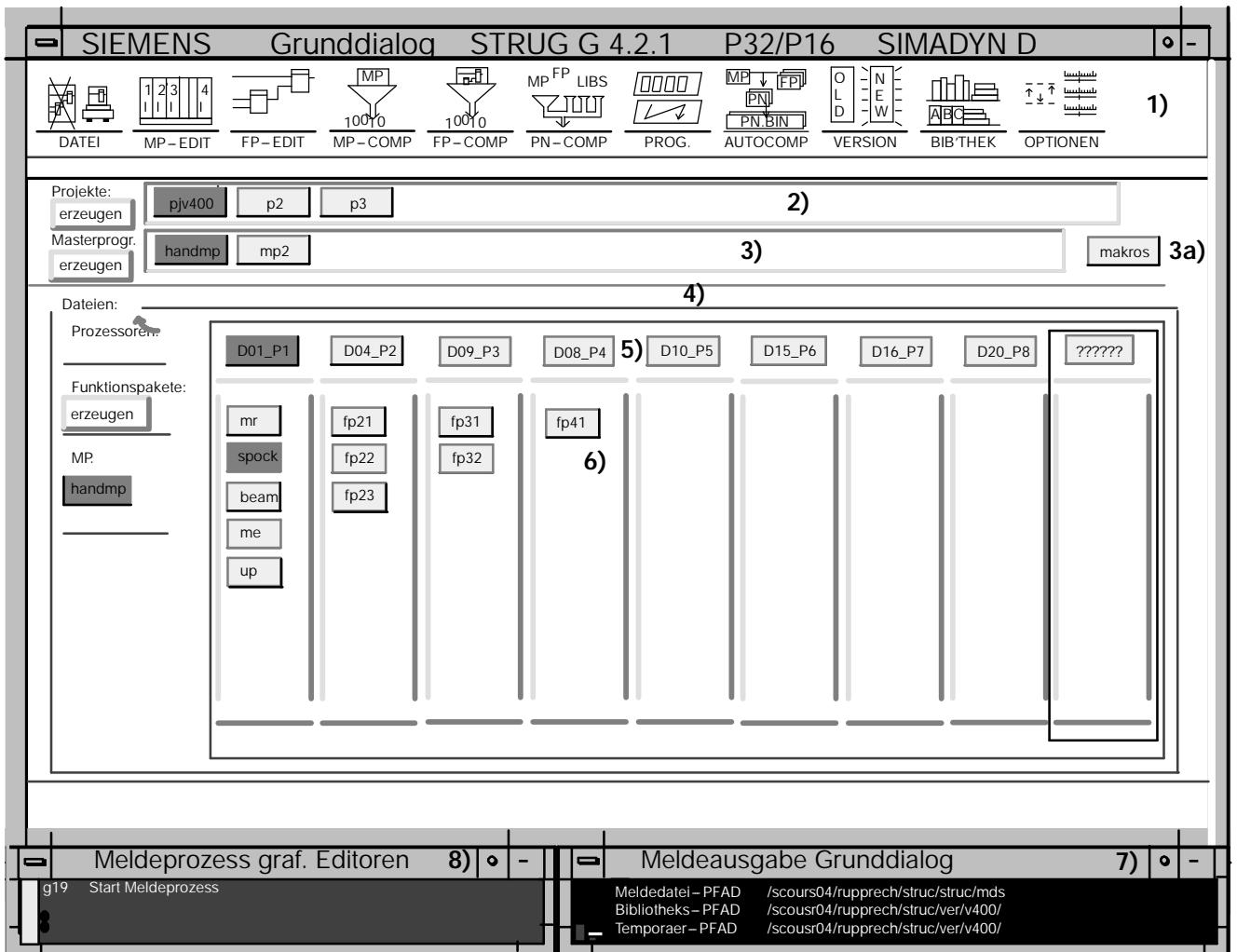


- 1) Static Menu
- 2) Project Line
- 3) Master Program Line
- 3a) Button for Macros
- 4) Files Line
- 5) Processor Line
- 6) Functions Packet Line
- 7) Basic Dialog and STRUC – Compiler Message Process
- 8) MP/FP – Editors Message Process

Diagram 1/2: STRUC Basic Dialog Window – Funktions Packet Overview



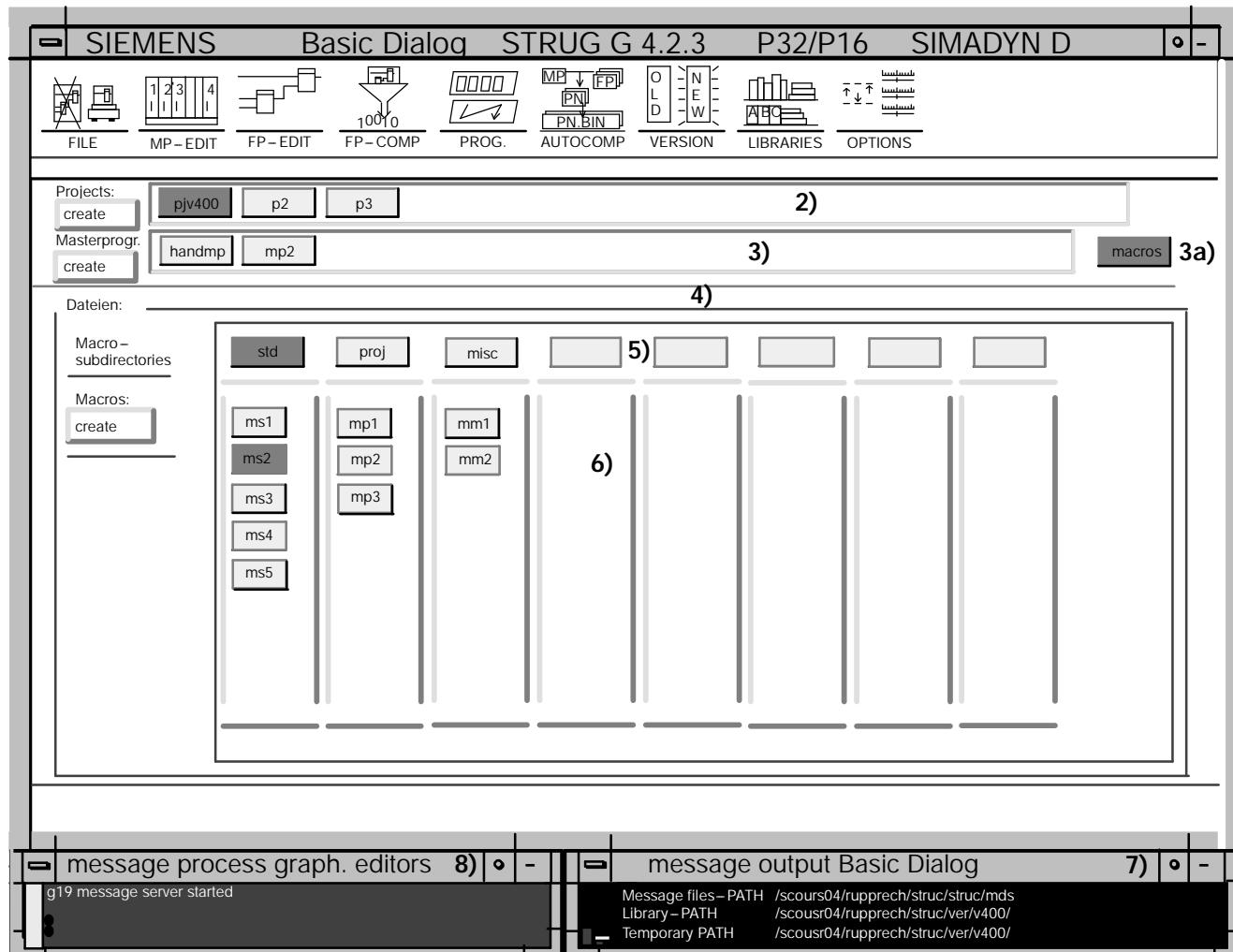
- 1) Statisches Menü
- 2) Projektzeile
- 3) Masterprogrammzeile
- 3a) Auswahlbutton für Makros
- 4) Dateienzeile
- 5) Prozessorenzeile
- 6) Funktionspaketbereich
- 7) Meldeprozeß des Grunddialogs und der STRUC - Compiler
- 8) Meldeprozeß des MP/FP - Editors



