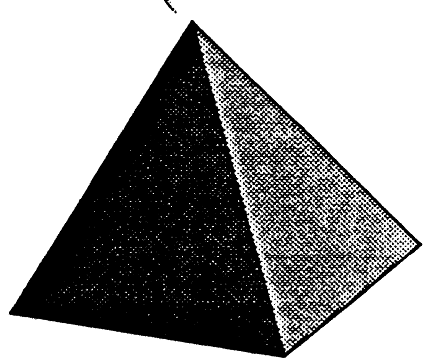
CALAY

CIS BOARD



C O N T E N T S

C I S - L A Y O U T

I N T R O D U C T I O N

ABOUT THE MANUAL

1. General

The concept of this manual is that it has been designed as a reference manual. This means that the manual concentrates on describing the menu functions; apart from describing the use and procedure of the functions, the manual also contains hints to interdependences with other functions or program parts.
The order of the described functions is oriented at the hierarchical organisation of the user menus and at a sensible order of using the functions.

In order to provide a direct access to the documentation, the manual has also been implemented online on your system.

2. Concept

The CIS documentation has been divided into three binders which correspond to the three units SCHEMATICS, BOARD and CAM.
The unit is indicated in the left corner of the header.

The division of the units into chapters is oriented at the functions of the main menu.
The appropriate main menu function appears in the center of the header.
Every chapter contains its own number which precedes the page number.

The division of the separate chapters is also oriented at the sequence of the functions within the menus.
The appropriate function is indicated in the right corner of the header.

3. Agreements

a. Keys

- characters in parenthesis are keyboard or mouse keys:
  <CR> = the RETURN key
  <LMB> = the left mouse button
  <MMB> = the middle mouse button
  <RMB> = the right mouse button

b. Abbreviations

CIS = CALAY Interactive System
SCS = Schematic
PCB = Board (=Layout)
CAM = Preparations for Production
RTI = Real Time Integration
I/O field = Input/Output field in the menus

c. quotation marks " ":

- menu entries to be made via the keyboard
- menu functions
- single characters, e.g. "@"

are put into quotation marks

d. inverted commas '':

- the figurative sense of a text
- a specific expression

are put into inverted commas

e. asterisk * :

indicates an "either/or" listing,
e.g.: entry via the * mouse:...
* keyboard:...

f. slash /:

The slash is used with functions for the separation of operation steps which are to be executed one after the other (e.g. "SYMBOL get/modify/TERMINAL get/mod. graphic/digitize/text")

OPERATION of CIS

1. General

CIS (CALAY Interactive System) is the program package of the PRISMA system for the interactive generation of the layout.

CIS is divided into:
- SCS = for the generation of the schematic
- PCB = for the generation of the layout
- CAM = for additional preparatory work for the manufacturing process

The operation of the program (= function selection/function activation/data entry) is screen oriented and is done via

- the optical mouse with the cursor on the screen
- the keyboard

2. Windows

a. defined windows

For the work with CIS the screen is divided into three windows:

- the function window (right side) -> function menus
- the status window (top) -> menu with i/o fields
- the graphic window (rest of the screen) -> graphic generation

Note:
A precondition for working within a certain window is that the cursor be placed within that window.

b. Temporary windows

Apart from these windows two other windows may be opened:
- the <u>pop-up window</u> with the pop-up menu which may often be used as an alternative for the function selection via the function window (see point 4)
- the <u>list window</u> which is automatically opened when lists are output (function "list")

Note:
As long as a pop-up or a list window is active, it is not possible to work outside this window!

3. Working with the mouse

A precondition for using the mouse keys is positioning the mouse cursor onto the field which is to be activated.

Cursor representation in the screen windows:
- status window: arrow
- function window: arrow
- graphic window: crosshair
- pop-up window: bar
- list window:
  > 'hand-on-key' for the "ready-for-entry" status
  > 'hour-glass' for the "searching for objects" status
  > 'double square' form, if the cursor is outside the opened window, i.e. if an entry into the window is not possible.

<u>Some examples for using the mouse:</u>

a. left mouse button <LMB>
  > confirmation of an entry (corresponds to <CR>)
  > in the function window: activation of the function onto which the cursor has been positioned
  > in the status window: activation of the i/o field onto which the cursor has been positioned, for data entry
  > in the graphic window: execution of the activated function (e.g. getting a graphic element at the cursor position or coordinate entry according to the cursor position for the graphic generation, etc.).

b. middle mouse button <MMB>
  > move window (<MMB> onto window edge/press/keep it/move)
  > in the status window: "toggling", i.e. conversion of the entry into its opposite with alternative input possibilities in i/o fields (e.g. "Y" <-> "N" or "marked with a cross" <-> "not marked with a cross").
  > in the graphic window:
    - digitize T-connection in the SCS part
    - entry of the rotation during the placement of objects (symbols, components, terminals)
    - digitizing and activation of vias in the PCB section "TRACKS".

c. right mouse button <RMB>
  > activation of the pop-up menu (if existing and if set "active")
  > in the graphic window, if the pop-up menu is switched off:
    - 'mirroring' of components, i.e. placement onto the opposite layer (=PCB part)
    - in the PCB part: delete tracks in the opposite sequence of their generation.
    - in the SCS part: with the subsequent generation of equal terminals it has the same functionality like the "continue" function

- for calling the generatable text types with the function "mod. graphic/TEXT/digitize" in all parts

Note:
The examples for using the mouse buttons are not complete and do not apply 'always and everywhere' (motto: try it)

4. Pop-Up Menu

Advantages of the pop-up menu:
- The pop-up menu can be activated at the current cursor position, i.e. the cursor does not have to be moved into the function window.
- in contrast to the menu window the pop-up window only offers implemented functionalities

Disadvantage:
- With the pop-up switched on the right mouse key is blocked for any other function (see point 3).

Operation Instructions:

a. the pop-up menu is called with <RMB>;
b. the functions of the pop-up menu are selected by moving the bar to the appropriate function and <RMB>
c. the availability of the pop-up menu can be switched on/off with the <R11> key;
d. if the availability of the pop-up is switched on, the right mouse button cannot be used for any other function.
e. as long as the pop-up is activated, working <u>outside</u> the pop-up window is not possible;
f. the availability of the pop-up menu is shown in the right upper corner of the status window ("POP-UP=ON" or "POP-UP=OFF");
g. the pop-up menu may be exited by positioning the cursor outside the pop-up window and pressing <RMB>.

5. Menu Operation

The menus in the status window serve for entering and/or outputting data for the currently activated function.
Some fields are mere output fields, i.e. the data are output only for information; they are generated/edited in another menu which belongs to a different function (e.g. in the SCS menu of the function "Symbol/mod. param." the function type names are output; but they are generated in the PCB part; in the SCS part they may neither be generated nor edited).

Entries are possible via the keyboard as well as via the mouse (e.g. in contrast to user-defined names, coordinates may be specified via the keyboard and via the mouse).

A precondition for making entries in i/o-fields is that the cursor be positioned in this field.

Cursor Positioning:

* with the mouse: place the mouse cursor to the beginning of the i/o-field and press <LMB>.
* with the cursor keys of the right keyboard block:

R8	=	cursor jumps back one input field
R14 (also TAB)	=	cursor jumps to the next input field
R10	=	cursor jumps back one character
R12	=	cursor jumps to the next character

6. Functionkeys for CIS (standard)

F1	= modify graphic (requires activated object)
F2	= switch from SCS to PCB and viceversa
F3	= modify parameter (requires activated object)
F4	= free
F5	= foreground for layers (only PCB)
F6	= background for layers (only PCB)
F7	= SHELL-ESCAPE ('open' new shell window)
F8	= graphic foreground/background
F9	= Subwindow foreground/background (SCS: "Miscellaneous/Subwindow")
L1	= reserved for SUN
L2	= SUBMENU on/off (only SCS)
L3	= modify symbols/components
L4	= modify signals
L5	= digitize (requires activated "mod. graphic" function)
L6	= delete (requires activated object)
L7	= move (requires activated object)
	= 'open' <-> 'close' window (precondition: mouse-cursor on edge of window)
L8	= insert (requires activated object)
L9	= get (requires activated object)
L10	= copy (requires activated object)
R1	= TTY-REPRINT (= refresh of the status window)
R2	= HELP (yet to be implemented)
R3	= TTY-HOME (yet to be implemented)
R4	= HARDCOPY (yet to be implemented)
R5	= TTY(=status)-Window foreground/background
R6	= MENU-Window foreground/background
R7	= fixing of a menu field for user-defined cursor positioning (move cursor to required field with arrow-buttons and press R7)

R8 = menu-IO-field (arrow up)
R9 = panning on/off (requires activated panning mode - see "modes" function)
R10 = menu-IO-field (arrow left)
R11 = POP-UP on/off
R12 = menu-IO-field (arrow right)
R13 = list
R14 = menu-IO-field (arrow down)
R15 = window-center (see "graphic setup/CENTER")

REALTIME INTEGRATION (RTI) BETWEEN SCHEMATIC AND PCB

REALTIME INTEGRATION makes it possible to handle both schematic and PCB-design changes simultaneously. Any modification done in the schematic or layout is realized automatically throughout the entire design, thus guaranteeing 100 % consistency between the schematic and PCB layout. There is no longer a need for forward and back annotation. Any PIN or GATE SWAP within the PCB-layout is simultaneously done within the schematic.

The basis for the REALTIME INTEGRATION is a fixed relationship between the schematic symbol and the physical shape. This fixed relation is realized within the COLUMBINE LIBRARY SERVER. For example, a component such as a 7400 copied out of the COLUMBINE LIBRARY SERVER includes 4 gates (NAND) and a physical shape (DIP14). By placing only one gate there will be 3 spare gates. These gates are not displayed on the actual page, they are reserved for later use. Using the function COPY, one of these reserved gates will now be used.

REALTIME INTEGRATION works also with signals. Any signal created or modified in the schematic will automatically be created or modified in the PCB layout and vice-versa.

Components which are not yet in the COLUMBINE LIBRARY SERVER may be defined locally. For that you have to create one (or more) schematic symbols and a physical shape, then build the relation between them.

Working with PRISMA now means, working on a schematic and PCB-layout at the same time. Some parameters of the PCB layout (track width, layers, critical signal etc.) may be defined during the work in the schematic.

If need be, the REALTIME INTEGRATION may, of course, be switched off (in the SCS part as well as in the PCB part in the "modes" menu).

STARTING THE PROGRAM AND READING IN AN .ADB FILE

The program is started by
* clicking the Icon SCHEMA/BOARD or
* entering "cis" plus <CR>

Loading this comprehensive program needs some time.
When the loading has been finished, a menu appears in the status window where the name of the file to be read in has to be specified - without extension.

If it is a standard technology file, you still have to enter "Y" in the field "read standard technology file?" (standard technology files contain 'mere' technology defaults for the SCS and the PCB part - see chapter "technology" in the manual).

After entering <CR> the file is read in. The same is achieved with the pop-up menu function "read file".

Precondition:

the specified file has to
* exist on the **adb**-subdirectory of the activated project, if it is a user-file.
* exist on the **projects/std/tec** subdirectory if "Y" has been specified in "read in standard technology file?".

Notes:

a. As long as you are in the first menu, the program can be exited by pressing <ESC><ESC> and answering the query "exit program?" with "Y";

b. with the pop-up function "list" you can list the contents of the appropriate .adb-subdirectory in the status window.
In order to list the existing standard technologies, you have to enter "Y" in "read standard technology file?" before activating the list function.
Exit the list function with <ESC><ESC>.
Paging down in the lists is done with <CR>.
The list window is closed by selecting a file from the list (with the keyboard or with <MMB>) and by then pressing <CR>; the file name is automatically entered into the menu.

c. When the file name has been entered, press <CR> or activate the pop-up function "read file" to read the file.
The read in is finished when the function menu appears in the right part of the screen.
The read in time is proportional to the size of the file - big files require some patience!

L A Y O U T

If after reading an .adb file (e.g. a standard technology file like "startup"), you activate the menu function "LAYOUT", you get into the main menu for the selection of the superior functions of the layout part in CIS.

Layout Main Menu

MODIFICATION

By activating this function a menu is called which contains the function selection for the actual work at the layout.
With the function of the modification menu you can generate components and their terminals or 'import' them from the library, generate the netlist, digitize tracks, generate copper and restricted areas, etc.

GRAPHIC SETUP

By activating this function it is possible to adjust and control the graphic output of the layout via various function selections as well as via the specification of parameter values in the status window menus.
Apart from selecting the layers and elements to be represented the user can assign the colours, the lines and other parameters for the exact determination of the representation (e.g. "filled" or "contour"). The user can also adjust different grids as a help for the interactive work during the placement and digitizing. The same functionality for the graphic adjustment is also available in the modification menu; it is called via the "GRAPHIC" function.

CAM

leads to the main menu of the CAM part

FINAL CHECK

This function serves for checking the tracks of a routed layout (= the tracks which have been routed by the router or interactively) for interrupts, shorts, etc.

SCHEMATIC

leads to the main menu of the schematic part

TOOLS

After this function has been called - and if the appropriate licence exists - the following programs which do not belong to CIS can be called from within CIS:
- AutoCAD
- PACIFIC NUMERIX (=thermal analysis)

The program is called via the pop-up.

GENERAL

In the selection menu of this main function the following functions may be called at present:
- "report" (for the generation of different lists and statistics from the current .adb file)
- "C-shell" (for calling a c-shell window)
- "plot" (for the generation of a 'hardcopy' of the current graphic window into a plot file)

MODES

After calling this function you may define a variety of parameter defaults for the work at the current layout by entering the specifications in the menus, e.g.:
- the measurement (mm or inch or 1/20")
- real-time-integration on/off
- pop-up on/off
- graphic mode (normal or panning mode)
- lower-case letters allowed for names yes/no
- different defaults for the digitizing (e.g. allowed gradients, auto-routing on/off, etc.)

and others

TECHNOLOGY

In the area below this function the whole technology for the current layout is defined, e.g.
- size and form of the PCB
- layer plan
- the grid for the routing, the component placement and the placement of vias
- width and form of the single via and track types
- diameter and restring size for the single drill types

etc.

With the technology function "reference component" 'global' terminal graphics are generated, which then serve as copies for the generation of component terminals in the modification part.

CHANGE FILE

This function serves for finishing the work at the current .adb-file (with storing the data) or for exiting the file (without storing the data) and for reading in a new .adb-file, without leaving the CIS program.

SAVE FILE

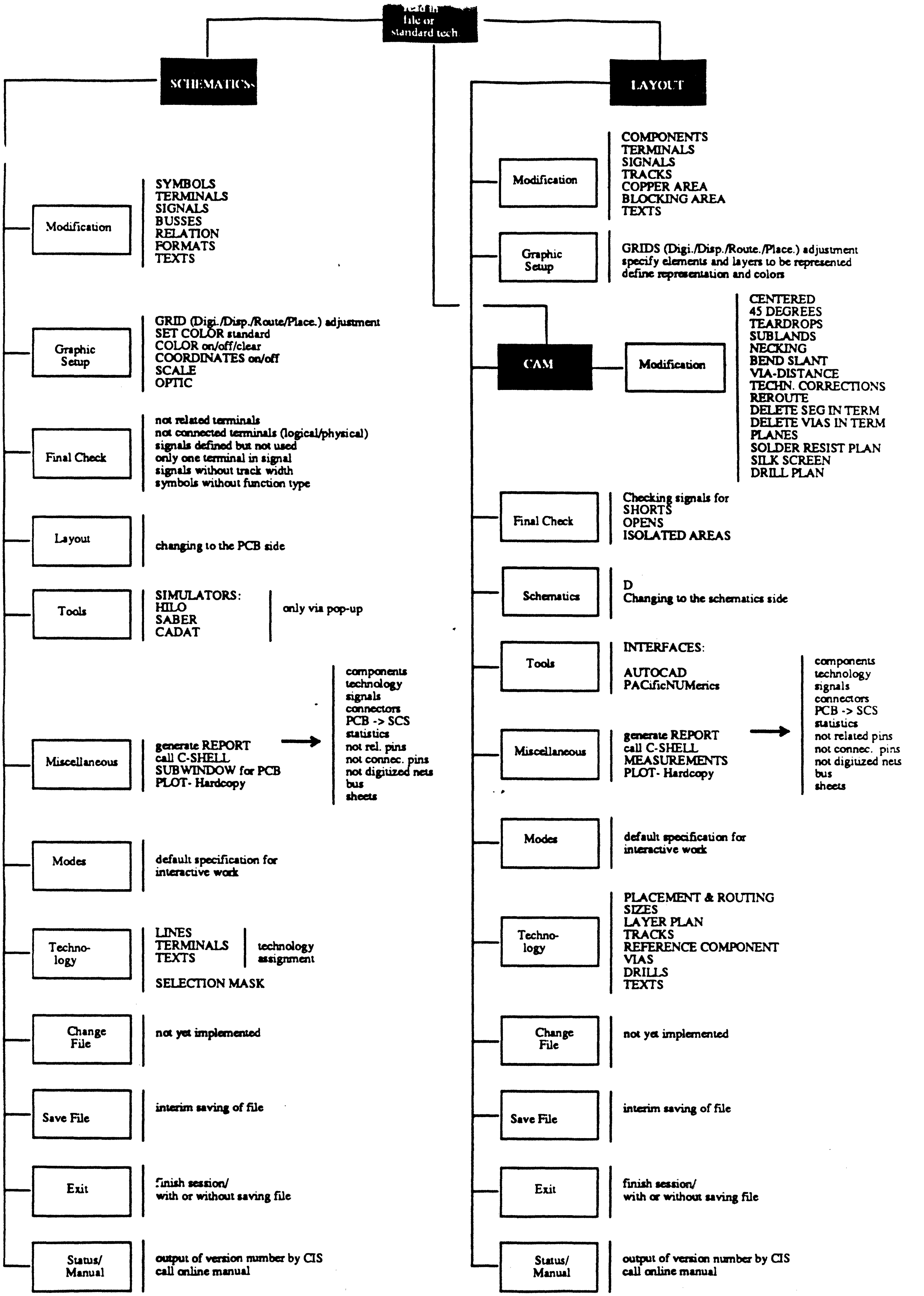
This function stores the current status of the .adb-file; then you can continue your work.

EXIT

With this function the user can exit the CIS program - with or without storing the data.

STATUS/MANUAL

After this function is called, the current version number of the CIS software is displayed in the status window.
It is also possible to call the online manual from there.

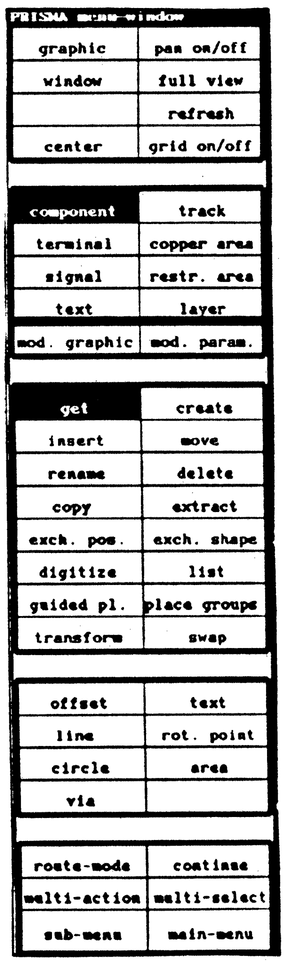

file or
standard tech
SCHEMATICS
LAYOUT
Modification
SYMBOLS
TERMINALS
SIGNALS
BUSSES
RELATION
FORMATS
TEXTS
Graphic Setup
GRID (Digi./Disp./Route/Place.) adjustment
SET COLOR standard
COLOR on/off/clear
COORDINATES on/off
SCALE
OPTIC
Final Check
not related terminals
not connected terminals (logical/physical)
signals defined but not used
only one terminal in signal
signals without track width
symbols without function type
Layout
changing to the PCB side
Tools
SIMULATORS:
HILO
SABER
CADAT
only via pop-up
Miscellaneous
generate REPORT
call C-SHELL
SUBWINDOW for PCB
PLOT- Hardcopy
components
technology
signals
connectors
PCB -> SCS
statistics
not rel. pins
not connec. pins
not digitized nets
bus
sheets
Modes
default specification for
interactive work
Techno-logy
LINES
TERMINALS
TEXTS
technology assignment
SELECTION MASK
Change File
not yet implemented
Save File
interim saving of file
Exit
finish session/
with or without saving file
Status/ Manual
output of version number by CIS
call online manual
Modification
COMPONENTS
TERMINALS
SIGNALS
TRACKS
COPPER AREA
BLOCKING AREA
TEXTS
Graphic Setup
GRIDS (Digi./Disp./Route./Place.) adjustment
specify elements and layers to be represented
define representation and colors
CAM
Modification
CENTERED
45 DEGREES
TEARDROPS
SUBLANDS
NECKING
BEND SLANT
VIA-DISTANCE
TECHN. CORRECTIONS
REROUTE
DELETE SEG IN TERM
DELETE VIAS IN TERM
PLANES
SOLDER RESIST PLAN
SILK SCREEN
DRILL PLAN
Final Check
Checking signals for
SHORTS
OPENS
ISOLATED AREAS
Schematics
D
Changing to the schematics side
Tools
INTERFACES:
AUTOCAD
PACificNUMerics
Miscellaneous
generate REPORT
call C-SHELL
MEASUREMENTS
PLOT- Hardcopy
components
technology
signals
connectors
PCB -> SCS
statistics
not related pins
not connec. pins
not digitized nets
bus
sheets
Modes
default specification for
interactive work
Techno-logy
PLACEMENT & ROUTING
SIZES
LAYER PLAN
TRACKS
REFERENCE COMPONENT
VIAS
DRILLS
TEXTS
Change File
not yet implemented
Save File
interim saving of file
Exit
finish session/
with or without saving file
Status/ Manual
output of version number by CIS
call online manual

M O D I F I C A T I O N

INTRODUCTION

In this menu you will do most of your interactive work for the generation of the layout.
Defaults and standard values for generating the layout are specified in the menus "Graphic setup" (grid, colours, etc.) "Technology" (layer assignment, track widths, etc.) and "Modes" (cursor representation, standard height for texts, etc.) which are selected from the layout main menu.

The functions are not organized in any hierarchy so that the user can select any executable function at any time. These functions can be combined in six units:



(1) Help for the graphic display (graphic parameters, grids, windows, etc).

(2) Objects (components, signals, copper areas, blocking areas, tracks and texts)

(3) Superior functions for modification of existing objects ("modify graphic" and "modify parameters")

(4) Actions that can be carried out with the objects

(5) Digitizing tools (for the generation or modification of graphic elements)

(6) Action modes (continue, process object groups, routing-like digitizing) and change to other menus

Note:

The option "Graphic" in the work menu corresponds to the function "Graphic Setup" in the main menu --> see chapter "graphic setup".

C O M P O N E N T S

General

After reading a standard technology file, the PC board appearing on the PCB-graphic window is 'empty'.

If, before working in the PCB part, you had worked in the SCS part and selected symbols from the library which belong to related function types, then the physical equivalents of these symbols, i.e. the physical shapes (called "component in the following description), will be displayed in the right upper corner of the PCB graphic window after changing into the PCB part. By getting these components "collectively" (with **"multi-select"/"multi-action"**) and after activating the function **"guided pl."** these are then automatically offered for placement. Instead of using the "guided" placement, it is also possible to use the automatic placement function (see **"place groups"**).

If, however, you begin your work in the PCB part or if you used in the SCS part only self-generated symbols or library symbols not related to a function type, then you have either to import the components from the library with the function **"insert"** (as physical shapes or complete function types), or to generate them first (function **"create"**).

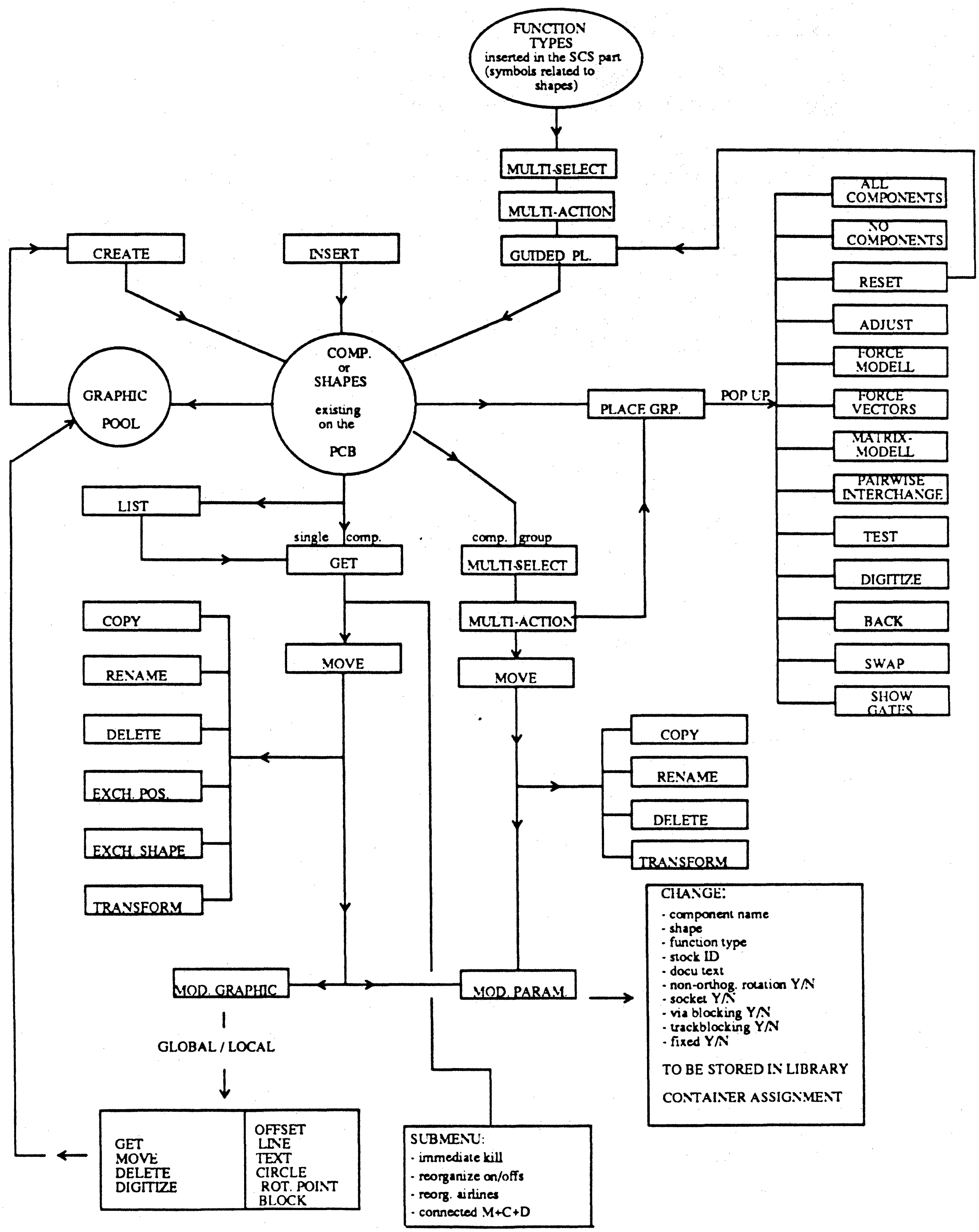
The following operations may be carried out with the single components in the graphic window:

- **"move"** = change the placement
- **"copy"**
- **"delete"**
- **"rename"** = change the component name
- **"exch.pos."** = exchange the positions of two components
- **"exch.shape"** = replace the shape of a component by another shape existing on the PC board
- **"list"** = list of the components on the PCB
- **"transform"** = generating a related symbol belonging to the shape by copying the shape graphic to the SCS side (onto page 0)

It is also possible to modify or generate graphic elements with **"mod. graphic"** and to modify or generate component parameters with **"mod. param."**.

With the function **"place groups"** it is possible to activate an automatic placement for all components or for "multi-select"ed groups of components.

I. COMPONENTS



CREATE COMPONENT

General

A component (or rather: a shape) consists of:

a. the graphic
b. the terminals

A 'virtual' component (in contrast to a 'pseudo component' used as a dummy or as a construction aid for drill plan macros) has been completely defined if the graphic and the terminals have been defined.

With the function "COMPONENT/create" **only the component graphic** (without terminals) is generated for the current layout, is named and placed.
The terminals belonging to the component are generated with the function "TERMINALS/create" after the generation and, if necessary, modification of the component frame.

A component graphic is always stored on the **SHAPE docu layer** belonging to the placement layer and consists of:

a. a component **frame**
b. a component **text field**
c. if necessary, **digitized graphic elements** (with function "mod. graphic")

The textfield with the component name is automatically generated by the system and placed into the component frame.

The component frame graphic is generated by the user; for this, there are two possibilities:

a. the graphic is transferred onto the new component from the graphic pool;

b. or the graphic is newly generated.

graphic pool:

this term describes an "invisible pool" which contains all graphics defined up to this point in this .adb-file stored in the background (i.e. not only those existing on the board, but also those deleted).

When using pool graphics for the component definition it is possible to assign two graphics to one component:
- one graphic for the component side ('**first graphic**')
- one graphic for the solder side, if the component is transferred to the solder side ('**second graphic**')

How to procede for generating a component graphic:

1. Activate function "Component/Create" with <LMB>.

2. If applicable, replace the default name with your desired placement name for the shape in the status window in "Component:".

3. a) For graphics to be transferred from the pool enter in the status window:
   - in "graphic"
     > in the first I/O field the name of the pool graphic the component is to get if placed on the component side
     > if applicable, in the second I/O field the name of the pool graphic the component is to get if transferred or copied from the component to the solder side
   - "N" in "create new graphic:"
   - if applicable, the placement coordinate for the lower left corner of the component frame in "coord:" (reference point is the production origin - see "Sizes", chapter "Technologies");

   b. For components to be newly generated enter in the status window:
   - enter in "graphic" in the first I/O field the name under which the newly generated component frame graphic is to be stored in the graphic pool
   - "Y" in "create new graphic:"

   Note:

   - For generating a component "COMP" has to be entered in "layer:" and "O" has to be entered in "rotation" in the status window!

   - Since at present there is no list function for the graphics contained in the "pool", it is important to remember or to note down the graphic names, for the component cannot be accessed without the name specification - this is necessary above all if the components are deleted after being generated.

4. place the component
   a) for graphics transferred from the pool (ref 3.a)
      * either via the mouse:
        - place the cursor onto the desired placement point and place the component with <LMB>
      * or via the keyboard:
        - press <CR> after specifying the placement point in "coord.:"
   b) for newly generated graphics (ref. 3b)
      * either via the mouse:

- place the cursor onto the starting point and press <LMB>
- open the frame with the mouse to the desired size and place it with <LMB>

* or via the keyboard
  - enter in the status window:
    the coordinates of the starting point in "coord.:"
    (reference point is the production origin - see "Sizes", chapter "Technology") and
    x and y size in "size"
  - enter <CR>;

Then the component frame - with the lower left corner on the placement coordinate - is placed; the text field with the component name is automatically placed into the component frame.

