

# 1 Compiler, Prom, Versions, Libraries, Options

## 1.1 STRUC – Compiler

### 1.1.1 MP–COMP: Compile Master Program

This function is only active, i.e. the sub-menu text is not ghosted, when a master program is selected. If no master program is selected, then the text in the sub-menu is ghosted.

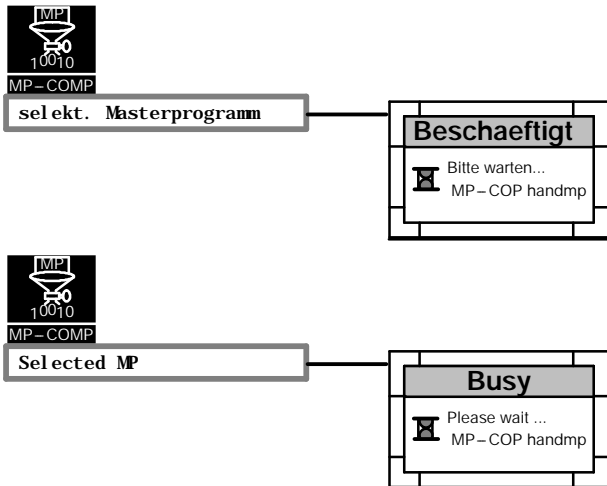


Diagram 1/1: Compile Master Program

#### MP--COMP / Selected MP

The selected master program is compiled using this function.  
The function is immediately implemented, i.e. no further prompting occurs.

Configuring errors, with regard to the SIMADYN D regulations, are detected by the compiler and the number of "Errors" and "Warnings" reported.

The MP EDIT must be re-invoked to correct these errors.

Non compiled master programs or master programs with errors are saved as \*.ump.

An error free compiled master program is saved as \*.cmp.

#### Remark:

The STRUC compilation will not be correct if \*.ump and \*.cmp extensions of a master program name exist simultaneously. This can occur when a master program has been copied into a project directory, at the operating system shell, in which a master program of the same name already exists. Therefore please check whether a master program of the same name already exists before carrying out a copy operation. If the answer is 'Yes', then rename (\*.omp) or erase.

Messages from the compilation process are displayed in the window 'Message Output Basic Dialog'.

### 1.1.2 FP-COMP: Compile Function Packet

This function is only active, i.e. the text of the sub-menu is not ghosted, when a function packet is selected. If no function packet is selected, then the text of the sub-menu is ghosted.

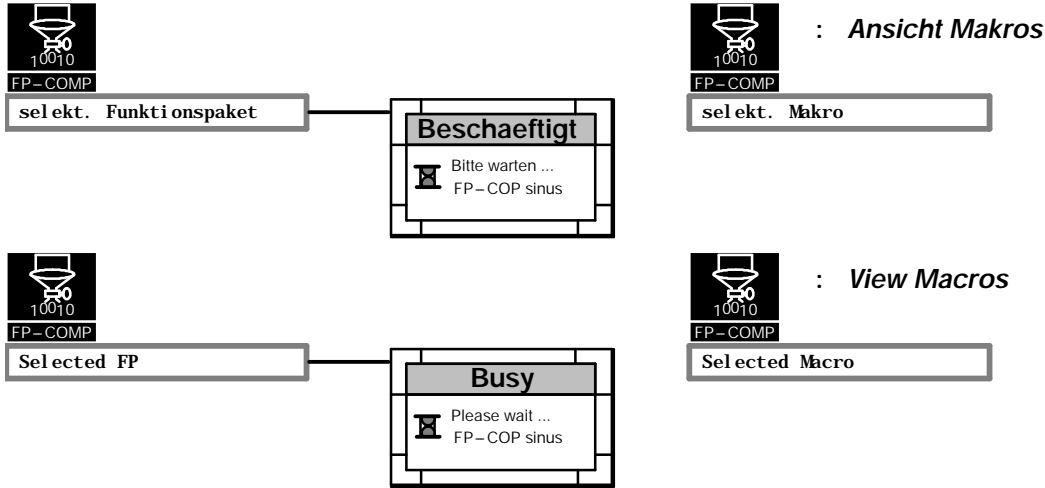


Diagram 1/2: Compile Function Packet

Compile Macro

#### FP-COMP / Selected FP

The selected function packet is compiled using this function.

The function is immediately implemented, i.e. no further prompting occurs.

Configuring errors, with regard to the SIMADYN D regulations, are detected by the compiler and the number of "Errors" and "Warnings" reported.

The FP-EDIT must be re-invoked to correct these errors.

Function packets containing errors or uncompiled function packets are saved as \*.ufp.

An error free compiled function packet is saved as \*.cfp.

#### Remark:

The STRUC compilation will not be correct when \*.ufp and \*.cfp extensions of a function packet name exist simultaneously. This can occur when a function packet is copied into a master program directory, at the operating system shell, in which a function packet of the same name already exists. Therefore please check whether a function packet of the same name already exists before implementing a copy procedure. When the answer is 'Yes', then rename (\*.ofp) or erase.

Compilation procedure messages are displayed in the window 'Message Output Basic Dialog'.

#### FP-COMP / Selected Macro

The selected macro is compiled using this function.

The function is immediately implemented, i.e. no further prompting occurs.

#### Remark:

Only when a macro has been compiled can the function packet, in which the macro is utilized, be compiled.

Configuring errors with regard to the SIMADYN D regulations are detected by the compiler and the number of "Errors" and "Warnings" reported.

The FP EDIT must be re-invoked to correct the errors.

Macros with errors or uncompiled macros are saved as \*.umk.

An error free compiled macro is saved as \*.cmk.

**Remark:**

The STRUC compilation will not be correct when \*.umk and \*.cmk extensions of a macro name exists simultaneously. This can occur when a macro is copied into a sub-directory, at the operating systemhell, in which a macro of the same name already exists. Therefore please check whether a macro of the same name exists before starting the copying procedure. If the answer is 'Yes', then rename (\*.omk) or erase.

### 1.1.3 PN-COMP: Compile Processor Program

This function is only active, i.e. the text in the sub-menu is not ghosted, when a processor program is selected. If no processor program is selected, then the text in the sub-menu is ghosted.

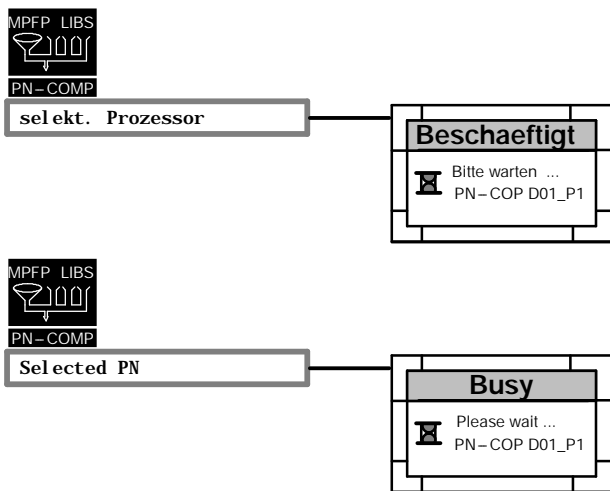


Diagram 1/3: Compile Processor Program

#### PN--COMP / Selected PN

The selected processor program is compiled using this function.

The function is immediately started, i.e. no further prompts are made.

The corresponding error window is popped onto screen, if errors occur during compilation.

The window '**Message Output Basic Dialog**' displays the compilation process messages.

### 1.1.4 AUTO-COMP: Compile Selected Master or Processor Program

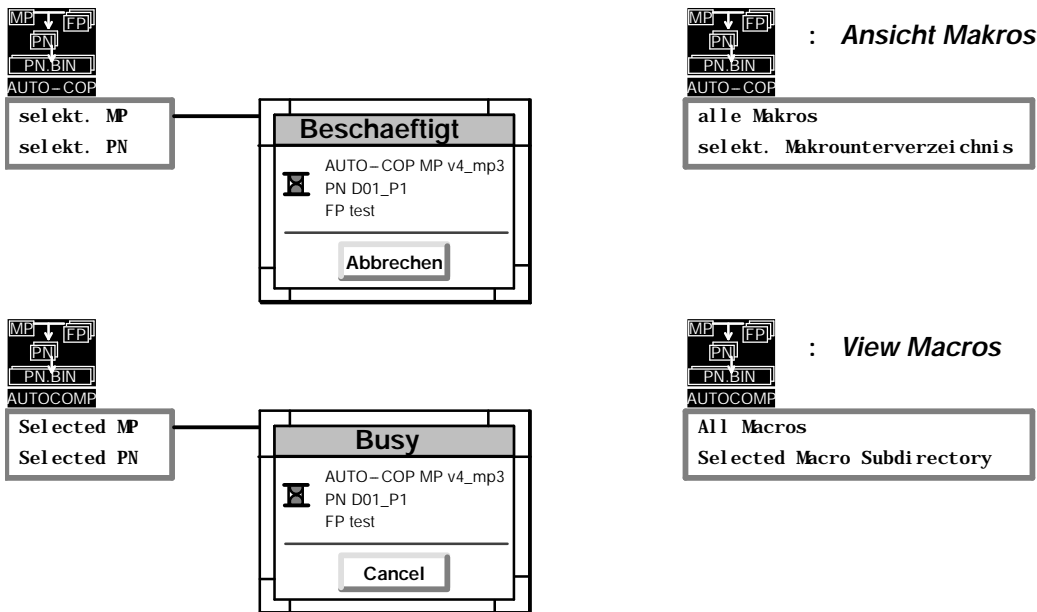


Diagram 1/4: Compile Selected Master or Processor Program

#### AUTOCOMP / Selected PN

This function is only active, i.e. the text of the sub-menu is not ghosted, when a processor program is selected. If no processor program is selected, then the text of the sub-menu is ghosted.

This function compiles the master program (if not compiled yet), all function packets of the selected processor program and the selected processor program.

An error window (Diagram 1/5) is popped onto screen when errors occur during the compilation. The compilation errors in the master programs, function packets or processor programs are listed with the corresponding number of detected error and warning messages.

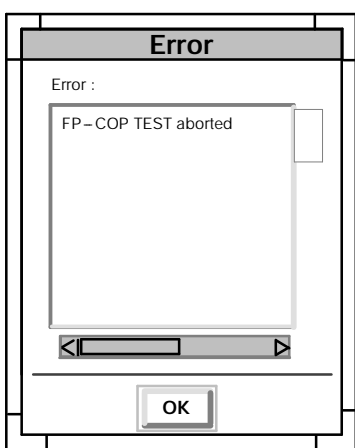


Diagram 1/5: Error Window in 'AUTOCOMP'

**AUTOCOMP / Selected MP**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a master program is selected. If no master program is selected, then the text of the sub-menu is ghosted.

This function compiles the master program, all function packets and all processor programs.

An error window (Diagram 1/5) is popped onto screen when errors occur during compilation. The window displays the compilation errors in the master programs, function packets or processor programs with a listing of the corresponding number of error and warning messages.

**AUTOCOMP / All Macros**

This function is only active, i.e. the text of the sub-menu is not ghosted, when the macro directory is selected. All macros of the selected project are compiled using this function.

An error window (Diagram 1/5) is popped onto screen when errors occur during the compilation. The compilation errors in the macros are listed with the correspondingly detected number of error and warning messages.

**AUTOCOMP / Selected Macro Subdirectory**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a macro sub-directory is selected. If no macro sub-directory is selected, then the text of the sub-menu is ghosted.

All macros of the selected macro sub-directory are compiled using this function.

An error window (Diagram 1/5) is popped onto screen when errors occur during the compilation. The compilation errors in the macros are listed with the correspondingly detected number of error and warning messages.

## 1.2 PROG.: Load Program Memory Sub-Module / Download

### 1.2.1 Load Program Memory Sub-Module

This function is only active, i.e. the text of the sub-menu is not ghosted, when a processor program is selected. If no processor program is selected, then the text of the sub-menu is ghosted.