- 1) Statisches Menü
- 2) Projektzeile
- 3) Masterprogrammzeile
- 3a) Auswahlbutton für Makros
- 4) Dateienzeile
- 5) Prozessorenzeile
- 6) Funktionspaketbereich
- 7) Meldeprozeß des Grunddialogs und der STRUC – Compiler
- 8) Meldeprozeß des MP/FP – Editors

Screen Window with Macro Overview

2/3 shows the basic dialog screen window, when the **macro directory** of a project is selected.



- 1) Static Menu
- 2) Project Line
- 3) Master Program Line
- 3a) Button for Macros
- 4) Files Line
- 5) Macro Sub–directory Line
- 6) Macro Area
- 7) Basic Dialog and der STRUC – Compiler Message Process
- 8) MP/FP – Editor Message Process

Diagram 1/3: *STRUC – Basic Dialog – Window – Macro Overview*

General Basic Dialog Operations

Each basic dialog function must be selected via the cursor.

Selecting with the cursor

Selection is implemented by placing the cursor on top of the desired function button using the mouse.

A sub-menu is popped onto screen when the **left-hand mouse key is pressed and held**.

The desired sub-menu function is selected with pressed and held left-hand mouse key. This function is selected when the mouse key is released.

Remove the cursor pointer from the sub-menu and then release the mouse key if no function in the sub-menu is to be executed.

This type of function selection is valid for all functions in the basic dialog.

Certain processing phases exist in which not all functions of a sub-menu can be executed, since they are either not possible, not practical or not permitted. These functions are then disabled by STRUC. This is indicated to the user by ghosting the text of this menu line on the screen.

1.3 Basic Dialog Screen Layout and File Functions

The individual sub-sections of the screen window (2/2 or 2/3) show an overview of the configuring data. The displayed objects simultaneously represent **function buttons** that are layed over further sub-menus containing the functions to process the object.

1.3.1 Project Line

All user projects are indicated here. The currently selected project is color emphasized.

All project names simultaneously represent **function buttons**, overlaying sub-menus which contain functions necessary for processing the projects.

Project -- Selection

The selected project name is color emphasized and the corresponding master program is inserted into the master program line. No master program has yet been selected and therefore the processor lines and function packet area are empty.

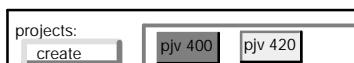


Diagram 1/4: Select Project

Project -- Create

A form is popped onto screen, after selecting this function, in which the name of the new project must be inserted (2/5). The **project name** consists of one to six characters (lower case letters, numbers and underscores), whereby the first character must be a letter.

Clicking the 'OK' button with the left-hand mouse key creates the project; the function is aborted with the 'Cancel' button, i.e. no project is created.'

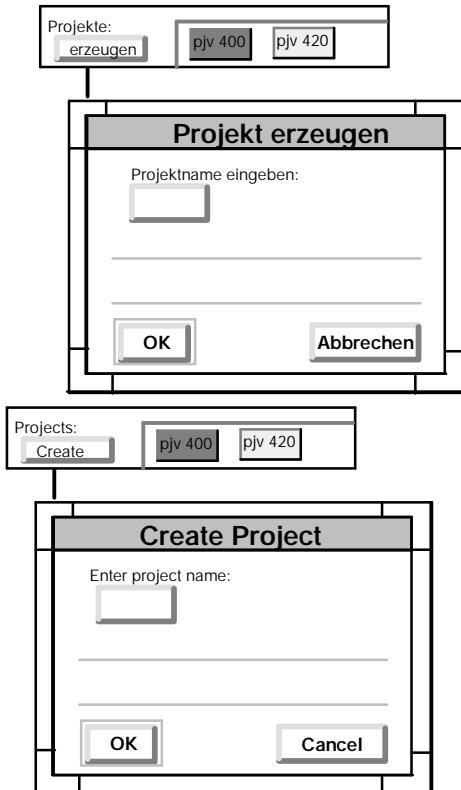


Diagram 1/5: Sub-Menu for "Project Create"

1.3.2 Master Program Line

All master programs as well as the macro directory "Macros" of the selected project are displayed here. The currently selected master program is color emphasized.

All master program names and the macro directory are simultaneously function buttons which overlay sub-menus containing functions required for their processing.

Masterprogram -- Selection

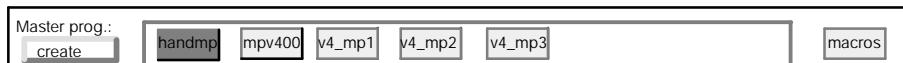


Diagram 1/6: Select Master Program

Clicking the button for the master program names with the left – hand mouse key (2/6) selects the master program. The selected master programm is color emphasized. The processors, projected in the master program, are entered in the processor line and the function packets are entered in the function packet column of those processors, where they are projected.

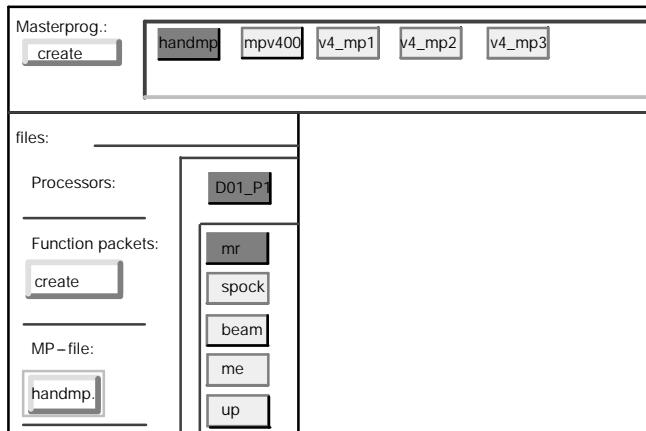


Diagram 1/7: Selected Master Program File

If a masterprogram is selected, then the functions for the master program compiler in the Static menu can be selected.

Master Program -- Create

A form is popped onto screen when this function is selected, in which the name of the new master program must be inserted (2/8).

The syntax of the **MP name** is an analogous to that of the project name, i.e. one to six characters (lower case letters, numbers and underscore), whereby the first character must be a letter.