The system now defaults to "Terminal/create/continue". The generation of terminals is described in the chapter "Terminals".

Notes:

a. When creating components or terminals it is useful to adjust the grid and to switch it on together with the coordinates (see grid adjustment and function **"grid on/off"** and **"coos on/off"** of the graphic menu).

b. **Component names** may consist of up to 10 characters, but may not begin with $, since this is reserved by the system.

c. After calling the function "create", the system automatically enters a placement name for the component to be created; the names are generated as follows:
   > if prior to the function "create" a component has been modified, the root of this component name is supplemented with the first number available for this combination.
   > otherwise the component gets the name **"$"**, supplemented with the first number available for this combination.

d. If the component is to be **rotated or placed onto the solder side,** these operations may be carried out only after generating the complete component (with function "COMPONENT/move").

e. The **rotation point** of the component generated is automatically placed onto the lower left corner of the component frame; it is visible only during "modify graphic".
   When using the functions "Terminal" or "COMPONENT/mod. graphic", the rotation point is the coordinate origin for your local work.

f. New component frame graphics may at first only be generated as **rectangles**; they may then be modified with the function "mod.

graphic".

g. Every graphic newly generated is automatically also stored in the **graphic pool** and is available for the generation of further component graphics - even if the component defined during the graphic generation has been deleted.
Since at present there is no list function for the names of the pool graphics, it is important to **remember** or to note down the **names of the generated graphics,** above all if the component is deleted, for the pool graphic may not be accessed without any correct name specification.

h. If necessary an existing component graphic may be modified **locally** (i.e. only for the activated component) or **globally** (i.e. as pool graphic, for all components which have this graphic) (see function "**mod. graphic**").

i. **Function types** may be generated by **relating** user-defined physical shapes with symbols generated in the SCS part (see SCS part, chapter "relation").

j. The user-defined physical shapes as well as the function types generated by relating them may be exported into the **library** and from there be imported into any .adb-file (see functions "mod. parameter" in the SCS and PCB part as well as chapter "data transfer" of the COLUMBINE documentation).

INSERT COMPONENT

General

With this function, it is possible to import components existing in the library either as "pure" physical shapes or as function types (i.e. related to symbols) into the current .adb-file.

How to procede

1. Activate "Component/Insert" in the menu window with <LMB>.

2. If required, replace the name entered automatically by the desired placement name for the component in the status window in "component" (max: 10 characters).

3. Specify the name of the element to be read from the library in the status window in "search s." - you may use wildcards (e.g. "DIP*" for search criterion "B", or "*LS*" for search criterion "P" or "74*" for search criterion "L" or "*" for any search criterion).

4. Specify search criteria and sequence with the respective abbreviations in the entry fields of the respective search sequence numbers.
   The four search criteria refer to the four fields under which it is possible to store component information in the library. The search criteria may be used separately or in combinations (i.e., for a sequence of search operations to be carried out one after the other).

   L: logical name (e.g. "7400", "74F138", etc.)
   P: function type containing this logical function (e.g. "74LS00", 74HC138)
   S: stock ID
   B: physical shape (e.g. "DIP16")

   For a multiple specification of the search sequence, the number indicates the priority of the respective search criterion.
   Example:
   Search sequence (LPSB): 1=P 2=B 3= 4=L
   means that at first, the function types of the library are searched for the name specified under "search s."; if the search is not successful, then the physical shapes are searched, if this is unsuccessful, too, the symbols (related to shapes) are searched for the name specified.

5. "COMP" has to be entered in "layer".

6. If the component is to be rotated, specify the rotation in 1/10 degrees in "rotation" (e.g. "900" for a rotation of 90 degrees). When inserting components, only orthogonal rotations may be executed (i.e. 90, 180 or 270 degrees).

7. If the element to be entered from the library is a "pure" physical shape (without symbol assignment), and if the shape graphic is to be automatically copied as a symbol onto page 1 of the SCS side, specify "Y" in "trans.".

8. If you would like to change the current default entry for the following search procedure, then specify the respective abbreviation in "Def=".

9. <CR>
   a. If the complete name has been specified in "search s." and if the search has been successful, the respective physical "shape" is attached to the crosshair and may be placed with <LMB>.
   b. If, however, wildcards have been used, then all library elements concerned are listed in a list window. By clicking the appropriate name with <LMB>, it is entered into the appropriate field in the status window; the respective physical shape is attached to the crosshair and may be placed with <LMB>.

   Attention:
   If "P" has been specified as search criterion, and if a function type has been found which exists on several physical shapes, the shape is used which has been related first. In order to use another shape, specify the respective name in "opt. shape". (A list functionality is yet to be implemented).

Notes:

a. After calling the function "insert", the system automatically enters a placement name for the component; the names are generated as follows:
   > if prior to the function "insert" a component has been modified, the root of this component name is supplemented with the first number available for this combination (e.g. "IC").
   > otherwise the component gets the name "$", supplemented with the first number available for this combination.

b. Component names may exist of up to 10 characters, but may not begin with $, since this character is reserved by the system.

c. If the imported shape is related with symbols to a function type in the library, the respective symbols are automatically stored on page 0 in the SCS part (in the upper left corner);
with <MMB> it is possible to list them together with all other free gates, if the function "symbol/get" is activated.

GET COMPONENT

General

This function always precedes any operation with a placed but not activated component. To "get" means to activate the component for further action and is done as follows:

How to procede:

1. Activate "Component/get" in the menu window with <LMB>.

2. Get component
   a: either via the mouse:
      by "clicking" with <LMB>
   b: or via the keyboard:
      by entering the placement name in the status window and <CR>.

The activated component is highlighted (the standard colour for an activated component is blue).
A "copy" of the component is attached to the crosshair and moves as the crosshair moves.
The system defaults to "COMPONENT/move".

You may either move the component or invoke one of the following functions with the <LMB>:

- "rename"
- "copy"
- "delete"
- "list"

- "exch.pos."
- "exch.shape"

- "transform"

- "mod. graphic"
- "mod. param."

Notes:

a. Groups of components are activated with function "multi-select" (see description "multi-select").

b. If a group of symbols is activated in the schematic with the "multi" function, this selection can be transferred to the appropriate components in the PCB part:
   > activate "COMPONENT/get/multi-select"
   > enter **"SCS-select"** in the field "element" in the menu, <CR>.
   The appropriate components are surrounded by a selection frame and are highlighted.

MOVE COMPONENT(S)

General

With this function, a component of the layout may be moved and/or rotated and/or transferred to the other side (component side -> solder side and vice-versa).
A component group activated with "multi-select" may be moved to another position on the same layer.

How to procede:

1. Get component
   a. single component
      or
   b. component group:
      When activating a group of components with "multi-select", the components are highlighted; after activating "multi-action", the respective selection frame is attached to the crosshair; the system defaults to "move".

2. Rotate component if applicable (not applicable for component groups)
   enter rotation:
   * either via the keyboard
     the angle is specified in 1/10 degrees in the status window in "rotat." (e.g. "900" for an angle of 90 degrees).
   * via the mouse
     with <MMB> in steps of 90 degrees
   The rotation direction is counter clockwise.

3. Change layer if applicable (not applicable for component groups)
   * either via the keyboard
     by specification of the respective layer (SOLD or COMP).
   * or via the mouse
     with <RMB>.

4. Place component or component group
   * via the keyboard
     by specification of the new placement coordinate in "coord:" and <CR> (for group operations the placement coordinate is always the centre of the selection frame)
   * or via the mouse
     by placing the crosshair at the respective position and <LMB>.

Notes:

a. If a **movement** from the component side to the **solder side** has been specified ("SOLD" entered in "layer:"), the system expects that a **second graphic** has been assigned for the representation of the component frame (see "define components") and of the terminals (see "define terminals") on the solder side; if this is not the case, the system outputs the message "second component/pin graphic does not exist".
By pressing <CR> or <LMB> a second time it is possible to make the system generate a second graphic by mirroring the first graphic onto the opposite layer.
After the second graphic has been generated, it is also available in the graphic pool (2nd graphic name = 1st graphic name + apostrophe)!
If not the mirror of the first graphic but a graphic existing in the pool is to be assigned as second graphic, the movement has to be interrupted. Then the graphic has to be assigned and then the graphics have to be moved to the opposite outer layer. The assignment of the second graphic is done with the function **"mod. graphic"** (for the component frame see "modification/component/mod. graphic", for the terminals see "technology-/reference component/mod. graphic).

b. A component may be rotated only in steps of 90 degrees (values in "rotation": 900, 1800 or 2700), as long as the default "N" in **"n-orth. rot."** has not been changed to "Y".

   In order to execute non-orthogonal rotations the following has to be done before the movement:
   - get the component
   - activate "mod. parameter"
   - enter "Y" in "n-orth. rot." in the status window, then <CR>

   After leaving the "mod. parameter" function and after getting the component anew the movement may now be executed with non-orthogonal values (e.g. "rotation:450" for a rotation of 45°).

c. When placing components it is useful to adjust the representation grid and to switch it on together with the coordinates (see grid adjustment and function **"grid on/off"** and **"coos on/off"** of the graphic menu).

d. "Collisions" with other components entail an error message in the status window; by pressing <CR> or <LMB> twice the placement may be enforced despite the collision.

e. With "COMPONENT/move" all elements are moved which belong to the component or to its terminals (incl. **additionally digitized graphic elements and texts**).
   Existing **signal connections** are moved as airlines.

f. When moving component groups, also the **tracks** of the signals connected to the terminals of the selected components are moved, if they are inside the selection frame.

RENAME COMPONENT(S)

General

Existing component names (generated by the system or by the user) may be changed
- for single components
- for component groups (activated with "multi-select") and
- for components activated continuously with the "continue" function being activated.

How to procede:

1. Get component
   or
   get component group with "multi-select" and activate component group with "multi-action"
   or
   activate "continue", get the first component.

2. Activate "rename" (click the function in the menu window).

3. Specify the new name in the status window under "component:"
   or
   specify the name root of the new group name (e.g. "IC ")

4. <CR>.

5. If the "continue" function is being used, click the next component to be renamed by pressing <LMB>, etc.

Notes:

a. **Component names** may consist of a maximum of 10 characters, but may not begin with "$", since this character is reserved for the system.

b. When renaming **component groups** the root specified is applied to all selected components; the numbers remain unchanged.

c. When components are renamed with the "continue" function, the frist activated component gets the name specified in the status window under "component:"; for the next activated component, this name is incremented by one, etc.
   (e.g. before renaming: A1, X3, C19
   new name: BT
   after the renaming: BT, BT1, BT2.)

DELETE COMPONENT(S)

General

With this function, it is possible to delete single components or groups of selected components.

How to procede:

1. Get component
   or
   get component group with "multi-select" and activate the group with "multi-action".
2. Activate "delete" (click the function in the menu window or enter "D" in "fct." in the status window).
3. If "immediate kill" has been activated in the submenu, the deletion is carried out at once, otherwise the system expects a confirmation through "Y" after "really delete" and <CR>.

Attention:

- If a component is deleted, then all elements belonging to this component are also deleted (terminals, texts, tracks, etc.). The respective symbols and terminals disappear in the SCS part.
- The graphic of a deleted component still exists in the graphic pool and may be transferred to other components.
  Since at present there is no list function for the graphics contained in the pool, it is important to remember or to note down the names of graphics of deleted components, for an access to the pool graphics is possible only with an exact name specification!

Notes:

a. If before or after the function "get" and before activating the function "delete", **"continue"** is activated, it is possible to delete several components simply by clicking the components with <LMB> after the deletion of the first component. A precondition is that **"immediate kill"** has been activated in the **submenu**; otherwise every deletion has to be confirmed with "Y" in "really delete?" and <CR>.

COPY COMPONENT(S)

General

With this function it is possible to copy a selected component to any free space in the layout. The copy may be rotated or copied onto the opposite outer layer.
A group of components selected with "multi-select" may be copied to any free space on the same layer of the layout.

How to procede:

1. Get
   a. a single component
      or
   b. a component group:
      When activating a component group, the selected components are highlighted. After "multi-action" the respective selection frame is attached to the crosshair.

2. Activate the function "copy"
   if more than one copies are required, also activate "continue" (only applies to single components)

3. Only for copying single components:
   if applicable, replace the component name automatically output by the system (= name of the original incremented by one) with the desired name in the status window under "component:".

4. If applicable, rotate the component (not applicable to component groups!)
   Specify rotation:
   * via the keyboard
     the angle specification is done in steps of 1/10 degrees in the status window in "rotation" (e.g. "900" for an angle of 90 degrees)
   * via the mouse
     with <MMB> in steps of 90 degrees
   The rotation direction is counter clockwise.

5. Change layer if applicable (not applicable to component groups)
   * via the keyboard
     by specifying the respective layer (SOLD or COMP).
   * via the mouse
     with <RMB>.

6. Place component or group
   * either via the keyboard:
     by specifying a new placement coordinate in "coord:" and <CR>;
     (for group operations the placement coordinate is always the centre of the selection frame)
     or
   * via the mouse:
     by moving the crosshair to the desired position and <LMB>.

7. If "Continue" has been activated, as many copies as required may be placed one after the other.

Notes:

a. If a **movement** from the component side to the **solder side** has been specified ("SOLD" entered in "layer:"), the system expects that a **second graphic** has been assigned for the representation of the component (see "define components") and of the terminals (see "define terminals") on the solder side; if this is not the case, the system outputs the message "second component/terminal graphic does not exist".
   By pressing <CR> or <LMB> a second time it is possible to make the system generate a second graphic by mirroring the first graphic onto the opposite layer.
   After the second graphic has been generated, it is also available in the graphic pool (2nd graphic name = 1st graphic name + apostrophe)!
   If not the mirror of the first graphic but a graphic existing in the pool is to be assigned as second graphic, the copying has to be interrupted. Then the graphic has to be assigned and the graphics are copied onto the opposite layer.
   The assignment of the second graphic is done with the function **"mod. graphic"** (for the component frame see "Modification/component/mod. graphic", for the terminals see "technology-/reference component/mod. graphic").

b. When copying the component copy may be rotated only in steps of 90 degrees (i.e. possible values in "rotation:" are 900, 1800, 2700).
   After the copy has been placed, it may also be rotated non-orhtogonally, if the parameter **"n-orth. rot."** has been set to "Y" before copying the original; if not, the parameter "n-orth. rot." of the copy has to be set to "Y" (compare "MOVE COMPONENT", note b.)

c. **"Collisions"** with other elements entail an error message in the status window; by pressing <CR> or <LMB> twice, the placement can be enforced despite the collision.

d. When copying a component, **all elements belonging to that component are copied** as well (component graphic, terminals, incl. additionally digitized graphic elements and texts); Existing **signal connections** are only copied (as airlines) if "M+C+D connected" has been activated in the submenu.

e. When copying **groups of components** also the tracks of the signals connected to the terminals of the selected components are copied, if they are inside the selection frame and if "M+C+D connected" has been activated in the submenu.

f. When copying groups of components the copies automatically get the name root "$".

g. If a component copied in the layout exists also on the SCS side (if the physical shape belongs to a function type), the respective **symbols** are also copied. (They can be found on page 0 and may be listed with <MMB>.

EXCHANGE POSITION

General

With this function it is possible to exchange the positions of two components in the layout. The rotation point of the first component is placed onto the placement coordinate of the second component and vice-versa.
Existing connections are moved as airlines.
The exchange procedure has no effect in the SCS part.
If the exchange procedure would entail a collision, it is not carried out and the warning "component collision" is displayed in the status window.

How to procede:

1. get first component
2. activate "exchange position"
3. select exchange partner
   * via the keaboard
     specify the respective component name in the status window with <CR>
   * or with the mouse
     click with <LMB>

Notes:

a. All elements belonging to the component are transferred together with the component to the new place (component graphic, terminals, texts, additionally digitized graphic elements, etc.)

EXCHANGE SHAPE

General

This function allows for replacing the physical shape of an existing component by the copy of another shape existing on the PCB.
Connections of the "old" shape are transferred to the "new" shape. The exchange has no effects in the SCS part.

Attention:
This action may only be carried out with equal shapes (i.e. same pin number and identical gates) (e.g. replacement of DIP shapes by equal SMD shapes and vice versa).

How to procede:

1. Get "Master component"

2. Activate "exchange shape"

3. select the shape to be replaced:
   a. via the keyboard
      specify the respective component name in the status window and <CR>;
      or
   b. via the mouse
      click on the respective component with <LMB>

LIST COMPONENT(S)

General

With this function it is possible to list the names of the components **existing on the PC board.**

How to procede

By clicking the respective name with <LMB> the component is activated for further action.

For the "list" function it is necessary that one of the functions

a. "get", "create", or "guided placement"
or
b. "move", "rename", "delete", "copy", "exchange position", "exchange shape" ("get" has to be activated prior to all of these functions)

has already been activated.

In the first case (a) the respective component is activated after clicking it in the list; the system defaults to "move".

In the second case (b) the respective component is activated instead of the current one after clicking it in the list; the function activated prior to the list function is now activated for the new component.

The list is displayed in a shell-window on the graphic window. The window cannot be moved.
By clicking the window-fields "previous" or "continue" you may page up and page down.

You may continue your work in the layout only if a component name has been chosen from the list or if you leave the window with "quit".

TRANSFORM COMPONENT

General

Newly generated components for which symbols do not yet exist in the SCS part, i.e., for which a relation does not yet exist, may be "copied" into the SCS part as symbols. Symbol and physical shape have the same graphic and their terminals are automatically related.

In this way it is possible to generate **signal connections directly in the PCB part** (without any previous connection definition in the SCS part). The physical shape graphic transferred to the symbol may be modified in the SCS part with the function "symbol/mod. graphic".

Attention:
The "transform" function may only be applied to components which, apart from the component graphic, also have terminals!

How to procede:

1. get the component to be transferred
   * via the keyboard
     by specifying the component name in the status window
   * or via the mouse
     by clicking the component with <LMB>

2. activate the function "transform"

GUIDED PLACEMENT

General

With this function, the system automatically offers components for placement.
The system marks the optimum placement field for the component offered (with green on the component side, red on the solder side) (disregarding any collisions!).
The only criterion is the less cost-intensive realisation of the nets; therefore, the components with the most connections are offered first; components without connections are offered last.

Preconditions:

1. the system has to offer the components in the right upper corner of the graphic window (outside the PCB); this applies to the following cases:
   a. if in the SCS part components are imported from the library with the function "insert" and if these are related with physical shapes to function types;
      in the PCB part the physical shapes belonging to these symbols appear in the right upper corner of the graphic window, with names generated automatically by the system.
   b. if components existing on the board are selected for a new placement with the "reset" function of the pop-up menu of "place groups".

2. the components have to be activated as a group (either through "multi-select"/"multi-action" or through "all components" in the pop-up menu of "place groups").

How to procede:

Case 1a
(shapes generated by importing related symbols, not yet placed)

1. Activate "multi-select" and select the components not yet placed
   * via the keyboard
     through specification in the status window and <CR>
   or
   * via the mouse
     through opening a selection frame with <MMB>

2. Activate "multi-action" and then "guided placement"; the component with the most connections is now attached to the crosshair;

3. Specify the rotation if applicable
   Specification of the rotation (only orthogonal rotation possible)
   * via the keyboard
     through the specification of the angle in the status window in 1/10 degrees (e.g. "900" for an angle of 90 degrees).
   * via the mouse
     with <MMB> in steps of 90 degrees

4. Change layer if applicable (default is "COMP")
   by entering the layer specification ("SOLD");
   Note:
   - <RMB> does not result in a change of the layer but in resetting the component into the right upper corner of the graphic window.
   - changing the layer is possible if the component already has a second graphic or if a system-internal generation is enforced by pressing <CR> or <LMB> twice.

5. Place the component
   a. by entering a new placement coordinate in "coord:" and <CR>
   or
   b. by moving the crosshair onto the desired position and <LMB>.

6. After placing the first component, the next one will automatically be attached to the crosshair and may be placed (if convenient place it into the placement frame proposed by the system or into the near of it).

etc.

Case 1b.
(Placed components set into the right upper corner with the command "reset")

1. Independent of whether the component had been activated with "multi-select" or with the subfunction "all components" of the pop-up menu, the selection remains active after the "reset" command. This applies only if immediately after the "reset" the function "guided placement" is activated (may be done only by clicking the function "component" in the function menu!).

2. After clicking the function "component", the function "guided placement" is automatically activated; the function "multi-action" remains active.
   The first component (with most connections) is attached to the crosshair for placement.

The further proceding corresponds to case 1, point 3, ff.

PLACE COMPONENT GROUPS

General

This function provides for automatic placement tools.

How to procede

The actual function with "place groups" may only be carried out via the popup menu.
The menu is activated with <RMB> after activating "place groups".
Place the cursor onto the function to be carried out and press <RMB>.

Function Description

The following functions are available in the popup menu:

- all components
- no component
- reset
- adjust
- force model
- force vectors
- matrix model
- pairwise interchange
- digital
- previous
- swap
- show gates
- main menu

Note: A differentiated selection of components may be done with the function "multi-select"/"multi-action" prior to calling the popup menu. Then call the function "place groups", but do not select the function "all components" of the popup, since this would deactivate the first selection (all components would be selected).

all components

This function selects all existing components for further action - provided they have not been fixed. (as regards fixed components, see "component/mod. Param.", parameter "fixed".)

no component

This is an undo function for a selection which has been carried out with "all components" or with "multi-select".

reset

places the components which have been selected with "multi-select"/"multi-action" or with "all components" outside the PCB into the right upper corner.

adjust

adjusts the selected components.

Necessary parameters (to be entered into the status window):

adjustment axis:

x the components are placed on an axis which is parallel to the x-axis. The x-coordinates of the components are not changed, the y-coordinates are entered into the field "coord.".
y see above, but adjustment to an axis parallel to the y-axis.

coord.

x and/or y coordinate of the adjustment axis, which may be specified either in the status window or with the <LMB>.
If both values have been entered (when using the mouse, this is done automatically), the system takes the value convenient for the specification in "adjustment axis".

mode:

0 the components are placed onto the adjustment axis
1 the components are placed so that the lower (or: the left) edge is placed on the adjustment axis.
2 like "1" but the upper (right) edge is placed onto the adjustment axis.

check:

N there is no check for collisions, collisions may occur.
J a check for collisions is carried out.
The adjustment is carried out only for that part of the selected components for which it does not entail any collision.

Force Model

All components not placed (which have been activated with "multi-select"/"multi-action" or with "all components" and which have been reset) are placed onto the PCB. The placement is carried out according to the so-called force model, which tries to minimize the track lengths to components which have already been placed.
A precondition for a sensible use of this function is that there are some fixed components on the PCB (e.g. connectors, CPU, Memory).
Component overlappings may still occur.

Force Vectors

Based on the sum of the airlines the system calculates a "force vector" for every component; it shows the main orientation and the average length of the airlines.
The direction and length of the vectors give the user helpful hints for an optimization of the placement (for possibly short connections). With a refresh the vectors disappear.

Matrix Model

Components selected with "multi-select"/"multi-action" or with "all components" may be pre-placed with the help of a matrix which the system generates internally.

The system divides the available space into an even number of rectangles of the same size, one for each component.

As a default, the area available is the board outline (a rectangle).
The area may be limited by the user.

* via the keyboard
  by specifying the starting and end coordinate in the status window under "area"

or

* via the mouse
  by specifying a frame with <MMB>.

The number of rows (=first number) and the number of columns (=second number) may be specified under "matrix:__/__".

The user may also specify the rotation of components with the respective specifications in the status window under "rotat.:"

"V" = the matrix as well as all components concerned (independent of their current rotation position) are placed vertically, i.e. with a rotation of 90 degrees counter clockwise

"H" = the matrix as well as all components concerned (independent of their current rotation position) are placed horizontally, i.e. with rotation 0.

"-" = all components concerned remain in their current rotation position; the matrix is placed horizontally.

" " = means "no entry", i.e. if an entry exists, it has to be deleted!
in this case all components concerned are placed with the rotation specified, independent of their current rotation position (0=zero degrees, 900=90 degrees, 1800=180 degrees, 2700=270 degrees); the matrix is placed according to the rotation;

The colour of the matrix corresponds to the colour of the respective layer (green = component side, red = solder side).

A refresh deletes the matrix lines from the screen.

Note:
- In order to avoid component overlappings, the components should be of nearly the same size, should be placed on the same layer and should have the same rotation.
- If components exist on both sides of the PCB, the "matrix model" function should be carried out for each side separately (with function "multi-select").

Pairw. Interch.

The system changes the placement of the components with the aim to minimize the overall length of the connections.
A "clock" is displayed as long as this function is active.

Test

This function has been reserved for internal requirements.

Digital

Before beginning your digitizing work, you should activate this function. Existing components the placement of which has not yet been registered by the system get the "placed" mark.
The airlines which are not visible in the "unplaced" status, are made visible.

Previous

By "previous" or by clicking the menu field "component", the function "Place groups" may be left.

Swap (identical to "Swap" in the work menu)

This function can only be used together with "Place groups".
With this function, gates of components or pins within a gate can be exchanged.

How to procede:

1. Click Swap
2. Click a pin
3. Either
   a. click a pin of the same gate; the two pins change the signals

   or

   b. click a pin of another gate; the two gates change their positions within the component.

A precondition is that the pins to be exchanged are of the same swap class (see parameter "swap class" in the SCS part, function "TERMINAL/mod. param.") or that the gates are of the same type.

Show gates

This function shows (in the colour of the respective layer) which pins belong to a gate.

Main menu

leads back to the main menu.

SUBMENU of MODIFY COMPONENTS

immediate kill

If this function is activated, a component is deleted immediately with the "delete" function after it has been activated. If this function is not activated, "Y" has to be entered in the status window under "really delete?" (with <MMB> in the entry field, the user can toggle), then <LMB> or <CR>.

Reorg on/off

After activating this function, the airlines are optimized after every move of a component.

Reorg airlines

The system optimizes all existing airlines.

connected M+C+D

When copying (C) a component or component group, existing signal connections are copied (as airlines), only if this function has been activated in the submenu.

MODIFY COMPONENT GRAPHIC

General

The function "mod. graphic" switches to the work area of digitizing the graphic and provides for various help functions for changing the component graphic (incl. the texts).

Apart from **lines, circles and texts** it is also possible to generate **glue points** for components (with the "block" function). In addition, the **rotation point** of the components may be moved.

By entering "Y" or "N" in "glob." in the status window, the user may determine this function as

a. **"global"**, i.e. with effect on the pool graphic, that means with effect on all components with this graphic,
   or

b. **"local"**, i.e. with effect only on the activated component.

Note: 'Globally' modified graphics keep their original name, whereas the 'locally' modified ones get a new, system-generated name: original name supplemented by the tilde character and an appropriately incremented number (e.g. DIP14~1).

As a rule, graphic elements which belong to components may **only be stored on docu layers.**
The component graphics are usually on the docu layer with the default name SHAPE-1 (for the component side) and/or SHAPE-0 (for the solder side).

How to procede:

The function may be called in two ways:

a. click "component"
   activate "mod. graphic"
   get the component with <LMB> or through entering the component name in the status window and <CR>; the system defaults to "offset"

b. get the component
   activate "mod. graphic"; the system defaults to "offset".

The activated component is highlighted (blue).

The rotation point is marked with a cross - for further action with the function "mod. graphic", the **rotation point is the temporary coordinate origin**; if it is moved with function "rotation point", the coordinate origin is also moved.

You may now either

1. **replace the existing graphic(s)**
   by other graphic(s) from the graphic pool

   > specify the name of the 'new' graphic in the status window under "graph." and press <CR>.
   (the 1st field is for the name of the first graphic, the 2nd field is for the name of the second graphic).

2. **modify existing graphic elements**
   (lines, circle segments, texts, glue points)

a. **get** (function "get")

   > activate function "get",
   > if applicable, specify the layer number of the element in "layer:"
   > click the element with <LMB>

b. **move** (function "move");

   > get the element
   > texts
   - may be rotated (with <MMB> or via a specification in the status window) and/or
   - a frame may be assigned to the text (by marking the field "frame:" with a cross in the status window and <CR>) and/or
   - the text may be right-adjustified (by marking " " with a cross and pressing <CR>)

c. **delete** (function "delete")

or

3. **Digitize new graphic elements**
   (after activating function "digitize")

a. set the starting point of the element to be digitized (line or circle segment) with the "**offset**" function or interrupt the current digitizing process;

   > set the offset with <LMB> or with an entry in "coord." in the status window, the offset always refers to the rotation point;

b. use the "**line**" function to digitize lines of any gradient;

   > the starting point is the point set with "offset" or the rotation point, if an offset has not been set.

   > the line width is specified with a respective track type (defined in the track technology) in "trk type" in the status window;

c. use the "**circle**" function to digitize circle segments:

> first set the starting point for the arc of the circle with "offset" and activate "circle"
or
activate "circle" and set the starting point by pressing <LMB>;
> beginning from the starting point a circle arc of 90 degrees is opened by moving the mouse:
- a movement to the left results in a positive angle,
- a movement to the right results in a negative angle;
- the crosshair centre is the centre of the circle; the radius is the distance between starting point and crosshair centre;
- if you move the crosshair so that its x-line is on the starting point of the arc (movement only into x-direction), then the starting point and the end of the arc are exactly on the lines of the crosshair;
> with <RMB> the sector is "folded" (the same angle with the opposite sign)
by pressing <RMB> again the arc is folded back into its original position and increased by 90 degrees;
> the same may be achieved with the appropriate entries in the status window:
enter the starting point coordinate with reference to the rotation point in the first line of the status window;
enter the coordinates of the arc centre with reference to the rotation point in "0/0"
- enter the length of the radius is "radius"
- enter the arc angle in "angle:" in 1/10 degrees;
> the segment width is specified through entering the respective track type (defined in the track technology) in the status window under "trk-type".
> the circle arc attached to the cursor is placed by entering
- <MMB>, if the specifications have been made with the mouse
- <CR>, if the specifications have been made via the keyboard;

d. the "**TEXT**" function serves for generating additional text fields:

> activate "digitize/"TEXT"
> specify in the status window:
- the text type
via the keyboard (by entering the respective text field number) or
by clicking it with <RMB> (by repeating <RMB> all available text types may be called, if the cursor is within the graphic window);
- if text type 0 (free text) has been specified, enter the desired text under "text:.."
note:
if the text is to be represented in small letters, the parameter "small letters" of the "modes" menu has to be activated; after the text generation this parameter should

be switched off again!
- if the text field is to get a frame, mark "frame" with a cross
- if the text is not to be "centered" in the frame, but "left-justified", mark " " with a cross
- if it is an "object-specific" text type (e.g. PCB name), and if the text field width is to change according to the changed object name (e.g. after renaming the component), delete the cross in "fixed:".
- specify a rotation, if required, in steps of 1/10 degrees (e.g. "900" for 90 degrees) in "rotation"
- if required, specify the placement and size in the first "coord." line and in "size"

> <CR>

or

> if placement and size have not been specified with coordinates in the status window, set the starting point of the text field with <LMB>;
depending on whether "size autom." has been activated in the "modes" menu, the text field size
-> is generated by the system (based on the size specified in "standard height" of the "modes" menu
-> is generated by the user through moving the mouse until the text field has the desired size;
place the field with <LMB>

e. the **block** function serves for generating **glue points**

> activate "digitize/block"
the field "layer" of the status window contains the name of the GLUE layer defined in the layer plan for the current component placement layer; (an appropriate layer definition has to exist in the layer plan).