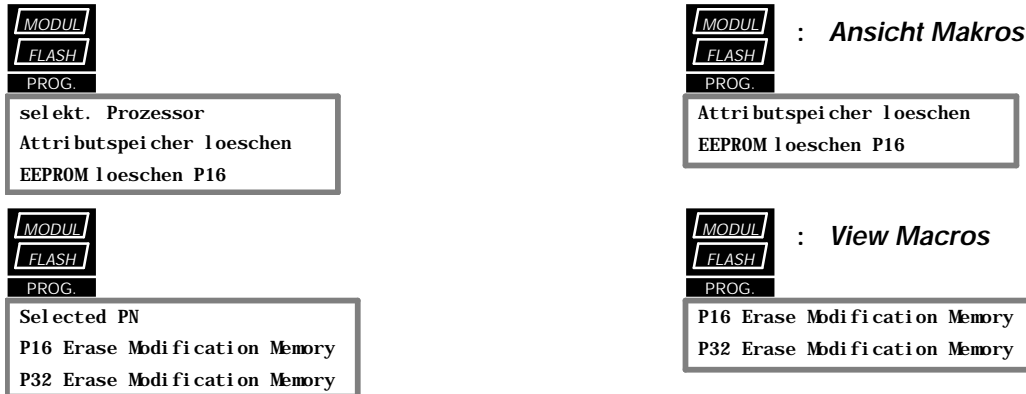


Diagram 1/6: Load Program Memory Sub-Module

#### PROG. / Selected PN

The selected processor program is loaded into a memory sub-module using this function. An acknowledge window is popped onto screen if either no program memory sub-module is installed or a program memory sub-module of an incorrect type has been installed when this function is invoked:

Install Module !

If the program memory sub-module of the type MS4x has not had its memory erased with UV light, then an acknowledge window is also popped onto screen:

Module Not Erased !  
Erase Attribute Memory ?

#### PROG. / P16 Erase Modification Memory / P32 Erase Modification Memory

The functions 'PROG. -- P16 Erase Modification Memory' or 'P32 Erase Modification Memory' erases the modifiable memory area of a program memory sub-module for P32 or P16 processors. The function is immediately implemented, i.e. no further prompting occurs.

## 1.2.2 Download

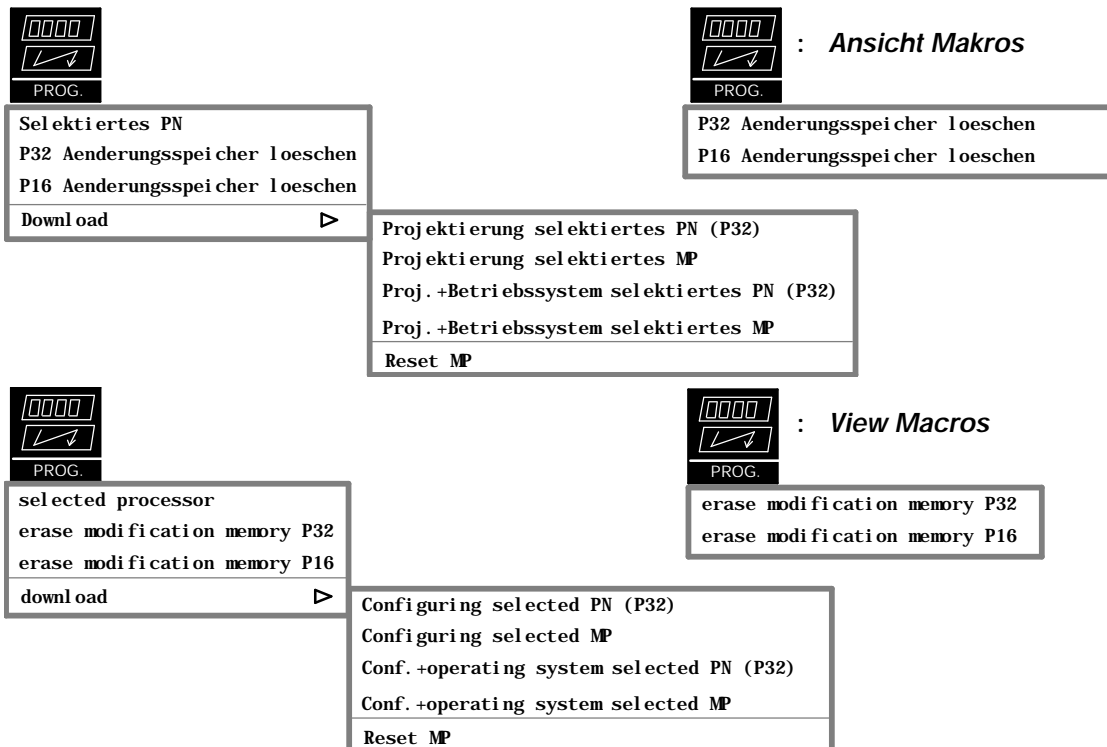


Diagram 1/7: Download

### PROG. / Download / Configuring selected PN (P32) / Configuring selected MP:

The functions 'Download / Configuring selected PN (P32)' or 'Download / Configuring selected MP' initiates a download of the configuring software of a P32–PN or all P32–PN's of a Master program. A prompt will be displayed, before the actual start, to acknowledge whether the download to the selected processor should really be started.

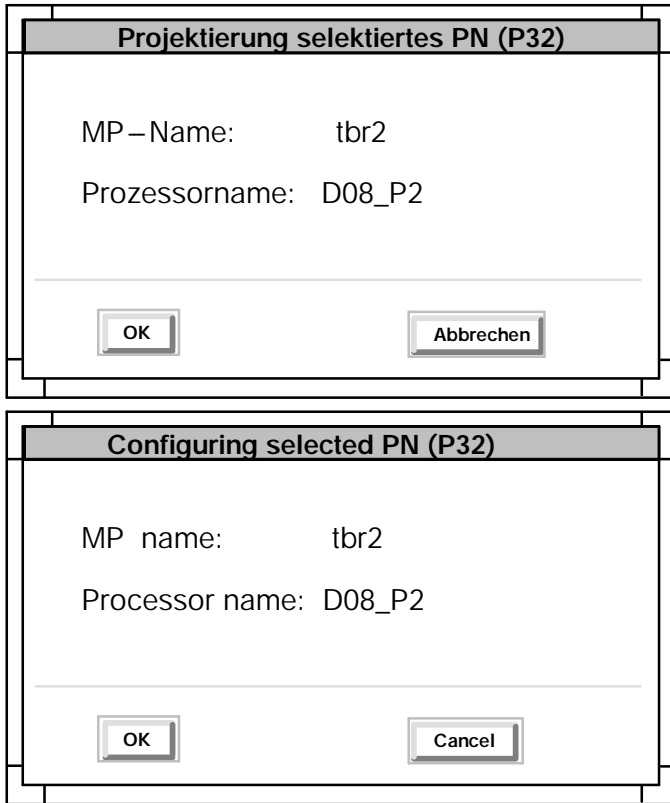


Diagram 1/8: Download, Configuring selected PN

Select 'Cancel' to return to the basic dialog. The download is started by selecting 'OK'. The following window is popped onto screen during the download procedure and contains information regarding the progress of the load:



Download läuft			
MP-Name: tbr2			
PN-Name	Speicher kByte		Ladefortschritt in %
	belegt	maximal	
D08_P2	648	2048	28
<input type="button" value="Reset MP"/> <input type="button" value="Schliessen"/> <input type="button" value="Abbrechen"/>			

Download running			
MP name: tbr2			
PN name	Memory kByte		Load progress in %
	used	maximum	
D08_P2	648	2048	28
<input type="button" value="Reset MP"/> <input type="button" value="Close"/> <input type="button" value="Cancel"/>			

Diagram 1/9: Information Window during the Download Procedure

The current progress of the load procedure is displayed on screen in %. The 'Reset', 'Close' and 'Cancel' buttons of a processor are switched **inactive** during an active load. The procedure can be cancelled between the loads of individual processors (the 'Cancel' button is active) when loading an MP (several processors). Reset the rack ('Reset MP') or exit the procedure ('Close') after completion of the download.

When an error occurs, during the download procedure, then an error window is popped onto screen that must be acknowledged:

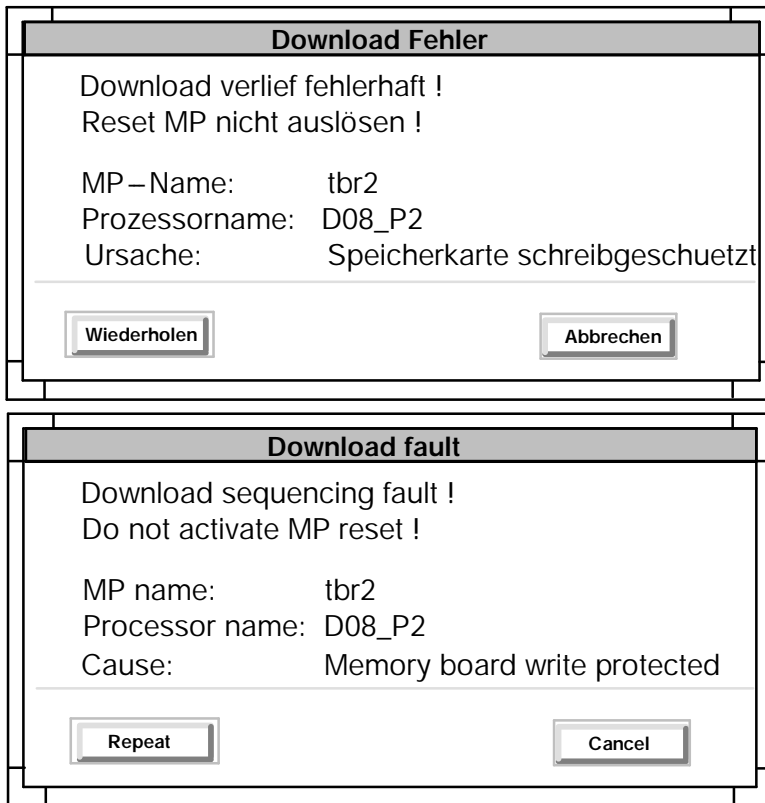


Diagram 1/10: Error Window

The download can be repeated after the occurrence of an error ('Repeat') or terminated ('Cancel'). It should also be noted that, due to the interrupted download caused by the error, the memory contents can possibly be inconsistent and that **no reset** is to be activated.

**PROG. / Download / Conf. + Operating System selected PN (P32) / Conf. + Operating System selected MP:**

The functions '**Download / Conf. + Operating System selected PN (P32)**' or '**Download / Conf. + Operating System selected MP**' initiates a download of the complete program (configuring software and Operating system) of a P32-PN or all P32-PN's of a Master program.

The subsequent dialog sequencing is the same as the functions '**Download / Configuring selected PN (P32)**' or '**Download / Configuring selected MP**'.

**PROG. / Download / Reset MP:**

The function '**Download / Reset MP**' activates a rack reset. This function will also prompt for a repeated intention to reset the rack.

### 1.3 VERSION: Version Matching / FP – Transformation P16 <-> P32

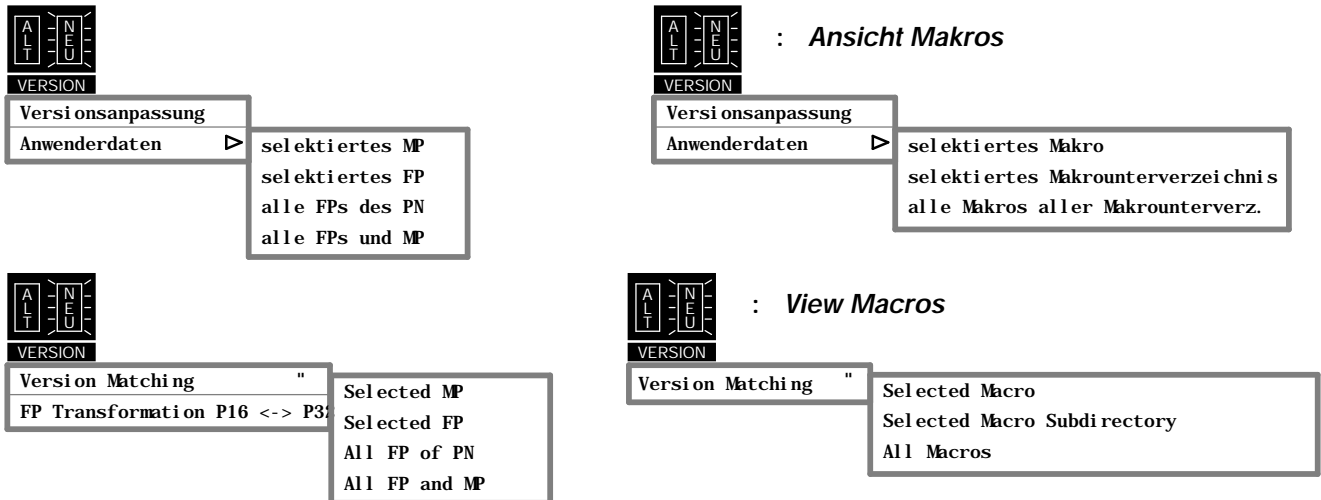


Diagram 1/11: Sub-Menu Title: Version Update

A version update can be implemented using the function **'VERSION / Version Matching'**.