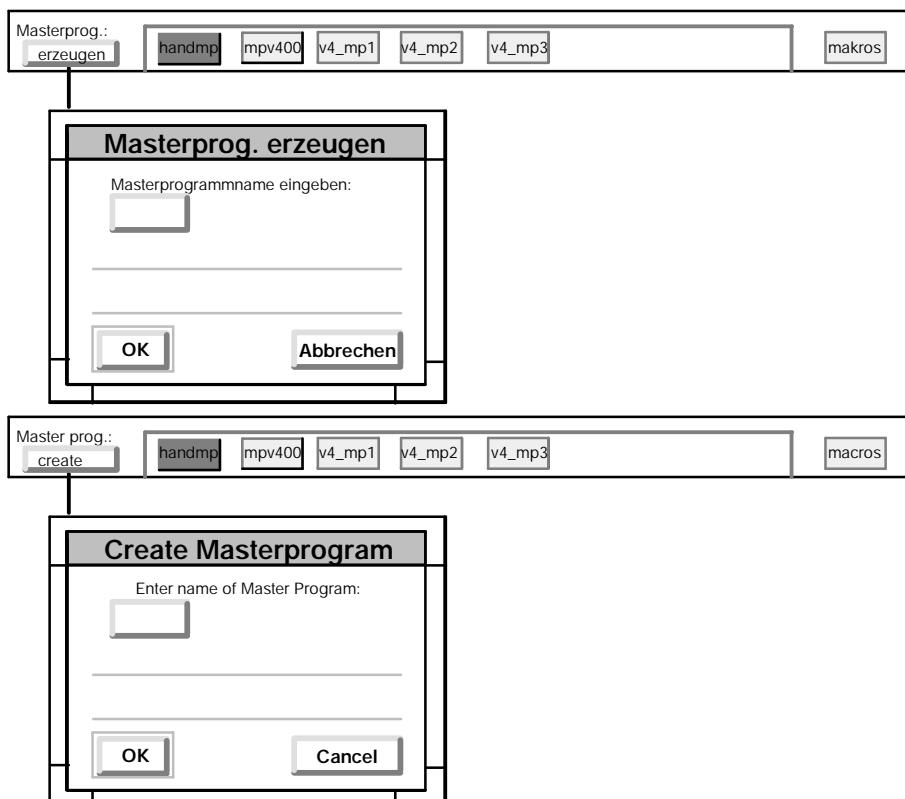


Diagram 1/8: Sub-Menu for 'Create Master Program'

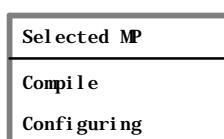
Clicking the 'OK' button with the left-hand mouse key creates the master program; the function is aborted with the 'Cancel' button, i.e. no master program is created.

First the function 'Create Master Program' creates the master program directory and an "empty" master program file.

The names of the master program file and its directory are identical.

Direct MP--Editor respectively MP--Compiler -- Invoke

The pop-menu is popped onto screen by clicking and keeping the right hand mouse key pushed in the master program:



The MP-COMP function is invoked for the current selected master program by positioning the cursor at projecting and letting the right-hand mouse key loose.

The MP-EDIT function is invoked for the current selected master program by positioning the cursor at projecting and letting the right-hand mouse key loose.

Macro Directory -- Select

Clicking the makro directory name 'Macros' with the left-hand mouse key selects the makro directory.

The macro directory is color emphasized. The basic dialog image changes to the makro view (2/3). The macros of the selected projects are displayed in individual sub-directories columns.

If the macro directory is selected, then the functions for macro compiler in the Static menu can be selected.

Macro Sub--Directory and Macro -- Select

Clicking the macro sub--menu name with the left-hand mouse key displays the currently selected macro sub--directory color emphasized (2/9).



Diagram 1/9: Select Macro Sub-Directories

1.3.3 Files Line

If a master program is selected in the master program line, then the master program file as well as the processors and function packets, configured in this master program, are displayed in the files line (2/10).

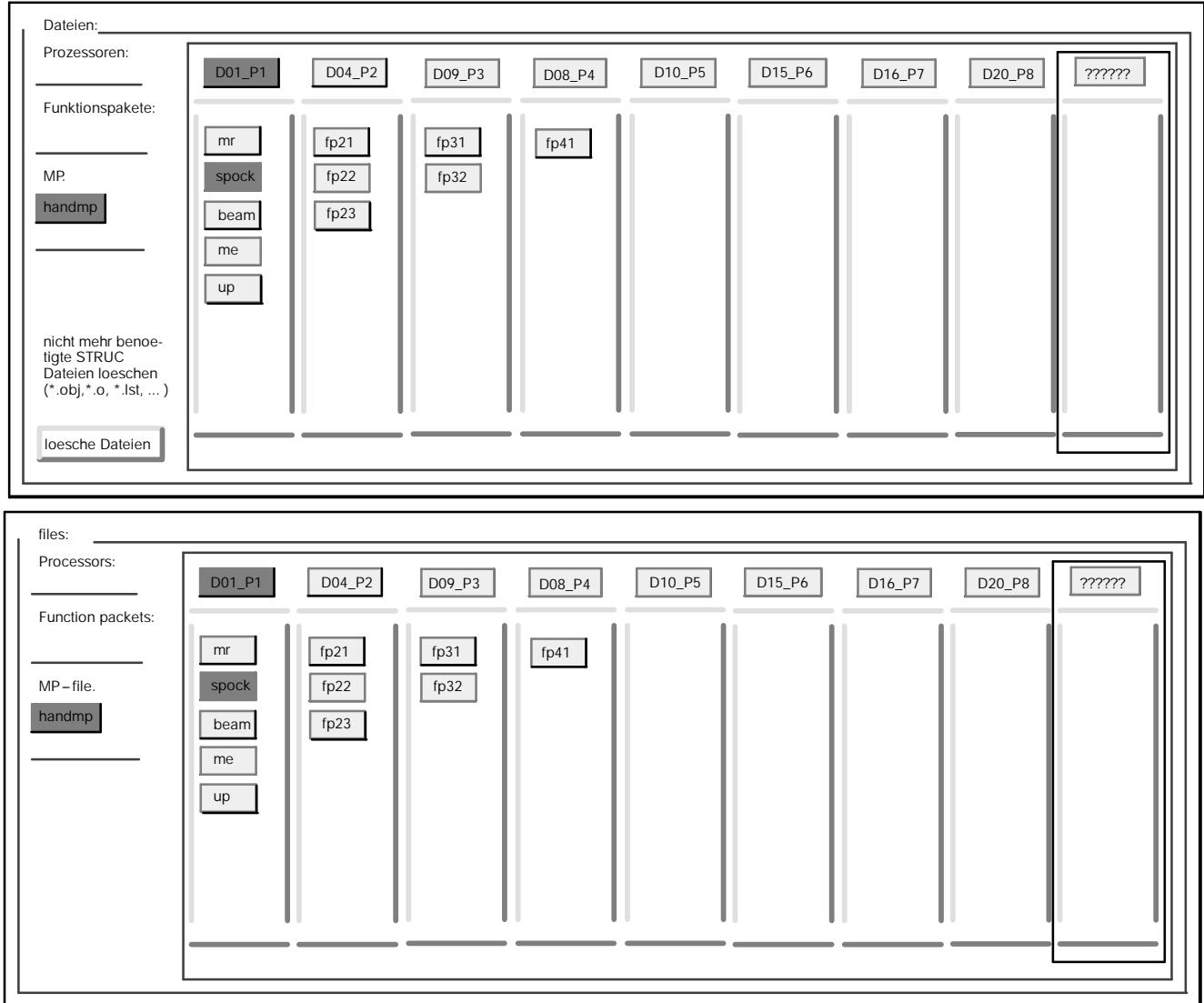


Diagram 1/10: Processing Files – Function Packets Overview

If the macro directory is selected in the master program line, then the macro sub-directories and the macros , contained in there, are displayed in the files line (2/11).