> enter the glue point diameter in "glue point" of the status window

> place the glue point
* via the keyboard: enter the placement coordinate (referred to the rot. point) in "coord." in the first input line and press <CR>, or
* via the mouse: set the mouse onto the placement point and press <LMB>.

The glue point is represented by a via from the via technology which has been asigned to the approriate glue layer. That via type is used the diameter of which comes closest to the value entered in "glue point".

or

4. **move** the **rotation point** (and thus the temporary coordinate origin) (activate function "ROTATION POINT" and enter the new placement coordinate via the mouse or via the keyboard).

   Attention:
   If the funciton "mod. graphic" is used for moving the rotation point, the option "glob.:" in the status window has to be set to "N", otherwise the rotation point will be moved for all components with this graphic!

Notes:

a. If you would like to leave the function "mod. graphic" without storing the changes, enter <CTRL-C>; then answer the query "ignore changes? (Y/N)" with "Y" and press <CR>.

b. If you would like to leave the function "mod. graphic" <u>and</u> to store the changes, enter <ESC><ESC>, or activate a superior function.

MODIFY COMPONENT PARAMETERS

General

This function serves for assigning attributes to components or for changing existing attributes (parameters) of activated single components and component groups.

How to procede:

This function may be called in two ways:

a. Click function "component";
   activate "mod. parameters";
   get the component (with <LMB> or through entering the component name in the status window and <CR>) or the component group (with "multi-select/multi-action")
b. get the component or the component group
   activate "mod. parameters"

The activated component(s) is/are highlighted (default=blue).

The parameter assignment is done in the status window.

By clicking another component with <LMB> during the work with "mod. graphic" the parameters of the newly activated component are entered into the status window and are ready for modification.

Part of the parameters are only relevant to the local work at the board, i.e. when transferring this component to the library, these parameters are not stored.
Some fields are mere "output" fields; the data they contain cannot be altered; they serve only for information.

Parameters Relevant for the Library

The following parameters are "relevant to the library", because the specifications in these fields may be transferred to the library together with the component.

graphic : name of the component graphic; existing graphic names may be changed here - the change is of a 'global' nature.

techn. : Specification of the manufacturing technology

The user may specify a mark for the component technology relevant for him (e.g. TTL, CMOS, etc.). If the component is transferred to the library, this is also transferred.
(at present not implemented)

shape : Name of the shape

With components imported from the library the respective physical component name is output in this field, the name can be altered.
With user-defined components a name has to be specified in this field, if the component is to be transferred to the library.

function type : name of the function type

With components imported from the library, this field contains the name of the function type; this name can be changed.
With user-defined components, the respective name of the function type has to be specified here, if the component is to be transferred to the library.

stock ID :

With components imported from the library which have an ID number, this number is output in this field. The number can be altered.
With user-defined components the ID number has to be specified in this field, if the component is to be transferred to the library with its ID number.

d-Text : user-defined text

the user may assign text to the activated component. When enabling this component for the transfer to the library, this text is also stored.

via-restr. : mark for a via-restriction

"Y" means: vias going through the placement layers may not be placed under that component when digitizing or routing, i.e. drilled-through vias may not be placed and "buried vias" may not be placed onto the layer pair containing the placement layer.

trk-restr. : mark for track restriction

"Y" means: on the placement layer of the component it is not allowed to place tracks under the component during both digitizing and routing.

type : container assignment

In addition to the component parameters described above any other parameters (mechanical / electrical parameters, supplier, etc) and component reference names may be combined to so-called containers and may then be assigned to components.
A 'container' is a kind of 'parameters' box'. The definition of the different container parameters (incl. the reference names) and the generation of containers is done in the file **cisini.cmd**

on the **lib/sys** directory.
In CIS a container may be assigned to any activated component or symbol by activating the function **"mod. parameters"** and entering the desired container number in **"type"**.

extended list : output of container parameters

After specifying "Y" in "extended list" the active parameters of the container specified in "type" appear in the part below. For each of these parameters a field appears where definite specifications can be made (e.g. specifications regarding the field, value, price, etc.).

Parameters for Enabling a Component for the Transfer to the Library

lib. : the following parameters in this line are all library-specific

mod.: imported library component has been modified

Applies only to components imported from the library: information whether ("Y") or not ("N") the component has been graphically modified. If a component imported from the library is changed with "mod. graphic", "Y" is automatically entered.

trf.: enabline the component for transfer to the library

With "Y" the current component is enabled for the transfer to the library - from the PCB side. The following fields determine which parameters are to be transferred to the library.

-rel: enabling the pin relation shape <--> symbol for transfer to the library

By entering "Y" or "N" the user determines whether or not the current relation is to be transferred into the library, if the parameter trf. has been set to "Y".

If a user-defined and related component is to be transferred to the library, "Y" has to be entered in "-rel", in "-typ" and in "-shp", and a function type name has to be entered in "fct. type" and a shape name has to be entered in "shape:".
In the SCS part, the symbol has to be enabled for the transfer to the library, too (see SCS part, function "SYMBOL/mod. param.").

Note:
if "-rel" is set to "Y", but the component has not been related, this parameter is automatically set to "N" after a <CR>; the message "log. symbol(s) not completely defined" is output.

-typ: function type assignment

If "trf." has been set to "Y" and if the function type assignment is to be transferred, "Y" has to be entered in "typ".
Note:
If "-typ" has been set to "Y", but the component has no function type, the parameter is automatically set to "N" after a <CR>; the message "need function type for library" is output.

-shp: enabling the shape for the transfer to the library

If "trf." has been set to "Y" and if the shape is to be transferred into the library, "Y" has to be entered in "-shp". If the shape is not to be transferred to the library, "N" has to be entered.
If it is a user-defined shape, this parameter has to be set to "Y", a shape name has to be entered in "shape:" and a function type has to be specified.
Note:
If the component does not have any shape name, this parameter is automatically set to "N" after a <CR>; the message "need shape for library" is output.

Note:

The "Y/N" specification for the parameters "-rel", "-typ" and "-shp" has been implemented, in order to provide for the possibility to selectively transfer changes of library imports.
Example:
The DIP shape of a component imported from the library has been replaced by a PLCC shape; the relations and the function type are still the same.
In order to transfer this component with its current constellation, just set the parameters "trf." and "-shp" to "Y"; the parameters "-rel" and "-typ" may be set to "N".

Parameters Not Relevant for the Library (=local parameters)

component : current placement name of the component

By specifying a new name and <CR> it is still possible to change the original placement name.
For the future it is planned that by entering the name of another component existing on the PCB, this component is activated for altering its attributes.

non-orth. rot. mark for non-orthogonal component rotation

"Y" means: the component may also be rotated non-orthogonally, i.e. in any angle.
"N" (= default entry) means: the component may only be rotated in steps of 90 degrees (i.e. 90, 180, 270 degrees allowed).

fixed : mark for "fixed" components

"Y" means: the placement of the component may not be changed; when trying to move a "fixed" component, the message "component is fixed" is output. "N" means the placement can be altered as required.

socket : mark for components to be set on a socket

"Y" means that the component is to be placed onto a socket.

Mere "Output" Fields

Some of the fields serve only for outputting information, they cannot be altered.

"coord."	-	contains the placement coordinate of the component
"layer"	-	contains the placement layer
"rotat."	-	contains the current rotation of the component
"modified"	-	contains whether or not a component read from the library has been changed in the current file
"height"	-	at present this field still shows an unchangable number without any relevance for the future it is planned that this field is to contain the component height
"placed"	-	placement mark "N" means: the component exists, but has not yet been placed on the PCB; this applies to components which are still in the right upper corner of the PCB "waiting" to be placed (see the "reset" function).

CONTINUE

When this function is activated together with one of the functions "copy", "delete" or "rename", the function may be repeated until another function is activated or until the "continue" function is switched off by clicking it again.

MULTI-SELECT/MULTI-ACTION

General

With this combination a group of components is selected and activated for the "collective" execution of a function.

The following functions may be executed for "multi-selected" components:
- "move"
- "copy"
- "delete"
- "rename"
- "place groups"
- "transform"
- "mod. param."

(see the detailed function descriptions)

How to procede

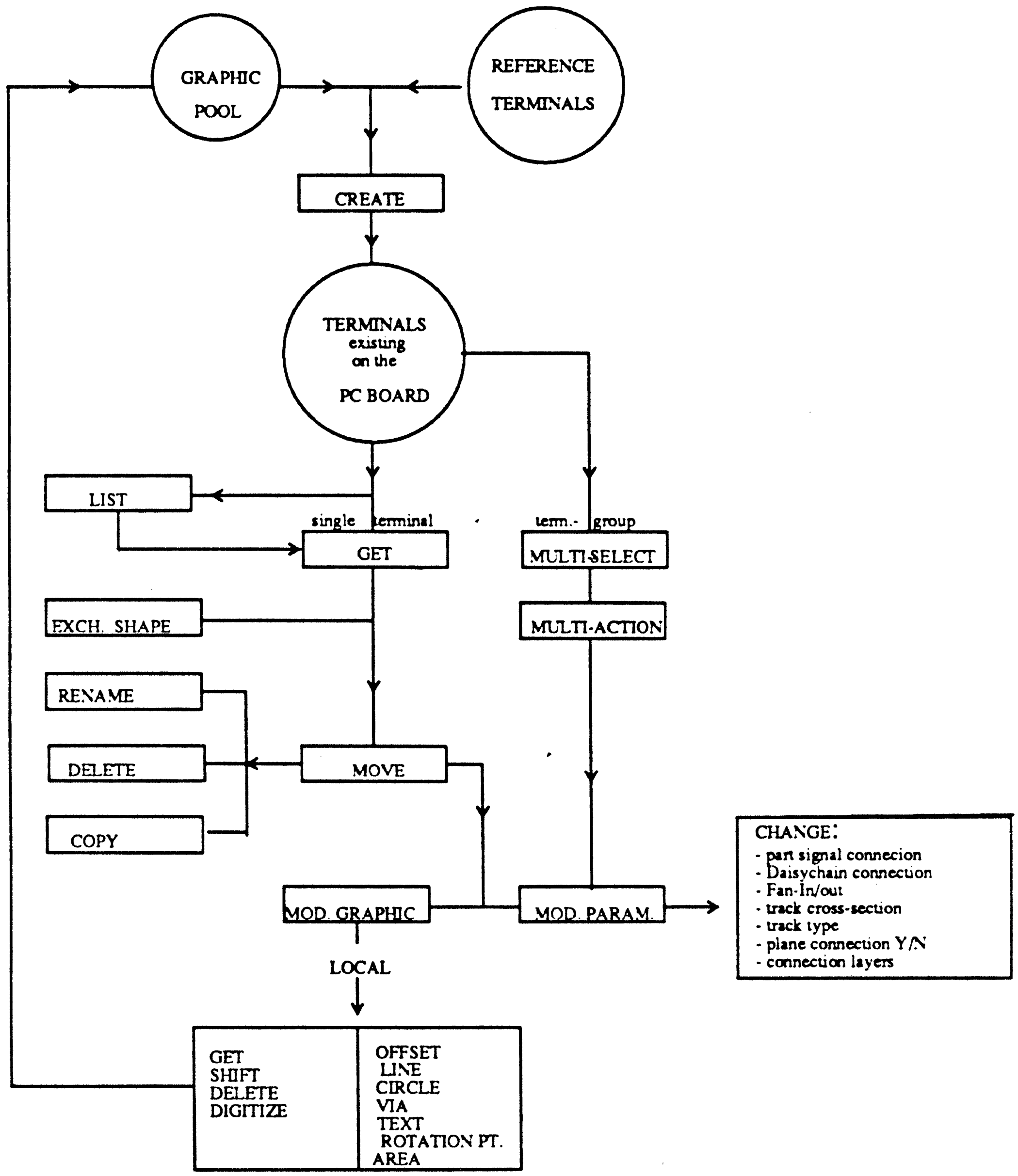
When calling the MULTI-SELECT-function, a form appears in the status window in which it is possible to make a detailed selection based on a differentiated selection mask (which corresponds to the component parameters definable in function "component/mod. parameters"), based on an appropriate name specification (wild-cards may be used) and based on the respective coordinate specifications (via mouse or keyboard).
As regards the selection criteria see the respective parameters of function "COMPONENT/mod. parameters".

The components may as well be selected by clicking them with <LMB> or by specifying their placement coordinates in the status window in "coord." and <CR> - under consideration of the selection criteria activated by the user.

It is also possible to select components with a selection frame (to be opened and closed with <MMB> or to be specified with the starting and end coordinate in the status window under "area").
With "I" or "O" you determine whether the components inside the selection frame or outside the selection frame are to be selected.
With "Y" or "N" under "cutting" it is determined whether or not components which are not completely covered by the selection frame are also to be selected.

When selecting a component group with "multi-select" the selected components are highlighted. After activating "multi-action" the selection frame is attached to the crosshair; the system defaults to "move".

TERMINALS


GRAPHIC POOL
REFERENCE TERMINALS
CREATE
TERMINALS existing on the PC BOARD
LIST
single terminal
GET
term.- group
MULTI-SELECT
EXCH. SHAPE
MULTI-ACTION
RENAME
DELETE
MOVE
COPY
CHANGE:
- part signal connecion
- Daisychain connection
- Fan-In/out
- track cross-section
- track type
- plane connection Y/N
- connection layers
MOD. GRAPHIC
MOD. PARAM.
LOCAL
GET
SHIFT
DELETE
DIGITIZE
OFFSET
LINE
CIRCLE
VIA
TEXT
ROTATION PT.
AREA

CREATE TERMINALS

General

A terminal is always part of a component and consists of the following elements:

a: the **placement point = terminal rotation point** (referred to the component rotation point)

b: the **terminal graphic**, i.e. the pad representation for the signal and docu layers and additional graphic elements (e.g. digitized texts)

During the procedure of generating a component the function "TERMINAL/create" is automatically called after generating the component graphic.

A. Terminal Generation with Reference Terminal Graphic

The terminal definition in the PCB work menu usually consists of specifying the placement coordinate and of assigning a reference terminal graphic for its representation.

The reference terminal graphics have to exist in the graphic pool before generating the board terminals.
Reference terminals are user-defined standard terminals which - together with their graphic - are stored in the "background" and are assigned to the so-called **reference component** (see chapter "reference component" of the PCB technology description).
Every reference terminal graphic is available in the **graphic pool**, is of a **global** nature and may be used as a copy for the graphic of newly created terminals.
("Graphic pool" is an invisible pool, where the graphics of all components and terminals defined in the current .adb-file are stored in the background - not only the graphics of the components and terminals existing on the board, but also of those already deleted.

How to procede for A

1. After generating the component frame (see description "COMPONENT/create") get the component and call the function "TERMINAL/create";

2. specify in the status window
   a. if applicable, replace the automatic pin number "1" under "pin" by a terminal name of your choice.
   b. if applicable, enter the placement coordinate for the first terminal in "coord.:" (if the placement is to be done via the mouse, this specification is not necessary)
   c. if a terminal row is to be generated, enter
      -> in "dist." the distance between the terminals in the appropriate measurement (e.g. in mm:"2.54/0.00" for a row to be generated in the positive x-direction; "0.00/-2.54" for a row to be generated in the negative y-direction)
      -> the number of the terminals to be generated in "number"
      -> in "series" e.g. "1" for a continuous numbering of the pins (1, 2, 3, etc) or "2" for a series like 1, 3, 5, etc.
   d. enter the name of the reference terminal graphic in the first input field under "graphic:" (for placement on the component side);
   e. if it is an SMD terminal which is to get a two-sided graphic
      > enter "Y" in "2 sided"
      > specify the name of the reference terminal graphic in the second input field under "graph.:" for the placement on the solder side;

      Notes:
      - if you specified "Y" in "2 sided", but you did not specify any name, the system automatically searches for a second graphic name of the standard form: name of the first graphic supplemented with an apostrophe (e.g. 1st graphic: SMDTERM, 2nd graphic: SMDTERM'); if such a graphic name does not exist, the error message "terminal graphic not found" is output;
      - as a rule the second graphic on the solder side is the mirror of the graphic on the component side; it is visible only if the terminal is copied or moved to the solder side;

3. Place the first terminal with the mouse or enter <CR>, if the coordinates are specified in the status window;
   if the specifications for the generation of a terminal row have been entered, the terminal row is now automatically generated.

B. Terminal Generation with 'Local' Definition

Terminal graphics which are used only for the generation of single board terminals may be generated 'locally', i.e. directly in the modification part.

In contrast to the 'global' reference terminal graphics the 'local' terminal graphics have effect only on the board terminal to which they belong.

The generation of 'local' terminal graphics grossly corresponds to the generation described in the chapter "reference component" of the technology description - the differences are the following:

- The generation is not done in the reference component menu, but - with the same functions - in the modification menu (with "TERMINAL/create" and "TERMINAL/mod. graphic").
- Before calling the function "TERMINAL/create" in the modification part, the component to which the terminal is to belong has to be activated (if it does not yet exist, it has to be generated).
- Do not change the name in "graph.", but leave the name "LOCAL" automatically entered by the system unchanged during the terminal generation.
- After finishing your work with the function "mod. graphic" the generated (or modified) 'local' graphic automatically gets a new graphic name: original name supplemented with the tilde character "˜" and an appropriately incremented number (e.g. "STDN25˜1).
  if "Y" has been specified in "2 sided" the second graphic gets the name of the first graphic supplemented by an apostrophe (e.g. STDN6').

How to procede for B

The graphic of the terminal has to contain all graphic elements which are necessary for its pad representation on the different layers in the layout:
- the pad graphic for the signal layer(s)
- if applicable the pad graphic for the solder resist plan on the respective RESIST docu layers
- if applicable the heat trap graphic on the plane layer(s)
- if applicable the graphic for the area of the solder paste to be laid on the SMD pad on the respective docu layer (for the SMD reflow method)

1. get the component to which the terminal is to belong and activate the function "TERMINAL/create"

2. specify in the status window:

   a. if applicable enter a terminal name of your choice in "pin" instead of the automatic pin number
   b. if the terminal is not to be placed with the mouse, enter the placement coordinate in "coord."
      Note:
      The reference point for the placement coordinate is the rotation point!
   c. do not change the name in "graph.", but leave the name "LOCAL" entered by the system unchanged;
      Note:
      After finishing your work with "mod. graphic" the 'local' graphic automatically gets a name consisting of the name root "STDN", followed by an appropriately incremented number;
   d. if it is an SMD terminal
      > enter "Y" in "2 sided"
      > do not make any entries in "graph." in the second input line
      Notes:
      - the second graphic is automatically generated by the system; it represents the mirror of the graphic on the solder side generated by the user on the component side and at first only exists in the graphic pool.
      - it is visible only if the terminal is moved or copied to the solder side;
      - the second graphic automatically gets the name of the first graphic supplemented by an apostrophe (e.g. STDN6');
      - the name of the second graphic only appears in the menu (in the second i/o graphic field), if the second graphic has been activated, i.e., after the terminal (or the component to which the terminal belongs) has been moved or copied to the solder side;

3. after <CR> or <LMB> the query "generate new graphic?" appears in the status window;
   > if you answer "N", you return to the previous menu and may modify the specifications
   > if you answer "Y", you may place the terminal;

4. placement of the terminal
   * with <CR>, if the placement coordinate has been specified in the status window
   * or with the mouse
   The terminal still consists only of a placement point marked with a cross;

5. activate the function "TERMINAL/get", get the terminal
   * with the mouse
   * or via specifying the terminal name in "pin" in the status window and <CR>;

6. activate the function "mod. graphic" / "digitize"
   Notes:
   - while working with the function "mod. graphic" the placement point (=rotation point) of the activated terminal is the (temporary) coordinate origin!
   - to make the digitizing work easier it is sensible to open a "window" around the terminal; if convenient you may also activate the "grid on" function for the grid output;

7. Digitizing the pad graphic for the signal layers

   a. for SMD pads
      * specify in the status window
        - in "layer" the number of the layer where the track segment is to be digitized as pad graphic;
        - in "trk.type" the number of the track type (from the track definition of the respective layer technology) with which the segment is to be digitized;
      * use the function "offset" and the coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point for the track segment;
      * activate function "LINE"
      * digitize the segment for the representation of the SMD pad on the component side

   b. for drilled-through pads
      * if applicable use the "offset" function and the coordinate specification (via the keyboard in the status window or via the mouse) to set the placement coordinate for the via;
      * activate the function "VIA"
      * specify in the status window
      - in "via-type" the number of the via type (from the via definition of the respective layer technology) with which the via is to be digitized on the layer(s) specified in "via-lay."
      - in "via-lay." the layer where the via is to be stored as pad graphic to the terminal;
        (e.g. "*" for "all layers" or "3,4" for two inner signal layers)
      * place the via
      * if different pad graphics are to be assigned to the terminal on different signal layers or signal layer pairs (="pad stack"), repeat the procedure for the other layers (the via type used has to be defined in the assigned layer technologies in the desired size).

8. Digitizing the pad graphic for the solder resist plan

   a. for SMD pads
      * specify in the status window
      - in "layer:" the number of the RESIST layer belonging to the component side where the solder resist track segment is to be digitized (as a rule RESIST-1 or layer number 20);
      - in "trk.-type" the number of the track type (from the track definition of the respective SOLD-RESIST-layer technology) with which the segment is to be digitized;
      * if applicable use the "offset" function and the respective coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point of the track segment onto the starting point of the track segment of the signal layer pad graphic;
      * activate the function "LINE"
      * digitize the segment for the solder resist plan representation of the SMD pad

   b. for drilled-through pads
      * use the "offset" function and the respective coordinate specification (via the keyboard in the status window or via the mouse) to set the placement coordinate onto the one for the signal layer pad graphic;
      * activate the function "VIA"
      * specify in the status window
      - in "via-type" the number of the via type (from the via definition of the respective SOLD-RESIST layer technology) with which the via is to be digitized;
      - in "via-lay." the number(s) of the RESIST-layer (as a rule 19 and/or 20, e.g. "19,20") where the via is to be stored as a terminal graphic element;
      * place the via;

9. Digitizing the heat trap graphic for a possible plane connection

   * specify in the status window
   - in "layer" the number of the PLANE layer defined in the layer plan where the track segments for the heat traps are to be digitized as terminal graphic elements;
     (if you would like to generate the postprocessing data for this plane, the respective layer has to be specified in the CAM menu "Planes" under "layers:")
   - in "trk.-type" the number of the track type with which the segment is to be digitized (from the track definition of the layer technology, which is assigned to the respective docu layer);
   - enter "Y" in "heat trap"
   - enter "Y" in "show heat traps"
   * use the "offset" function and the respective coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point for the first track segment;

* activate the function "LINE"
* digitize the first track segment of the heat trap representation
* use the "offset" function to set the starting point for the second track segment of the heat trap representation, digitize the segment, etc.
  Note:
  the heat trap graphic is output only for those terminals which are connected to the plane (the system "recognizes" and considers this);

10. Digitizing the paste graphic for the reflow solder plan

* specify in the status window
- in "layer:" the number of the PASTE docu layer belonging to the component side where the track segment is to be digitized as a terminal graphic element;
- in "trk.-type" the number of the track type with which the segment is to be digitized (from the track definition of the respective PASTE-layer technology);
* if applicable, use the "offset" function and the respective coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point onto the paste track segment;
* activate the function "LINE"
* digitize the segment for the representation of the solder paste area via the signal layer representation of the SMD pad

11. After finishing all graphic elements for the terminal store the data with <ESC><ESC>;

    result:
    the generated terminal graphic now exists as a pool graphic with the STDN-name(s) assigned by the system (ref. point 2 c/d)
    if the system-internal generation of a second graphic had been activated, this will be visible only if the terminal is shifted or copied onto the solder side;
    the 2nd graphic contains all elements of the 1st one in a mirrored form on the respective mirror layers, i.e. on the solder side and, if applicable, on the docu layers belonging to the solder side (as a rule RESIST-0, PASTE-0)
    Note:
    - Remember or note down the names of the generated graphics, since at present there is no list function for the names of the graphics existing in the 'pool'.
    - If a graphic generated 'locally' (e.g. a 2nd graphic generated as the mirror of the 1st graphic, which has then to be modified) is modified with the function "mod. graphic", it gets a name of the following form: original name supplemented by the tilde character "~" and and appropriately incremented number (e.g. STDN'~1). It is stored in the graphic pool; the original graphic remains unchanged in the graphic pool;

C. Generation of the NC-routing terminals and contours

The information needed for the NC-routing of the PCB is created by generating so-called "NC-routing terminals" and by digitizing the appropriate NC-routing contours to these terminals.
The NC-routing terminals represent the starting points for the NC-routing tools.
The NC-routing contour starting from the NC-routing terminal represents the NC-routing line with the appropriate tool.

For the generation of the NC-routing terminals 32 predefined reference terminal graphics are available in the graphic pool. To each of these graphics one drill type between no. 255 and 224 from the drill technology has been assigned:

graphic name	drill type
NC-GRAPHIC0	255
NC-GRAPHIC1	254
NC-GRAPHIC2	253
NC-GRAPHIC3	252
.	.
.	.
.	.
NC-GRAPHIC29	226
NC-GRAPHIC30	225
NC-GRAPHIC31	224

The diameter which is assigned to the appropriate drill type in the drill technology represents the diameter of the NC-routing tool which is to be used for the NC-routing contour, i.e. the NC-routing width.
In order to define an NC-routing contour with a certain width, this diameter has to be assigned to one of the drill types between 255 - 224; the reference terminal graphic assigned to this drill type has then to be used for the generation of the appropriate terminal.

How to procede

In order to define an NC-routing contour with a starting point and with a certain width, procede as follows:

1. Assign a diameter to the drill type for determining the NC-routing width:
   * call the function "drills" in the technology menu
   * call a drill type (224-255) and assign the value for the width in "diameter" (e.g. drill type 254 with diameter 1.00mm);
   * store the definition with the (pop-up) function "save";

2. Generate the starting point

   * generate a special component with an appropriate name (for easy identification) in the "modification" menu
   * generate a terminal ("NC-routing terminal") to that component, by making the following entries in the status window:
     - an appropriate terminal name for marking the special function (e.g. NCROUT1)
     - "F" in "t-typ"
     - in "graph:" the name of the reference terminal graphic which has been assigned to the drill type defined under (1) (e.g. NC-GRAPHIC1);
     - place the NC-routing terminal at the desired position;

   Note:
   After the generation of the necessary NC-routing terminals you may delete the component frame and the text field with the component name with the help of the function "Component/mod. graphic".

3. Digitize the NC-routing contour

   * activate the function "mod. graphic" and the subfunction "line"
   * digitize the NC-routing contour

Notes:

- An NC-routing contour must always consist of connected track segments!

- The NC-routing terminal must always be placed at the beginning of a track segment. It must neither be placed on a segment nor at a T-shaped intersection of two segments.

- For each NC-routing contour, a separate terminal must be defined!

- For generating several NC-routing contours with different widths, you have to assign the reference terminal graphic to each of the terminals; the drill type assigned to the terminal graphic has to have the diameter of the desired NC-routing width.

- For creating perforation NC-routings, a separate terminal must be defined for each NC-routing segment (specify the same reference terminal graphic for all terminals).

- To ensure that NC-routing of board contours is recognized by OUTPUT, the diameter of the drill type assigned via the reference terminal graphic (see point 1) has to be defined with the value 0.00.
  Only if that precondition is fulfilled, OUTPUT definitely executes this NC-routing as the last NC-routing procedure.

In the 'starup'technology file the drill type 255 is defined with diameter 0.00 - this drill type is assigned to the reference terminal graphic NC-GRAPHICO.

- In an .adb file a maximum of 32 different NC-routing diameters may be used, since the number of the drill types which may be used is limited to 32.
- During the digitizing, the NC-routing contours are automatically placed on the solder side (=layer 1). The default for the digitizing is track type 1.
- If an NC-routing contour in the signal layer plot on the solder side is to be represented in its real width, you have to enter a track type with the respective width in "Trk-type" in the status window in CIS.
- In order to avoid that the tracks cannot be routed or digitized in NC-routing areas, the NC-routing contours must be within restricted areas - on all layers!
  The area <u>outside</u> the board outline (created with the function 'Sizes'of the technology menu) is a restricted area on all layers. (If necessary, you can activate this restricted area and change it with the function 'mod. graphics'.)
  If you create additional NC-routing contours <u>within</u> the board area, they must be surrounded with 'normal' restricted areas <u>on all layers</u> (to be generated with the function 'restricted areas' of the modification menu).

NC-routing direction:

The NC-routing direction depends on the course of the track segments from the starting point (= NC-routing terminal). At a joint with more than two segments, the direction is determined as follows:

1. In the X direction if an X segment exists (regardless of whether Y or non-orthogonal segments exist).
2. In the Y direction if a Y segment but no X segment exists.
3. In the direction of the non-orthogonal segment if a non-orthogonal segment but no X or Y segment exists.

The preferred direction is always the X direction.

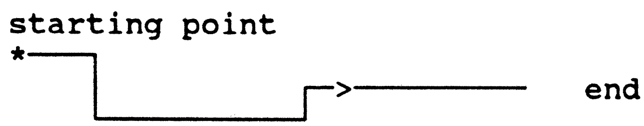
The knowledge of the NC-routing direction rules is of importance where the determination of the right NC-routing direction is a precondition for a correct use of the self-compensation.

Example: inner NC-routing

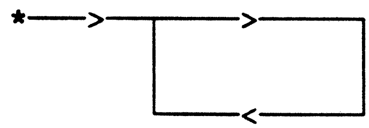
End of a NC-routing process

The NC-routing process is finished if

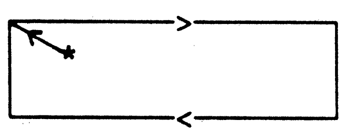
1. a track ends (or is interrupted)



2. a track ends in a T-connection



3. an already processed point is touched again (inner NC-routing)



Sequence of the NC-routing operations:

The sequence depends on whether or not the 'contour mark' exists.