A version update of the user data is necessary when either the release (or the version) of the standard or user libraries, utilized in the configuring, is modified or a macro utilized in a function packet is once again changed after already being invoked in a function packet.

If the layout (visual structure) of the corresponding block or macro has not been altered, then it is automatically revised during the version update.

Blocks, whose layout have changed, such that no automatic version update is possible, are flagged with "?" and must be revised using the function **Configure block / SUBMSK** in the function packet editor.

A layout modification can be:

- New or deleted connectors,
- Modification of the type of connector,
- Block was deleted from the library.

The re-compilation of the processor program is mandatory, subsequent to a version update: AUTOCOMP.

An error window (review Diagram 1/5) is popped onto screen when errors occur during the version update. The window contains a listing of the corresponding master programs or function packets with the number of detected blocks marked as faulty with "?".

#### Remark:

If the libraries have been modified and no version update is carried out, then the remark: **"Version Incompatible"** is displayed when the **list editor** is invoked. If the libraries have only one release modification, then an additional prompt is displayed: "Version Update Y/N?". If this prompt is acknowledged with "Y", then a version update is carried out before invoking the editor. If this prompt is acknowledged with "N", then no version update is carried out and the MP/FP can only be displayed but not modified or the list editor can possibly be immediately aborted. If the version is modified, then only an up – wards compatible up – date version is possible, i.e. <system software (i.e. V400<V420). If however the version has been changed, then the remark: "Version Update Not Possible" is displayed and the MP/FP can only be displayed but not modified or the list editor can possibly be immediately aborted.

If the libraries have been modified and no version update has been carried out, then no version update is forced when the **graphics function packet editor** is invoked. The editor prevents blocks being inserted into the function packet from the library in question.

#### **VERSION / Version Matching / Selected MP**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a master program is selected. If no master program is selected, then the text of the sub-menu is ghosted.

This function implements a version update for the selected master program file.

#### **VERSION / Version Matching / Selected FP**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a function packet is selected. If no function packet is selected, then the text of the sub-menu is ghosted.

This function implements a version update for the selected function packet file.

#### **VERSION / Version Matching / All FP of PN**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a processor program is selected. If no processor program is selected, then the text of the sub-menu is ghosted.

This function implements a version update for all function packets of the selected processor program.

#### **VERSION / Version Matching / All FP and MP**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a master program is selected. If no master program is selected, then the text of the sub-menu is ghosted.

This function implements a version update for the selected master program and all function packets within this selected master program.

#### **VERSION / Version Matching / Selected Macro**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a macro is selected. If no macro is selected, then the text of the sub-menu is ghosted.

This function implements a version update for the selected macro file using this function.

#### **VERSION / Version Matching / Selected Macro Subdirectory**

This function is only active, i.e. the text of the sub-menu is not ghosted, when a macro sub-directory is selected. If no macro sub-directory is selected, then the text of the sub-menu is ghosted.

This function implements a version update for all macros of the selected macro sub-directory.

#### **VERSION / Version Matching / All Macros**

This function is only active, i.e. the text of the sub-menu is not ghosted, when the macro directory is selected. This function implements a version update for all macros of the selected project.

#### **VERSION / FP-- Transformation P16< -- >P32**

The FP–transformation is a special kind of version up–date. This function is only active, i.e. the text of the sub–menu is not ghosted, when the function packet is selected. If no funktion packet is selected, then the text of the sub–menu is ghosted.

If the selected processor is a P32 processor and the function packet is a P16 function packet, then the function packet is transformed in a P32 function packet.

If the selected processor is a P16 processor and the function packet is a P32 function packet, then the function packet is transformed in a P16 function packet.

If the selected function packet is not attached to any processor (?????? PN), then this function transforms a P16 function packet to a P32 function packet or a P32 function packet to a P16 function packet.

If the function packet, that must be selected, was created by an older system software version, then a version adaptaion is additionally carried out while the transformation. If the maske layout of a transformed block has changed, then this block is marked with "?" as the version adaptation. While projecting the block must be replaced with the function **Configure Block / SUBMSK**.

## 1.4 LIBRARIES: Select Libraries



Diagram 1/12: Select Libraries

The library selection window is popped onto screen when this function is invoked.

The user can select three of the available libraries, in addition to the standard block rack 'FBSLIB' or the block library 'IBSLIB', for configuring his project.

The standard library is always present, even when no additional libraries have been selected.

It is part of the STRUC scope of supply and cannot be modified.

**Bibliotheken selektieren**

Grundsoftware GSW    V420    940901

FB-Bibliotheken	Version	Stand	P16	P32	IB-Bibliotheken	Version	Stand	P16	P32
FBSLIB	V420	940901	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	IBLSIB	V420	940901	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
A: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	B: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	C: frei	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

OK

---

**Select Libraries**

Basic Software GSW    V420    950401

FB-libraries	Version	Release	P16	P32	IB-libraries	Version	Release	P16	P32
FBSLIB	V420	950401	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	IBLSIB	V420	950401	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
A: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	A: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	B: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	C: Empty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

OK

Diagram 1/13: Select Libraries

The selection is implemented by activating the buttons in one of the library lines A:, B: or C: (column FB libraries or IB libraries). A sub-menu is allocated to each button:

Select  
Deselect

The function "**Select**" is only active when the button is designated with "Empty". The available libraries are listed in a selection window when "**Select**" is invoked. A library can only be allocated once, i.e. this library is no longer listed as available when select is re-invoked. The FB libraries must be selected for the configuring of function packets or the IB libraries for the configuring of master programs.

The allocation of the FB libraries must be same for all function packets of a processor program.

One of the library lines A, B or C allocated libraries can be deselected by selecting the function "**Deselect**" in the popped up sub-menu. The library is then no longer active. The button is then designated with "Empty".

The version of the **supplementary libraries** must always be **the same** as the version of the GSW (Basic Software). The release can however differ. The following applies:

P16

Block are available for P16 Function Packets.

P32

Blocks are available for P32 Function Packets.

Blocks for the P16 and for the P32 function packets are normally available.

**Libraries** consists of **several inter-dependent** files (shown in the example for the library FBSL03):

P16:	fbsl03.m86	P32:	fbsl03.mr3
	fbsl03.l86		fbsl03.a
	fesl03.l86		fesl03.a

If, for example, after the installation of a **new user library**, the library name fbaxxx does not appear in the selection window of the available libraries, then individual files are possibly missing. This can be checked with the UNIX command:

**ls -al \$HOME/struc/ver/v400/f?axxx\***

Clicking on the 'OK' button exits the selection function.

## 1.5 OPTIONS: Address Book, FP-(UN)Lock, Save Error Image, Internal Graphics Editor, Communication Island

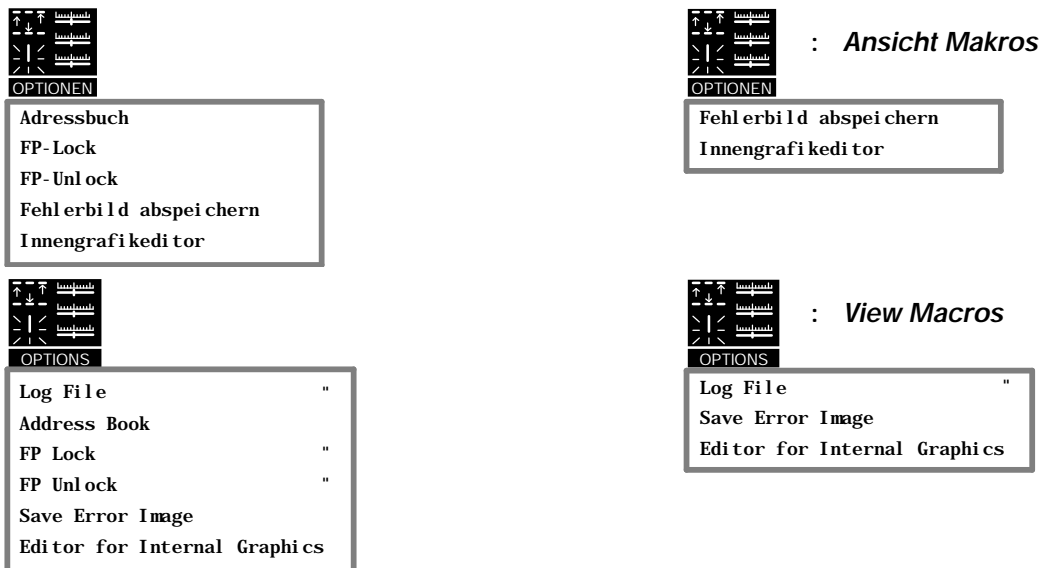


Diagram 1/14: OPTIONS

### 1.5.1 Log File

The error message, created in COMP functions, and the listen editor invokes are collected in an log file. This log file can be loaded with the function '**OPTIONS / Log File / View**' and erased with the function '**OPTIONS / Log File / Delete**'.

This function is only active, i.e. the text of the sub-menu is not ghosted, when a function packet is selected. If no function packet is selected, then the text of the sub-menu is ghosted.

### 1.5.2 Address Book

A directory of all connector addresses, of the selected P32 processor program, is created using this function. This function is only active, i.e. the text of the sub-menu is not ghosted, when a P32 processor is selected and a processor program does exist. If this function is invoked for a P16 processor program, then it will abort with an error message.

### 1.5.3 FP Lock

This function is only active, i.e. the text of the sub-menu is not ghosted, when a function packet is selected. If no function packet is selected, then the text of the sub-menu is ghosted.

This function disables the selected function packet for retro-documentation. A disabled function packet cannot be loaded again from the binary file (know-how-protection).

### 1.5.4 FP Unlock

This function is only active, i.e. the text of the sub-menu is not ghosted, when a function packet is selected. If no function packet is selected, then the text of the sub-menu is ghosted.

An FP lock disable is removed using this function.

### 1.5.5 Save Error Image

If errors have occurred in the graphics editor during the processing of a master program or function packet, then you should first save the corresponding error image and send it to your service center before re-invoking the editor.

The configuring steps up to the occurrence of the error and normally also the error itself can be exactly reproduced with the aid of this error image. This is the prerequisite for the fastest possible error analysis and resolution.

The error image contains:

- Implemented editor commands and their effects since invoking STRUC G,
- The last selected function packet and master program file,
- Page lists, created by the editor, for the currently selected function packet.
- Cross reference lists of the processor program, created by the compiler
- The core-dump file after a program plunge

#### Create Error Window

An error image can be created by invoking the function '**OPTIONS / Save Error Image**' after exiting the graphics function packet or master program editor.

The error image is inserted as a file in the directory "\$HOME/struc/lnk\_tmp/fel". An information window containing the error image file names is displayed after the error image has been generated.

If the function is invoked, when a previous error image had already been saved, then a dialog window is displayed containing the prompt to either overwrite the old error image or cancel the function.

#### Downloading the Error Image

Save the error image file with "Betrieb " system commands on a DOS-disc:

```
$cd$HOME/struc/lnk_tmp/fel
$dos
```

```
c:\      \struc\lnk_tmp\fel>copy *.* a:          dos-disc in Laufwerk a:
```

The data carrier should be sent, together with the "SIMADYN D Report", to the SIMADYN-D-Hotline ASI1 V122 PTS. The Error image is an archiv file.