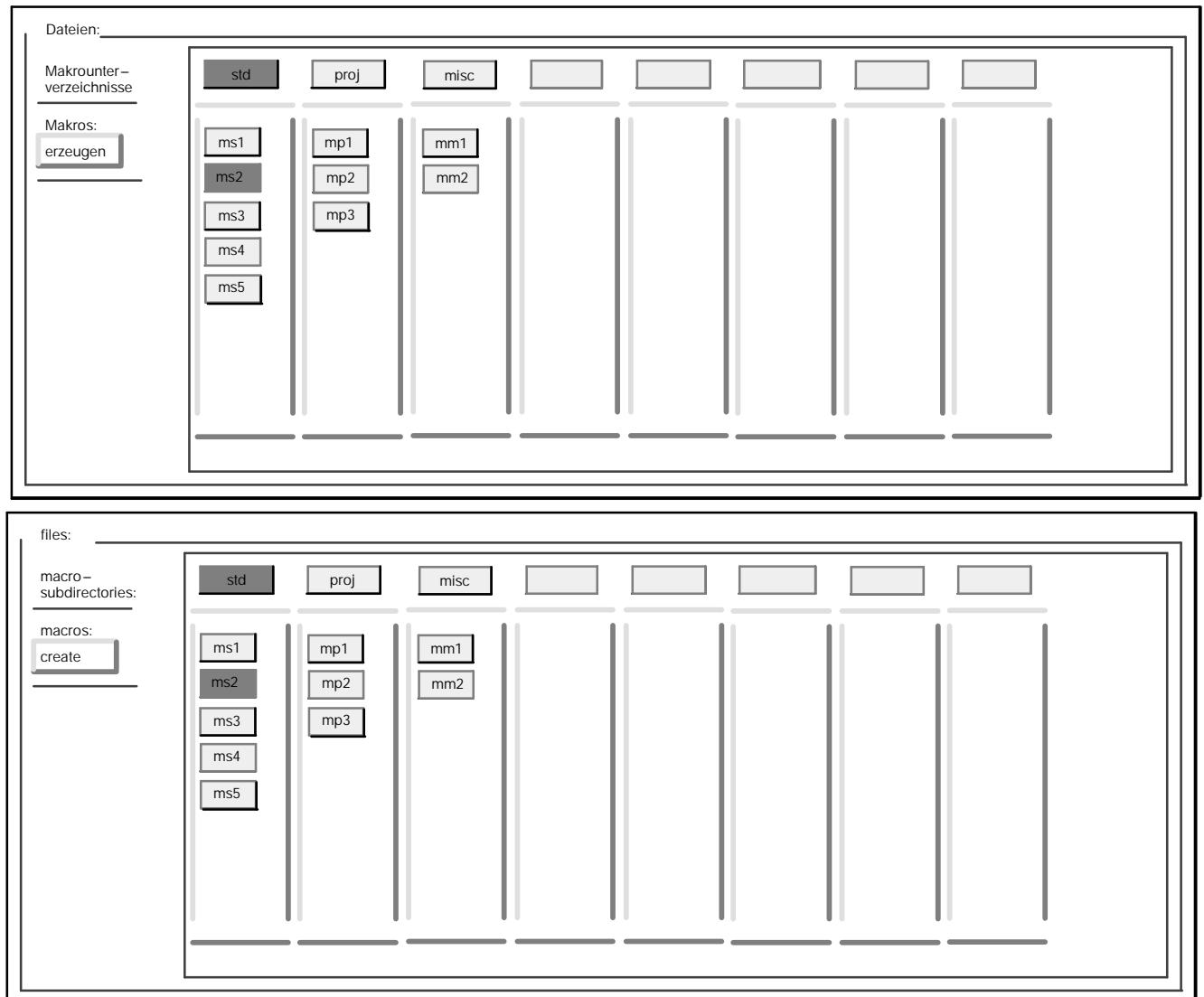


Diagram 1/11: Processing Files – Macros Overview

The names displayed in the project and master program line are path names according to the UNIX file structure.

The name of the selected master program is displayed in the basic dialog on two function buttons.

The master program name is located in the **master program line** (2/6) and represents the sum of all the STRUC files assigned to a rack, whereas this name in the **files line** (2/7) represents only the master program file.

1.3.4 Processors Line

All processors of the selected master programs are displayed here in the **function packet overview**. The currently selected processor is displayed color emphasized.

All processor names simultaneously represent function buttons, overlaying sub-menus containing the functions required for processing the processor programs.

The **macro overview** displays the macro sub-directories of the selected project in the processors line. The currently selected directory is displayed color emphasized.

Processor Program -- Select

Clicking the processor program name with the left-hand mouse key selects the processor program.

The selected processor program is color emphasized on screen.

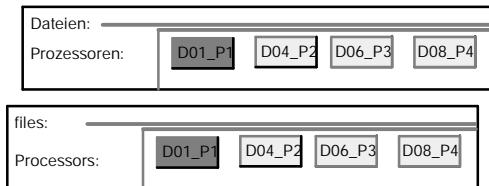
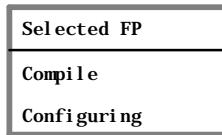


Diagram 1/12: Select Processor Program

Direct FP--Editor respectively FP--Compiler -- Invoke

The pop-menu is popped onto screen by clicking and keeping the right hand mouse key pushed in the master program:



The FP-COMP function is invoked for the currently selected function packet by positioning the cursor at projecting and letting the right-hand mouse key loose.

The FP-EDIT function is invoked for the currently selected function packet by positioning the cursor at projecting and letting the right-hand mouse key loose.

1.3.5 Function Packet Area

All function packets of a selected master program are displayed, sorted into columns according to processors, in the **function packet overview**. The selected function packet is displayed color emphasized.

All function packet names simultaneously represent function buttons, overlaying sub-menus containing the functions required for processing the function packet.

All macros of a selected project are displayed, sorted into columns according to sub-directories, in the **macro overview**. The selected macro is displayed color emphasized.

Function Packet -- Select

The selected macro is displayed color emphasized by clicking the function packet name with the left-hand mouse key.

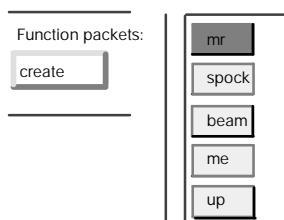


Diagram 1/13: Select Function Packet

The functions for the function packet processing can be selected in the Static menu and its sub-menus, as soon as the function packet is selected.

Function Packet -- Create

After selecting this function, the user must initially specify whether the function packet is to be created for a 16 bit or 32 bit processor. A form is subsequently popped onto screen in which the function packet name of the new function packet must be entered.

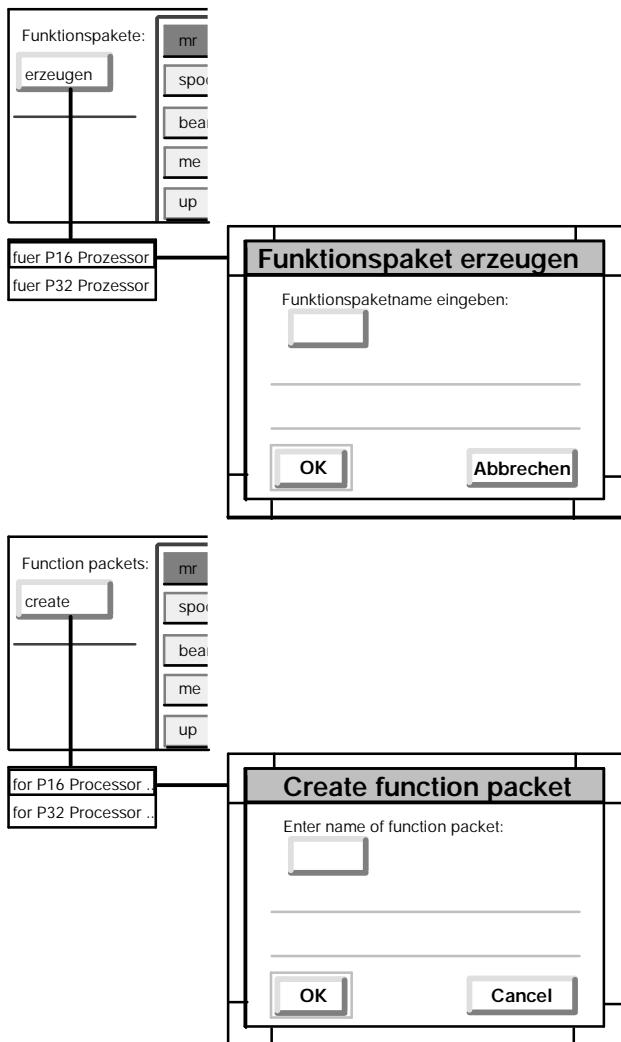


Diagram 1/14: Create Function Packet

The syntax of the **FP name** is analogous to that of the project name, i.e. one to six characters (lower case letters, numbers and underscore), whereby the first character must be a letter (or a '@' for special function packets).