1. The NC-routing starts with the tracks the NC-routing terminals of which were created with reference terminal graphics, the assigned drill types of which are defined with a diameter unequal 0 (i.e. have no 'contour mark')

2. Next, the tracks are considered that have a contour mark (i.e. that were assigned to a drill type with diameter=0).

GET TERMINALS

In order to modify an existing terminal it has first to be activated with the function "get".

Get the terminal
* either by "clicking" it with <LMB>
* or by specifying the name in "pin" and the component in "comp." in the status window and <CR>;

The system defaults to "TERMINAL/move".

Note:
Activating groups of terminals is done with the function "multi-select" (see function description multi-select at the end of this chapter).

MOVE TERMINALS

This function serves for moving activated terminals to a new position; at the same time, the terminal may be rotated and/or moved to the opposite outer layer.

How to procede:

1. get the terminal

2. if the terminal is to be moved onto the opposite outer layer, enter the layer number ("1" or "2") in "layer:"

3. if the terminal is to be rotated, enter the rotation in "rotat." in 1/10 degrees (e.g. "1800" for a rotation of 180 degrees); or rotate the terminal with <MMB>

4. the position is either specified by moving the crosshair to the desired position and <LMB> or by entering the coordinates in the status window under "coord." and <CR>.

It is not possible to place terminals above one another, this is reported as an error ("pin collision").

RENAME TERMINALS

How to procede:

1. get the terminal

2. click the function "rename"

3. replace the current name in "pin" in the status window by the desired name and confirm with <CR>.

DELETE TERMINALS

Terminals may only be deleted separately.
Deleting groups of terminals is not possible.

How to procede:

1. get the terminal to be deleted

2. activate the "delete" function
   depending on whether the option "immediate delete" has been activated in the menu "modes",
   - the terminal is immediately deleted or
   - in the status window the query "really delete?" appears; if you answer "Y" and confirm with <LMB> or <CR>, the terminal is deleted.

Note:

a. If a terminal is deleted in the PCB part, its counterpart in the SCS part is also deleted, unless the parameter "no auto-delete" has been activated in the menu of the function "mod. parameter" (in the SCS modification menu) (activation by marking this parameter with a cross and confirming with <CR>).

b. The graphic of the PCB-terminal remains available in the graphic pool after the deletion.

c. There is no undo function for the deletion of a terminal!

COPY TERMINALS

This function serves for copying an activated terminal to a desired new position - with the copy belonging to the same component.
If required, it is also possible to generate a terminal row with the copy function.

How to procede

1. get the terminal to be copied with the function "TERMINAL/get"

2. activate the function "copy"

3. specify in the status window

   a. if applicable, replace the automatic pin number in "pin:" for the (first) copy by a terminal name of your choice
   b. if it is an SMD terminal and if the copy is to be placed onto the solder side, enter the number of the solder side in "layer:" (="1");
      Note:
      in order to copy the terminal onto the solder side, it has to have a 2nd graphic (i.e., during its generation you had to enter "Y" in "2 sided" and a name of a reference terminal graphic in the second i/o field in "graph.")
   c. the placement coordinate for the (first) copy in "coord." (if the terminal is to be placed via the mouse, this specification is not necessary);
      Note:
      the reference point for the placement coordinate is the component rotation point!
   d. if a terminal row is to be generated with the copy function:
      > enter in "dist." the distance between the terminals in the respective measurement (e.g. in mm: "2.54/0.00" for a row to be generated in the positive x-direction; "0.00/-2.54" for a row to be generated into the negative y-direction)
      > enter the number of the terminals to be generated in "number:"
      > in "series" e.g. "1" for a continuous row of pin number (like "1, 2, 3..."), or
        "2" for a row like, e.g., 1, 3, 5, etc.
   e. if the copy(ies) is (are) to be rotated, enter the rotation in 1/10 degrees in "rotat." (e.g. "1800" for a rotation of 180 degrees); the same is achieved with <MMB>; terminals may only be rotated orthogonally, i.e. in steps of 90 degrees;

4. place the (first) copy with the mouse or press <CR>, if coordinates have been specified in the status window;
   if the necessary specifications have been made for generating a terminal row, this row will now be generated automatically.

TERMINAL / exchange shape

General

This function serves for transferring a reference terminal graphic to an existing board terminal.
If the terminal has already a graphic, it is replaced by the reference terminal graphic.

How to procede:

1. Activate the function "TERMINAL/get"

2. specify in the status window:
   - the name of the reference terminal in "pin:"
   - the reference component name **"!PCBREFGRP"** in "component:"

3. after <CR> the function "exchange shape" is automatically activated

4. click the terminals the graphics of which are to be replaced with <LMB>; the graphic of the activated terminal is replaced by the one of the reference terminal

TERMINAL/list

This function serves for listing the terminals of a component. A terminal may be activated by clicking its name with <LMB>.

How to procede:

1. activate "TERMINAL/get"

2. get the component the terminals of which are to be listed

3. chose "TERMINAL/list"

4. move the cursor in the list window onto the terminal name and press <LMB> to activate the terminal (if applicable, click "continue" or "previous")

5. you may continue your work only after leaving the list window; it is left either after clicking a terminal name or, if no terminal name has been selected, by clicking "quit".

MODIFY TERMINAL GRAPHIC

General

The function "mod. graphic" switches to the work area of digitizing the graphic and provides for different digitizing aids for generating new terminal graphics or for modifying existing ones.

Apart from **lines, circles, vias and texts**, it is also possible to generate or modify **copper areas** to the terminal (with function "area").
In addition, the **rotation point** of the terminal may be moved.
With the "block" function **via blocks** for the router may be generated around terminals.

In contrast to the "reference component" of the PCB technology part, modifications at the terminal graphic made in the modification part are not of a "global" nature, but of an exclusively "local" nature, i.e. the changes only affect the activated terminal.

Terminal graphic elements may be stored on signal layers as well as on docu layers.

How to procede:

The function may be called in two ways:

a. click "TERMINAL";
   activate "mod. graphic"
   get the terminal with <LMB> or via entering the terminal and component names in the status window and pressing <CR>; the system defaults to "offset".
b. get the terminal
   activate "mod. graphic"; the system defaults to "offset".

The component belonging to the activated terminal is highlighted (default colour=blue).

The rotation point of the terminal is marked with a cross - during your work with "mod. graphic" the **terminal rotation point** is the temporary **coordinate origin**;
if it is moved with the function "rotation point", the coordinate origin is also moved!

You may now either:

1. **modify an existing graphic**
   (lines, circle segments, vias, texts, areas)

   a. **get** (function "get")

      > activate function "get",
      > if applicable specify the layer number of the element in "layer:"
      > click the element with <LMB>

   b. **move** (function "move");

      > get the element
      > texts may be
        - rotated (with <MMB> or via a specification in the status window) and/or
        - frames may be assigned to texts (by marking the field "frame:" with a cross in the status window and <CR>)
        - the text may be left-adjusted (by marking " " in the status window with a cross and pressing <CR>)

   c. **delete** (function "delete")

or

2. **digitize new graphic elements**
   (after activating the function "digitize")

   a. the **"offset"** function is used for setting a starting point of an element to be digitized (line, circle segment, via or copper area outline) or for interrupting a current digitizing process;

      > the offset is set with <LMB> or via an entry in the menu in "coord." (first input line) and always refers to the component rotation point;

   b. the function **"LINE"** serves for digitizing lines of any gradient;

      > the starting point is the point set with "offset" or the rotation point, if no offset has been set;

      > the line width is specified with an appropriate track type in the status window under "trk.-type" (defined in the track technology of that layer)

c. the function **"CIRCLE"** serves for digitizing circle segments:

> set the starting point of the circle arc with "offset" and activate the function "CIRCLE"
> or
> activate "CIRCLE" directly and set the starting point with <LMB>;

> beginning at the starting point a circle arc of 90 degrees is opened by moving the crosshair:
> - a movement to the left results in a positive angle
> - a movement to the right results in a negative angle
> - the crosshair center is the center of the circle; the radius is the distance between the starting point and the crosshair center
> - if you move the crosshair so that its x-line is on the starting point (i.e. movement only in x-direction), then the starting point and the end of the circle arc are exactly on the crosshair line.

> with <RMB> the circle segment may be "folded" (=same angle but with the opposite sign);
> by pressing <RMB> again, the angle is "folded" back and is enlarged by 90 degrees;

> the necessary specifications may as well be done directly in the status window:
> - enter in the first input line in "coord." the starting point coordinate referred to the component rotation point;
> - enter in "0/0" the coordinates of the arc center referred to the component rotation point
> - enter the length of the radius in "radius:"
> - enter the arc angle in 1/10 degrees in "angle:"

> the segment width is determined through the appropriate track type in "trk.-type" in the status window (defined in the track technology of this layer)

d. the function **VIA** serves for placing vias:

> activate "digitize/via"

> enter in the status window
> - in "via-type" the via type the graphic representation of which is to be generated (if applicable as a layer-specific pad stack);
> - in "via-layer" the layer(s) where the (first) via representation is to be generated; ("*" stands for "all layers"; single layer numbers may be lined up, separated by commas, a "from-to" specification is also possible (e.g. "6-8"));
> - if applicable specify the placement coordinate in the first input line in "coord." (referred to the rotation point)

> or set the placement point with "offset" and <LMB>
> place the via with <LMB> or with <CR>;
> for pad-stack graphics generate the next via representation (of the same via type!) on the respective layer(s), etc.

e. the function **"TEXT"** serves for generating additional text fields

> activate "digitize/TEXT"
> specify in the status window
  - the text type
    via the keyboard (by entering the respective text type number)
    or by calling the text type with <RMB>
    (by repeating <RMB> all available text types may be called, if the cursor is within the graphic window);
  - if text type 0 (=free text) has been specified, type the desired text in "text:"
    Note:
    if the text is to be represented with small letters, the parameter "small letters" of the "modes" function has to be activated; after the text generation this parameter should be switched off again!
  - if the text field is to get a frame, mark the field "frame:" with a cross;
  - if the text is not to be "centered" but "left-justified" in the text frame, mark "   " with a cross
  - if it is an object-specific text type (e.g. signal name), and if the text field width is to change according to the object name (e.g. after renaming the component), delete the cross in the field "fixed" in the status window
  - if applicable specify the rotation in 1/10 degrees (e.g. "900") in "rotat."
  - if applicable specify the placement and size through a coordinate specification in the first "coord." line and through a specification in "size"

> <CR>
  or
> if placement and size have not been specified in the status window, set the starting point of the text field with <LMB>, open the text field to the desired size with the mouse and place it with <LMB>.

f. the function **"AREA"** serves for generating copper areas belonging to the terminal

> activate "digitize/AREA"
> the system defaults to "offset", set the starting point for the area outline;
> the system defaults to "LINE"; digitize now the outline of the copper area as a <u>closed</u> outline;

> activate the "STOP" function; if the outline is correctly closed, the copper area is filled (as regards the preconditions for the filled or cross-hatched copper area see chapter "copper areas")

g. with the **"block"** function via blocks may be generated to a terminal.
> activate the functions "digitize" and "block"
> click the positions to be blocked for (router and digitized) vias with <LMB> - the blocked positions are marked with a cross.

or

3. **Move the rotation point** (and thus the temporary coordinate origin) (activate the function "rotation point" and specify the new placement coordinate with the mouse or via the keyboard)

Notes:

a. If you would like to leave the function "mod. graphic" without storing the changes, enter <CTRL-C>; then answer the query "ignore changes? (Y/N)" with "Y" and press <CR>.

b. If you would like to leave the function "mod. graphic" and store the changes, either enter <ESC><ESC>, or activate a superior function.

c. To modification of terminal graphics applies: if elements have been digitized which are not connected to the existing graphic, the warning "graphic not ??" is output after the "STOP" function is activated (= because the copper elements belonging to the terminal are not connected to each other).
If you would like to store the graphic in this form, answer the query "ignore errors" which appears after pressing <ESC><ESC> with "Y".

d. Modified terminal graphics get a new, system-generated name; the system-generated terminal graphic names contain the root "STDN", supplemented by an appropriate number.

MODIFY TERMINAL PARAMETERS

General

This function serves for assigning attributes to terminals or for modifying existing attributes (parameters) for activated terminals or terminal groups.

How to procede:

The function may be called in two ways:

a. activate the function "TERMINAL/mod. parameter";
   get the terminal (with <LMB> or via entering the terminal and the component name in the status window and <CR>); or get the terminal group (with "multi-select/multi-action")
b. get the terminal or the terminal group
   activate "mod. parameters";

The components belonging to the activated terminal are highlighted (default colour=blue).

The parameter assignment is done through entering the appropriate specifications in the status window.

For single terminals applies: by clicking another terminal with <LMB> during your work with the function "mod. parameters" the parameters of the newly activated terminal are read into the status window and are ready for modification.

Part of the terminal parameters is "relevant to the library", i.e. such a parameter is stored in the library together with the component.
Most of the parameters are not relevant to the library; they are valid only for the activated .adb-file.
Some fields of the status window are mere "output" fields, the data may not be changed, they only serve for information.

Parameters relevant to the library

graphic: : name of the terminal graphic; existing graphic names may be changed - the change is of a "global" nature

Fan In/Out : input loading factor / output loading factor

The first field stands for the "Fan/in" factor.
An integer number specifies the factor by which the input current of the input is bigger than the one of the standard gate.

The second field stands for the "Fan/out" factor.
An appropriate integer number specifies the output loading factor. It indicates how many inputs of the same standard gate may be driven from the respective exit.

Relation name : Relation to the logical element

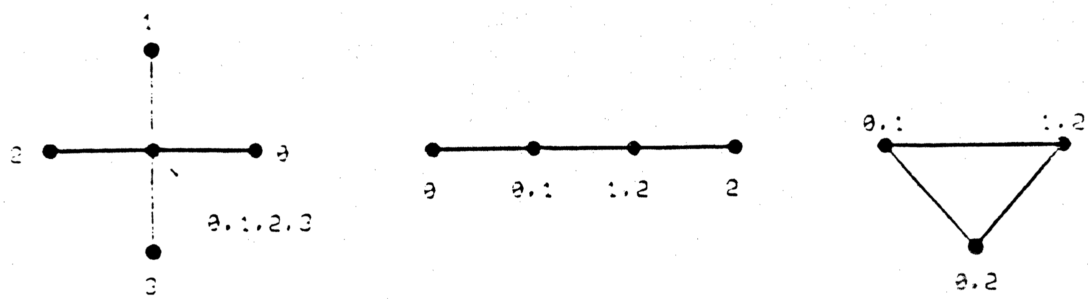
In this field you may specify a relation name for relating the current physical terminal to its logical counterpart in the SCS part.
The relation name consists of the name of the respective gate, followed by the name of the respective logical pin (e.g. "NAND2").
Note:
For terminals connected to GND or VCC planes which are <u>not related</u> specify the respective signal name as relation name!

Parameters not relevant to the library

part sig. : part signal number

If the router is to be given the structure of a multiple connection (star, chain or ring connection), an appropriate part signal number has to be assigned to every of the respective terminals.
In order to assign a part signal number to a terminal, the terminal has at first to be inserted into the main signal (fct. "SIGNAL/insert"). Then get the terminal, activate the function "TERMINAL/mod. parameters" and enter the part signal number in the status window under "part sig." (see fig. 1)

A maximum of 14 part signals may be assigned to one signal (0-13).
Within a part signal (except for part signal 0) only one track width is allowed, i.e. all terminals belonging to the part signal have to have the same track type in "trk.-type". The part signal number 0 (=main signal) may contain different track widths.
In addition to that, the condition
track width part signal n $\leq$ track width part signal n-1
has to be adhered to.
The router routes the signals beginning with the part signal number 0.
At first the single part signals are routed, then they are connected to the main signal.
For the specification of the part signal, the single numbers are separated by a comma; you may as well make "from-to" specifications (e.g. "1-3"); part signal 0 is the actual signal (=main signal).



star connection chain connection ring connection

fig. 1

con-type : daisy chain connection

It is possible to connect a 'daisy chain' to a terminal which is connected to a signal or a part signal. A 'daisy chain' is a chain with as many connections as required <u>without any defined connection sequence</u>.
In contrast to the chains of fig. 1 the connection points of a daisy chain do <u>not</u> represent different part signals, but <u>have to</u> belong to <u>one</u> signal or part signal (see fig. 2).
For generating a 'daisy chain', the terminals have first to be inserted into the respective signal (function "SIGNAL/insert").
Then activate the function "TERMINAL/mod. param." and mark every daisy chain terminal in the "con-type" field of the status window:

- "D" for every 'normal' daisy chain node
- "E" for the ending node (i.e. the last connection point)

Enter the (same!) part signal number for every daisy chain signal in "part sig."

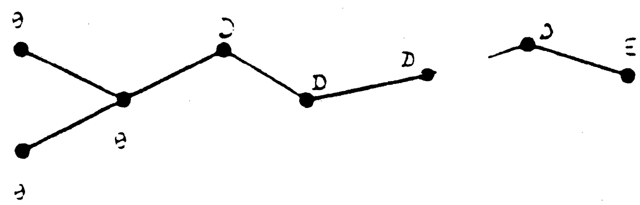


fig.2

connection : layers allowed for connecting the terminal

Here, you may specify the layers where the terminal <u>may</u> be connected.
The specification is done on the basis of the layer number(s).
Single layer numbers are separated by a comma; ("*" stands for "all layers").

pl.- connection : plane connection

"Y" has to be entered for terminals which are to be connected to planes! The plane mark is necessary to differentiate system-internally between plane terminals and 'ordinary' terminals (the information is needed for the routing, for the output of heat traps, etc.)
Terminals with a plane mark are not routed, if they exist on all layers and if they are drilled-through, since they may be connected directly to the plane. But the router connects terminals to the plane which do not exist on all layers or which are not drilled-through; this is done with the help of vias or of terminals directly connectible to the plane.

track diameter : minimum diameter for the track to be connected

The user <u>may</u> specify a minimum diameter of a track to be connected.

-type : track type of the track to be connected

Here, the user <u>has to</u> specify the track type for the track to be connected.

Note:

For some of the changable terminal parameters ("connection", "track diameter", "-type", "plane") the respective signal parameter values have a higher priority when generating the netlist (ref. "SIGNAL/mod. param.", parameters: "layers", "trck diameter" "trk.-type", "pl-connec.").
Therefore, after generating the netlist it is recommendable to use the function "TERMINAL/mod. param." so that terminal parameter values which differ from the respective signal parameters are transferred into the netlist.

CONTINUE

When activating this function together with the function "copy", the copy action may be repeated until another function is activated or until "CONTINUE" is deactivated by clicking it again. "CONTINUE" should be used together with "multi-select" if more than one selection is to be carried out (by opening a selection frame or by specifying the coordinates), since otherwise "multi action" is automatically activated after the first selection.

MULTI-SELECT/MULTI-ACTION

General

With this combination a group of terminals may be selected for a "collective" execution of the function "mod. parameters".

How to procede:

After calling the multi-select function you may select the desired terminals
- by selecting single terminals one after the other
- or by determining the inner or outer area of selection frame

Single terminals may be selected by continuously clicking the them with <LMB> or by specifying their placement coordinates in the status window in "coord." and <CR>.

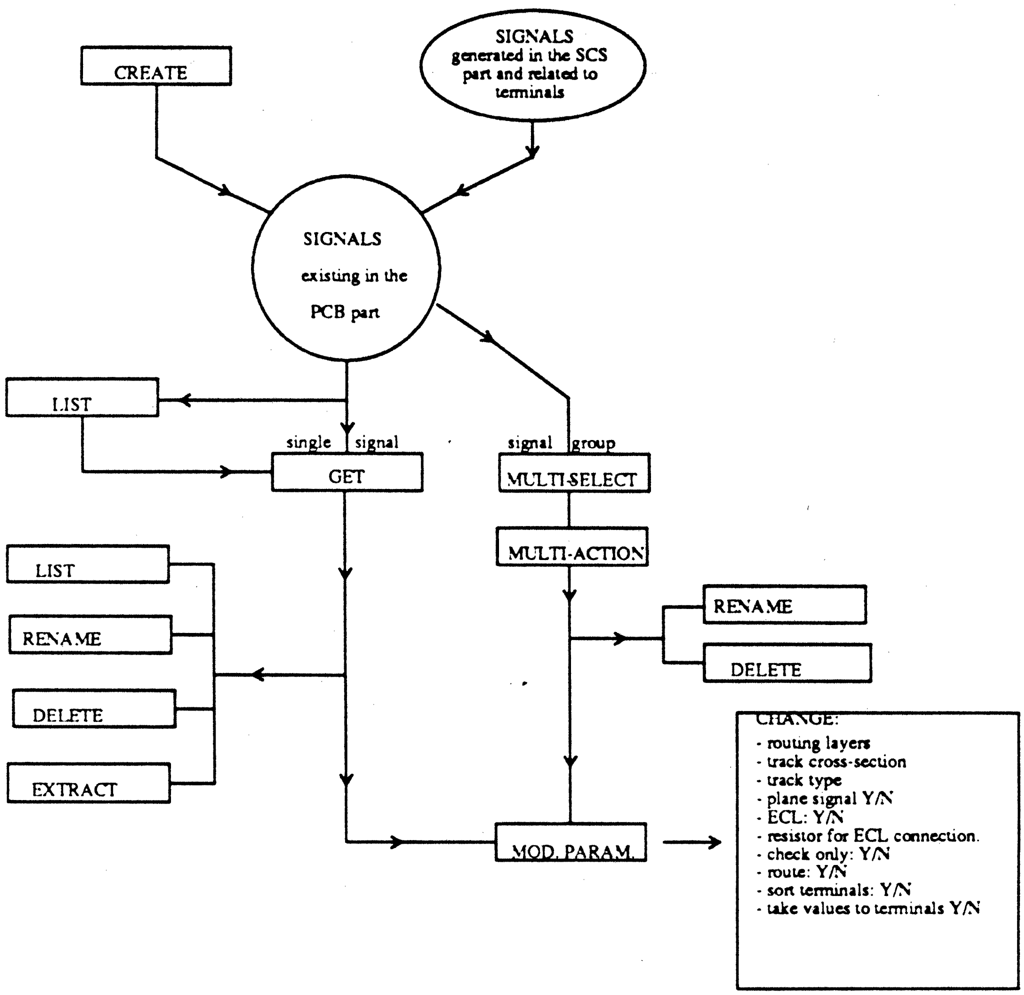
A selection is done
- by opening a selection frame with <MMB> or by entering the starting and end coordinates in "area"
- by specifying the selection letter in "I_nside/O_utside"
- entering "Y" or "N" in "cutting" for including or not those terminals which are not wholly covered by the selection frame.

Note:

- When activating a group of terminals with multi-select the activated terminals are highlighted.

- If more than one selection is to be carried out (by opening a selection frame or by specifying the coordinates) "continue" should be activated after the function "multi-select", since otherwise "multi-action" is automatically activated after the first selection.

SIGNALS


CREATE
SIGNALS generated in the SCS part and related to terminals
SIGNALS existing in the PCB part
LIST
single signal
GET
signal group
MULTI-SELECT
MULTI-ACTION
LIST
RENAME
DELETE
EXTRACT
RENAME
DELETE
MOD. PARAM.
CHANGE:
- routing layers
- track cross-section
- track type
- plane signal Y/N
- ECL: Y/N
- resistor for ECL connection.
- check only: Y/N
- route: Y/N
- sort terminals: Y/N
- take values to terminals Y/N

SIGNAL/CREATE

General

This function serves for defining new signals for the netlist.

Newly generated signals do not contain any other elements than the signal name.
The specification of the connections for the signals is done with the function "SIGNAL/insert".

How to procede:

1. call "SIGNAL/create"

2. if applicable, change the name under "signal:";

3. confirm with <CR>;

   the system defaults to "insert" for the specification of the connections (see "SIGNAL/INSERT");

Notes:

a. Signals generated in the SCS part exist under the same name in the PCB part; in the SCS part the signal connections have counterparts in the PCB part only if the symbol terminals connected to the signal are related to terminals of physical shapes from the PCB part.

b. signal names
   - may consist of up to 6 characters;
   - may not contain the characters "#", "&" or "$", since these are reserved for the automatic generation of names;
   - may not contain any free spaces or tabs;

c. After calling the function "create", the system automatically enters a name for the generated signal; the names are generated as follows:
   > if prior to the function "create" a signal has been modified, the root of that signal name ("e.g. "SIG") is supplemented by the first number available for this combination.
   > otherwise the signal gets the standard name "#", supplemented by the first number available for this combination.

SIGNAL/GET

General

This function activates an existing signal for further processing with one of the functions

- "insert"
- "rename"
- "delete"
- "extract".

How to procede:

There are 3 possibilities of getting a signal:

1. by clicking an element belonging to the signal (terminal, airline or track) with <LMB>;
   newly generated signals which do not yet contain any connections cannot be activated this way;

2. by specifying the signal name in the status window under "signal:" and <CR>

3. by activating the function "list" and clicking the respective signal name in the list window.

Notes:

a. If the signal already contains elements (terminals, airlines or track segments), these are highlighted after activating the signal (default colour = blue);

b. after a signal has been activated, the system defaults to "insert";

c. For activating a group of signals use the function "multi-select/multi-action" (see function description at the end of the "signals" chapter).
   With "multi-select"ed signals only the functions "rename" and "delete" can be executed.

SIGNAL/INSERT

General

This function serves for inserting terminals or copper areas into signals, i.e. they are automatically included into the netlist.
Precondition:
Terminals which are to be inserted into a signal in the PCB part have to be related to a respective symbol terminal in the SCS part.
If this is not the case, the error message "pin or component not related" is output when trying to insert the terminal.

How to procede:

1. get the respective signal
   the system defaults to "insert"

2. insert the terminal or copper area by
   * clicking it with <LMB>
   * for terminals: by specifying the terminal name and the appropriate component name in "pin:" and "component:" in the status window plus <CR>;

3. if applicable, insert further terminals and/or copper areas in the same way like described under point 2;

Notes:

a. The terminals inserted into the current signal are highlighted (default color = blue).

b. The terminals and copper areas inserted into the current signal are connected to the nearest terminal or copper area of that signal by airlines.

c. If terminals or copper areas belonging to other signals are specified or clicked, the error message "pin is already connected" is output.

d. If the component the terminals of which are to be inserted has not yet any counterpart in the SCS part, you may use the function "COMPONENT/transform" to copy the component to a symbol with the same graphic onto the SCS side .
   The terminals of the component are automatically related to those of the symbol and may now be inserted into signals without any problems.

SIGNAL/RENAME

General

Existing signal names (generated by the system or by the user) may be changed for single components as well as for component groups (activated with the function "multi-select).

How to procede:

1. Get the signal
   or
   get the signal group with "mulit-select" and activate the group with "mulit-action"
2. activate the function "rename" by clicking it in the menu window;
3. enter the new name in "signal:" in the status window
   or
   enter the root of the new group name (e.g. "SIG")
4. <CR>.

Notes:

a. Signal names may consist of up to 10 characters, but may not begin with "#", since this is reserved by the system.

b. When renaming signal groups the specified name root is transferred to all selected signals;
the numbering remains unchanged.

SIGNAL/DELETE

General

This function serves for deleting single signals or "multi-select"ed signal groups.

How to procede:

1. Get the signal
   or
   get the signal group with "multi-select" and activate the group with "multi-action"

2. activate "delete" (= clicking the function in the menu window or entering "L" in "fct." in the status window).

Attention:

- If "immediate kill" has been marked with a cross in "modes" of the PCB part, the delete function for signals is executed without any warning.
  If it is not marked with a cross, the query "real delete" appears before the function is executed. In this case the deletion is executed only if answering the query with "Y" and pressing <CR>.

Note:

- The names of deleted signals may not be used a second time for generating symbols.

SIGNAL/EXTRACT

General

This function is a complement to the function "Signal/insert". It removes the assignment of a terminal to a signal.

How to procede:

1. get the signal;
   the system defaults to "insert";

2. activate the function "extract":
   - either by clicking "extract"
   - or with <MMB>

3. click the terminal to be extracted with <LMB>;

Notes:

a. After it's activation the function "extract" remains active as long as no other function is activated;

b. with <MMB> it is possible to 'toggle' between the complementary functions "insert" and "extract".

SIGNAL/LIST

General

This function serves for listing the names of

a. all the existing signals

b. all the terminals connected to the activated signal

How to procede:

One precondition for the "list" function is that one of the functions

a. "get"
   or
b. "insert" ("get" has to be activated prior to this function)

are activated.

In the first case (a) the list of the existing signal names is output after activating the "list" function.
After clicking a signal name in the list the respective signal is activated;

In the second case (b) the list of the terminals connected to the activated signal is output after activating the "list" function.
Output form for the connections: component name/terminal name.

The list is output in a shell window which is placed over the graphic window.
The list window can be moved with <MMB> (go to the window edge, press <MMB> and keep it, and move the window).
By clicking the fields "continue" or "previous" you may page down or page up.

The work in the layout may be continued only if a name has been activated by clicking it with <LMB> in the list window or if the window has been closed with "quit".

SIGNAL/MODIFY PARAMETERS

General

This function serves for assigning attributes or for changing existing attributes (parameters) for activated single signals or signal groups.

How to procede:

The function may be called in two ways:

a. activate the function "SIGNAL/mod. parameters"; get the signal (with <LMB> or by specifying the signal name in the status window and <CR>) or get the signal group (with "multi-select/multi-action");
b. get the signal or the signal group; activate the function "mod. parameters";

Elements belonging to activated signals (terminals, tracks, airlines) are highlighted (default colour = blue).

The parameter assignment is done by entries in the status window.

For the modification of single signals applies: when clicking another signal with <LMB> during the work with the function "mod. param." the parameters of the new signal are read into the status window and are ready for modification.

Parameters:

layers : layers allowed for routing the signal

Here, you may specify the layers where the signal <u>may be</u> routed.
The specification is done on the basis of the layer number(s). Single layer numbers are separated by a comma; ("*" stands for "all layers").

trk.-diam. : minimum cross-section of the signal routing track

Here, you <u>may</u> specify the minimum cross-section of the track with which the signal is to be routed.
If the diameter of the track type specified in "trk.-type" is smaller than the minimum cross-section, the message "track type not wide enough" appears; the parameter modification for this signal may be continued or finished only if a track type with an appropriate width or if an appropriate minimum cross-section has been specified.

Note:
If in the SCS part a value is entered in the signal parameter "Q:", then in the PCB part this value appears in the field "track cross section".

trk.-type : track type for routing that signal

Here, the track type with which the signal is to be routed has to be specified (default is track type 1).

Notes:

- If the signal has been generated in the SCS part and if a certain track type has been assigned for its realisation in the PCB part with the function "mod. parameters" in the field "for PCB t-type", this track type automatically appears here in the field "trk.-type".