With the following UNIX command the content of the archiv file is loaded:

```
tar -tvf f_<fp-name>      or      m_<mp-name>
```

With the following UNIX command error image files of the archiv file are loaded :

```
tar -xvf f_<fp-name>
```

If the error image is not immediately downloaded, then it will be overwritten by the next error image, since:

**The directory \$HOME/struc/lnk\_tmp/fel can only hold one error image.**

**You should therefore immediately download a generated error image and subsequently delete it in the directory '\$HOME/struc/lnk\_tmp/fel'.**

### 1.5.6 Internal Graphics Editor

The chapter describes the internal graphics editor of STRUC G. It is designed to process the internal graphics that symbolize the block functionality in the graphics function packet editor and during plotting of the function packets (Diagram 1/15).



The internal graphics editor is invoked from the basic dialog via the menu item 'OPTIONS – Internal Graphics Editor' and operates within its own screen window (Diagram 1/17).

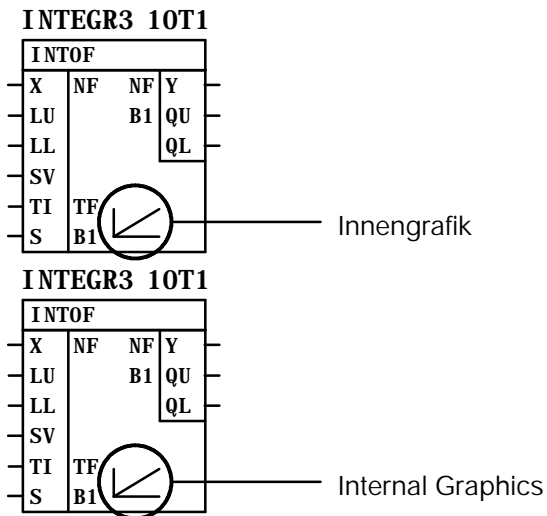


Diagram 1/15: Internal Graphics of a Function Block

Internal graphics for the standard block library ('FBSLIB') and also for macros are defined after the installation of STRUC.

The application of an internal graphics editor is practical when special internal graphics are to be created for **macros** or blocks from **user libraries**.

The internal graphics of the function blocks are held in their own **internal graphics libraries**. A host internal graphics library can be created for each block library and for macros. It is possible to differentiate between individual SIMADYN versions even here.

### 1.5.6.1 Library Names

The names of the internal graphics libraries are generated according to the following pattern:

- D 'fbxxx.vy.igr' (for internal graphics of function block libraries),
- D 'macros.vy.igr' (for internal graphics of macros),

where fbxxx is the name of the associated block library, vy the version of the library.

Examples:

- D The internal graphics library for the block library 'FBA123' from version 4.2 must receive the name 'fba123.v4.igr'.
- D The internal graphics library for the block library 'FBSL03' from version 3.2 must receive the name 'fbsl03.v3.igr'.

The function packet editor searches for internal graphics libraries in the directories:

- D '\$HOME/struc/lnk\_igr'  
The user can insert his own internal graphics libraries here.
- D '\$HOME/struc/lnk\_font'  
The internal graphics libraries supplied with the STRUC installation are inserted here. These libraries should not be modified.

Note: '\$HOME' designates the users home directory.

### 1.5.6.2 Block Classes and Identifier Characters

Each function block of a block library is allocated its internal graphics using the **block class** (1 – 9) and an **identifier character** (a–z or A–Z).

An internal graphics library can therefore contain up to 52 individual internal graphics per block class.

The function blocks from the standard library are divided into the following block classes, (review 'Technical Description: Standard Function Blocks P16 / P32'):

- D Class 1: Control Blocks
- D Class 2: Power Converter Specific Blocks
- D Class 3: Arithmetic Blocks
- D Class 4: Logic and Switching Blocks
- D Class 5: Input / Output Blocks
- D Class 6: Communication Blocks
- D Class 7: Conversion Blocks
- D Class 8: Service / Diagnostic Blocks
- D Class 9: Special Communications and Message Blocks.

The definition of block class and internal graphics identifier character is implemented for **user function blocks**, in a user block generator. It is recommended to maintain, as far as practical, the scheme utilized for the standard function blocks.

**Macros** are allocated the **block class 1**. The internal graphics identifier character can be configured during the processing of the macro.

### 1.5.6.3 Files of an Internal Graphics Library

The following files are inserted in each internal graphics library 'fbxxx.vy.igr':

- D A file 'fbxxx.vy.igr'
- D Files 'fbxxx.vy.bstz' for each block class z, for which internal graphics are defined in the library.

### 1.5.6.4 Layout of an Internal Graphic

The individual internal graphics consist of **31\*25** grid points (Diagram 1/16). Symbols consisting of individual **lines** (vector graphics) are generated within this grid. This permits internal graphics to be displayed on various output devices in any magnifying factor.

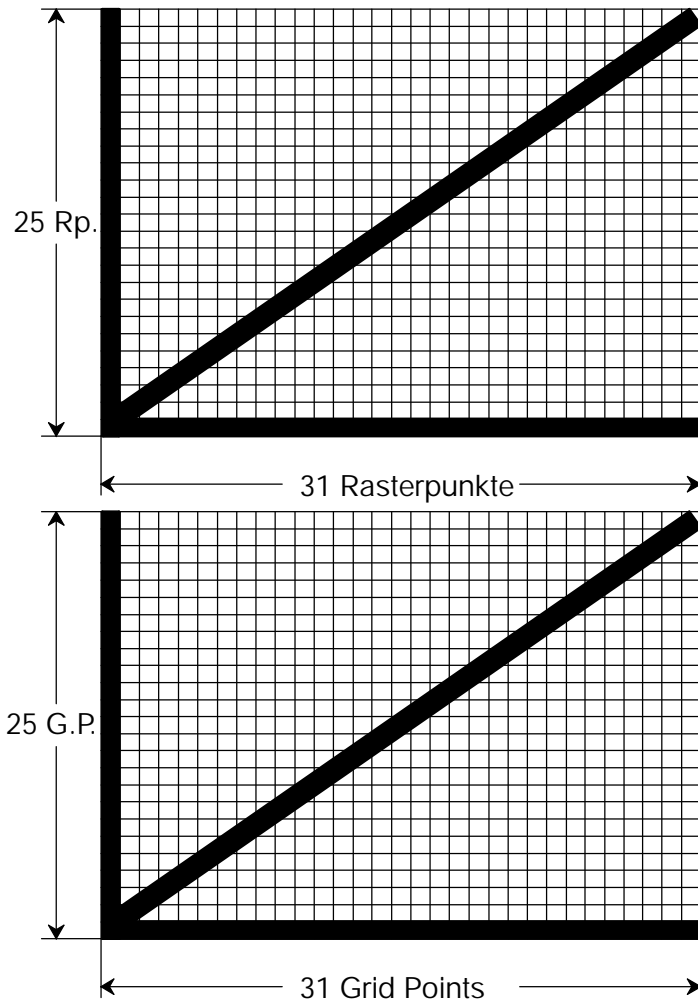


Diagram 1/16: Internal Graphics Symbol in the Grid 31\*25

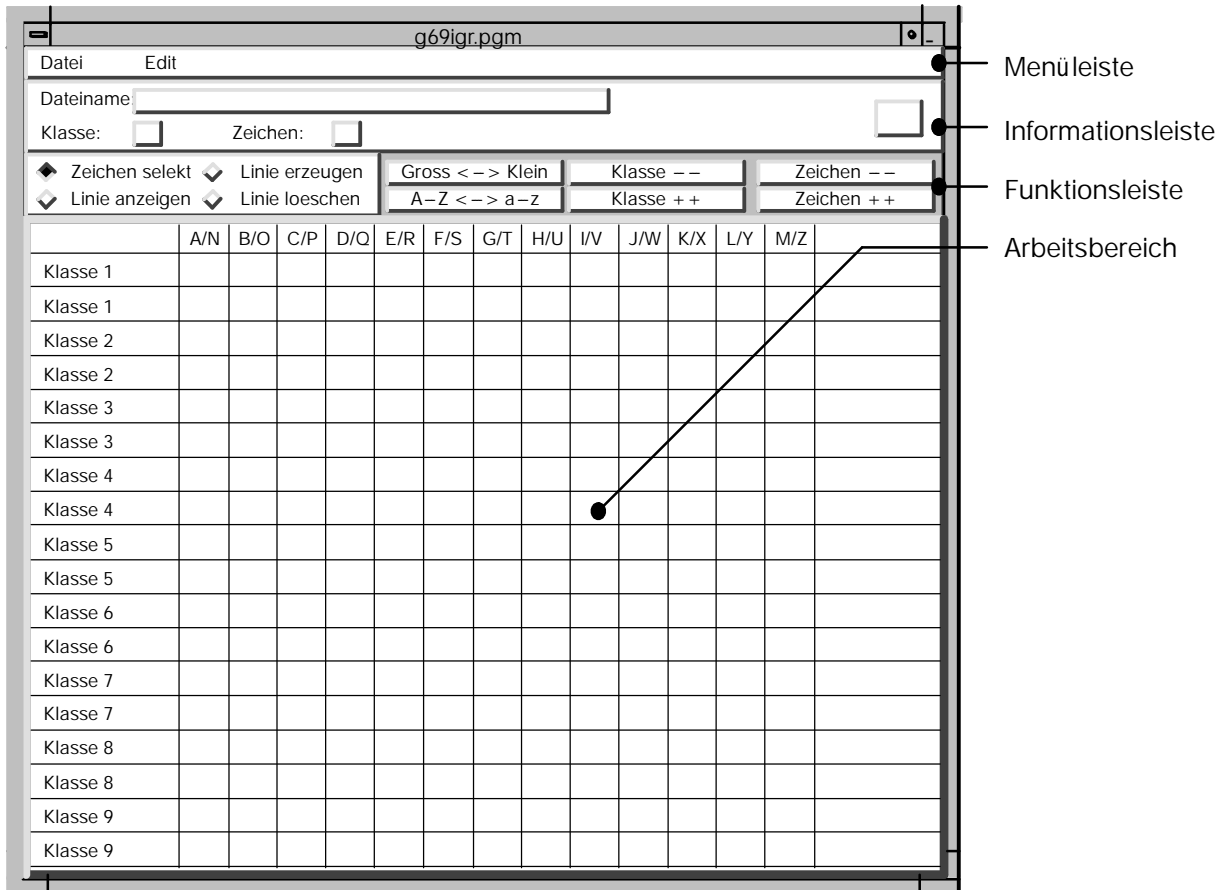
### 1.5.7 The Screen Window of the Internal Graphics Editor

The screen window of the editor consists of :

- D The menu bar,
- D The information strip,
- D The function strip and
- D a working area.

An **overview mode** of the editor permits all the internal graphics contained within the library to be displayed. Individual internal graphics are processed in the working area in the **editing mode** (see Diagram 1/31).

Diagrammm 7/13 shows the screen window in the overview mode after invoking the editor. No internal graphics library has been loaded as yet for processing.