Click on the 'OK' button with the left-hand mouse key to create the function packet. The 'Cancel' button aborts the function, i.e. no function packet is created.

The newly created function packet is entered into the '???????' column in the basic dialog, if it has not yet been allocated to a processor program.

The allocation of the function packets to processors is implemented in the module parameter drawing of the master program editor.

If function packets, without function packet files yet, are attached to a processor in the master program, then these function packets are color emphasized differently on screen. Additionally they must be installed with the function 'Create function packet'. If you enter the name of such a function packet as a name in 'Create function packet', then its color changes and it can now be compiled.

Macro -- Select

The selected macro is displayed color emphasized by clicking the macro name with the left-hand mouse key.

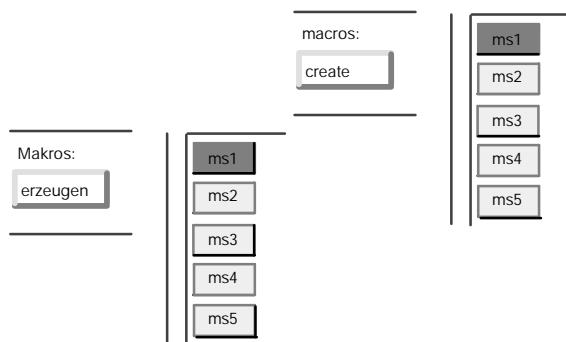


Diagram 1/15: Select Macro

Macro -- Create

Selecting this function causes the form shown in 2/16 to be popped onto screen. The name of the new macro is entered at this location. The macro sub-directory, in which the macro is to be created, must also be selected.

The syntax of the macro name is analogous to that of the project name, i.e. one to six characters (lower case letters, numbers and underscore), whereby the first character must be a letter.

Click on the 'OK' button with the left-hand mouse key to create the macro. The 'Cancel' button aborts the function, i.e. no macro is created.

If the basic dialog detects that a macro with the desired name already exists in one of the macro sub-directories, then an error window is popped onto screen; a new macro is then not created.

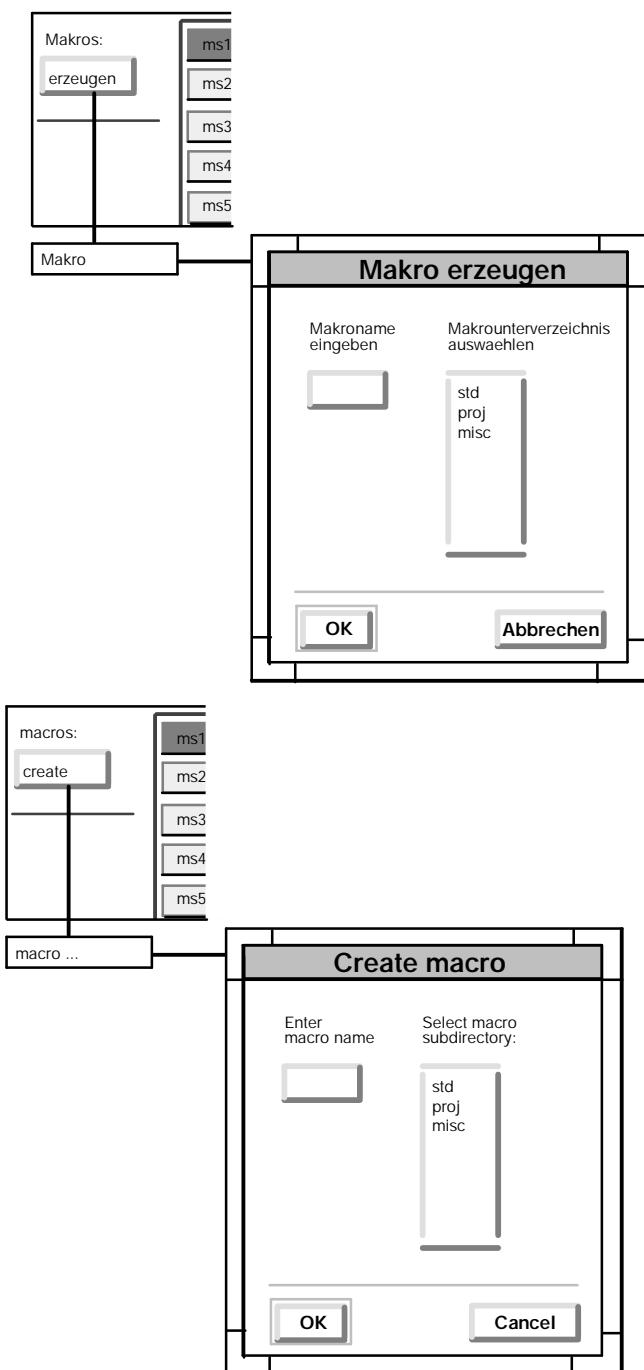


Diagram 1/16: Create Sub-menu

1.4 Configuring Functions in the Static Menu

The static menu differs only slightly between the **function packets overview** (2/2) and the **macros overview** (2/3). The following text describes only the function packets overview. Deviations in the macros overview are flagged with “: **Macros Overview**”.

FILE:	Compile Basic Dialog	
MP-EDIT:	Process Master Program	
FP-EDIT:	Process Functions Packet	
MP-COMP:	Compile Master Program	(not available in : macro overview)
FP-COMP:	Compile Function Packet	
PN-COMP:	Compile Processor Program	(not available in : macro overview)
PROG.:	Load Program Memory Sub-Module	
AUTOCOMP:	Compile Selected Master or Processor Program	
VERSION:	Update User Data Version	
LIBRARIES:	Select Libraries	
OPTIONS:	Address Book, Disable Retro-Documentation, Save Error Image, Internal Graphics Editor	

1.5 Configuring Data Structure

2/17 indicates the structure of the user data ‘lnk_awd’ as well as the various levels of the **projects PJ**, **master programs MP** and **function packets FP** (see also chapter “Overview” of this manual).

The temporary data bin ‘lnk_tmp’, that holds the function packet / master program files during the editor processing, exists in addition to the user data. The temporary data bin is allocated to the corresponding user. The area ‘ver’ holds the libraries corresponding to a particular STRUC L version.

The processed function packet / master program files are saved from the temporary data bin ‘lnk_tmp’ to the user data area ‘lnk_awd’ after exiting the editor.