- The track type entered in this field is automatically used for the inserted terminals as connection width (for the parameter "-type", see "TERMINALS/mod. parameters").

Attention:

The connection track type which is assigned to the target terminal in the netlist (see "TERMINALS/mod. param.", parameter "-type") is decisive for which track is used for routing a signal between two terminals. If the parameter "-type" is subsequently changed for a terminal which belongs to the signal (i.e. after inserting the terminal into the signal and, if applicable, after modifying the signal with the function "SIGNAL/mod. param."), then this changed track type will be entered into the netlist.

plane : plane connection mark

For every signal which is to be connected to a plane, "Y" has to be specified - a specification directly after the generation would be the best!
If the plane mark is set only after the terminal is inserted into the signal, "Y" should also be specified in the field "term.: take values:", so that the plane mark is transferred to all terminals belonging to the signal.
The plane mark is a precondition for the system-internal differentiation between plane terminals and 'ordinary' terminals (this information is needed for the routing, for the output of the heat traps, etc.).
If the terminals with a plane mark exist on all layers and if they are drilled-through, they are not routed, since they may be connected directly to the plane. The router connects terminals to the plane which do not exist on all layers or which are not drilled through; this is done with the help of vias or terminals directly connectable to the plane.

* Note:
If in the SCS part the signal parameter "signal on own layer" has been marked with a cross, then "Y" is automatically entered in the field "plane" in the PCB part.

Attention:
The existence of this mark with the appropriate terminals (see "TERMINALS/mod. param.", parameter "pl.-connect.") is decisive for the existence of the plane mark in the netlist.
If the parameter "pl.-connection" of a terminal belonging to a plane signal is subsequently changed from "Y" to "N" (i.e. after inserting the terminals into the signal and after modifying the signal with the function "SIGNAL/mod. parameters"), then this latest value will be entered into the netlist.

Routing:
Critical : mark for preferred routing

Signals for which this parameter is set to "Y" are placed at the beginning of the netlist and are thus routed first.
Note:
If in the SCS part the signal parameter "to be routed critic." is marked with a cross, then "Y" is automatically entered in the field "Routing:critical" of the PCB part.

check only :

route: routing mark

If a signal is to be automatically routed, "Y" <u>has to be</u> entered in "route".
The default for this parameter is "Y".
Signals for which the user sets the "route" parameter to "N" are ignored by the router.
Note:
If in the SCS part the signal parameter "not to be routed" is marked with a cross, then "N" is automatically entered in "route" of the PCB part.

Term:
take values: Transfer of the signal parameter values onto terminals

If the parameters "layers", "trk.-diameter", "trk.-type" or "plane" have been changed with the function "SIGNAL/mod. parameters", these changes are transferred to all the terminals of the signal only if "Y" plus <CR> is specified in this field. (The signal parameters corresponding to the above terminal parameters are: "connect.-layers", "track cross-section", "-type", "pl.-connection")

Attention:
If the signal parameters "layers", "trk.-diameter", "trk.-type" "plane" have been changed and if the signal parameter "term: take values" is not set to "Y",
- all the signal terminals keep their original parameter values,

- whereas all terminals inserted subsequently get the changed parameter values of the signal.

ecl: marking signals as ECL signals
sort: suppression of the terminal sorting for ECL signals
res.: specification of the terminating resistors for ECL signals

These three parameters serve for automatically generating ECL connections to resistors.

For signals activated with "multi-select/multi-action"
- which are marked as ECL signals
- for which the name root of the terminating resistors is specified in "res." and for which "N" is specified in "sort:",

and the parameters of which are stored, the system starts an automatic connection procedure which connects the ECL signals via airlines to the nearest terminating resistor, after the ECL terminal to be connected first has been clicked with <LMB>.

How to procede for automatically generating ECL connections

a. place the terminating resistors to the desired position; name the resistors so that they all have the same root (e.g. ABW1, ABW2, ABW3, etc.);
   for single resistors: connect the terminal which is not to be connected to the ECL signal to the respective signal!
b. generate the ECL signals with an identical name root (e.g. ECL1, ECL2, etc.);
   activate the function "SIGNAL/insert" and connect the ECL signals only to those signal terminals which do not belong to the terminating resistors.
c. get the ECL signals with "multi-select" (name specification with the help of wildcards, e.g. "ECL*") and activate "mod. param.";
   enter "Y" in "ECL" in the status window and "N" in "sort."; use wildcards to enter the name specification for all resistors (e.g. "ABW*"); and press <CR>;
   now every ECL signal is automatically connected to the nearest terminating resistor with an airline (the shortest way is automatically calculated);

Note:

Before the automatic connection of the terminating resistors can be activated, these resistors have to be connected to a signal at one of their two terminals (e.g. GND, VCC, etc.), otherwise the system automatically connects ECL signals to both terminals!

This does not apply to resistor networks.

CONTINUE

"CONTINUE" should be used together with "multi-select" if more than one selection is to be carried out (by a names' specification, by opening a selection frame or by specifying the coordinates), since otherwise "multi action" is automatically activated after the first selection.

MULTI-SELECT / MULTI-ACTION

General

With this combination a group of signals may be selected and may be activated for the "collective" execution of the functions "rename", "delete" or "mod. parameters".

How to procede:

When calling the multi-select function a menu appears in the status window where based on a differentiated selection mask (according to the parameters definable in "SIGNAL/mod. param.") and based on a name specification (if applicable, wildcards may be used) and/or based on the specification of a section (via mouse or keyboard) a detailed selection can be executed.
As regards the selection criteria see the respective parameters in "SIGNAL/mod. param.".

Single signals may be selected by
- specifying the name under "signal:" in the status window plus <CR>
- clicking them with <LMB>
- specifying a coordinate of a signal element (terminal, track, airline) in the field "coord." plus <CR>

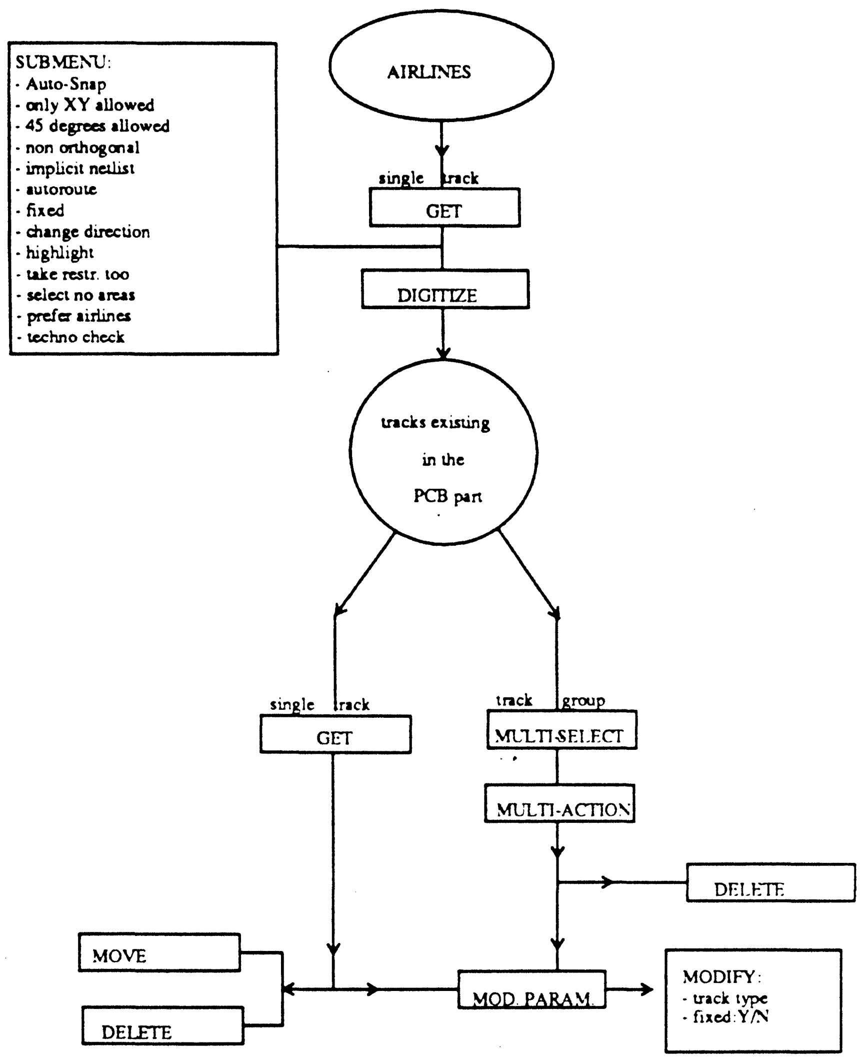
these signals may then be "collectively" activated with "multi-action". The way of proceding requires the activation of the "CONTINUE" function.

A selection with the help of a selection frame is done as follows:
- specification of the selection letter in "I_nside/O_utside",
- specification of "Y" or "N" in "cutting" as regards signals only partly covered by the selection frame
- opening a selection frame with <MMB> or specifying the starting and ending coordinates in "area" plus <CR>.

Hints:

- Signals activated with "multi-select" are highlighted.
- if prior to "multi-action" more than one multi selection is to be carried out, "CONTINUE" should be activated after "multi-select" since otherwise the system defaults to "multi action" after the first selection.

TRACKS


SUBMENU:
- Auto-Snap
- only XY allowed
- 45 degrees allowed
- non orthogonal
- implicit netlist
- autoroute
- fixed
- change direction
- highlight
- take restr. too
- select no areas
- prefer airlines
- techno check
AIRLINES
single track
GET
DIGITIZE
tracks existing in the PCB part
single track
GET
track group
MULTI-SELECT
MULTI-ACTION
DELETE
MOVE
DELETE
MOD. PARAM.
MODIFY:
- track type
- fixed:Y/N

INTRODUCTION

As a rule, the signals which are generated in the SCS or PCB part appear as airlines between the terminals in the PCB part.

The physical realisation of these airlines are the tracks.

The interactive generation of tracks instead of airlines is done by 'digitizing' with the mouse.
The automatic routing of tracks is done by the auto-router.

Digitized tracks may be routed as "variable" or "fixed" tracks.
"Variable" tracks may be "touched" and changed by the router.
"Fixed" tracks may not be changed; but they may be specified as a target for the automatic routing process.

The vias to other layers are summarized under the term "track", since they are an integrated part of the tracks.

TRACKS / GET

General

In order to generate or modify a track the airline or the respective track element has to be activated with the "get" function.

How to procede:

If you click the function "TRACK", "get" is automatically activated, since this operation preceedes any further action.

It is possible to get:

a. a physical element of an existing track, i.e. a "line" (= track segment) or a "via" which is subsequently moved or deleted.

   - get lines with <LMB>
   - get vias with <MMB>;

   after an element is activated, the system defaults to "move";

b. or an airline which becomes a track by digitizing the connection.

   after an airline is activated, the system defaults to "digitize";
   That end of the airline which is nearer to the crosshair when getting the airline, is the automatic starting point of the track; now a track segment appears from that starting point to the crosshair (the colour is defined in the graphic menu for the digitizing layer), this airline moves as the crosshair moves.
   That end of the airline which is farer away from the crosshair when getting the airline is the ending point of the digitized track; an airline appears from the crosshair to this ending point which moves as the crosshair moves.

Note:
   For activating a group of tracks (which may belong to different signals) the combination "multi-select"/"multi-action" is used. With "multi-select"ed signals only the functions "delete" and "mod. parameters" may be executed.

TRACKS / MOVE

Move Line

Attention:
Only orthogonal track segments may be moved!

After an orthogonal track segment (= a line) is activated with <LMB>, it may be moved parallely to its current position.
An airline from the activated point to the crosshair appears which moves as the crosshair moves. That part of the line which is orthogonal to the activated track determines the distance of the movement; <LMB> executes the movement.
The adjacent track segments are also moved, i.e. they are made shorter or longer.

Attention:
When segments are moved, vias are fixed points, i.e., if required, vias have to be moved separately.

Move via

In order to activate a via, move the crosshair to the position and press <MMB>.
The system defaults to "move";
now it is possible to place the via with <LMB> or with <MMB> at the desired position.
The track segments connected to the via are made longer or shorter.
If the track is yellow, then the tracks lie on the same positions on the complementary digitizing layers.

Note:
After the deletion of a via, the remaining Z-airline marks (see below "via/delete") may be activated with <MMB> and may be moved; after the movement the Z-airline marks are represented again as vias!

TRACKS / DELETE

Delete Line

After a track segment (= a line) has been activated with <LMB>, it may be deleted with <RMB>.
A deleted track segment is replaced by an airline.

Note:
After a track segment has been activated and deleted with <RMB>, you may delete other elements of the same track (segments <u>and</u> vias) just by pressing <RMB> again.

Delete Vias

An activated via is deleted with <RMB>.
In order to activate a via, move the crosshair to its placement coordinate and press <MMB>.

After the deletion the placement point of the via is marked with a cross as a point which is yet to be connected (so-called "Z-airline", i.e. airline in the third dimension), if the points to be connected are not on the same layer, i.e. if a via is necessary.
The Z-airline mark disappears only if a new via is placed onto this coordinate or if by a subsequent digitizing a via is placed at the other end of the connection, i.e. if a "via" is no longer necessary.

Note:
Vias as well as the Z-airline marks remaining after the deletion are treated like terminals in connection with the "TRACK" function.
It is possible to get them with the function "TERMINAL/get" and to move or delete them.
- during the movement the connection to the signal is moved as an airline (i.e. the original and the new placement point are connected with an airline)
- with the deletion the adjacent track segments also disappear; the interruption is bridged with an airline.

Delete tracks 'from ... to'

In order to delete a track between two points (incl. the vias), activate the function "TRACK/delete"; click the starting and the ending point of the segment with <LMB>.

Delete track in segments

A track may be deleted in segments as follows:
get the last track segment with <LMB> and press <RMB>;
by presing <RMB> again, it is now possible to delete the other elements (segments <u>and</u> vias) (in the opposite order of their generation)

Delete 'multi-selected' tracks

a. Within an area the tracks may be deleted as follows:
   - activate "multi-select", specify the appropriate information in the status window for the desired selection, open a selection frame with <MMB>,
   - activate "multi-action", then "delete".

b. Variable tracks (not the "fixed" ones) may be deleted as follows:
   - activate "multi-select" and press <CR>
   - activate "multi-action", then "delete"

c. Fixed tracks may be deleted as follows:
   - activate "multi-select", specify "Y" in "fixed:" in the status window and press <CR>,
   - activate "multi-select", then "delete".

TRACKS / DIGITIZE

General

This function serves for routing tracks interactively (= transferring an airline into a physical connection).

Before the digitizing, the airline has to be activated. Then the system defaults to "digitize".

Note:
You should activate the connection to be digitized only via the airline, not via existing track segments (when activating existing track segments the system expects a movement or a deletion)

How to procede

1. Get the airline in the near of that end which is to serve as the starting point for the digitizing.
   Notes:
   - after the airline is activated, the system defaults to "digitize".
   - that end of the airline which is nearer to the crosshair when getting the airline, is the automatic starting point of the track; now a track segment appears from that starting point to the crosshair (the colour is defined in the graphic menu for the digitizing layer), this airline moves as the crosshair moves.
   - that end of the airline which is farer away from the crosshair when getting the airline is the ending point of the digitized track; an airline appears from the crosshair to this ending point which moves as the crosshair moves.

2. the following entries can be made in the status window:

   a. in **"trk.-type"** the track type which is to be used for the digitizing;

   Note:
   - the track type specification refers to the track technology which is assigned to the layer specified in "layer";
   - as a rule the track type assigned to the starting point appears in this field:
     > if the starting point is a terminal, the track type entered in the terminal parameter "-type" appears in this field
     > if the starting point is an existing track segment, then the track type of that segment appears in this field;

b. in **"viatype"** the via type to be used may be specified;
Note:
- the via type specification refers to the via technology which is assigned to the layer specified in "layer:";
- as a rule, the via type appears which is assigned to the track type specified in "trk.type" in the appropriate layer technology.

Attention:
if an existing reference terminal name is specified in "term.name", this is used as a via and not the via specified in "via type"!

c. if you would like to use an existing reference terminal as a via instead of the via type specified in "via type", specify that name in **"term. name"** (= name used in the PCB technology/reference component in the field "pin:")

d. in **"coord."** you may specify the coordinates of the point to which the next track segment is to be routed; a subsequent <CR> (not <LMB>) routes the track segment - if there is not any hinderence or invalid gradient.

e. if you specify "Y" in **"fixed"**, the track is fixed, i.e., it may not be altered by the router; for a better differentiation between fixed and variable tracks the fixed tracks are represented with dotted lines.

f. if you specify "Y" in **"impl. netlist"**, a terminal which does not yet belong to any signal becomes automatically part of the signal (i.e. of the netlist) when clicking it.
Note:
this also applies to copper areas, if they are cross-hatched, and if "select no areas" has not been activated in the "TRACK" submenu;
If "impl. netlist" is set to "N" (= default) the system outputs the message "pin is not connected" if such a terminal is clicked;

g. in **"layer"** you may specify the digitizing layer (according to the layer plan); please bear in mind that the starting point (= terminal or track segment) has to exist on that layer!

2. Digitizing:

   a. move the crosshair to the point to which the first track segment is to be routed and press <LMB>;
   between the starting point and the crosshair position a track segment is routed, the track which is still to be routed to the ending point is represented as an airline;

   b. if required, you may place a via (via or reference terminal, see 1b, 1c) at the crosshair position by pressing <MMB> which is then automatically connected to the last digitized track segment; after that you are on the opposite layer of the layer pair and you may now continue to route the track;
   If you want to get to a layer which is not part of the current layer pair, you first have to enter the appropriate layer number in "layer" and then place the via with <MMB>; in this case, a via going through all layers is automatically used.
   Note:
   In order to use a **buried via** as a via from one layer of the layer pair to the opposite layer, the following has to be provided:
   - the field "buried:" has to be marked with a cross in the menu "sizes" of the PCB technology
   - in the via technology the used via type has to be marked with a "b" (for "buried") instead of "D" (for "going through") (in the t-mode) or the field "buried:" has to be marked with a cross (in the v-mode).

   c. thus you may route the whole track step by step; the gradients allowed are determined in the submenu (if, for example, "only X-Y allowed" is activated, only horizontal and vertical track segments may be routed - see below).

   d. if you reached the ending point, the system defaults to "track/get" and you may activate the next airline to be digitized;

   e. if you activate "get" before reaching the ending point, the routing is interrupted; an airline connects the ending point and the last routed segment or the last via;

   f. **<RMB> deletes** the last routed segment or via;
   repeated <RMB> deletes the routed elements - in the opposite order of their creation;

   Note:
   When routing a track segment you may use different track and/or via types, the respective track or via type has to be entered in the status window before digitizing the respective element.

TRACKS / MODIFY PARAMETER

General

This function serves for subsequently changing the track type or the parameter "fixed:Y/N" for single tracks or for groups of tracks.
A precondition for the modification is the activation of the respective track(s).

How to procede:

The function may be called in two ways:
a. activate the function "TRACK/mod. param." ;
   get the track (with <LMB> or by entering the coordinates in the status window and <CR>) or the group of tracks (with "multi-select"/"multi-action");
b. get the track or the group of tracks
   activate "mod. param.";

Elements belonging to the tracks (segments, vias) are highlighted (default=blau).
When single tracks are modified and if a new track is activated with <LMB> while working with the function "mod. param.", the parameters "trk.-type" and "fixed" of the new track are entered into the status window and are ready for modification.

After the desired track type and/or the desired option for the "fixed" parameter has been entered plus <CR>, the modification is executed for the activated track(s).

ROUTE-MODE

With this option it is possible to switch on or off the automatic routing of tracks when moving vias.

If in the "Modes" menu the parameter "digit:auto-route" has been marked with a cross (= activated), the function "ROUTE-MODE" is automatically activated when calling the function "TRACK/digitize".

The function "ROUTE-MODE" has the same effect like the option "autoroute" of the TRACK submenu. The activation of "auto-route" automatically activates the "ROUTE-MODE" and vice versa.

CONTINUE

"CONTINUE" should be used together with "multi-select" if more than one selection is to be carried out (by opening a selection frame), since otherwise "multi action" is automatically activated after the first selection.

MULTI-SELECT / MULTI-ACTION

General

With this combination a group of tracks may be selected and may be selected for the "collective" execution of the function "delete" or "mod. param.".

How to procede:

The multi-select function activates a menu in the status window where based on a selection mask and, if applicable, based on the definition of a section (via mouse or keyboard) a detailed selection may be executed.
The selection mask offers the following criteria for the track and via selection:
- "trk.-type", "fixed:Y/N"
- "viatype", "fixed:Y/N"

Single tracks may be selected by
- clicking them with <LMB>
- or by specifying the coordinate where the element is placed in "coord." plus <CR>

they may then be activated "collectively" with "multi-action". The precondition is the activation of the CONTINUE function.

The selection with a selection frame is done as follows:
- Specification of the respective selection letter in "I_nside/O_utside",
- Specification of "Y" or "N" in "cutting:" for the confirmation or rejection of the tracks only covered partly by the selection frame
- open a selection frame with <MMB> or specify the starting and the ending coordinate of the area in "area" plus <CR>.

Notes:

- When activating a signal group with "multi-select" the activated signals are highlighted.
- If prior to activating "multi-action" more than one multi-selection is to be carried out, "CONTINUE" has also to be activated after "multi-select", since otherwise the system automatically activates "multi-action".

SUBMENU

Auto-Snap

When tracks are digitized, the user may specify certain gradients (see "45 degrees valid", "only XY valid"). Other gradients are then not allowed.
With the auto-snap-mode switched on the gradients specified don't have to be exactly adhered to, since the system automatically 'pulls' the track to the next possible gradient.
Thus digitizing is considerably made easier for the user.
With the auto-snap-mode switched off a not exact digitizing of tracks leads to the error message: "invalid grade".

Only XY valid

Only 90-degree gradients are allowed, i.e. 90, 180, ...degrees. That means only orthogonal tracks may be routed.

45 degrees valid

Only tracks with gradients of a multiple of 45 degrees are allowed, i.e. gradients of 45, 90, 135, 180 ... degrees.

Non-orthogonal

Any gradients are allowed. This provides for the possibility to digitize vectors of any gradient.
Note:

In order to digitize non-orthogonally, auto-snap has to be switched off!

impl. netlist

This option has the same effect like setting the parameter "impl. netlist" in the status window of the work menu "TRACKS" to "Y". If during the digitizing you hit terminals which are not yet assigned to any signal, they will be included into the current netlist if "impl. netlist" is activated.

autoroute

This option has the same effect like the function "route-mode" in the work menu "TRACKS".
This option serves for switching on/off the routing of tracks when vias are moved.

fix

This option has the same effect like the parameter "fixed" in the status window of the work menu "TRACKS".
If "fixed" is activated the digitized tracks are fixed, i.e. they may not be altered by the router.

change direction

exchanges the starting and the ending point for the digitizing. This is possible only if an airline has already been activated.

highlight

The complete signal to which the activated track belongs, is highlighted.

take restr. too

Only concerns files converted from CALAY-V04.Rev.5n with SCSCOM: track segments belonging to outlines of blocking areas are assigned to a pseudo terminal generated internally during the conversion.
This terminal can only be activated if in the submenu the option "take restr. too" has been activated.

select no areas

If "impl. netlist" is activated, cross-hatched copper areas are treated like terminals if activated during the digitizing, i.e. they are included into the signal.
By activating this option the integration of copper areas into signals is deactivated.

prefer airlines

If this option is activated, airlines are preferred when getting adjacent airlines and tracks, otherwise the tracks are preferred.

techno check

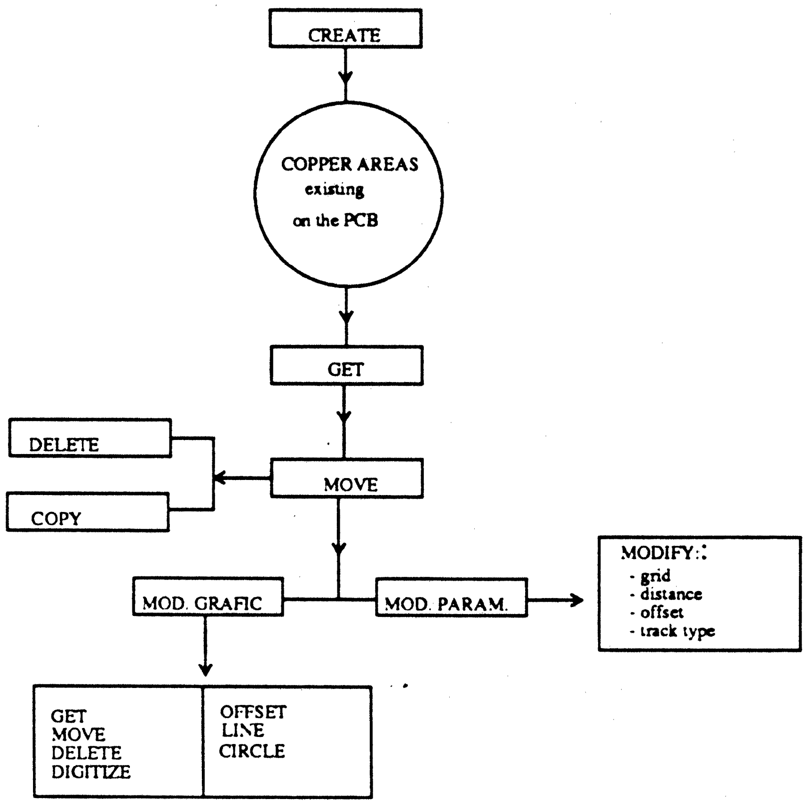
This function serves for checking an activated track for
- interrupts
- shorts and
- drill hole collisions.

work menu

leads back to the "TRACKS" menu

COPPER AREAS

(function selection also via pop-up)



Defaults

- modes - "areas auto-close:X"
- graphic selection - "copper areas:X"
- graphic selection - "copper areas:RM"

Outline Generation and Modification

After executing the function "COPPER AREAS/create" it is possible to open a filled rectangle with the left mouse button. All the tracks and vias in this rectangle are spared with the minimum clearance.

After activating "COPPER AREAS/get", clicking the area with <LMB> and activating "mod. graphic" the representation changes from filled to 'contour'.
Now you may get and delete single lines of the contour and then newly digitize the outline (tools: offset, line, circle).
The contour has always to be closed, separate lines must not remain.
The "STOP" function automatically closes the last gap (if applicable) and executes a check.
If the area is not properly defined, an error message appears.
If the modified copper area has been correctly defined, it is filled again, as soon as the function "COPPER AREA" is newly activated.

cross-hatching

After activating a copper area with "COPPER AREAS/get" and "mod. parameters" the following grid parameters appear in the status window:

<u>grid:</u>
This serves for defining a so-called basis grid. The default is the routing grid.

<u>dist.</u>
The distance to be entered here is the multiplication factor for the grid: copper grid = distance * basis grid

<u>offset:</u>
movement of the copper grid in basis grid units (to be specified in x and y direction)

trk.-type

By specifying the track type you determine the track width with which the grid is generated.
The track type refers to the track technology which is assigned to the layer where the copper exists.

An example of a grid in 1/40" would be: grid=0.1588 mm, dist.=8, offset=4/4

After the above parameters have been adjusted a <CR> will execute the cross-hatching.

Deactivating cross-hatching

The cross-hatching of a copper area may be deactivated with "COPPER AREA/mod. param." by clicking the copper area and pressing <CR>. By entering new parameters and by pressing <CR> twice the cross-hatching may be modified until the desired result is achieved.
After graphic operations, as described under "changing the cross-hatching" it is no longer possible to deactivate the grid.

Changing the cross-hatching

A cross-hatched copper area can be subsequently modified with "mod. graphic"; e.g. it is possible to delete islands or to digitize additional grid cells. For getting a thus modified area you have to activate the origin (if necessary, set "copper area" in the graphic menu/colours to "KM", so that the origin becomes visible).

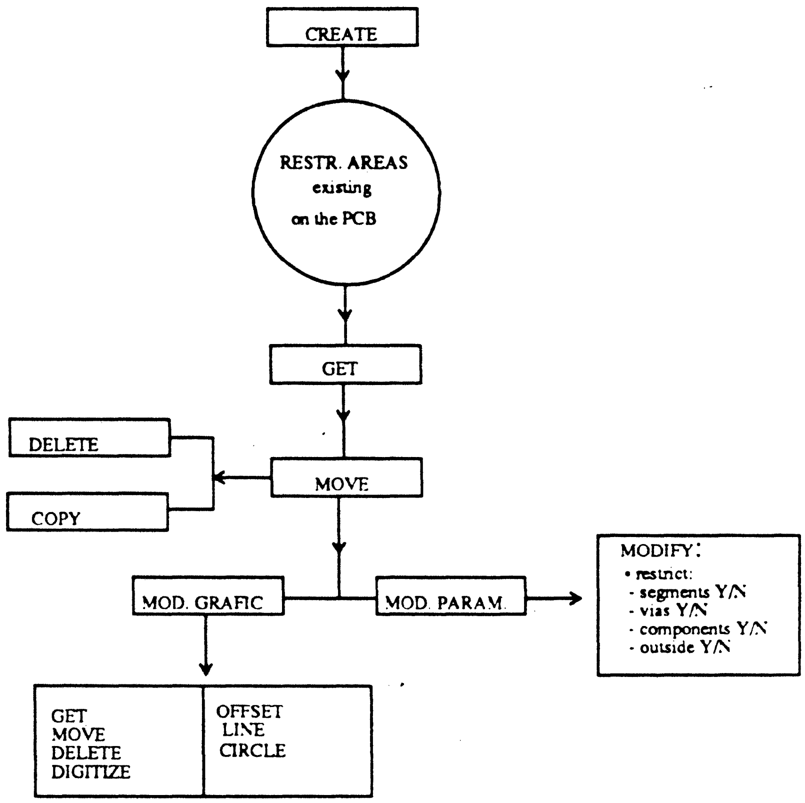
Signal Assignment

By getting a signal, activating the function "SIGNAL/insert" and clicking a copper area, the area is assigned to the signal. An airline appears from the area origin to the respective signal.

Further Operations with Copper Areas

1. move
2. copy
3. delete
4. rotate with <MMB>
5. transfer to the other layers of the layer pair with <RMB>, then place it with <LMB>
6. mirror with status entry "mirrored:Y" and <CR>

RESTRICTED AREAS (selection also via pop-up)



Defaults

- modes - "areas auto-close: X"
- graphic selection - "restricted areas: X"
- graphic selection - colours - "restricted areas: RM"

Outline Generation and Modification

After executing the menu function "RESTRICTED AREAS/create" it is possible to open a rectangle with the left mouse button at a free position.