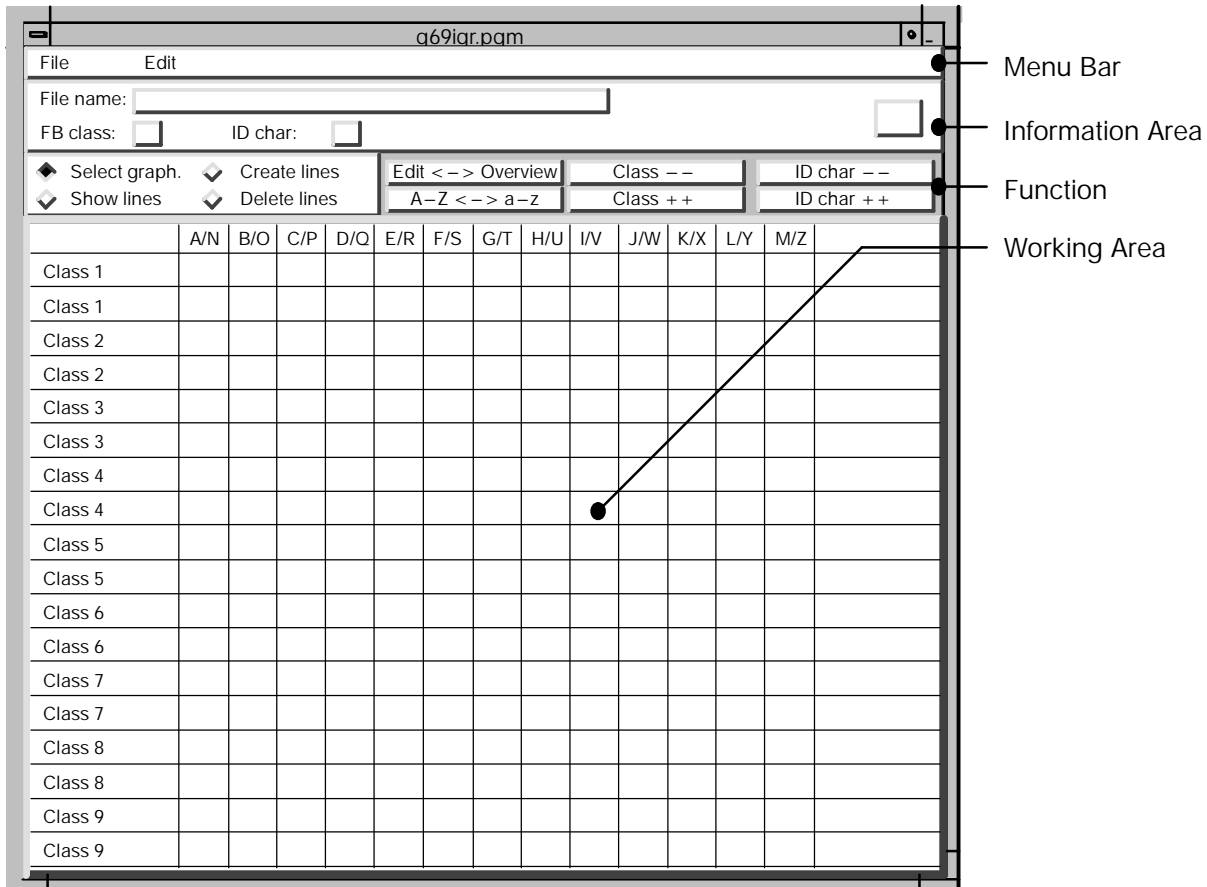


Diagram 1/17: The Differing Screen Window Areas

## 1.5.8 The Menu Bar of the Internal Graphics Editor

The menu bar of the internal graphics editor consists of the two menus 'File' and 'Edit'.

### 1.5.8.1 The File Menu

The file menu contains functions for loading and saving internal graphics libraries as well as the menu item for exiting the editor.

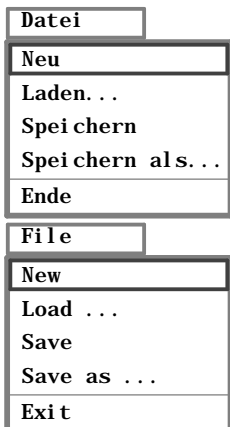
**File -- New**

Diagram 1/18: *Create New Internal Graphics Library*

Selection of the menu item 'New' causes an empty internal graphics library to be created and loaded. The name of the library is only defined during the saving procedure.

If a library is currently being processed that has been modified but not saved, then the function 'File – New' prompts whether the modified library should be saved first.

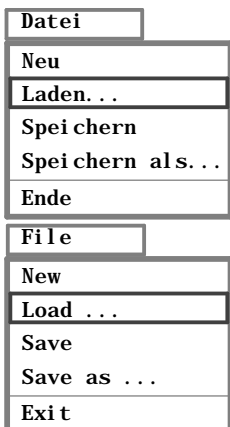
**File -- Load**

Diagram 1/19: *Load Internal Graphics Library*

This function loads an already existing internal graphics library for processing in the editor. A **file selection window** (Diagram 1/20) is popped onto screen for the selection of this library.

If a library is currently being processed, that has been modified and not saved, then the function 'File – Load' prompts whether the modified library should first be saved before the new library is loaded.

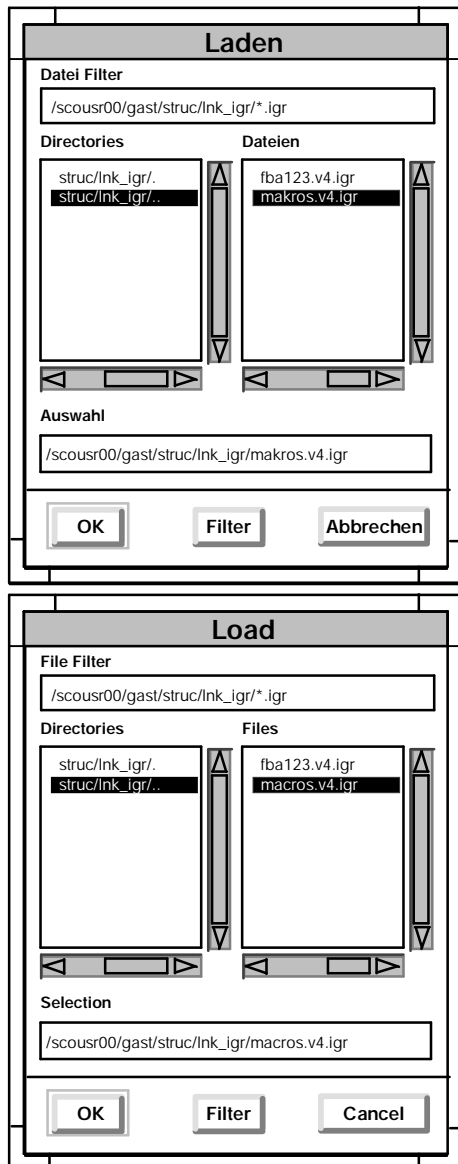


Diagram 1/20: File Selection Window When Loading an Internal Graphics Library

The panels 'Directories' of the selection window requires the selection of a directory in which the internal graphics library is located. The selection is implemented by clicking on the corresponding entry with the left-hand mouse key and subsequently activating the 'Filter' button (or simply double clicking the desired entry).

If the directory has been selected, then the desired internal graphics library is selected in the panel 'Files' and the window exited with the 'OK' button.

The total path name is displayed in the panel 'Selection'. Entries are also possible into this panel.

Loading can be aborted by activating the 'Cancel' button.

#### File -- Save

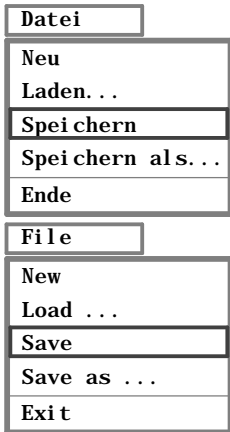


Diagram 1/21: Save Internal Graphics Library

This function saves the processed internal graphics library.

If the library has not yet been defined with a name (after invoking the function 'File – New'), then the function 'File: Save as' pops a file selection window onto screen (Diagram 1/23).

#### File -- Save As

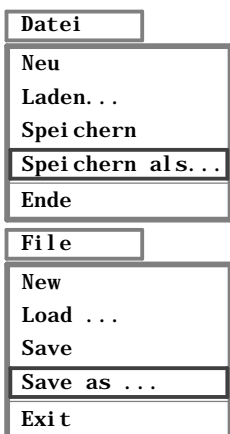


Diagram 1/22: Save Internal Graphics Library under Another Name

This function is designed to save the processed internal graphics library under another name. The file selection window (Diagram 1/23) is therefore popped onto screen.



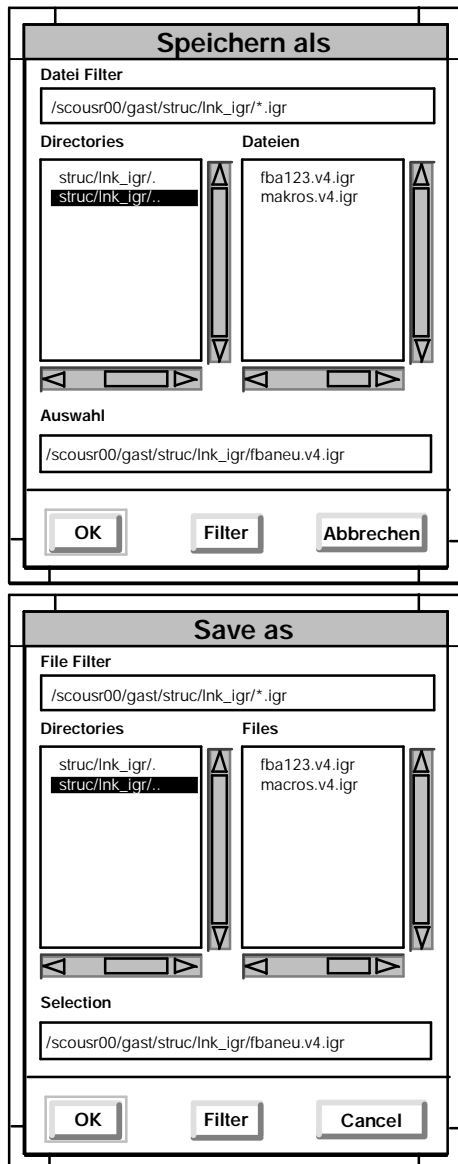


Diagram 1/23: File Selection Window During Saving of an Internal Graphics Library

This window permits either an existing library name to be selected (in the panel 'Files') or a new name to be entered (in the panel 'Selection'), analogous to the fast selection during loading of an internal graphics library.

**Warning:** If a library exists with the selected name, then this will be over-written **without further prompting** !

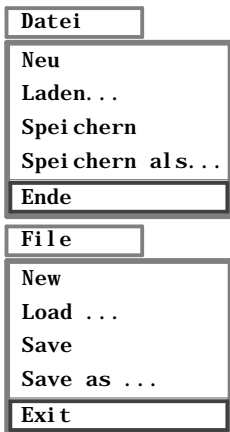
**File -- Exit**

Diagram 1/24: *Exiting the Internal Graphics Editor*

This function exits the internal graphics editor. The editor prompts whether the currently processed library should be saved first.

**1.5.8.2 The Edit Menu**

The edit menu contains functions for deleting and copying the currently processed internal graphic.

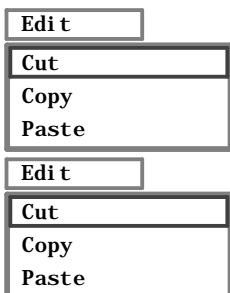
**Edit -- Cut**

Diagram 1/25: *Cut Internal Graphic and Insert into the Data Bin*

The function 'Edit – Cut' deletes the current internal graphic and inserts it into the data bin.

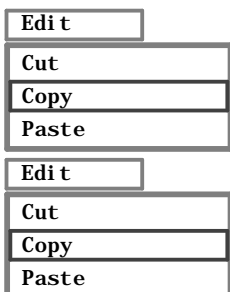
**Edit -- Copy**

Diagram 1/26: *Copy the Internal Graphic into the Data Bin*

The function 'Edit – Copy' inserts a copy of the current internal graphic into the data bin.

## Edit -- Paste

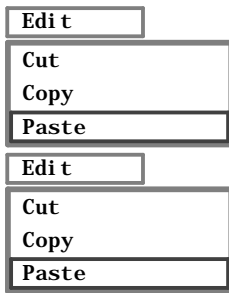


Diagram 1/27: Extract Internal Graphic from Data Bin

The function 'Edit – Paste' inserts the contents of the data bin into the current internal graphic. The data bin is not deleted, such that this internal graphic can be inserted as often as required.

The instruction sequence 'Cut' and 'Paste' moves therefore an internal graphics symbol, the sequence 'Copy' and 'Paste' duplicates it.

### 1.5.9 The Information Strip of the Internal Graphics Editor

The information strip of the editor (Diagram 1/28) displays the file name of the processed internal graphics library as well as the block class and the identifier character of the current internal graphic.