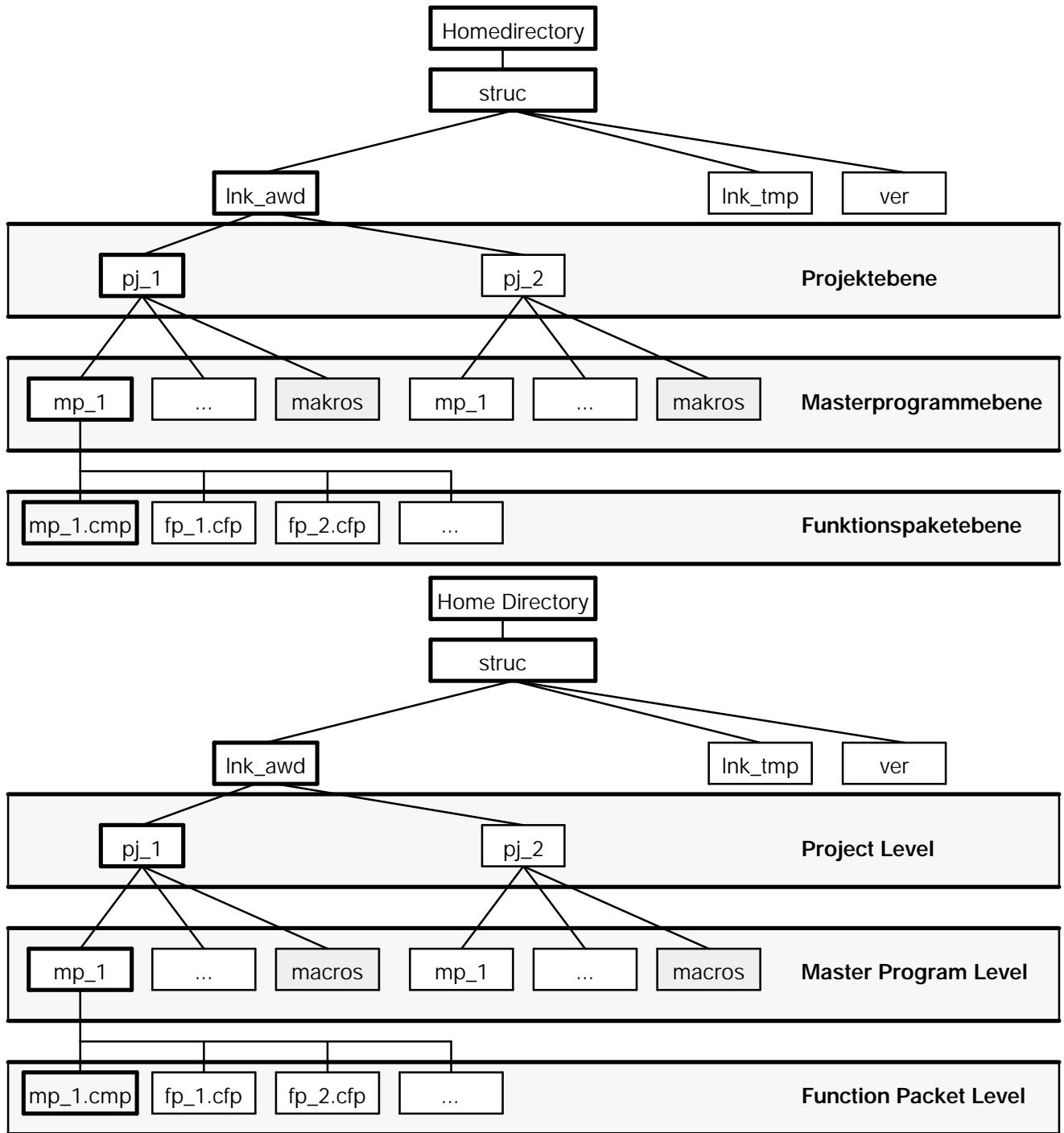


Diagram 1/17: User Data Structure

Each **project** contains the **macro directory** "Macros", next to a series of master programs, into which the project allocated macro files are inserted (2/18).

The macro directory contains (up to 8) **macro sub-directories**, into which the actual **macro files** are saved. Three macro sub-directories with the names "std", "proj" and "misc" are already created when a project is generated (see page 3 – page 8). Further macro sub-directories can be created with the UNIX command **mkdir r**. Their names must correspond to the STRUC name convention, i.e. one to six characters (lower case letters, numbers and underscore), whereby the first character must be a letter.