After activating "RESTRICTED AREAS/get", clicking the area with <LMB> and activating "mod. graphic" you may get single lines of the outline and delete them and then digitize a new outline according to your requirements (tools: offset, line, circle).
The outline has always to be closed, separate lines must not remain.
The "STOP" function automatically closes the last gap and checks the area.
If the area has not been defined correctly, the error message "restricted area not closed - hit <CR> to close" appears; if you try to leave the function "digitize" of an uncorrectly defined area and to store the data, the query "ignore error(s): Y/N:_" appears.

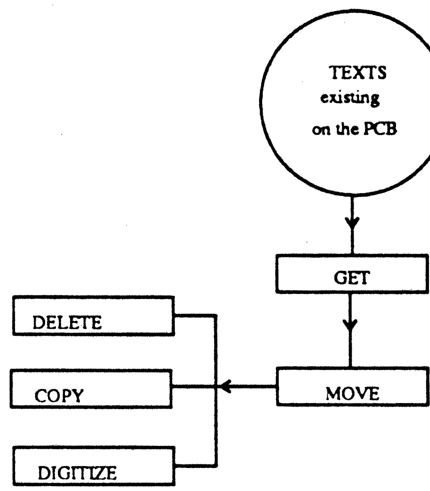
Further operations with restricted areas

1. move
2. copy
3. delete
4. rotate with <MMB>
5. transfer to the other layer of the layer pair with <RMB>, then place it with <LMB>
6. mirror with status entry "mirrored:Y" and <CR>
7. modify parameters:
   - by entering "Y" or "N" in the fields "segm.:", "vias", "comp." you may selectively determine for which components the area is to be a restricted area;
   - through the specification of "Y" or "N" in the field "outside:" you determine whether the area inside or outside the frame is to be the restricted area.

Note:

If unallowed components are at the position where a restricted area is to be defined (see parameters, point 7), it may not be placed. the warning "alien elements in restricted area" is output; by pressing <LMB> or <CR> a second time, you may enforce the placement, although the technology is violated.

TEXT



General

The TEXT function exclusively refers to text fields belonging to components or terminals.
It corresponds to the subfunction "TEXT" of the "mod. graphic" function for components and terminals.

Note:
With the function "mod. graphic" component graphics and their texts may be modified "globally" or "locally", whereas the function "TEXT" may only be used for "local" modifications.

With the function "TEXT" either existing component or terminal texts may be modified or further component or terminal texts may be generated.

Modify existing text

a. **get** (function "get")

> activate function "get"
> if applicable, specify the layer number where the text field is stored in "layer:"
> click the text field with <LMB> or specify the placement coordinate in "coord." and press <CR>

b. **move** (function "move")

> get the text field, the system defaults to function "move";
> you may now
- change the rotation (with <MMB> or through an entry in the status window)
- define a frame around the text (by marking "frame:" in the status window with a cross) or delete an existing frame (by deleting a cross in "frame:")
- change the size of the text (by entering the respective x and y values in "size:")
- change the position (by clicking the new position with <LMB> or by entering the new placement coordinate in "coord.:" plus <CR>;

c. **delete** (function "delete")

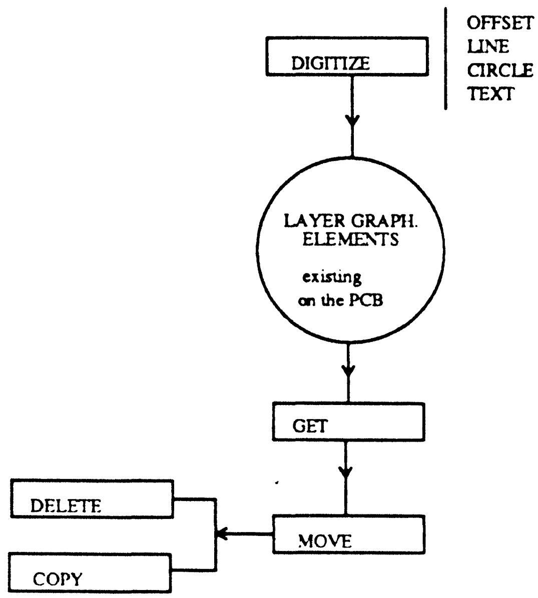
> get the element
> activate "delete"

Digitize additional text

In order to digitize additional text to a component or a terminal, an existing text belonging to the respective element has at first to be activated. Then the subfunction "digitize" may be activated.

> get the existing text field
> activate "digitize"
> specify in the status window
  - the text type
    with the keyboard (by specifying the text field number)
    or by activating it with <RMB>
    (by repeatedly pressing <RMB> any available text types may be called, if the cursor is positioned within the graphic window);
  - if text type:0 (=free text) has been specified, type the desired text in "text:";
  - if the text field is to get a visible frame, mark the field "frame:" with a cross;
  - if applicable specify the rotation in 1/10 degrees (e.g. "900") in "rotat." (may also be specified with the placement with <MMB>)
  - if applicable, specify placement and size by entering the coordinates in the first "coord." line and in "size:"
> <CR>
  or
> if placement and size have not been specified with their coordinates in the status window, set the starting point of the text field with <LMB>, open the text field to the desired size by moving the mouse and place it with <LMB>

LAYERS


DIGITIZE
OFFSET
LINE
CIRCLE
TEXT
LAYER GRAPH.
ELEMENTS
existing
on the PCB
GET
MOVE
DELETE
COPY

General

This function serves for generating or modifying layer-specific graphic elements (lines, arcs, texts)

You may also generate graphic belonging to the drill legend or modify component texts of the silk screen in a drill plan or silk screen generated with the CAM function (see CAM, chapter "drill plan" and "silk screen").

In addition to that you may copy any graphic generated on a master layer to any other layer by means of the "LAYER" function.

Generate graphic

a. enter in the status window
   > in "layer:" the layer number (according to the layer plan) where the graphic elements are to be digitized
   > in "lay.type"
   either
   "F" if the graphic is to be specified as layer-specific
   or
   "L" if those layers are concerned where the drill plan is stored and if the graphic is not to be specified as layer-specific but as drill legend-specific;
   or
   "S" if those layers are concerned where the silk screen is stored and if the graphic is not to be specified as layer-specific but as silk screen-specific;
   > if lines are to be digitized, enter the number of the track type to be used in "trk.type" (from the layer technology belonging to the layer)

b. then activate the function "digitize";
   you may now digitize the desired graphic with the functions "LINE", "CIRCLE" or "TEXT".

Note:
   Layer-specific graphic remains unchanged on the same layer after a new generation of the drill plan or the silk screen, whereas the legend-specific or silk screen-specific graphic disappears.

Modify existing graphic elements
(lines, circle segments, texts)

a. **get** (function "get")

> activate the function "get",
> if necessary, specify the layer number where the element to be activated is stored in "layer:"
> enter "L" or "S" in "lay.type" if a drill plan or silk screen layer is concerned or if drill legend-specific or silk screen-specific graphic is to be activated,
> click the element with <LMB>;

b. **move** (function "move")

> get the element, then the system defaults to "move"
> texts may be rotated (with <MMB> or via a specification in the status window) and /or a frame may be assigned to the text (by marking "frame:" with a cross in the status window) or an existing frame may be deleted (by deleting the cross in "frame:")

c. **delete** (function "delete")

> get the element
> activate "delete"

Copying the graphic from the master layer to another layer

How to procede:

> activate "layer"
> specify the number of the layer to which the graphic is to be copied in "layer:"
> specify the master layer in "master layer:"
> press <CR>

Note:

The PCB **border** which is digitized in the PCB technology menu "sizes" is visible on the graphic screen on all layers, but it is <u>not plotted</u> in the postprocessing, since in CIS it is automatically generated with track type 0 which is not considered by the postprocessing.
If you would like to have the PCB border plotted, you have to digitize the outline of the PC border (= restricted area contour) - digitized in the menu "sizes" - with an appropriate track type using the function "LAYERS/digitize".

Now you may transfer the PCB border generated as a layer-specific graphic from the respective layer to all the layers to be plotted; this is to be done as described under "Copying the master layer to another layer".

G R A P H I C

GENERAL

The graphic output of the layout on the screen is set up and controlled via a variety of parameters and functions.

The graphic functions and parameters may be called from the PCB/CAM main menu by selecting the function "graphic setup" or from the PCB modification menu by selecting the function "graphic". (Exception: the "pan on/off" function which only appears in the PCB modification menu)

When the graphic function is called, the following appears:
- in the menu window the graphic functions appear; they are selected via the mouse
- in the status window the main menu appears; the parameters are adjusted via the keyboard (the user determines which elements and layers are represented).

The following functions placed in the upper menu window are not only selectable from the graphic menu, but also directly selectable from the PCB main menu function "modification":
- "window"
- "origin"
- "refresh"
- "center"
- "grid on/off"

"pan on/off" is selectable <u>only</u> from the modification menu

When the following graphic menu functions are called, an additional menu appears in the status window for the parameter specification:
- grid functions (TEMP, DISP, DIGIT, PLACE-GRID)
- coordinate origin function ("0/0")
- function "colours"

The graphic chapter is divided into the following parts:
"global functions"
"grid adjustment"
"main menu"
"colours".

GLOBAL FUNCTIONS

work menu (in the graphic menu also via pop-up function "<==")

leads back to the work menu

main-menu (in the graphic menu also via pop-up function "main menu")

leads back to the main menu

pan on/off

If in the menu "modes" the panning mode is switched on (see chapter "modes" of the PCB part), the user may pan through the pixel memory after activating this function. The visible part of the graphic may be activated at the desired position with <LMB>, and with the <LMB> button pressed down the graphic window may be moved into positive and negative X and Y direction.
The size of the panning area is specified in the menu "modes" in the field "panning area".
The position of the graphic window within the panning area is displayed in the lower left corner of the graphic window.

window

serves for specifying a new window

Procedure:
Move the cursor to the position where the lower left corner of the window is to be placed.
Fix the corner with <LMB>. Then open the window with the mouse to the desired size. Fix the window size with <LMB>.

full view

Display of the overall view in the graphic window, i.e. of the whole work sheet (= counterpart to the "window" function).

refresh

Deletes the screen and newly displays the graphic (with the same window).

center

The crosshair position clicked becomes the new center of the graphic window.
Note that the window cannot be moved further than the lower or the left edge (the lower left corner is the coordinate origin).

grid on/off

The representation grid is switched on/off.
The work grid is not affected.
The representation grid may be represented dotted or solid.
The dots or lines are selected in the additional menu of the graphic function "colour" in the field "grid:" (1=solid, 2=dotted).
The distance is specified with function "disp-grid" (see below, "grid")
If the represetation grid is too fine, it is suppressed, since otherwise the graphic would be white.
If the grid is not needed, it should be switched off, since it has to be drawn again with every refresh (increases the refresh time).

restore

is similar to the "refresh" function, but the graphic content is not newly calculated, but is newly written with the help of the pixel memory (is faster).

Zoom

The graphic elements of the clicked area are scaled (enlarged or reduced).
Procedure:
Specify the point to be "zoomed" with the crosshair. This point is then scaled. Enlarge with <LMB>, reduce with <MMB> and finish the function with <RMB> or by clicking "zoom" a second time.

0/0

The following may be changed for the interactive work in the work menu:

a. the coordinate reference point which is "globally" valid independent of the objects modified (by specifying the X and Y distance to the absolute origin in the menu or by clicking the position with <LMB>).

b. the "current" coordinate reference point valid only temporarily for the work within the same object area (by specifying the X and Y distance to the absolute origin in the menu).

Note:

- The temporary coordinate origin is, above all, helpful for defining terminals to the components.
- The actual coordinate origin for the complete PCB data is still the production origin defined in "sizes" of the PCB technology; when the .adb file is stored, all the layout coordinates are stored in relation to this point.
- as long as the "global reference point" is not changed with "0/0" its coordinates are identical to those of the production origin.
- as long as the "current reference point" is not changed, its coordiantes are identical
  - > either to the production origin, if the function "mod. graphic" is not activated,
  - > or to the rotation point of the modified object, if the function "mod. graphic" is activated (exception: restricted areas)
- the "absolute" origin is identical to the lower left corner of the work sheet (see PCB technology description, chapter "sizes"); the coordinates of the "global reference point", the coordinates of the "current reference point" and the coordinates of the production origin refer to this point.

coos on/off

The X and Y coordiantes of the current crosshair position are displayed in the status window.

GRIDS

The grid points are the only coordinates for placing or routing elements.

In addition to the routing grid defined in the technology menu (function "placement & routing") to which the router is bound for routing tracks or vias, the following grids are provided for interactive work:

temp-grid (temporary work grid)

By changing the default values the work grid may be changed for the interactive work within one object area ("COMPONENT", "TERMINAL", "TRACK" etc.).

It is possible to specify different values for the X and Y value.

The default for the temporary grid is always the current work grid, i.e.

- for modifying <u>components</u> (except for the function "mod. graphic") it is the value of the placement grid (defined in the technology menu, function "placement & routing").
- for <u>any other objects</u> as well as for "mod. graphic" it is the value of the digitizing grid (default: same value like routing grid).

If a new object area is called, the temporary work grid is automatically set back to the work grid value valid for this object area.

The temporary grid is useful above all for defining components and terminals to the components.

The work grid is displayed within the work menu in every form in the right upper corner.

disp-grid (= display grid)

It is the display grid which may be switched on/off with "grid on/off" (see "grid on/off").

The display grid has been designed as a help to orientate for the interactive work; if it is set to the same values like the placement grid, then the grid points are the positions valid for placing components.

It is possible to specify different X and Y values.

The display grid may be represented dotted or solid. This is set up in the additional menu of the graphic function "colours" in the field "grid:" (1=solid, 2=dotted).

digit-grid (digitizing grid)

This is the grid for any digitizing actions (for routing tracks and for digitizing graphic elements of components, terminals, areas and layers).

The default is the value of the routing grid (defined in the technology menu, function "placement & routing").

Note:

> When tracks are digitized, the digitizing grid should be the routing grid or a multiple of it; when moved tracks are routed the digitizing grid has to be the routing grid or a multiple of it, since otherwise it is not possible to digitize correctly! (The movement always refers to the routing grid, and the system is always oriented to one single grid).

place-grid (placement grid)

is the grid for placing components.
The default are the values of the placement grid defined in the technology menu (function "placement & routing).

Procedure

When one of the grid functions is activated, a menu appears in the status window where the current values for the different grids are displayed.
The measurements (inch or mm=millimeters or 1/20"=1/20 inch) are defined in the menu "modes".

The current measurement is displayed in the status window under "MS". (When the measurement is changed, the system automatically calculates the new values).

The menu offers ten different default grid values which may be changed via the keyboard and which are selected via the numbers of the function menu by clicking these using the mouse.

Grid values deviating from the default values may be directly entered into the input field in the status window (e.g. in "disp. grid").

If a certain grid is to be changed, first activate the appropriate menu function (e.g. "disp. grid"); the appropriate grid field is displayed in the status window by an arrow. The desired values may now be entered by selecting the appropriate digit or by directly specifying it via the keyboard.

MAIN MENU

After the menu function "graphic" is called, the menu "PCB: graphic selection" appears in the status window where the PCB elements (components, terminals, segments, vias, etc.) and layers are defined.

a. in the first part of the menu **the layout elements to be represented are specified**
The possible specifications are listed behind the input fields for the elements.
The differentiated inputs allow a detailed selection of the layout elements to be represented.

b. The **possible inputs** for the detailed specification of the element parameters are listed on the right side of the menu.

Specifications:

X=all	any aspect of the element is to be represented (corresponds to the specification of all letters in parenthesis) - e.g. all components = fixed and variable components.
Y=free	representation of "free" restricted areas, i.e. restr. areas which are not assigned to any signal or component, when "restr. areas" are represented
C=comp.	representation of the components when "frame" and "text" are represented (i.e. representation of the component frames and texts).
S=sign.	yet to be implemented
P=pads	representation of vias/segments/texts belonging to pads when "segments", "vias", "texts" are represented.
T=term.	representation of the segments/texts belonging to terminals when "segments" or "texts" are represented.
F=fix	representation of the "fixed" components/ tracks /vias when "components", "segments" or "vias" are represented.

V=var.	representation of the "variable" (=not specified as "fixed") components/tracks/vias when "components", "segments" or "vias" are represented
U=proposals	representation of the proposals (proposals for the router) when "segments" or "vias" are represented.
H=heat trap	representation of the heat trap segments when "segments" are represented.
A=adjustments	for files converted from the .gph into the .adb format: representation of the adjustment marks when "terminals" are represented
G=glue spots	representation of the glue spots when "frame" is represented; for files converted from the .gph into the .adb format: representation of the glue spots when the "terminals" are represented.
Q=test spots	for files converted from the .gph into the .adb format: representation of the test spots when the "terminals" are represented
M=milling sp.	representation of the NC routing spots when the "terminals" are represented and representation of the NC routing contour when the "frame" is represented
B=via block	representation of the via block marks (see TERMINAL/"mod. graphic") when "restricted areas" are represented
Z=sheet	representation of the work sheet elements when "frame" or "text" is represented (i.e. representation of the sheet outline and the title block)
R=rout.	representation of the work sheet elements when the "frame" is represented (i.e. representation of the routing area outline)
O=outline	representation of the board contours when the "frame" is represented.

c. **layer:**

The middle menu part contains four columns with fields for the specification of the layers to be represented.
By marking the appropriate field with a cross layers entered are activated for the representation on the screen.
Whether or not an activated layer is visible depends on whether it is in the foreground or in the background (see parameter "top layers:" in the "colours" menu).
The colour with which the elements of a layer are represented depends on the assignment specified in the "colours" menu (see below).

The default of the first layer fields are the names of the signal layers defined in the layer plan (marked with a cross). As a default, the outer layers are in the foreground; the default colour of the elements on the component side is green, the default of the elements on the solder side is red (colour no. 2 and 1 of the pcb_palette-file).
If inner layers are represented in the foreground, green and red are also used as default colours (depending on whether the layer is the upper or lower side of a layer pair).
The default colour for docu layers is colour no. 80 (=grey) of the pcb_palette-file.

The two fields below the field "remaining" contain the component graphic docu layers (SHAPE-1 and SHAPE-0). If these two fields are marked with a cross, the layers entered are <u>always</u> represented in the colour no. 8 of the pcb_palette-file (default=white) and in the foreground.
Note:
If the component frames are not to be represented white, you have to
- deactivate the SHAPE layer (=delete the cross)
- enter the layer into one of the free fields of the first three columns and activate it (mark it with a cross), <CR>,
- call the function "colours" and assign the desired colour number to that layer, <CR>,
- activate "REFRESH",
- if desired, "toggle" with key F5 or F6 to bring the layer into the foreground.

In the free input fields of the first three columns you may add docu layer to be represented (by specifying the layer number or the layer name <u>according to the layer plan</u>).

If **"remaining"** is marked with a cross, all the remaining layers (which are not specified in the layer fields) are represented in the background (default colour black = colour 19 of the pcb_palette-file).

Notes:

- For PRISMA systems with UNIX standard operation system (3.2 or 3.5) applies: a maximum of 4 layers may be specified in the graphic menu under "layers:" and may be represented.
  Up to 4 layers may be represented in the foreground at a time.
  In order to represent another 4 layers, these have to be entered under "layers:" and have to be activated (REFRESH necessary).
- For PRISMA systems with UNIX OS4 operation system applies: up to 12 layers may be entered into the layer fields and may be marked with a cross; after the colour assignment the layers may be made visible by "toggling" with the F5 or F6 key.
  A maximum of 4 layers may be represented in the foreground at a time.

d. The last menu line contains the **component/pin search function.** By specifying the component name and, if applicable, the pin name and <CR> a refresh is automatically executed; the component and the pin are highlighted and are moved into the center of the graphic window.

e. The line above the component and pin search function shows the coordinates of the current window.

   Here, you may specify a window via the coordinates (with the keyboard) instead of specifying it with the mouse (with the "window" function). After the specification of the appropriate coordinates you may execute the refresh with <CR>.

COLOURS

Definition of Colours

The 83 available colours for the PCB graphic are defined on the lib/sys directory in the file "pcb_palette".
This file may be edited using the vi editor.

The pcb_palette-file has the following syntax:

colour no.	**red share, green share, blue share**	**colour designation (or element to be represented)**

The share of the three primary colours (red, green, blue) is defined by the appropriate numbers from 0.00 to 1.00:

```
0.0  -->  colour has no share
1.0  -->  colour is used to its full extent
```

Intermediate values are defined by the number behind the decimal point (= 1/10 units).

Examples:

```
green     -->  0.0, 1.0, 0.0
yellow    -->  1.0, 1.0, 0.0 (=red + green)
black     -->  0.0, 0.0, 0.0
white     -->  1.0, 1.0, 1.0
airline   -->  0.0, 0.6, 1.0 (=bluishgreen)
```

Note:
The following colours are reserved:
- colour no. 0 - for the background
- colour no. 17 - for the airlines
- colour no. 16 - for highlighting elements

Assignment of Colours and of Representation Forms to Board Elements

After the function "colours" is called in the graphic menu or via the pop-up, the menu "PCB: element representation & colours" appears where colours as well as 'forms' may be assigned to

- the layers to be represented (or to the tracks, vias, areas on them)
- the layer-specific elements (airlines and highlight)
- the 'frames' (component frame, text field frame and text, board outline, sheet outline).

a. Representation Form

In the first part of the menu the representation form of the elements is specified. The possible specifications are listed on the right side of the menu.
The possible entries are listed behind the appropriate input fields.

Explanation of some of the specifications:

C=cont. only the contours are represented

F=fill the element is filled

R=real only the minimum clearance specified in the menu "placement & routing" is considered in the graphic output

M=plus mark
the cross mark for rotation points (of components, terminals, copper areas) is displayed.

1/2/3/4 four different representation forms for lines

top layers:
number of layers which are to be represented in the foreground at a time:
1 -> only 1 layer in the foreground (beginning with the component side)
2 -> 2 layers (=1 layer pair) in the foreground (beginning with the two outer layers)
4 -> 4 layers(= 2 layer pairs) in the foreground (beginning with the 2 outer layer pairs)

The **F5** key toggles from the foreground to the inner layers of the board and the **F6** key vice versa.

grid: In this field you may specify the kind of the output of the representation grid:

1 -> solid
2 -> dotted (only the grid points are represented) = default

The output of the representatin grid (=DISP-GRID) is switched on/off via the menu function "grid on/off".

highl. techn. wrong highlighting technology violations (Y/N)

b. Colour Specification

The colour assignment is done in the lower part of the menu.

The colours are specified by the appropriate colour number in the pcb_palette-file.

Explanations:

Airlines and Highlight The colours defined in the pcb_palette-file for representing airlines and for highlighting elements are output only for information; they may not be changed in this menu.

frame Specify the colour number for the representation of the frames (component frames, text field frames), outlines (PCB outline, sheet outline) and texts.

remaining Specify the colour number for the representation of all the board elements which are not in the foreground (ref: "top layers"). The default is colour no. 19 (=black).

COMP SOLD etc. These fields contain the layer names specified in the graphic main menu under "layers:" within the first three columns and the colours assigned to these

Of a layer is in the foreground, the segments, vias and areas are represented with the colour assigned to that layer.
The default for the component layer is colour no. 2 (default=green), for the solder side colour no 1 (default=red) of the pcb_palette-file.
To the inner signal layers also colour no. 1 and 2 are assigned (depending on whether it is the upper or the lower layer of a layer pair).
The default for all the docu layers (except for the SHAPE layers) is colour no. 80 of the pcb_palette-file (default=grey).

Note:
For the representation of overlapping signal layer elements (e.g. through-holes or two-sided pads) applies: the colours overlap and result in an appropriately mixed colour (e.g. yellow=green+red).

F I N A L C H E C K

The "final check" checks the layout signals for

a. shorts
b. opens
c. drill hole collisions
d. terminals with disconnected graphics (containing "isolated" copper elements")
e. cross-hatched copper areas with disconnected elements (containing "isolated copper elements")
f. terminals the connection width of which does not correspond to the track diameter defined in the SCS part during the signal generation.

The check function(s) to be activated is/are activated by marking it/them with a cross.
The function is executed after <CR> is pressed.

The check may be executed for all signals or only for part of the signals.
The signal(s) may be selected by specifying the name in "signals:" and in "except" (wildcards are possible).

The signals specified in "except" are excluded from the check.

In addition to the error output on the screen, an error report can be output into a file onto the current home directory.
If an error report file is to be generated, "Y" has to be entered in "report (Y/N)".
The auto-created name of the error file can be changed by the user.

Error sources found are highlighted in the graphic output (sometimes the user has to execute a "refresh")

Notes:

a. A signal is free of interrupts only when all connections of the signal are connected by copper.

b. When drill collisions are listed, the collision coordinates are also listed.

M I S C E L L A N E O U S

After this function has been activated, a menu appears where the functions "report", "C-shell", "plot" and "measurement" are available.

report

After this function has been clicked, different list possibilities are made available. The single lists may be generated one after the other. In the status window you may specify whether the appropriate list is to be written as a file onto the tmp directory or whether it is to be directly output in a list window.

c-shell

This function opens a shell window ("shelltool-/bin/csh").

plot

With this function you may generate a hardcopy plot of the current screen graphic.
After the necessary data is specified in the status window plus <CR> the plot file is generated and is stored onto the **gdi**-directory.
From this file you may then generate the plot data with the appropriate postprocessing functions (see postprocessing description, point "output of plot data online/offline").

status window

name of plot file:	contains the name of the plot file to be generated, consisting of the name of the current .adb file, followed by the extension which has to begin with the letter "p", followed by the appropriate number.
scale:	contains the factor by which the plot is to be scaled (enlarged or reduced).
name of plotter:	Specify the name of the plotter to be used. The plotter specified has to exist in the Columbine library. Note: at present only pen plotters may be used for hardcopies.
window:	In these fields the coordinates of the current window are automatically entered.

If you would like to plot another window (not the current one), you may define this one either via a direct coordinate specification using the keyboard or via opening the window using <MMB>.

vias, components, airl., COMP, SOLD

In these fields plotter pens are assigned to the single elements or layers.

size:

has not yet been implemented

main menu

leads back to the main menu.

M O D E S

With this function a comprehensive menu is called where a variety of parameters for the interactive work in the layout are set. The specification is done by marking fields with a cross, by selecting certain letters, or by specifying the appropriate numbers.

Unit: (I)nch (M)m (C)=1/20":

By specifying the appropriate selection letter you define the measurement for the interactive work.
The system may be adjusted to inches, millimeteres, or to the 'old' CALAY measurement 1/20".

search range+-: x:_______ y:______

is the area around the crosshair which is searched for the appropriate layout element when the function "get" is executed.

Cursor:
element plus crossh.:_
plus terminals:_
digit.-cursor plus crosshair:_

By marking the field with a cross or by deleting the cross
- the crosshair representation for the activation of layout elements is switched on/off
- the terminal representation for the activation of layout elements is switched on/off
- the crosshair representation for the digitizing is switched on/off.

text: standard height:_______ automat. width:_______

is the default for the generation (digitizing) of texts

The specified height is used
- for all the name text fields generated by the system (e.g. component names)
- for the interactive generation (digitizing) of texts if no other heights are specified by the user in the appropriate menus.

"automat. width"
- marked with a cross: the width of the text is related to the width of the generated text field (width of the frame).
- not marked with a cross: the width is calculated system-internally in relation to the text height.

areas: auto-close:

For the digitizing of areas (copper or restr. areas) applies: if this function is activated (marked with a cross) the digitized track segment is automatically closed when the menu function "STOP" is activated.

digit:

auto-routing:_
gradients:_ (0=only X/Y; 1=X/Y+1:1; 2=no restr.)

the following defaults for digitizing tracks are set:
- switching on/off the "route-mode" (=routing of tracks when vias are moved) by marking the field with a cross or by deleting the cross
- setting the gradient allowed for the digitizing by specifying the appropriate selection number:
  0 = only orthogonal gradient allowed
  1 = orthogonal and 45° gradient allowed
  2 = any gradient allowed

The defaults may be changed in the submenu of the "TRACKS" menu for one digitzing action.

misc.:

RTI:_

This serves for switching on/off the real-time-integration (by marking the field with a cross or by deleting the cross).
Note:
With the RTI switched off, irreversible inconsistencies may occur between schematic and layout.

popups:_

This specifies whether the <RMB> key activates the popup or another function.
This switch is also possible with the <R11> key.

lowercase characters:

With this parameter activated the system differentiates between uppercase and lowercase characters, i.e. names or text fields can be generated, which contain lowercase characters.
Attention: if upper- and lowercase characters are used with the creation of user-defined names, the spelling has to be consistent with the one used with the genration of the element.

imm. delete:

With this option being activated, activated layout elements are immediately deleted without any query when the function "delete" is activated.
Note:

- Since there is no undo function for deletions, this option should not be activated.
- The default defined here can only be changed for components in the submenu of the "COMPONENT" function

autom.reorg.: _ yet to be implemented

standard-graphic-mode (P,W):_

Either the panning or the window mode is activated by means of the appropriate selection letter.

With the window mode the whole pixel memory, i.e. the "overall view" is output on the screen.
With the graphic function "WINDOW" a section may be defined and displayed; with the function "REFRESH" the graphic output is set back to the "overall view".

If the panning mode is switched on, an area of the size defined in "pan-area" is stored in a pixel memory in the background. With the pan function activated, the section visible on the screen may be moved through the panning area, and the appropriate section is displayed on the graphic screen. (see function "Pan on/of" of the chapter "graphic setup" of the PCB part).

auto-panning-size:_ or fullsize:_ yet to be implemented

panning area (pixels) in x:____ in y:____

specification of the panning area in pixels.
Behind the input fields of the pan area the lower and upper values are specified in parenthesis.

preference-value:____ pixels be adequate to ____ mm

yet to be implemented

P C B   T E C H N O L O G I E S

GENERAL

The first step for the creation of an .adb-file in CIS is the read in of an existing standard technology file where the necessary technology parameters are stored in the .adb format. The current CALAY standard technology is stored on the directory **projects/std/tec** in the file named **startup.adb.**

Some of the technology parameters (tracks, vias, clearance and copper height) are defined layer type specifically in so-called "layer technologies" (see layer technology). The parameter values of a certain layer technology only apply to the layers to which it is assigned, and not to the whole board. In addition to that there are technology parameters which are not layer-specific, but which apply to the whole board (e.g. routing grid, drills, etc.).

1. Defaults of the startup.adb

The technology of the startup.adb contains the following parameter defaults:

- grid (routing grid, placement grid for components, via routing grid, digitizing grid)

- sizes for the work sheet, board and routing area (the sizes refer to the euro-format)
- starting coordinates for board and routing area
- coordinates for the production origin
- the pre-digitized board outline

- layer technology (=layer specific parameters):
  > global clearance
  > copper height
  > track technology
  > via technology
- layer plan
- drill technology

The startup file contains 9 different layer technologies which are assigned to the single layers in the layer plan - according to the specific layer type (signal layer, solder resist layer, drill plan layer etc.).