The internal graphics processed in the working area during the editors editing mode is displayed here in normal size.

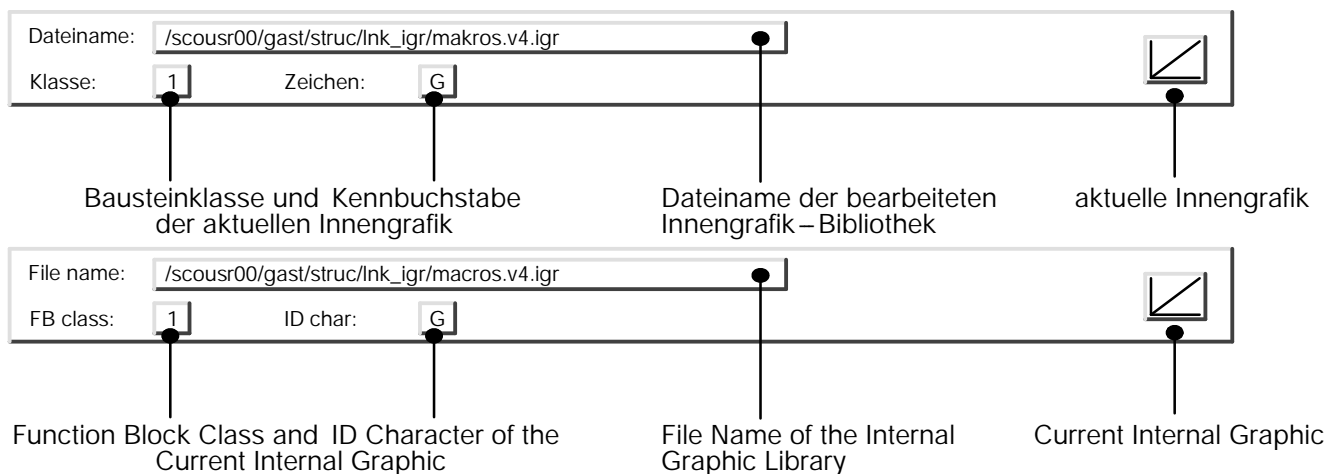


Diagram 1/28: The Information Strip

### 1.5.10 The Function Strip of the Internal Graphics Editor

The function strip of the editor (Diagram 1/29) contains the dialog elements for selecting the current internal graphic and for selecting the processing function.

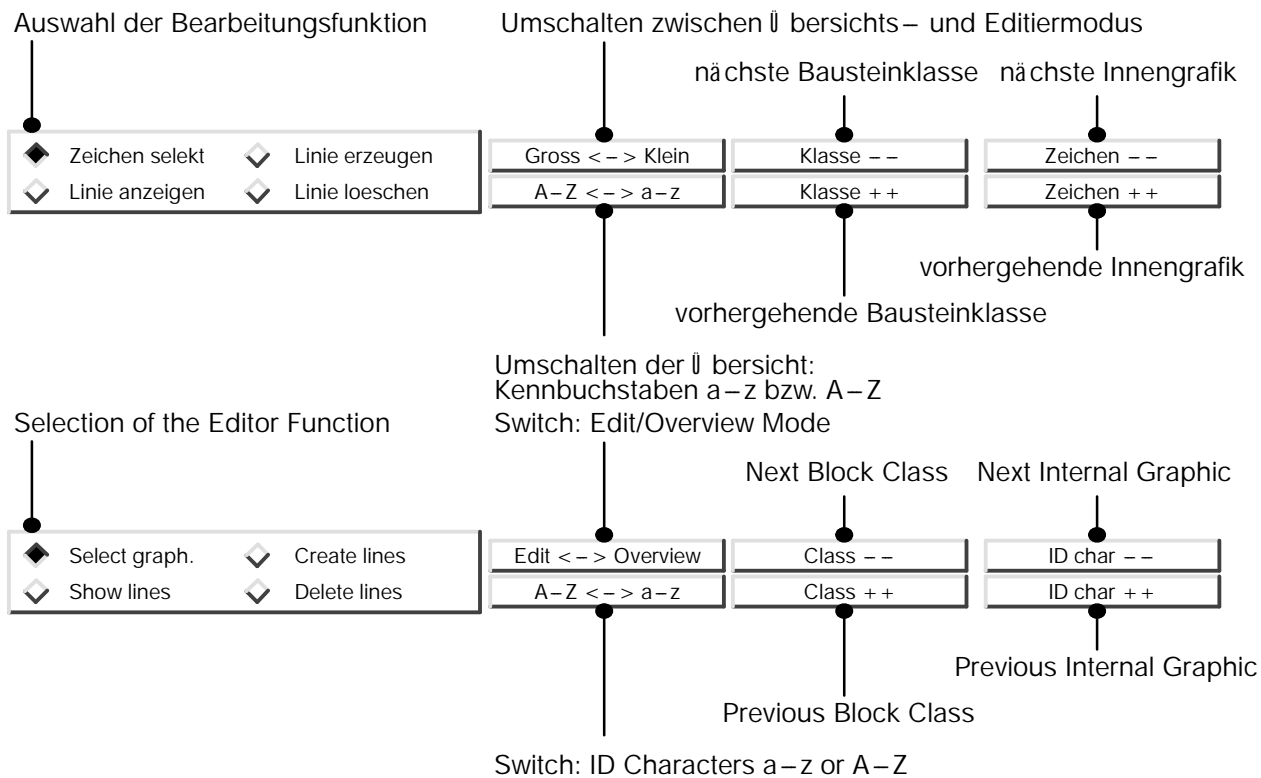


Diagram 1/29: The Function Strip

### 1.5.10.1 Selection Functions

The selection functions are operated via buttons in the right-hand section of the function strip.

#### Large <-> Small

This button switches between the **overview mode** (Diagram 1/30) and the **editing mode** (Diagram 1/31) of the editor.

The **overview mode** is utilized to select the internal graphic to be currently processed and the internal graphic to be deleted or copied using the 'Cut – Copy – Paste Mechanism'. The current internal graphic is framed within the working area.

The current internal graphic is processed in the **editing mode**.

#### A-Z <-> a-z

This button switches between the representation of the internal graphic with the identifying character a–z and those with the identifying character A–Z. This is necessary, since the overview mode only has enough space to display half of the internal graphics contained in the process library.

#### Class -- respectively Class ++

This button permits switching to the internal graphic with the same identifying characters of the previous respectively of the next block class.

#### Character -- respectively Character ++

This button switches to the internal graphic with the previous respectively with the next identifying character of the current block class.

### 1.5.10.2 Processing Functions

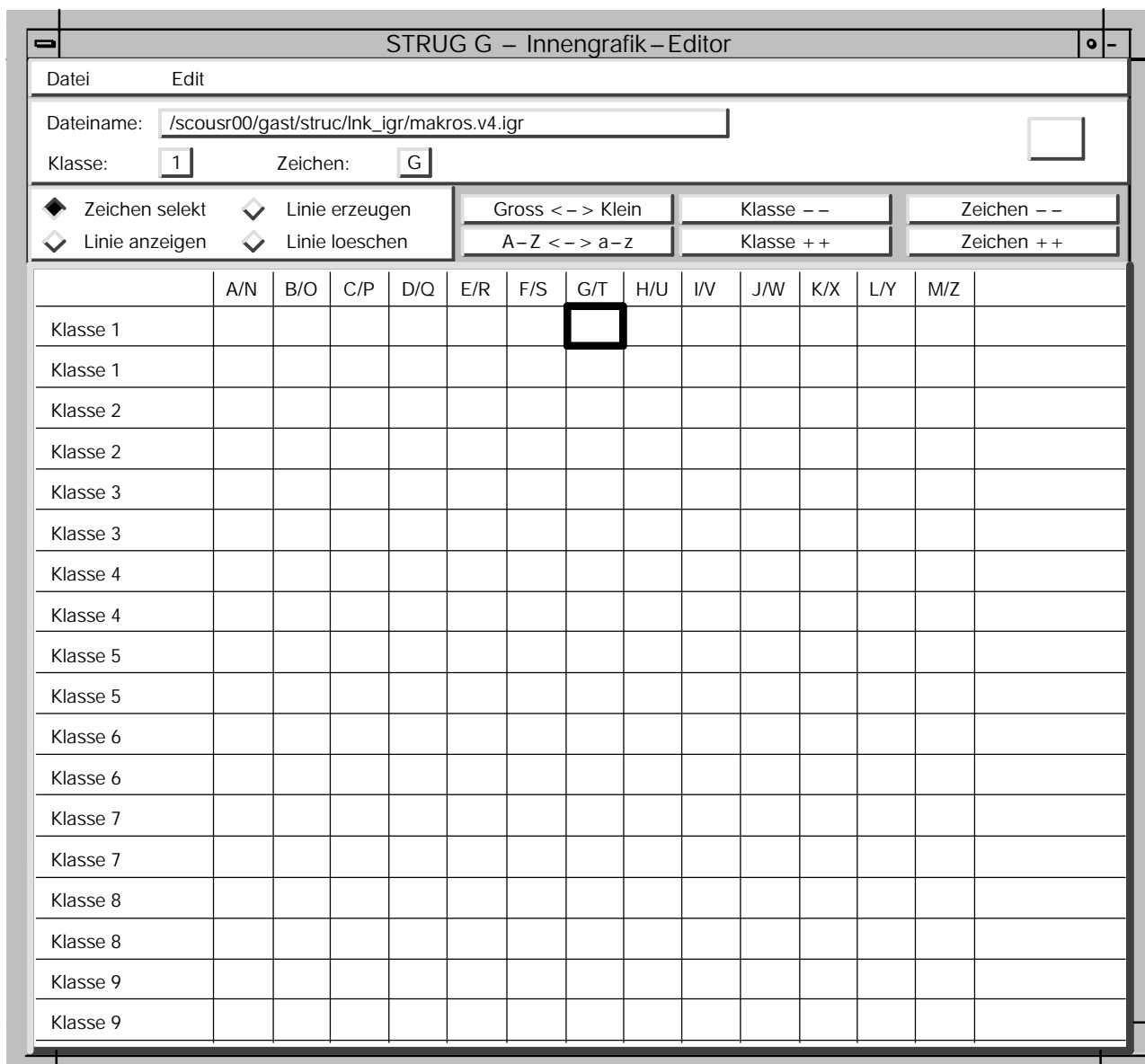
The processing functions are selected by radio buttons contained on the left-hand section of the function strip. The selected function is displayed by a filled in marking '◆' after selection of the corresponding button (by clicking with the left-hand mouse key).

#### Selecting Characters

The function 'Selecting Characters' is active in the editor overview mode.

The **current internal graphics** is selected in the overview mode by clicking on the desired internal graphics symbol in the working area with the left-hand mouse key.

Block class and identifying character of the current internal graphic are displayed in the information strip (see Diagram 1/29).