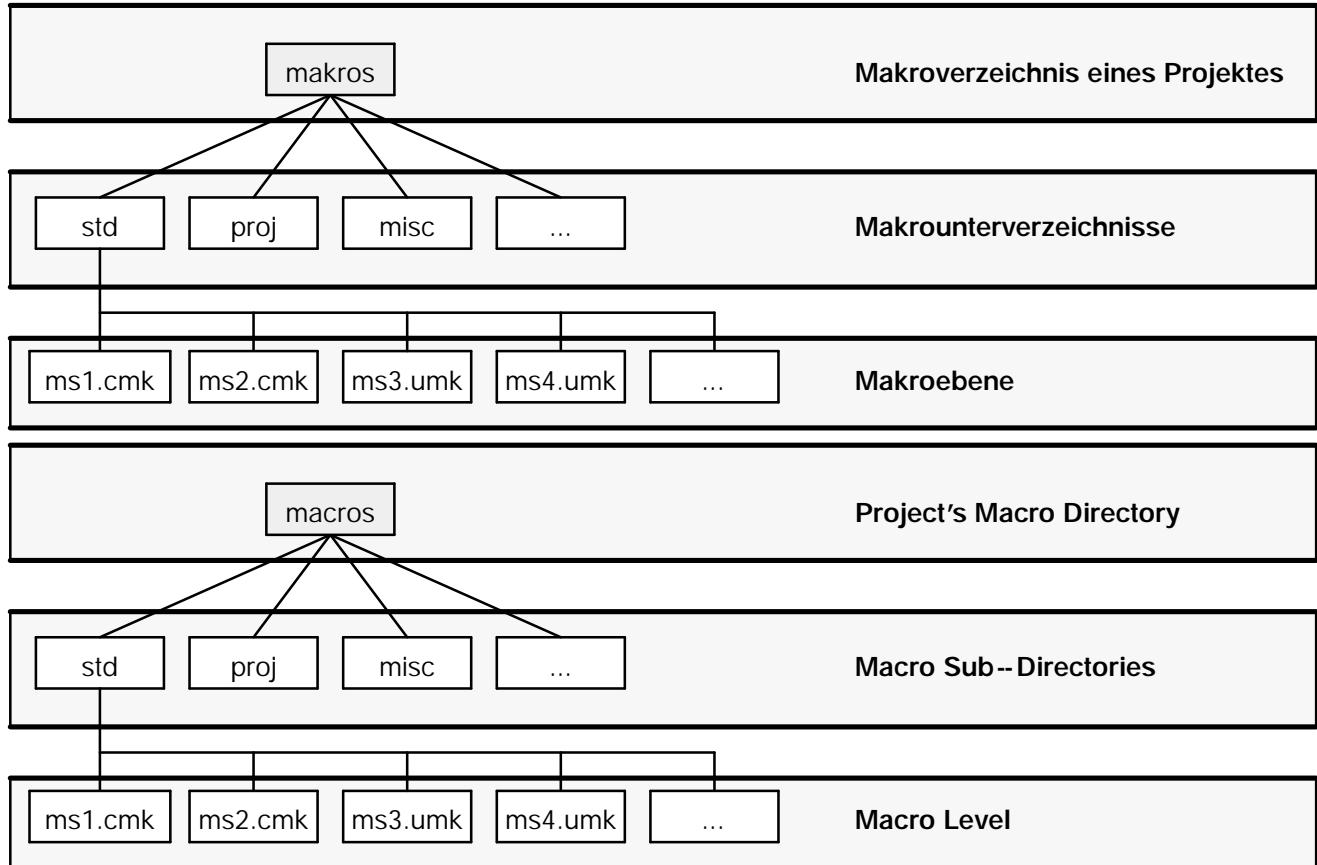
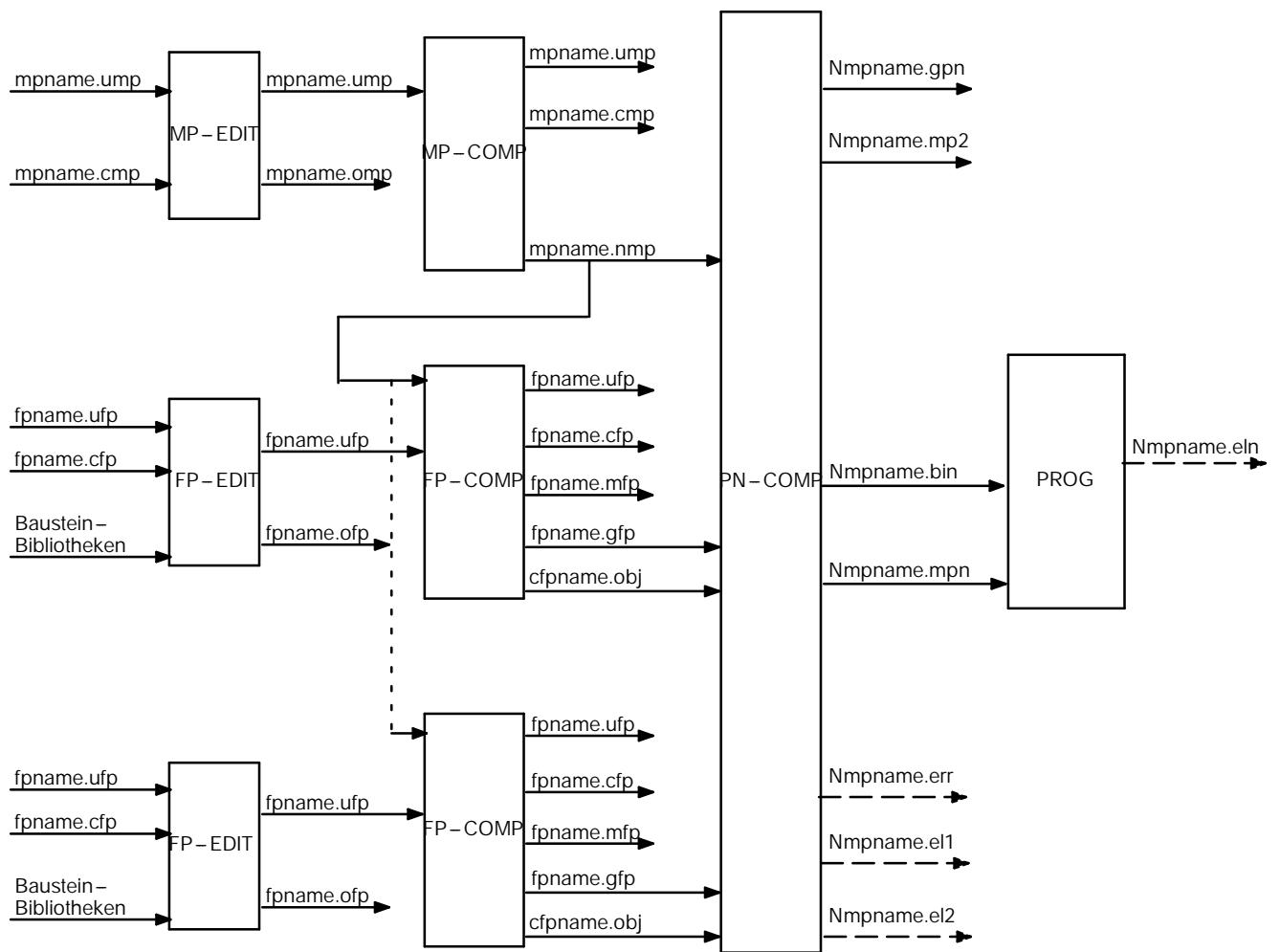
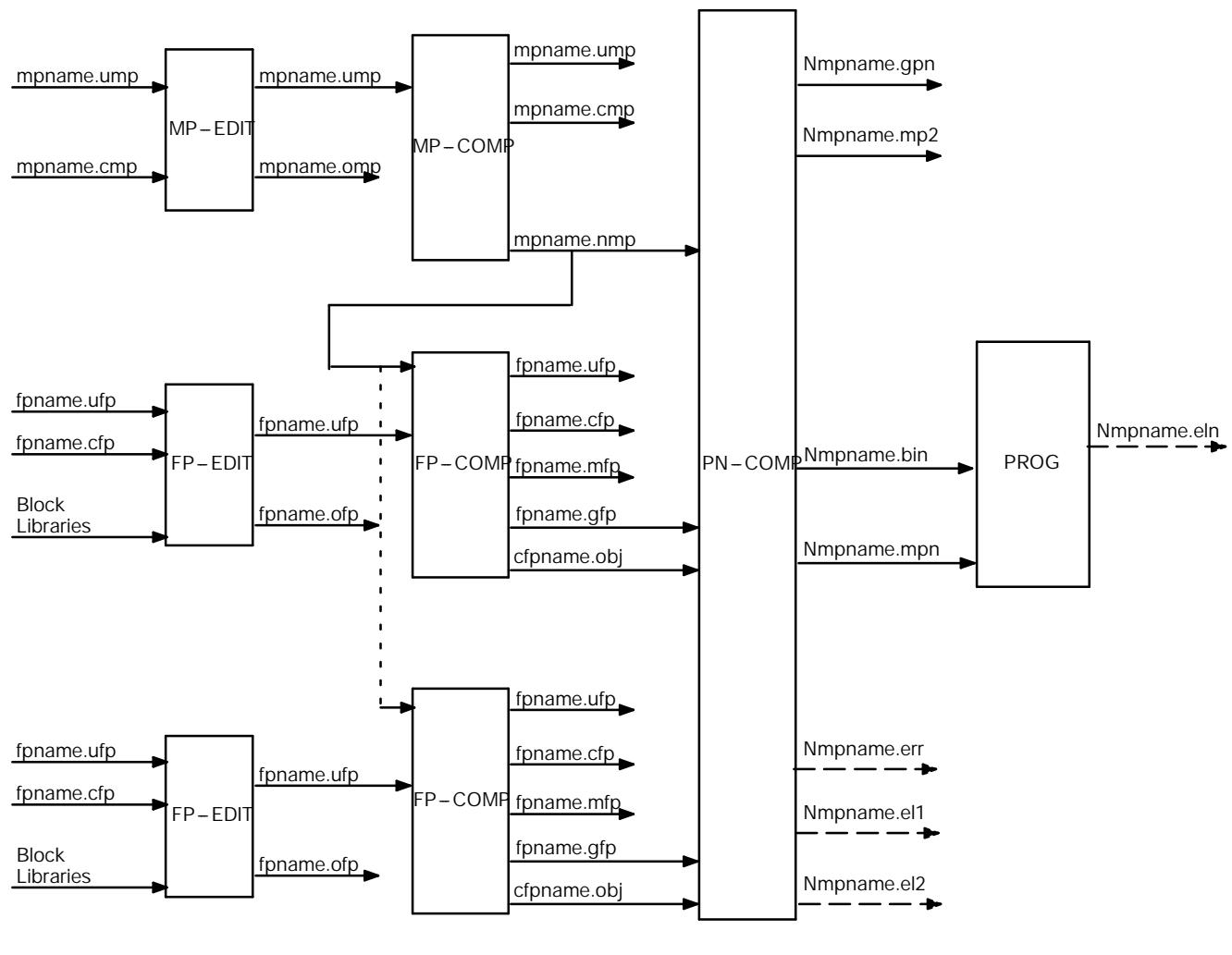