By changing this standard technology the user may generate one or more user-specific standard technologies and store these in the respective .adb files.

2. Generation of user-specific standard technologies

The definition of own technology standards should be done prior to the product on oriented work in CIS.
Take your time since there is considerable work to be done.

1. First collect all user-specific technology data you use every day.
   Try and find out which kind of standard layer technology is the most convenient for you:
   - a comprehensive 'allround' layer technology (like, e.g. DT1 or DT2 in the startup.adb), which may be used independently of the grid (due to the high number of definitions in small steps),

   or
   - several 'grid-specific' layer technologies which are used for special routing grids and, therefore, contain grid-specific definitions,

   or
   - 'allround' as well as 'grid-specific' layer technologies like contained in the startup.adb.
     (If you work with technology variations which often change it is sensible to also define an 'allround' technology on the basis of the layer technology standard DT1 and/or DT2).

2. Read in the startup-file.
   For reading in the CALAY standard technology file enter "startup" in "file" and "Y" in "read standard technology" and <CR>.

3. After reading the file, activate "LAYOUT" and in the layout menu the function "technology". In the technology menu then call the area where you would like to make the changes; you may enter the changes in the respective menu.

   Sensible definition sequence:

   a. Placement & Routing
   b. Sizes
   c. Drills
      Note: If you would like to use drill plan symbols which are not defined in the startup.adb, these have to be generated before the definition of the drill technology (see chap. "Drills").
   d. Layer Technology
   e. Layer Plan
   f. Vias in the different layer technologies
   g. Tracks in the different layer technologies
   h. reference component

Note:
After entering the changes in one menu, you should delete the superfluous definition lines.

4. After modifying the menus for tracks and vias for all layer technologies, you should delete all superfluous layer technology definitions.

5. Store the file as an own standard under an easy-to-identify and technology-specific name.

6. Shift or copy your standard technology from the .adb-directory to the standard technology directory projects/std/tec (with the UNIX command "mv" or "cp").

Note:

- If you would like to continue to use the 'old' technology standards (tec20, tec40, tec50, tec60, tec70, tec80) which are valid up to update 2.07, just read them into CIS; the system 'recognizes" the 'old' data format and converts it into the new layer technology format with the program **IOCONV**; Of course, you may also change these converted standard technologies and generate your own standards from these.

- If the default values of the startup-technology are changed (e.g. routing grid, clearance, track widths, via widths, etc), restrictions dictated by the router have to be adhered to which will be described under the respective parameter description.

3. General hints for the definition

In order to avoid problems with routing or inaccuracies in the router-internal representation of the technology sizes, the following hints should be considered for the definition of user-specific technologies:

a. in the track and via technologies not more than 16 types may be defined as routable, i.e., may be without attribute "N" (for "not routable"), since the router can only route a maximum of 16 types.
b. the routable track types have to be defined with radius=1/2 width; if tracks are defined with radius=0 or with R< 1/2 width, the router may violate the clearance at the segment ends when routing these tracks.

c. Routable vias may be exactly represented router-internally, if they have the same parameters (width, radius, additional clearance) like a routable track of the same layer technology.

d. If a routable track type is used neither in the netlist nor for digitizing, the attribute "N" should be entered in the definition line. (This enables the router to calculate its internal representation symbols with a better graduation).

e. Tracks and vias shifted by half a grid into negative x and y direction are always represented by an even number of grid cells.
This results in a router-internal minimum limit of one grid cell for the widths of shifted elements.

Note:
A control of the router-internal representation of the PCB technologies for vias and tracks is provided by the .ldr-file generated during the routing process of the .adb-file (see chapter ".ldr-file" of the router description).

4. Hints for Working with the Menu

1. After calling a menu, you are always in the edit mode (also after executing a function); the cursor is at the beginning of the first definition line.
Use the arrow keys of the right key block to move the cursor for menu entries (stepwise with keys R10 or R12 and blockwise with the keys R8 or R14).
The cursor may also be positioned with the mouse.
Set the cursor onto the input field and press <LMB> - the cursor is placed onto the beginning of the respective input field.

2. As a rule, <CR> only confirms the entries, but does not store them.
Exception:
In the menus "Placement & Routing" and "Sizes" a <CR> stores the entries! Leave these menus with <ESC><ESC>.

3. In all other menus (except for "Placement & Routing" and "Sizes") the data are only stored with the function "Save File".

4. <ESC><ESC> interrupts the menu editing;
-> if no entries have been made, you get back to the technology selection menu;
-> if entries have been made, the query "ignore changes" appears; if you answer with "Y", you leave the menu; if you answer "N", you remain in the menu and may continue your work.

Exception:
the menus "Placement & Routing" and "Sizes", see point 2!

PLACEMENT & ROUTING

```
PCB: modify technology            Grid: 0.653/0.653   MS:  mm        [985]

(P) plac. & routing  (S) sizes               (Y) layer-plan
(L) tracks           (R) reference - comp.   (V) vias
(D) drills           (T) texts                            fct:  P

 grid-base      1     /     33600        routing - base :  1" / 960

 routing-grid   x: 0.6350  y: 0.6350   offset       x: 0.0___  y: 0.0___
 placement-grid x: 1.270   y: 1.270    offset       x: 0.0___  y: 0.0___
 via-grid       x: 0.6350  y: 0.6350   offgrid routing: N

 clearance      x: 0.2903  y: 0.2903   copper height  : 0.0348        (mm)
 buried vias      : _
```

Menu Description

Fct. ("function")

By entering a function short form - without <CR> - the respective technology menu is read in (may as well be called via the mouse); if, for example, you enter "L", the menu for the definition of the track technology is read in.

Grid base

Specifies the minimum unit for the system internal calculation. This specification is only displayed for your information.

Routing base

The routing base is the smallest coordinate distance considered by the router. This unit may be changed.

The smallest routing base is 1/6000 inch, the biggest 1/100 inch. 1/960 inch is the standard routing base.

The routing grid must always be an integer multiple of the routing base with the multiplier (M) fulfilling the following condition:

$$2 \leq M \leq 50$$

If this is not the case, the specified routing base value will automatically be rounded off so that this condition is fulfilled.

Example:

Routing base: 1/960"; routing grid: 1/70";
i.e. M = 1/70 : 1/960 = 13.7
Change: routing base: 1/980" (M = 1/70 : 1/980 = 14).

Note:

- The maximum size of the work sheet (measurement unit in X and Y direction) which can be constructed with CIS depends on the routing base and can exactly be determined by multiplying the routing base by 32767.

Examples:

1/960 inch resolution --> 32767 X 1/960" ≈ 34" ≈ 866 mm
1/100 inch resolution --> 32767 X 1/100" ≈ 327" ≈ 7320 mm
1/6000 inch resolution --> 32767 X 1/6000" ≈ 5.5" ≈ 138 mm

Routing grid:

If a shifting does not exist, the tracks are routed along the grid lines and the vias are placed onto the grid points.
The routing grid has always to be an integer multiple of the routing base. If this is not the case, the routing base will internally be changed into a suitable value that fulfills this condition.
According to the technology and to the coordinate unit, the grid unit to be used by the router is automatically entered in "Grid" (e.g. "0.635" for 1/40" technology and unit mm). The table shows the current routing grid values.

unit grid	**mm**	**in.**
1/20"	1.27	0.050
1/40"	0.635	0.025
1/50"	0.508	0.020
1/60"	0.423	0.0166
1/70"	0.362	0.0142
1/80"	0.317	0.0125

Via grid:

By entering the grid element spacing (in the respective unit --> ref. to the function "Modes"), you can specify the distance (in grid cells) between the vias.

If you have a routing grid of x: 0.635 y: 0.635 and enter x: 1.27 y: 1.27 for the via grid, the router can at most place a via on every 2nd grid (for instance, in order to provide that tracks can still be digitized between the vias).
If the specifications in "Routing grid" and "Via grid" are equal, a via can be placed on every adjoining routing grid.

Attention:

The via grid does either have to be equal to the routing grid or has to be an integer multiple of it!
At present, the system does not check whether this condition is fulfilled!

Placement grid:

Since routing is bound to the grid, components should be placed on a grid which is equal to the routing grid or an integer multiple of it.
A disadvantageous choice of the routing grid may lead to a bad routing result since pins may be placed off-grid. Off-grid pins (i.e. pins which are not placed on a grid point) may effect that tracks cannot be digitized between them.
The most sensible grid for the placement of components is 1/10" since this makes all standard routing grids possible.

	sensible grid for the placement					
routing grid	1/10"	1/20"	1/40"	1/50"	1/60"	1/70"	1/80"
1/20"	X	X					
1/40"	X	X	X				
1/50"	X			X			
1/60"	X	X			X		
1/70"	X					X	
1/80"	X	X	X				X

In order to get an optimal bus routing, you should take care that the distance between components of memory structures is not larger than 0.3 inches (= 7.62 mm).

Offset:

By entering the respective values, you can define a shift of the placement grid in relation to the routing grid.
This is useful if elements that are 'on-grid' in the routing grid are 'off-grid' in the placement grid.

Shifting the routing grid has not yet been implemented.

Offgrid routing: Y/N

When entering "Y", the complete routing area with all elements will be shifted by half a routing grid into the negative X and Y direction.
This is necessary for the 1/60" routing in order to route 2 tracks of the width 0.21 mm between 2 IC-pins with a maximum diameter of 1.53 mm.

Clearance:

This field contains the value for the global clearance defined in the respective layer technology.

Copper height:

This fields contains the value for the copper height defined in the respective layer technology.
Based on the copper height and the track width, the track diameter is calculated (ref. to "SCS: Signals & connections"). This parameter is relevant to the simulation.
At present, the standard value 0.0348 mm appears here.

Buried vias:

This specification only applies to the interactive digitizing of tracks, not to the routing. By marking this field with a cross, you specify that buried vias shall be used during digitizing.

Operation Instruction

The data entered in the menu are stored with <CR>. Then enter <ESC><ESC> to leave the menu.

SIZES

```
PCB: modify technology          Grid: 0.635/0.635   MS:  mm      [775]

sheet size     x: 210.8003   y: 150.7996     boardname: ____________

pc-board:      x: 25.40___   y: 25.40___   /   x: 105.4003  y: 124.460

rout.-area     x: 25.40___   y: 25.40___   /   x: 105.4003  y: 124.460

prod.-origin   x: 25.40___   y: 25.40___

digit board shape: N
```

In this menu, the specification of the work sheet and routing area size and the digitized board outline is relevant to the router.

The size specifications are entered in the status window. Reference point for all coordinate specifications in this menu is the origin of the sheet.

Operation Instruction:

To store the changes of the size specification in this menu, press <CR>. Leave the menu with <ESC><ESC>.

Description of the Menu

Fct: (stands for "function")

By entering a respective function short form - without <CR> - the respective technology menu is called (may also be called via the mouse); for example, "L" calls the menu for the definition of the track technology.

Sheet size:

When specifying the work sheet sizes, the maximum size considered by the router must not be exceeded.
The maximum size of the work sheet in x and y direction depends on the routing base and can exactly be determined by multiplying the routing base by the number 32767 (ref. to page 20).
The sheet frame is displayed automatically on the graphic screen.

Board name:

Here, you may enter a user text for characterizing the current PC board. This text will always appear in this field when calling this file again.

PC board:

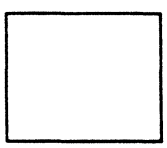
By specifying the coordinates of the origin and the upper right corner of the board, you determine the PC board size here.

Attention:
The contour of this area does <u>not</u> appear on the graphic screen!
If you would like to generate a visible board outline which can also be plotted in the postprocessing, you have to digitize it onto the respective layer(s) with the function "LAYER" in the modification part (see there) - independent of the sizes specified under "PC board" of this menu.

Rout.-area:

Here, you specify the routing area by entering the coordinates of the origin and the upper right corner of the routing area. The frame of the routing area is automatically displayed on the graphic screen.
The maximum size of the routing area in x and y direction depends on the routing grid and on the form of the routing area:

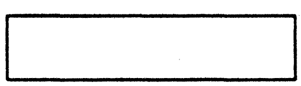
- square area



y = 2048 grid cells

x = 2048 grid cells

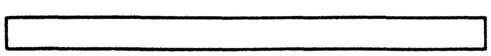
- rectangular area



y = 1024 grid cells

x = 4096 grid cells

- long rectangular area



y = 512 grid cells

x = 8192 grid cells

Attention:

Components cannot not be placed outside the routing area, otherwise the file cannot be routed (the program would be aborted with the message "Element outside area").

Prod.-origin:

Coordinate specification for the production origin.
This point is the **coordinate origin** for the interactive work in CIS-PCB as well as for the postprocessing and the coordinate display (when using the graphic menu function "Coos on/off").

Digitize border:

When entering "Y" and <CR>, the system defaults to "Mod. graphic -digitize" of the work menu.

If a standard technology is read in for generating an .adb-file, the board shape is already digitized - it appears as the inner frame on the screen.

This predefined border represents a blocking area and may be changed to the user's requirements by getting and deleting the respective lines first and digitizing the desired shape then.

Notes:

- When digitizing, take care that the board outline is perfectly closed (check with "STOP")!
  If the outline has not been digitized correctly, and if you press <CR>, the system queries "ignore errors?" - with "Y" the file will be stored despite the errors and you will leave the menu; with "N" the file will not be stored and the function "mod. graphic" remains active for the necessary corrections.

- The track segments representing the board outline are automatically transferred to all layers.

- Since track type 0 is used when using the contour and since this track type is not considered by the postprocessing, the board outline does not appear in any layer plot.

- The area <u>outside</u> the PCB border is a <u>blocking area</u> for the router.

- When changing the values in "PC board" do not have any influence onto the PCB border.

LAYER TECHNOLOGY

General Hints

With the new CIS technology concept the technology is no longer globally defined for the whole board - except for the drills - but separately for each layer type.
This means that the PCB technology is a combination of several specific layer technologies and one common drill technology.

"Layer technology" means all the specific parameters relevant for the layout generation of a certain layer type.

By assigning a layer technology to a PCB layer in the layer plan the respective layer technology is transferred onto that layer.

The definition of a layer technology comprises the following parameters:

a. technology name
b. global clearance
c. copper height
d. layer-specific track technology
e. layer-specific via technology

In the menu "layer technology" only the first three parameters are defined; this parameter constellation is the basis for a layer technology definition.

This 'rump' definition is supplemented by the appropriate track and via parameters in the menus of the PCb technology functions "tracks" and "vias" (see below).
(In these menus the name of the layer technology to which the parameter specification is assigned has to be specified for the definition of the track or via types.)

Description of the Menu

When generating a new .adb-file based on the standard file startup.adb, the following default definition appears on the first screen page when calling the function "layer technology":

```
PCB: Lagentechnologie aendern     Raster: 0.635/0.

Fkt: +   Technologie: ________________

Tec      Name          Min-Dist   Hoehe     Referenzeb
Nr.
-------------------------------------------------------
0   DT1__________      0.3001     0.0340    D5,GLUE-0,
1   SR1__________      0.3001     0.0348    RESIST-0,R
2   SR2__________      0.3001     0.0348
3   T20__________      0.3001     0.0348
4   T40__________      0.3001     0.0348    COMP,SOLD
5   T60__________      0.3001     0.0348
6   T80__________      0.3001     0.0348
7   DT2__________      0.3001     0.0348
8   DRILL1_______      0.3001     0.0348    DRILL
9   SHP1_________      0.3001     0.0348    SHAPE-0,SH
```

If required, this definition may be changed
-> by inserting additional 'rump' technology definitions (which have to be supplemented by further parameter definitions if need be)
-> by modifying existing 'rump' technology definitions through new field specifications
-> by deleting existing 'rump' technology definitions (automatically implies the deletion of the complete layer technology).

In an .adb file up to 32 'rump' technology definitions can be made (Tec no. 0 to 31), to which the track and/or via technologies can be assigned if required.

Fct: ("function")

Various functions may be called through entering the respective function short form.
These functions may as well be called via the popup-menu (to be activated with <RMB>).

Note:
The entry via function short forms does not require any <CR>! Any necessary specification (e.g. technology number or technology name for entries or deletions) have to be made prior to the function short forms (or prior to activating the function in the popup-menu)!

Following please find the function short forms, their functionality and their counterparts in the popup-menu.

+ (popup: "+")
pages down (if there is at least one more definition which is not visible on the screen)
one screen page may contain up to 7 definition lines

- (popup: "-")
pages up

I (popup "Insert")
inserts a new technology

precondition:
-> Specification of the technology number or technology name in "technology".

result:
if a technology name has been specified, it appears under "name", the technology number is automatically generated.
if a technology number has been specified, it is entered at the convenient position, the system now expects a technology name and the global parameters (clearance and copper height).

D (popup: "Delete")
deletes an existing technology

preconditions:
-> specification of the technology number or of the technology name in "technology:"
-> the technology to be deleted may not be assigned to any layer in the layer plan (i.e. existing assignments have to be deleted in the layer plan);

result:
all entries are deleted in the respective line;
if the layer technology also contained definitions for tracks, vias or drills, these are also deleted.

C (popup: "Check")
carries out a check of the current 'rump' technology definitions;
if, for example, a technology has not been assinged to a name, just to a number, a respective error message is output;
if no errors are found, the message "no technology violation found" appears.

W (popup: "Write")
stores the current technology definitions

at first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);
technology definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the technology definition;
this is to undo the changes made up to this point in the 'rump' technology;
the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the specification of the elements, layers, etc. to be represented

$ (popup: "<==")
leads back to the menu technology selection

/ (popup: "main menu")
leads back to the PCB or CAM main menu (according to the main menu the technology function has been called from)

Technology:

Here, the layer technology to be processed (= to be generated, to be deleted, to be edited) is specified through specification of the technology number or the technology name.

If a technology is to be generated (function "create") or to be deleted (function "delete") the specification has to be made prior to activating the function and without <CR>.

In order to edit the desired line directly (if, for example, it is not visible on the screen), specify the technology, then enter <CR>.
Note:

Since, as a rule, you are always in the edit mode, and since the cursor may be freely moved, calling the line to be edited through the specification in "technology" is only sensible if the definition line is not visible on the screen.

The technology name may contain up to 16 characters.

An already existing technology name may also be abbreviated (e.g. "DEF" instead of "DEFAULT-TEC").

Tec No.

Here, the technology number is entered (possible numbers 0 to 32).

Name

In order to clearly define a technology, a user name has to be generated for the technology (either by entering the name in "technology" during the generation, or by editing).
The name may consist of up to 16 characters.

Min-Dist

This value defines the global clearance between graphic elements (tracks, pads, vias, text fields) as well as between graphic elements and routing area or PCB outline for this layer technology.
This clearance then applies to all layers to which this technology has been assigned in the layer plan.

Note:
- In order to be able to route segments of the smallest track type on every grid line, the user should see to it that
  clearance + smallest track width ≤ routing grid
- a subsequent smaller definition of the clearance may result in short circuit messages if a technology check is carried out.
- For the single track and via types an **additional clearance** may be defined in the respective technology menus; this is added to the basic clearance when routing the respective track or via type.

Height

Here, the copper height is defined for this certain technology.
This value applies to all layers, to which this technology has been assigned in the layer plan.
The track diameter is calculated on the basis of the copper height and the track width (see the parameter "A" in the menu of the SCS functoins "SIGNAL/create" and "SIGANL/mod. param." as wel as the parameter "trk.-diameter" in the menu of the PCB function "SIGNALS/mod. param."; this parameter is relevant for simulation.
At present, the standard value of 0.0348 mm is output here.

Referencing Layers

Here, the layer names appear to which this technology has been assigned in the layer plan.

Remarks

At present, this field is not evaluated.

LAYER PLAN

When generating a new .adb-file, the layer plan for the 'physical' layers of a two-layer-board (here=signal layers) appears in the status window when calling the function "layer plan". When entering "D" in "layer types" in the menu for the 'physical layers' the first page of the appropriate docu layer plan appears.

```
PCB: modify layer plan          grid: 0.635/0.635   MS:  mm        [772]

fct: +  layer: __________     layer types: P       insert pos.: __

lay.  name      technology  type  rou  att  reference layers remarks
No.                               lay.
--------------------------------------------------------------------
0   COMP_____   DEFAULT-TEC  COM  1__  AP   SHAPE-1,RESIST-1 __________
1   SOLD_____   DEFAULT-TEC  SOL  0__  AP   SHAPE-0,RESIST-0 __________
```

```
PCB: modify layer plan          grid: 0.635/0.635   MS:  mm        [772]

fct: +  layer: __________     layer types: D       insert pos.: __

lay.  name      technology  type  du   att  reference layers remarks
No.                               al
--------------------------------------------------------------------
17  SHAPE-0__   SHP1______   SHP  18   A_   SOLD________   __________
18  SHAPE-1__   SHP1______   SHP  17   AP   COMP________   __________
19  RESIST-0_   SR1_______   RES  20   AP   SOLD________   __________
20  RESIST-1_   SR1_______   RES  19   A_   COMP________   __________
21  DRILL____   DRILL1____   DRL  21   AP   ____________   __________
22  D5_______   DT1_______   DOC  22   A_   ____________   __________
23  GLUE-0___   DT1_______   GLU  24   A_   SOLD________   __________
24  GLUE-1___   DT1_______   GLU  23   A_   COMP________   __________
25  D8_______   DT1_______   DOC  26   A_   ____________   __________
```

This default layer plan may be changed according to the user's requirements by the respective entries.

-> through inserting additional layers and through the assignment of respective layer technologies and parameters

-> through modifying existing layer definitions by writing over the entries

-> through deleting existing layer definitions from the layer plan.

It is possible to define up to 255 layers for one .adb-file. In the 'physical' part, a maximum of 32 signal layers - which have to be combined to 16 layer pairs - may be set "active" (i.e. may have the attribute "A"), since the router can only handle a maximum of 16 layer pairs.

The two outer layer are always a **layer pair.**
Going from the component side towards the solder side, two layers in the 'physical' layer plan form one **layer pair.**
Vias valid only between two layers, are called **buried vias.**

Menu Description

Fct. ("function")

By entering the respective function short forms, various functions may be called.
These functions may also be called via the popup-menu (to be activated with <RMB>.
Note:
Entering function short forms does not require any <CR>.
Possible specifications (e.g. layer number or layer name for entries or deletions) have to be made prior to entering the function short forms (or prior to activating the function in the popup-menu).

The following describes the function short forms, their functionality and their counterparts in the popup-menu.

+ (popup: "+")
pages down (if there is at least one more layer definition which is not visible on the screen);
one screen page comprises 7 definition lines;

- (popup: "-")
pages up

I (popup "Insert")
inserts a new layer

precondition:
-> Specification of the layer number or layer name in "layer".
-> when inserting 'physical' layers the insertion position has to be specified under "insert pos." (see "insert pos.")

result:
the new layer is inserted at the respective insertion position (with 'physical' layers) or in the first free line (with docu layers). The cursor is at the beginning of the line where the new layer is to be defined. The necessary assignments are made through the respective parameter specifications.

When inserting into the 'physical' layer plan part, the 'new' layer is inserted instead of the one specified under "insert pos."; the original layer and all following 'physical' layers are shifted down one line.

D (popup: "Delete")
deletes an existing layer definition

preconditions:
-> specification of the layer number or the layer name in "layer:"
-> deletion of the entries in the respective layer definition line.
result:
the definition line for the specified layer definition disappears from the screen, the following definition lines are shifted upwards one line.

C (popup: "Check")
checks whether the current layer plan definitions are correct; if an unvalid type or a non-existing technology has been specified, or if an unvalid layer assignment or a wrong routing layer assignment has been made, an error message appears.
if no errors are found, the message "no technology violation found" is output.

W (popup: "Write")
stores the current layer plan definition
At first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);
layer plan definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the layer plan;
this is to undo the changes made up to this point in the layer plan
the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the specification ofthe elements, layers, etc. to be represented;

$ (popup: "<==")
leads back to the menu technology selection

/ (popup: "main menu")
leads back to the PCB or CAM main menu (according to the main menu the technology function has been called from)

layer:

Here, the layer to be processed (= to be deleted or to be edited) is specified through specification of the layer number (defined in "lay.no.") or the technology name (defined in "name").
An existing layer name can also be abbreviated (e.g. "SOL" for "SOLDER") if the abbreviation implies a definite relation to a name.

The layer name may contain up to 16 characters.

If a layer is to be generated (function "create") or to be deleted (function "delete") the specification has to be made prior to activating the function and without <CR>.

If a layer definition is to be edited, a <CR> is necessary after the layer specification.

layer types:

Through entering the appropriate short forms you get into the signal layer part or the docu layer part of the layer plan.
P='physical' layers (signal layers, planes, insulation layers, substratum layers)
D= documentation layers

Insert pos.

This field is only valid if combined with the insert function. Here the position where a new layer is to be inserted is specified in terms of the layer number or the layer name. When inserting a new layer, the original layer plus the other layers are shifted downwards by one line.
The following only applies to docu layers:

At present, it is not possible to make any entries in "Insert pos.". The new layer is placed onto the first free position (corresponds to the first unused layer number), (as a rule, this is the first free line at the end of the docu layer plan).

lay no.

Here, the layer number is entered. Numbers between 1 and 255 may be used.
Number "1" can only be assigned to the solder side, number "2" to the component side.

Name

In oder to define a layer in the layer plan, a user-name has to exist in this field.

Technology

Here, a technology name defined in the layer technology has to be assigned.

type

Here, the correct type key for a definite identification of the layer has to be entered.
The keys to be used under "type" are defined as follows:

```
physical layers
     signal layers:
          COM  -->  component side
          SOL  -->  solder side
          SIG  -->  signal inner layer
          PLN  -->  copper layer (plane or "splitted" plane)
     production layer
          ISO  -->  insulation layer
          SUB  -->  substratum layer

docu layers
          SHP  -->  for shape layer plan
          SSC  -->  for silk screen
          RES  -->  for solder resist plan
          DRL  -->  for drill plan
          PST  -->  for SMD paste plan ("reflow"method)
          GLU  -->  for glue point plan
          TXT  -->  docu layer without specification
          DOK  -->  docu layer without specification
```

When assigning types to 'physical' layers the user has to see to it that
-> the "COM" layer is always 'on top',i.e. that it is in the first definition line;
-> the "SOL" layer is always 'at the bottom', i.e. that it is in the last definition line.

rou. lay. (only for signal layers)

Specifies the signal layer number the current layer is to be routed with.
For signal layers this information is indispensable for the router!

Also when routing tracks interactively the default for layer changes (with <MMB>) is the layer assigned here.

For signal layers to which a pair layer cannot (or is not to) be assigned the own layer number has to be entered. For ISO and PLN layers the pair assignment is omitted - "0" is automatically entered.

As regards the assignment it has to be seen to it that for each of the layers to be routed together the other layer is specified in "rou.lay.".

Note:

The layers to be routed together may belong to different layer pairs, but have to have the same layer technology. In such cases the vias going through all layers are to be used, since "from-to" vias cannot yet be handled by the router and buried vias only apply to single layer pairs.

dual (only for docu layers)

Specify the layer number of the docu layer which is to be used as the mirroring layer of the current layer. This information is indispensible for the generation of two-sided graphics. For docu layers to which a "dual" is not to be assigned, the own layer number has to be specified. For the assignment please see to it that for each of the "dual"s which belong together the respective counterpart is entered in "dual".

att

This field serves for assigning attributes to the layers.
At present the following attributes may be assigned:

P marks a layer as being "producable" for the postprocessing; without this mark the layer cannot be handled by the postprocessing.

A marks a layer as being "active" in the layer plan; if nothing is specified the layer is ignored by the system;
this might serve for changing an 8-layer-board into a 6 or 4-layer-board.

Note:

At present the deletion of the "active" identifier does <u>not</u> yet ignore the layer!
Layers which shall be considered by the system have to have the activated "A" identifier.

reference layers

- In the "physical" part of the layer plan, for layer 2 and 1, this field contains the layers assigned to layer 2 and 1 (component and solder side) in the documentation part of the layer plan.

- In the docu layer part enter the names of teh assigned outer layers in this field for those docu layers which need a definite assignment (e.g. SHAPE, RESIST, PASTE, SILK docu layers).

remarks
This field is not yet evaluated.

TRACK TECHNOLOGY

With the new technology concept a track technology is always bound to a certain layer technology.

After activating the function "Tracks" in the menu "Technology" of the PCB part, the first 'page' of the track definition of the default layer technology DT1, if the startup.adb file has been read in at the beginning.
This track technology standard contains comprehensive definitions (the values are defined in 1/10 mm steps) and has been planned as an 'alround' track technology set convenient for all grids.

```
|PCB: modify track technology                    grid:  0.635/0.635  MS:  mm   [990] |
|Fct: +   track w.: 0__  technology : DT1________                                    |
|                        layers     : D5,GLUE-0,GLUE-1,D8,D9,D10,D11,D12,D13,D14,D15 |
|                                                                                    |
|trk. width   radius   add.dist. diameter  att  via    remarks                       |
|no.                                                                                 |
|= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = |
|0    0.0__   0.0___    0.0__   0.0___     N_   0__    ____________________________ |
|1    0.0998  0.0499    0.0__   0.0___     _    1__    ____________________________ |
|2    0.2003  0.0998_   0.0__   0.0___     _    1__    ____________________________ |
|3    0.3001  0.1497_   0.0__   0.0___     _    1__    ____________________________ |
|4    0.3999  0.1988_   0.0__   0.0___     _    1__    ____________________________ |
|5    0.4997  0.2487_   0.0__   0.0___     _    1__    ____________________________ |
```

In order to call the track technology definition for another layer technology the name of the appropriate (defined!) layer technology has to be entered in "technology" with a <CR>.

If after the generation of a new layer technology (see function "layer technology") the menu for the track technology is called, and if the name of the nwe layer technology is entered in the field "technology" with a <CR>, the definitions entered in the track menu appear for the new teck technology.
Now, you may modify these track definitions.
After the changes, the data have to be stored. Thus, the changed track technology is part of the newly defined layer technology.

An existing track technology may be modified through
-> inserting additional track definitions
-> modifying existing track definitions by over-writing the entries
-> deleting existing track definitions

A maximum of 255 track types may be defined.

Since tracks also serve for representing SMDs, the track ends may be defined in different forms (see "radius").

Note:

In order to prevent wrong definitions the system checks whether

a. every change of an attribute in a track type is automatically transferred to that track type in all layer technologies;

b. every track type definition newly generated by the user with the "insert" function is automatically transferred to the track technology of all existing layer technologies;

c. a track type deleted in a layer technology by the user is automatically deleted in all existing layer technologies.