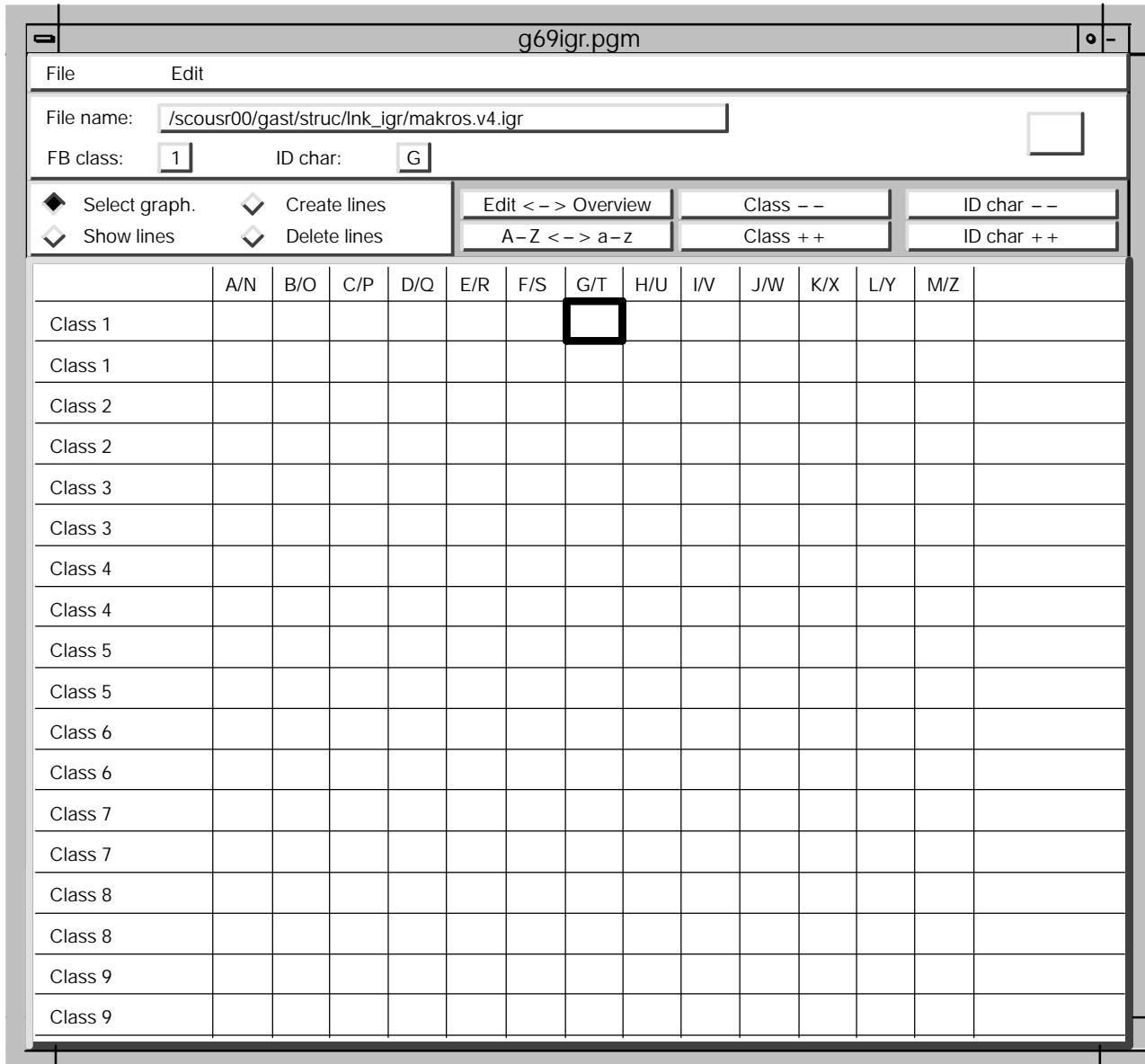
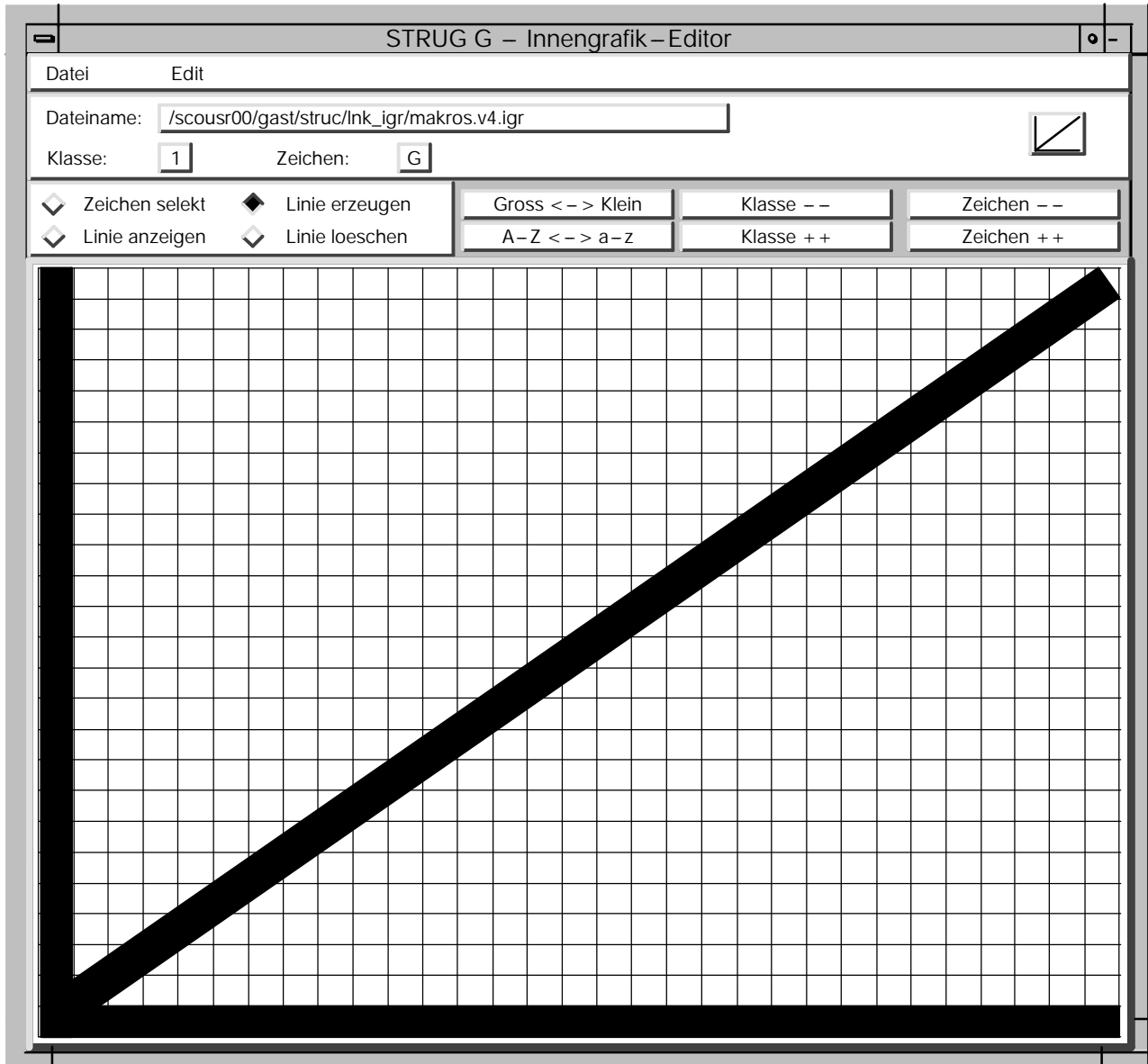


Diagram 1/30: The Window of the Internal Graphics Editor in the Overview Mode

## Display Line

The function 'Display Line' is activated after switching to the editing mode of the editor (Diagram 1/31).

The individual lines of the current internal graphic is drawn with a small white line if the left-hand mouse key is pressed and dragged across the working area in this function.



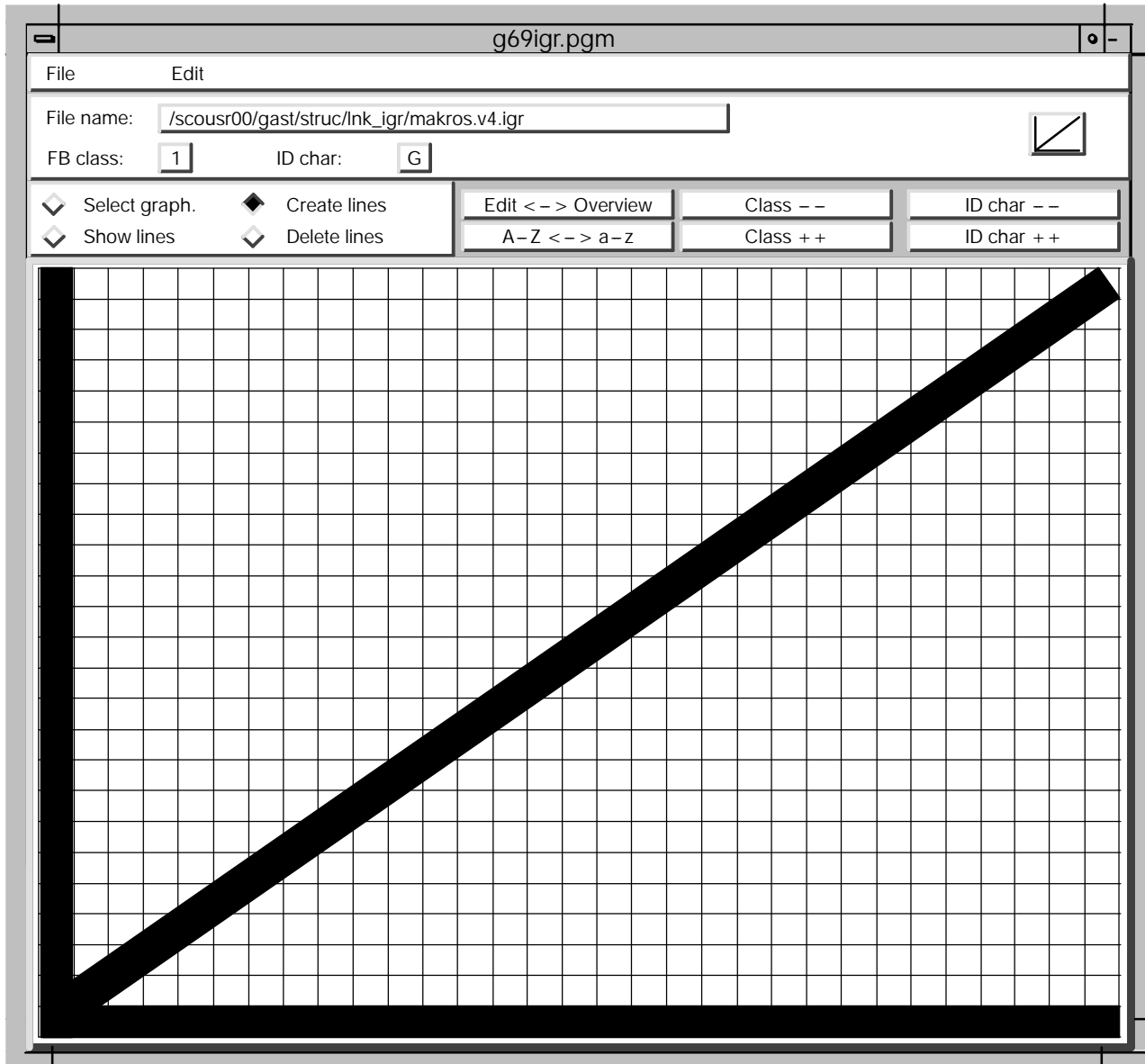


Diagram 1/31: The Window of the Internal Graphics Editor in Editing Mode

Lines are generated in the editing mode (Diagram 1/31) of the editor by selecting the processing function 'Generate Line'. This is implemented by sequentially selecting the start and end point of the line (by clicking with the left-hand mouse key) in the working area.

**Points** can be generated as lines of length 1 by selecting the start and end point to be the same. Filled in **areas** are drawn as lines lying next to one another. **Characters** are created from lines.

**Deleting** a line in the editing mode (Diagram 1/31) of the editor requires one of its two starting points to be selected by clicking with the left-hand mouse key. The line to be deleted will be marked by a small white line. Repeated clicking of the left-hand mouse key on the same point will acknowledge the delete intention and the line is removed.



## 1.5.11 Communication Island

### 1.5.11.1 Define Communication Island

Invoking this function is not necessary for simple plant configurations in which the island each only contain one rack.

To increase the communication activity, several SIMADYN racks can be interconnected with one another via the rack link to create a SIMADYN D network (review SIMADYN D user manual "Communication", chapter "25 Network"). This type of interconnection of racks and even an individual rack is designated as (SIMADYN D communication) island, where it can be accessed externally via at least one communication connection (i.e. SINEC H1 bus, serial DUST1 connection). STRUC G functions (i.e. "Function Packet Editor" - "IBS G") address these islands via a name, the island name.

The allocation of the rack (or a master program) to an island is implemented via the window "Define Communication Island" (see Diagram 7/28), that is opened by the STRUC G function of the same name. An island name and a reference channel number for each master program of the current project can be specified in this window. The reference channel name is utilized to shift the utilities channel numbers on the individual racks with respect to one another when communication is taking place via a peer to peer SIMADYN D network (review details in chapter 8: "Configuring Rules ..."). This value should be 0 in all other cases. The default island name is the master program name; the reference channel number is initialized with 0.