Diagram 1/18: *Macro Directory Structure*

Configuring Files -- P16



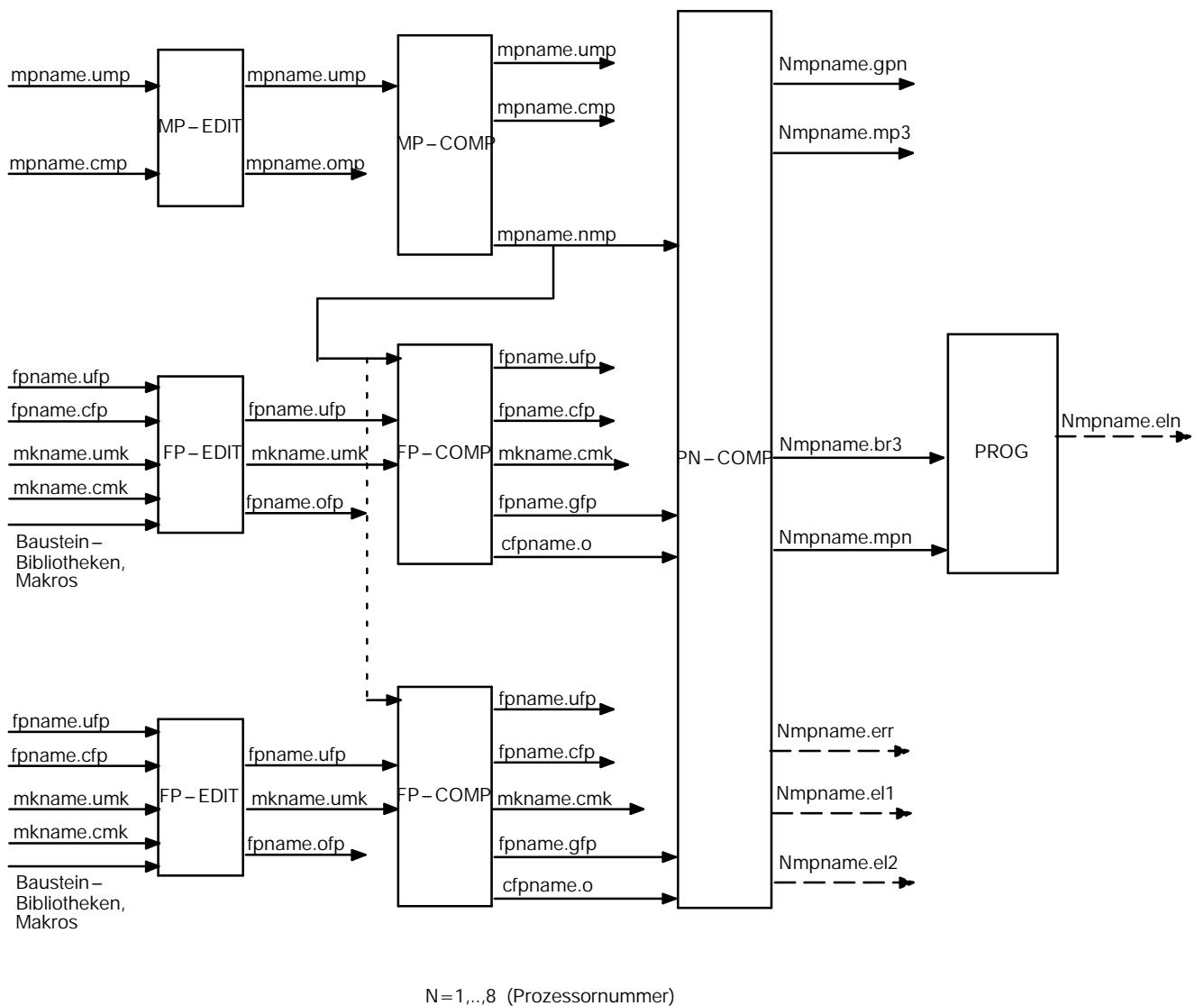
N=1...8 (Prozessornummer)

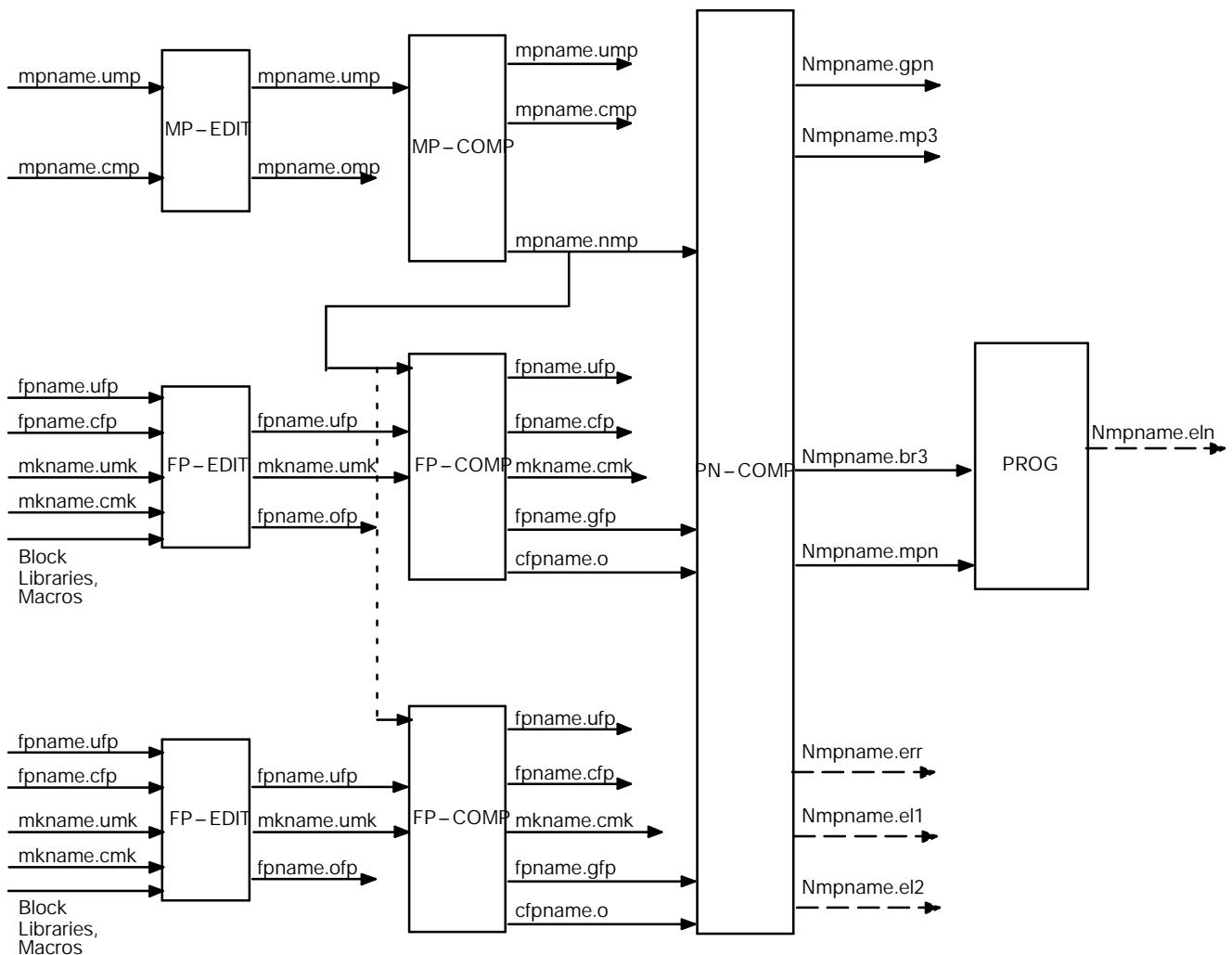


N=1,...,8 (Processor number)

Diagram 1/19: Configuring Files – P16

Configuring Files -- P32





N=1,...,8 (Processor number)

Diagram 1/20: Configuring Files – P32