To modify the allocation of a particular master program, select the master program in the selection box using the scroll bar. The selected data is repeated in 3 panels below the selection box, whereby the panels containing the island name and the reference channel number can be modified. Actuate the "Modify" button to accept the data in the selection box after entering the desired allocation. The "OK" button saves the selection box data for the project and returns to the basic dialog. The "Cancel" button returns to the basic dialog without modifications being made.

### 1.5.11.2 Select Communication Island

The STRUC G functions "Function Packet Editor - IBS G" and "Download" require the STRUC G processor to be connected with the configured SIMADYN D equipment via one of its serial interfaces or via the SINEC H1 bus. At the equipment end, one or several CSH11 boards are required when the SINEC H1 link is utilized or one CS7/SS4 combination when a serial link is utilized. Any of these peripheral boards will permit the access to a so called island. An island is a single rack or a group of several racks, that are interlinked to a SIMADYN D network via a rack link (see SIMADYN D user manual "Communication", chapter "25 Network").

The name of this type of island is specified by the function "Define Communication Island" (see section 7.5.11.6).

This function then defines which path the STRUC G processor takes to access the individual island in the SIMADYN D equipment.

After selection, the window "Select Communication Island" is displayed. Diagram 2/29 shows the layout of the window after it has been opened for the first time in a configuring session. The corresponding PC also has an SINEC H1 interface board, with the name H1, installed in addition to the serial interfaces. The column island of the panel "Addressing" shows all the lines allocated with names; i.e. a maximum of 16 islands can be accessed by STRUC G.

The displayed example defines that the island "CQB" can be accessed via the COM2 interface of the PC (i.e. via a DUST1 connection) with 9600 baud.

The panel "Interface" defines which physical interface (and also implicitly which protocol) will be utilized. Only one interconnection can be selected in this panel.

If the interface COM1 or COM2 is selected, then the corresponding serial interfaces on the UNIX-PC are intended. These interfaces correspond internally to the UNIX device `/dev/tty1a` and `/dev/tty2a`. The DUST1 protocol will be automatically utilized. The transmission speed must then be selected in the panel "Baud Rate". The destination island to which the interconnection leads from the serial interface must be entered by the user into the STRUC G system in the panel "Addressing" by selecting the corresponding island in column "C". Only one island can be accessed via a serial interface and therefore only one button can be selected. If the processes of a communication island (or their utilities) can be accessed via the routing SIMADYN D network, then "Network" must also be selected for this communication island.

If the interface H1 is selected, then the connection runs via the SINEC CP1411 board installed in the PC under this name. The layers 1 to 4 of the SINEC H1 network are utilized for the protocol. It is therefore possible to create several transport connections to various SINEC H1 network nodes (i.e. also to SIMADYN D CSH11 boards). Which island drives what SINEC H1 network node is defined in the column "Ethernet Address" of the "Addressing" panel. The same Ethernet address must be entered here as has been defined at the connector MAA of the @CSH11 function block, that initializes the CSH11 board of the corresponding island. A selection in the "S" column must now take place in order to enable the node (and therefore also the connection to the island) for use by STRUC G.

This separate enable, seems to be unnecessary, at first sight. It permits, however only the really necessary interconnections to be activated. Therefore less interconnections to the individual utilities need to be created at the start of applications (i.e. IBS-G), which leads to shorter initialization times for the application.

If the processors of an island (or their utilities) are accessible via the routing SIMADYN D network, then "Network" must be additionally selected for this island.

The specifications in the panel "Baud Rate" and in the column "C" can be ignored.

Errors in the window entry content will have causal effects in the application programs (IBS G, down load) such that certain SIMADYN D processor utilities seem not to be accessible.

Application programs usually wait to see if the utility will actually connect. This can cause an accumulation of these wait times, during start-up, into considerable delays, until the missing utility is detected and processed by the application program.

Even when these types of errors do not lead to system crashes, care should be taken when setting parameters in the tables.

Example:

Diagram 7/30 shows a somewhat complex example for a window "Select Communication Island". As you can see, the SIMADYN D equipment consists of 6 islands MPA, INSELB, INSELC, MPD, INSELE and INSELF. The individual islands MPD, INSELE and INSELF can be accessed via the SINEC H1 bus under the specified Ethernet address. The CP1411 interface H1 is utilized for this purpose. The utilities in the processor in the island INSELF can be accessed via the routing SIMADYN D network, whereas the islands MPD or INSELE are interconnected via fixed links (direct or peer to peer SIMADYN D network).

Define Communication Islands

Master Program	Island	Reference Channel
<b>MPA</b>	<b>MPA</b>	<b>0</b>
<b>MPA020</b>	<b>INSELB</b>	<b>0</b>
<b>MPA050</b>	<b>INSELC</b>	<b>0</b>
<b>MPA090</b>	<b>INSELE</b>	<b>0</b>
<b>MPA120</b>	<b>INSELF</b>	<b>0</b>
<b>MPB030</b>	<b>INSELB</b>	<b>16</b>
<b>MPB060</b>	<b>INSELC</b>	<b>0</b>
<b>MPB100</b>	<b>INSELE</b>	<b>16</b>
<b>MPB130</b>	<b>INSELF</b>	<b>0</b>
<b>MPC040</b>	<b>INSELB</b>	<b>0</b>

**MPB030**      **INSELB**      **16**

OK      Modify      Cancel

Diagram 7/28 : Define Communication Islands

Select Communication Island

Interface	Addressing																																																																																																		
<table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 60%;"><b>H1</b></td><td style="width: 40%; text-align: right;"><b>f</b></td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> <tr><td><b>COM1</b></td><td style="text-align: right;">f</td></tr> <tr><td><b>COM2</b></td><td style="text-align: right;">f</td></tr> <tr><td> </td><td style="text-align: right;">D</td></tr> </table>	<b>H1</b>	<b>f</b>							<b>COM1</b>	f	<b>COM2</b>	f		D	<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Island Netw</th> <th style="width: 5%;">C</th> <th style="width: 5%;">H</th> <th style="width: 75%;">Ethernet Addr.</th> </tr> </thead> <tbody> <tr><td><b>CQB</b></td><td style="text-align: center;">j</td><td style="text-align: center;">Ø</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>KQH</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>INSELBj</b></td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>INSELCj</b></td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>INSELEj</b></td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>INSELFj</b></td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MPA</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MPB</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MPG</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MPH</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MQA</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MQC</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MQD</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MQF</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>MQG</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> <tr><td><b>ZQE</b></td><td style="text-align: center;">j</td><td style="text-align: center;">f</td><td style="text-align: center;">j</td><td style="text-align: right;">[ ]</td></tr> </tbody> </table>	Island Netw	C	H	Ethernet Addr.	<b>CQB</b>	j	Ø	j	[ ]	<b>KQH</b>	j	f	j	[ ]	<b>INSELBj</b>	f	j	j	[ ]	<b>INSELCj</b>	f	j	j	[ ]	<b>INSELEj</b>	f	j	j	[ ]	<b>INSELFj</b>	f	j	j	[ ]	<b>MPA</b>	j	f	j	[ ]	<b>MPB</b>	j	f	j	[ ]	<b>MPG</b>	j	f	j	[ ]	<b>MPH</b>	j	f	j	[ ]	<b>MQA</b>	j	f	j	[ ]	<b>MQC</b>	j	f	j	[ ]	<b>MQD</b>	j	f	j	[ ]	<b>MQF</b>	j	f	j	[ ]	<b>MQG</b>	j	f	j	[ ]	<b>ZQE</b>	j	f	j	[ ]
<b>H1</b>	<b>f</b>																																																																																																		
<b>COM1</b>	f																																																																																																		
<b>COM2</b>	f																																																																																																		
	D																																																																																																		
Island Netw	C	H	Ethernet Addr.																																																																																																
<b>CQB</b>	j	Ø	j	[ ]																																																																																															
<b>KQH</b>	j	f	j	[ ]																																																																																															
<b>INSELBj</b>	f	j	j	[ ]																																																																																															
<b>INSELCj</b>	f	j	j	[ ]																																																																																															
<b>INSELEj</b>	f	j	j	[ ]																																																																																															
<b>INSELFj</b>	f	j	j	[ ]																																																																																															
<b>MPA</b>	j	f	j	[ ]																																																																																															
<b>MPB</b>	j	f	j	[ ]																																																																																															
<b>MPG</b>	j	f	j	[ ]																																																																																															
<b>MPH</b>	j	f	j	[ ]																																																																																															
<b>MQA</b>	j	f	j	[ ]																																																																																															
<b>MQC</b>	j	f	j	[ ]																																																																																															
<b>MQD</b>	j	f	j	[ ]																																																																																															
<b>MQF</b>	j	f	j	[ ]																																																																																															
<b>MQG</b>	j	f	j	[ ]																																																																																															
<b>ZQE</b>	j	f	j	[ ]																																																																																															
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 60%;">Baud Rate</th> <th style="width: 40%;"></th> </tr> </thead> <tbody> <tr><td>300</td><td style="text-align: right;">f</td></tr> <tr><td>600</td><td style="text-align: right;">f</td></tr> <tr><td>1200</td><td style="text-align: right;">f</td></tr> <tr><td>2400</td><td style="text-align: right;">f</td></tr> <tr><td>4800</td><td style="text-align: right;">f</td></tr> <tr><td>9600</td><td style="text-align: right;">Ø</td></tr> <tr><td>19200</td><td style="text-align: right;">f</td></tr> </tbody> </table>	Baud Rate		300	f	600	f	1200	f	2400	f	4800	f	9600	Ø	19200	f																																																																																			
Baud Rate																																																																																																			
300	f																																																																																																		
600	f																																																																																																		
1200	f																																																																																																		
2400	f																																																																																																		
4800	f																																																																																																		
9600	Ø																																																																																																		
19200	f																																																																																																		

OK
Cancel

Diagram 7/29 : Select Communication Island

Select Communication Island																																																																		
<b>Interface</b> <b>H1</b> <input type="checkbox"/> <b>f</b> <input type="checkbox"/> <b>D</b>  <b>COM1</b> <input type="checkbox"/> <b>f</b> <b>COM2</b> <input type="checkbox"/> <b>f</b>		<b>Addressing</b> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Island Netw</th> <th style="text-align: center;">C</th> <th style="text-align: center;">H</th> <th style="text-align: left;">Ethernet Addr.</th> </tr> </thead> <tbody> <tr> <td><b>INSELB</b></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td><b>INSELC</b></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td><b>INSELE</b></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td><b>080006010002</b></td> </tr> <tr> <td><b>INSELF</b></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td><b>080006010003</b></td> </tr> <tr> <td><b>MPA</b></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td><b>MPD</b></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input checked="" type="checkbox"/></td> <td><b>080006010001</b></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="checkbox"/></td> <td style="text-align: center;"><input type="checkbox"/></td> <td><input type="text"/></td> </tr> </tbody> </table>			Island Netw	C	H	Ethernet Addr.	<b>INSELB</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<b>INSELC</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<b>INSELE</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010002</b>	<b>INSELF</b>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010003</b>	<b>MPA</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<b>MPD</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010001</b>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>		<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<b>Baud Rate</b>  300 <input type="checkbox"/> <b>f</b> 600 <input type="checkbox"/> <b>f</b> 1200 <input type="checkbox"/> <b>f</b> 2400 <input type="checkbox"/> <b>f</b> 4800 <input type="checkbox"/> <b>f</b> 9600 <input checked="" type="checkbox"/> <b>f</b> 19200 <input type="checkbox"/> <b>f</b>	
Island Netw	C	H	Ethernet Addr.																																																															
<b>INSELB</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
<b>INSELC</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
<b>INSELE</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010002</b>																																																															
<b>INSELF</b>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010003</b>																																																															
<b>MPA</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
<b>MPD</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>080006010001</b>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>																																																															
<input type="button" value="OK"/>		<input type="button" value="Cancel"/>																																																																

Diagram 7/30 : Select Communication Island