Description of the Menu

fct. ("function")

Various functions may be called by entering the respective function short forms.
These functions may also be called via the popup-menu (to be activated with <RMB>).
Note:
The entry via function short forms does not require any <CR>! The specification of the track number for deleting or inserting has, therefore, to be made before entering the short forms (or before activating the function in the popup-menu).

The following shows the function short forms, their functionality and their counterparts in the pop-up menu.

+ (popup: "+")
pages down (if there is still at least one more track definition not visible on the screen)
one screen page comprises 7 definition lines.

\- (popup: "-")
pages up

I (popup: "Insert")
inserts a definition line for a new track type

precondition:
-> requires preceeding specification of track number in "track"

result:
The new track type is inserted at the specified position. The cursor is at the beginning of the line where the definition of the new track type is to be entered, the track type is specified through the size and parameter specifications.

D (popup: "Delete")
deletes an existing track type definition

precondition:
-> specification of the track number in "track"

result:
the definition line for the specified track type disappears from the screen, the following definition lines are shifted up by one line.

C (popup: "Check")
checks whether the current track technology has been defined correctly; if, for example, a via type is specified which has not been defined, or if invalid values are specified for the radius, an error message appears.
if no errors are found, the message "no technology violation found" appears

W (popup: "Write")
stores the current track technology;
at first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);
track technology definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the track technology;
this is to undo the changes made up to this point
the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the adjustment of an adequate graphic output (selection of the objects and layers to be represented, selection of the colours to be used, etc.)

$ (popup: "<==")
leads back to the technology selection menu

/ (popup: "main menu")
leads back to the PCB or CAM main menu (according to the main menu the technology function has been called from)

track:

Specifies the track type to be processed (to be generated, to be deleted or to be edited) through specification of the track number (defined in "trk.no.").

technology:

Specifies the layer technology (defined in "layer technology") to which the track technology to be edited belongs, or to which the new track technology to be generated is to belong.

layer:

Here, the names of those layers appear to which the layer technology is assigned in the layer plan; this layer technology is specified in the current form under "technology". For all of these layers, the current track technology applies.

trk. no.

is the track type to which the values and parameters of this line apply.

width

defines the width of the respective track type.
This value is used by the router as a basis for the internal representation of the track width.

For the definition of the track types, the following **restrictions** have to be observed:

1. minimum width
   - for unmoved tracks: 0
   - for moved tracks: 1 x routing grid

2. if
   the widest non-shifted track is wider than
   (2 x routing grid) minus (clearance)

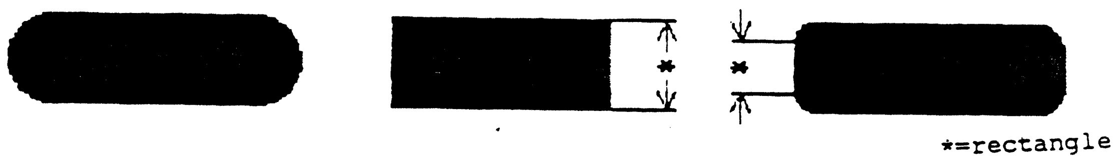
   then
   the smallest non-shifted track has to be wider than
   (0.83 x routing grid) minus (clearance).

radius

Here, the radius is specified with which the corners of the track ends are to be rounded. By determining the track end rounding, you determine the form of the track ends, and thus the form of the SMD pad.

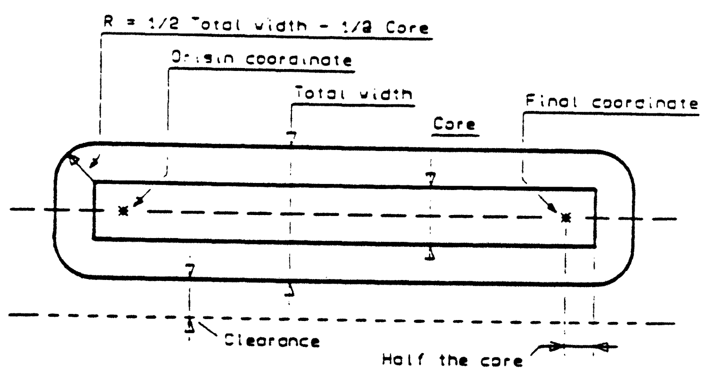
The following shapes are possible:

standard SMD-pad	rectangle pad	rounded rectangle pad



R = 1/2 width	R = 0	R < 1/2 width

The following figure is to explain the sizes more closely:



Note:
The **routable** track types have to be defined with <u>radius = 1/2 width</u>; if tracks are defined with radius R=0 or R < 1/2 width, it might occur that the router violates the clearance at the segment ends. You should, therefore, use radius values of R=0 and R<1/2 width only for track types for which the attribute "N" (not routable) is specified.

add.dst.

If you specify an additional clearance, the router adds this value to the global clearance when routing that track type; the global clearance is defined in the respective layer technology under "min-dist".

attribute:

S with "S" ("shifted") this track type is shifted by 1/2 routing grid into the negative x and y direction during routing. The shifting may sometimes make routing tracks between IC pins possible, where it would not be possible without the shifting. Whether a shift is helpful has to be calculated on the paper for each case and has then to be checked by digitizing or routing - with the appropriate grid.

N this parameter 'forbids' the routing of this track type.

Note:
- Since the router can work with a maximum of 16 routable track types, it has to be seen to it that a maximum of 16 track types do <u>not</u> contain the attribute "N"; i.e., except for the types to be routed, the attribute "N" has to be set for all types.
- track types used for generating SMD pads should have the attribute "N", otherwise it might occur that the router sets vias into the SMD pads!

via

The via type specified in this field (defined in the via technology, belonging to the same layer technology like the current track technology) is used for routing the track type defined in this line.

remarks

This field is not yet evaluated.

VIA TECHNOLOGY

Vias are used for "changing" to other layers when tracks are routed.
There are two kinds of vias:
* "drilled-through" vias (going through all layers of a board)
* "buried" vias (going through one single layer pair).

In the via technology new via types and their representation form and size are defined, and existing via definitions can be changed or deleted.

Every via technology is bound to a certain layer technology. The assignment of a layer technology to every layer in the layer plan guarantees a definite assignment of a via technology to every layer.

Up to 32 layer technologies with different via technology definitions can be defined for a max. of 255 via types.

The layer-specific via technology makes it possible to represent the same via type on different layers with different sizes - and even in different forms - and, thus, to create so-called 'pad-stacks'.

Remark:

- The solder resist plan technology for the via types of the 'physical' layer technology has to be stored in a separate 'docu' layer technology in the form of a via technology (default name "SR1" and "SR2" in the standard technologies); in this via technology the solder resist values are specified in "width" and "radius".
  (as regards the generation of the solder resist data for the pads of the physical terminals, see "reference component" of the technology description as well as chapter "terminals" of the LAYOUT/Modification description;
  as regards the generation of the solder resist data for the "vias" see "solder resist plan" in the CAM description).
- The graphics of the via types used for digitizing tracks are generated automatically on the basis of their technology definition and are stored in the terminal graphics pool; they have the standard name **VIAGRFn** (n is the via type number - e.g. VIAGRF1 for via type 1).

After activating the function "Vias" in the menu "Technology" of the PCB part, the first 'page' of the via definition for the default layer technology DT1 appears in the status window, if the startup.adb-file has been read in at the beginning.

A. Output in Menu Type "T"

If the menu is called for the first time, the type of the menu is always "T".

In this menu type, all via types are output in their numerical order with their definitions in the layer technology entered in "technology".

("V" activates the second output form for the via technology where one via type is output with all its layer technology definitions - see below, point "Output in Menu Type "V".)

The via technology standard of the layer technology DT1 contains very comprehensive definitions (the sizes are defined in steps of 1/10 mm) and has been designed as an 'allround' via technology set convenient for all grids.

```
PCB: modify via technology                         grid  : 0.635/0.635   MS:  mm   [990]

Fct: +   Via: 0__  technology : DT1____________     type of form: T (V/T)
                   layers     : D7,D8,D9,D10,D11,D12,D13,D14,D15

via  width   radius    additional   attr. drill
                       clearance          no.
= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
0    0.0___  0.0____   0.0___       N_D   0_    ____________________________
1    0.0998  0.0499    0.0___       __D   4_    ____________________________
2    0.2003  0.0998_   0.0___       __D   4_    ____________________________
3    0.3001  0.1497_   0.0___       __D   0_    ____________________________
4    0.3999  0.1988_   0.0___       __D   0_    ____________________________
5    0.4997  0.2487_   0.0___       __D   0_    ____________________________
6    0.6002  0.3001_   0.0___       __D   0_    ____________________________
7    0.70    0.350__   0.0___       __D   0_    ____________________________
8    0.7998  0.3999_   0.0___       __D   0_    ____________________________
9    0.9003  0.4498    0.0___       __D   0_    ____________________________
10   1.0001  0.4997    0.0___       __D   0_    ____________________________
attributes: S=shifted  A=active  N=not routable  D=drilled through  B=buried  F=free
```

In order to call the via technology definition for another layer technology, the name of the (defined!) layer technology has to be entered in "technology:" with <CR>.

If after the generation of a new layer technology (see function "layer technology") the "T" menu for the via technology is called, and if the name of the new layer technology is entered into the field "technology:" with a <CR>, the definitions which were entered last in the via menu are taken for the new via technology.

Then, you can change these via type definitions.
Afer the necessary changes have been entered, the data have to be stored. Thus, the changed via technology is part of the previously defined layer technology.

In menu type "T", an existing via technology may be modified through
-> inserting additional via definitions
-> modifying existing via definitions by overwriting existing entries
-> deleting existing via definitions

Notes:

1. To avoid wrong definitions, the system automatically checks that
   a. every change of an attribute or drill type of a via type is automatically transferred to this via type in all layer technologies;
   b. every via type definition generated with the "insert" function is automatically transferred into the via technologies of all existing layer technologies;
   c. a via type which the user deleted in one layer technology is automatically deleted in all existing layer technologies.

2. In the via technology a maximum of 255 via types can be defined.

3. For vias different forms can be defined (see "radius").

Description of the Menu

fct. ("function")

Various functions may be called by entering the respective function short forms.
These functions may also be called via the popup-menu (to be activated with <RMB>).
Note:
The entry via function short forms does not require any <CR>! The specification of the via number for deleting or inserting has, therefore, to be made <u>before</u> entering the short forms (or before activating the function in the popup-menu).

The following shows the function short forms, their functionality and their counterparts in the pop-up menu.

+ (popup: "+")
pages down (if there is still at least one more via definition which is not visible on the screen)
one screen page comprises 7 definition lines.

- (popup: "-")
pages up

I (popup: "Insert")
inserts a definition line for a new via type

precondition:
-> requires the preceeding specification of a via number in "via"

result:
The new via type is inserted at the specified position. The cursor is at the beginning of the line where the definition of the new via type is to be entered, the via type is specified through the size and parameter specifications.

D (popup: "Delete")
deletes an existing via type definition

precondition:
-> specification of the via number in "via"

result:
the definition line for the specified via type disappears from the screen, the following definition lines are shifted up by one line.

C (popup: "Check")
checks whether the current via technology has been defined correctly; if, for example, a drill type is specified which has not been defined, or if invalid values are specified for the radius, an error message appears.
if no errors are found, the message "no technology violation found" appears

W (popup: "Write")
stores the current via technology;
at first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);
via technology definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the via technology; this is to undo the changes made up to this point the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the selection of an adequate graphic output (selection of the objects, elements to be represented, selection of the colours to be used, etc.)

$ (popup: "<==")
leads back to the technology selection menu

/ (popup: "main menu")
leads back to the PCB or CAM main menu (according to the main menu the technology function has been called from)

via:

Specify the via type to be modified (to be generated, to be deleted or to be edited) through the specification of the via number (defined in "via no.").

technology:

Specify the layer technology (defined in the menu "layer technology") to which the via technology to be edited belongs, or to which the new via technology to be generated is to belong.

type of form:

If "V" is specified, the second output form for the via technology is activated where <u>one via type</u> is output with all its layer technology definitions (see below, point "output in menu type "V").

layers:

Here, the names of those layers appear to which the layer technology is assigned in the layer plan; this layer technology is specified in the current form under "technology". To all of these layers, the current via technology applies.

via

is the via type to which the values and parameters of this line apply.

width

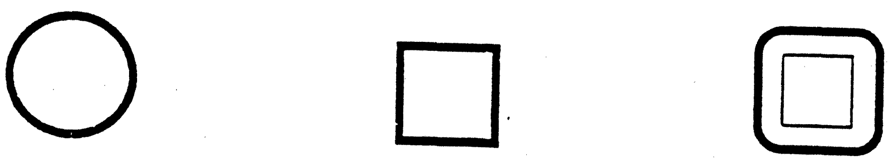
defines the width (or: the diameter) of the respective via type.

radius

Here, the radius is specified with which the corners of the vias are to be rounded. This is to determine the form of the vias.

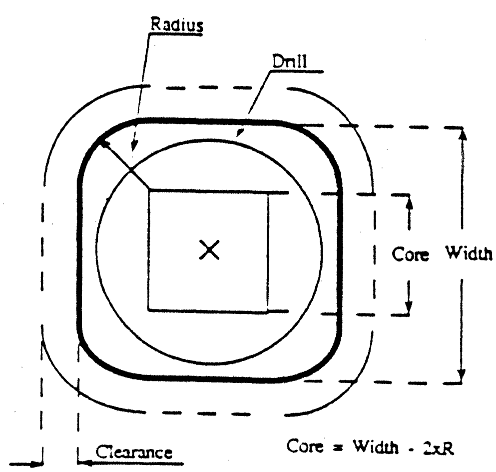
The following shapes are possible:

round via square via rounded square via



R = 1/2 width R = 0 R < 1/2 width

The following figure is to explain the sizes more closely:



additional distance

If you specify an additional clearance, this value will be added to the global clearance when routing that via type; the global clearance is defined in the respective layer technology under "min-dist".

attribute:

If an attribute is changed for a via type, this change is automatically transferred to the respective via type in all layer technologies.

S with "S" ("shifted") this via type is shifted by 1/2 routing grid into the negative x and y direction during routing

Note:
If a via type gets the attribute "S" in one layer technology, this via type is automatically shifted in all layer technologies.

A "A" ("active") activates the via type for the digitizing.

N this parameter "forbids" the routing of this track type.
Note:
Since the router can only handle a maximum of 16 via types, you have to take care that a maximum of 16 via types do <u>not</u> contain the attribute "N" (attribute "N" has to be set for all types except for the types to be routed).

D "D" determines that this via type is to go through all layers (for pads and vias from the component to the solder side). This information is important both for the router and for drilling the vias.

B This attribute defines that the via is to be used as a "buried via", i.e. it is to be used for only one layer pair.

drill no.

The drill type specified in this field (defined in the drill technology) is assigned to the via type defined in this line.
If the assigned drill type is modified, this modification is automatically transferred to the respective via type in all layer technologies.

B. Output of Menu Type "V"

If you are in the menu type "T", enter "V" in the field "type of menu" and press <CR>, the second output form for the via technology is activated, where one via type is output with all of its layer technology definitions.

With this output form you see all definitions for the via type entered in the field "via:" for all layers existing in the layer plan.

In addition to the sizes and attributes definable in the "T" menu, the "V" menu contains:
- the number of vias of this via type used in the current layout,
- all definitions of the drill type assigned to this via type as well as the number of the drills of this type used in the current layout.

If the technology standard file startup.adb has been read, the menu looks as follows:

```
|PCB: modify via technology                    grid:  0.635/0.635  MS: mm  [990] |
|                                                                                |
|fct: _   via: 1__          layer     : COMP_________   type of form: V (V/T)    |
|                                                                                |
|attr.: shifted:_                                                                |
|       routable.. : X   buried: _  | drill no.: 4__                             |
|       no. used on current design:0_ |   hole size:  0.7998 annular ring: 0.25202 |
|                                   |   search range:from.:  _____  to : ______   |
| next via:_                        |   drill macro.......:  DT4                  |
|                                   |   no. drills used in current design:   0    |
|                                                                                |
|                                          additional layer                      |
| layer      width      radius            clearance  technology                  |
|= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = |
| COMP       1.001_     0.4997_           0.0___     T40                         |
| SOLD       1.001_     0.4997_           0.0___     T40                         |
| SHAPE-0    0.0998     0.0499_           0.0___     SHP1                        |
| SHAPE-1    0.0998     0.0499_           0.0___     SHP1                        |
| RESIST-0   1.3002     0.6501_           0.0___     SR1                         |
| RESIST-1   1.3002     0.6501_           0.0___     SR1                         |
| DRILL      0.0998     0.0499_           0.0___     DRILL1                      |
| D5         0.0998     0.0499_           0.0___     DT1                         |
```

If the via technology is output in menu type "V", a via type definition can be changed by
-> modifying an existing via definition through overwriting the entries
-> deleting existing entries

New via types can be generated with the function "insert".

Notes:

1. Every change of a layer-bound size (width, radius, add. distance) is automatically transferred to all layers with the same layer technology,
   i.e.
   if the layer technology T40 is assigned to the COMP layer and if the width of this via type is changed, the new width is transferred to all layers assigned to layer technology T40.

   Note:
   At present, layer-specific sizes (width, radius, additional distance) can be changed only in the last definition line which contains this layer technology (otherwise the change is not put through all layer technologies),
   i.e.
   If in the definition of the current via type the layer technology T40 is assigned to several layers, a change of the via width for the layers with a T40 assignment can be stored only if the change is entered in the last definition line which contains a T40 assignment!

2. If a new via type is generated by the user with the "insert" function, the definition entries of the last activated via type remain in the menu; by editing these values (see notes, point 1) the new via type can be defined and can then be stored.

Description of the Menu

fct. ("function")

Various functions can be called by entering the respective function short forms.
These functions may also be called via the popup-menu (to be activated with <RMB>).

Note:

The entry via function short forms does not require any <CR>! The specification of the via number for deleting or inserting has, therefore, to be made <u>before</u> entering the short forms (or before activating the function in the popup-menu).

The following shows the function short forms, their functionality and their counterparts in the pop-up menu.

+ (popup: "+")
pages down (if there is still at least one more definition line which is not visible on the screen)

- (popup: "-")
pages up

I (popup: "Insert")
inserts a definition line for a new via type

precondition:
-> requires the preceeding specification of a via number in "via"

result:
The new via type is inserted; the last definition of the via menu is automatically transferred to the via type.
The via type can be changed through the size and parameter specifications.
After the specifications, the entries have to be stored.
Important: see the notes, point 1!

D (popup: "Delete")
deletes an existing via type definition

precondition:
-> specification of the via number in "via"

result:
the definition of the specified via type is deleted; the next via type is entered into the menu with its definitions.

C (popup: "Check")
checks whether the current via type has been defined correctly; if, for example, a drill type is specified which has not been defined, or if invalid values are specified for the radius, an error message appears.
if no errors are found, the message "no technology violation found" appears

W (popup: "Write")
stores the current via type definition;
at first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);
via type definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the via technology;
this is to undo the changes made up to this point
the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the selection of an adequate graphic output (selection of the objects, elements to be represented, selection of the colours to be used, etc.)

$ (popup: "<==")
leads back to the technology selection menu

/ (popup: "main menu")
leads back to the PCB or ÇAM main menu (according to the main menu the technology function has been called from)

via:

Specify the via type to be modified (to be generated, to be deleted or to be edited) through the specification of the via number (defined in "via no.").

layer:

By entering the layer and <CR> the definition line for this layer is called. For the editing, the cursor is placed onto the first input field of the definition line.

type of form:

If "T" is entered, the menu type "T" is called for the output of the via technology; <u>all via types</u> are output in their numerical order with their definitions in the layer technology which is specified in "technology".

attr.: **shifted:** (identical to attribute "S" in menu type "T")
routable: (identical to missing attribute "N" in menu type "T")
buried: (identical to attribute "B" in menu type "T")

The three attributes which can be assigned in the via technology are output here:
- marked with a cross: the attribute is activated
- not marked with a cross: the attribute is not activated;

Note:
"buried:" not marked with a cross means that this via type is defined as being "drilled through" (corresponds to attribute "D" in menu type "T").

The attribute assignment can be activated or deactivated by marking or not marking the field with a cross.

As regards the meaning of the attributes, see "attribute" descriptions under point A ("output in menu type "T").

no. used on current design:

Contains the number of the vias generated in the layout which have the current via type (spec. in field "via:").

next via:

By clicking this field with <LMB> the via definition for the next higher via type is entered into the menu.

drill no.

This field contains the drill type assigned to this via type. The drill type can be changed by entering the appropriate number (see PCB technology, function "Drills").

hole size:

Contains the diameter for the drill type entered in "drill no." which is defined in the drill technology.

annular ring:

Contains the annulus width for the drill type entered in "drill no." which is defined in the drill technology.

search range:

By specifying the appropriate drill diameters in the fields "from:" and/or "to:" you can search for a convenient drill type from the drill technology.
If the search is successful, the drill type is 'entered' into the menu.

precondition: the field "drill no." may not yet contain any drill type.

drill macro:

This field contains the name of the drill plan macro assigned to the drill type of the field "drill no." in the drill technology.

no. drill used in current design:

This field contains the number of the drills existing in the layout which have the current drill type (entered in "drill no.").

layer:

This field contains the names of the layers defined in the layer plan.

width:

The value entered here defines the width of the current via type for the layer technology entered at the end of the line - and thus for the layer specified at the beginning of the line

radius

This field contains the radius with which the corners of the vias are to be rounded.
This determines the form of the current via type for the layer technology entered at the end of the line - and thus for the layer entered at the beginning of the line (for more information, please see the section "radius" in "output of the menu type "T"").

additional clearance

The additional clearance is added to the global clearance of this layer technology when that via type is routed within the layer technlogy entered at the end of the line - and thus on the layer entered at the beginning of the line.

layer technology:

This field contains the name of the layer technology assigned to the layer entered in the layer plan at the beginning of the definition line.

DRILLS

The drill technology applies to the whole board, i.e., it is not defined layer-specifically (like the track and via technology).

After activating the function "Drills" in the menu "Technology", the first page of the table with the drill technology of the current .adb-file appears in the status window.

When creating a new .adb-file on the basis of the startup standard technology, the table contains the values stored in the startup.adb-file.

```
PCB: modify drill technology     grid  : 0.635/0.635   MS:  mm      [989]

fct: + drill

drl.  diameter   annulus    macro   used  att    remarks
no.
=========================================================================
1      0.0___    0.0___    DT1____   0    P     ____________________
2      0.6002_   0.2502_   DT2____   0    P     ____________________
3      0.0___    0.0___    DT3____   0    P     ____________________
4      0.7998_   0.2502_   DT4____   0    P     ____________________
5      0.0___    0.0___    DT5____   0    P     ____________________
6      0.9003_   0.2502_   DT6____   0    P     ____________________
7      0.0___    0.0___    DT7____   0    P     ____________________
10     1.0001_   0.2502_   DT8____   0    P     ____________________
```

A maximum of 255 drill types may be defined; to these any numbers between 1 and 255 may be assigned.

An existing drill technology may be modified through
-> inserting additional drill technologies
-> modifying existing drill definitions by overwriting existing entries
-> deleting existing drill definitions

Description of the Menu

fct. ("function")

Various functions may be called by entering the respective function short forms.
These functions may also be called via the popup-menu (to be activated with <RMB>).

Note:

The entry via function short forms does not require any <CR>! The specification of the drill number for deleting or inserting has, therefore, to be made before entering the short forms (or before activating the function in the popup-menu).

The following shows the function short forms, their functionality and their counterparts in the pop-up menu.

+ (popup: "+")
pages down (if there is still at least one more drill definition which is not visible on the screen)
one screen page comprises 7 definition lines.

- (popup: "-")
pages up

I (popup: "Insert")
inserts a definition line for a new drill type

precondition:
-> requires preceeding specification of drill number in "___"

result:
The new drill type is inserted at the specified position. The cursor is at the beginning of the line where the definition of the new drill type is to be entered, the drill type is specified through the size and parameter specifications.

D (popup: "Delete")
deletes an existing drill type definition

precondition:
-> specification of the drill number in "___"

result:
the definition line for the specified drill type disappears from the screen, the following definition lines are shifted up by one line.

C (popup: "Check")
checks whether the current drill technology has been defined correctly; if, for example, a macro is specified which has not been defined an error message appears.
if no errors are found, the message "no technology violation found" appears

W (popup: "Write")
stores the current drill technology;
at first, a check is carried out (it is not carried out if the "check" function has been activated before and if no error could be found);

drill technology definitions containing errors cannot be stored;

R (popup: "Restore")
Restores the unchanged version of the drill technology; this is to undo the changes made up to this point the restore function only applies to changes which have not yet been stored.

G (popup: "graphic")
calls the graphic menu for the selection of an adequate graphic output (selection of the objects and layers to be represented, selection of the colours to be used, etc.)

$ (popup: "<==")
leads back to the technology selection menu

/ (popup: "main menu")
leads back to the PCB or CAM main menu (according to the main menu the technology function has been called from)

drill

Specifies the drill type to be processed (to be generated, to be deleted or to be edited) through specification of the drill number (defined in "drl. no.").

drl. no.:

is the drill type to which the values and parameters of this line apply.

diameter:

defines the diameter of the respective drill type (see fig. 1)

annulus

defines the width of the annulus for the respective drill (see fig. 1)

macro

For every drill type a macro graphic existing in the pool component graphic has to be specified for its representation in the drill plan.

How to procede for the macro generation:

1. call the function "COMPONENT/create" in the PCB work menu
2. enter the macro name in "graphic" in the status window and "Y" in "create new graphic?";
3. open and set a component frame at a free position in the graphic window with the mouse (is done like creating a component)
4. get the component, activate the function "mod. graphic"; enter "Y" in "glob." in the status window
5. modify the component graphic so that it gets the desired form (i.e. digitizing the macro and deleting the original component frame)
6. get and delete the component name textfield
7. leave the menu "mod. graphic" with <ESC><ESC>
8. get and delete the component

The graphic now exists in the "graphic pool" under the name specified in the field "graphic" and may be specified as a macro in the drill technology.

Note:

- Do not leave the "mod. graphic" function before making sure that the macro exactly corresponds to your requirements! At present there is no delete or change function for pool graphics. In order to change a stored macro graphic you have to generate a new component with this macro graphic, then modify the component graphic to the desired shape with the function "mod. graphic", then enter "Y" in "glob." and store the changes with <ESC><ESC>; the last step is to delete the component.

- Do not forget to enter "Y" in "glob.", otherwise the graphic gets automatically a new name by the system (STDN..)!

- For digitizing the macro the coordinate origin is the lower left corner (= rotation point) of the original component.

- Remember or better note down the names of the generated macro grpahics, since at present there is no list function for the graphics existing in the 'pool'.

Note:
If the standard technology startup.adb has been read in at the beginning, there are already 30 predefined macros (names "DT1" to "DT30") in the graphic pool.

used

contains the number of the drills existing on a board of the appropriate drill type.

attribute:

P ("producable") activates the drill type for the post-processing.
Drill types not containing the "P" are ignored by the post-processing (in the drill plan as well as in the drill tape).

remarks

This field is not yet evaluated.

REFERENCE COMPONENT

This function serves for defining user-specific standard terminals (="reference terminals") for the current .adb-file which are stored in the 'background' as part of the technology and which may be called into the modification part.

-> Any graphic belonging to a reference terminal is available in the graphic pool and may be used in the Layout/Modification part as a copy for the graphic of a newly generated terminal;

-> In the Layout/Modification part any terminal of an existing component may be replaced by any reference terminal (function "TERMINAL/exchange shape").

-> Any subsequent change of an existing reference terminal graphic in the technology part is of a "global" nature, i.e., it applies to all terminals in the layout which have been generated on the basis of that reference terminal.
In contrast to this, terminal graphic changes in the modification part are always of a "local" nature, i.e., they apply only to the modified terminal.

Note:

- If you would like to use the terminals of the reference component not only for the current .adb-file, you may generate it in one of your standard technology files. If you then generate a new .adb-file with the help of this technology standard, you have direct access to the reference component also from the new file.

- The standard technology startup.adb contains 9 predefined reference terminals the forms, sizes and graphic names of which are output on the graphic screen when activating the technology function "reference component".

- The graphics of the via types used for digitizing tracks are generated automatically on the basis of their technology definition and are stored in the graphic pool; they have the standard name **VIAGRFn** (n is the number of the via type - e.g. VIAGRF1 for via type 1).

When activating the function "Reference component", the menu "Place terminals" appears in the status window. After activating the function "TERMINAL/create" you may define your reference terminals assigned to the reference component.
The reference component is named **!PCBREFGRP** and is not displayed in the layout.

Generation of the Reference Terminals
(with pad graphic for signal layers, solder resist layers, heat traps, paste layers and via blocks)

When defining a reference terminal, all graphic elements have to be generated by the user, these graphic elements are then used for the representation of the pads of the respective terminal type on the different layers in the layout, the elements are:
- the pad graphic for the signal layer(s)
- the pad graphic for the solder resist plan on the respective RESIST docu layers
- if applicable the heat trap graphic on the respective plane layer(s)
- if applicable the graphic for the area of the solder paste to be laid on the SMD pad on the respective docu layer(s) (for the SMD reflow method).
- if applicable, mark via blocks (for router vias and digitized vias) around the terminal

The solder resist plan pad graphic is generated with a respectively enlarged via (for drilled-through pads) or with an enlarged track (for SMD-pads) from the via or track technology which belongs to the SOLD-RESIST-layer technology. It is digitized "over" the signal layer graphic of the respective copper element on the assigned RESIST layer.
The heat traps have to be digitized on the respective plane layer(s) to the vias of the plane terminals.
For SMD pads where the SMD component is to be soldered with the reflow method, the solder graphic is generated with a respectively smaller track from the track technology belonging to the PASTE track technology; it is digitized "over" the signal layer graphic of the respective copper element on the appropriate PASTE docu layer.

How to procede:

1. call function "TERMINAL/create" in the menu "technology/reference component"
2. enter in the status window:
   a. if applicable, enter a terminal name of your choice in "pin" instead of the automatic pin number
   b. if the terminal is not to be placed with the mouse, enter the placement coordinats in "coord."
   c. a name for the terminal graphic in the first input field under "graph.";
   d. if it is an SMD terminal
      > "Y" in "2 sided:"
      > in "graph" in the second input field the name for the "second graphic" (for the terminal placement on the solder side)

the standard name for the second graphic is:
name of the first graphic supplemented by an apostrophe
(e.g. 1st graph.: SMDTERM, 2nd graph.: SMDTERM')

Notes:
- the second graphic is automatically generated by the system, if "Y" has been specified in "2-sided"; it is the user-defined graphic on the component side (possible also on the RESIST-1-layer and the PASTE-1-layer) mirrored to the solder side (also possible the RESIST-0-layer or the PASTE-0-layer);
  at first, it is only available in the graphic pool;
  it is visible only if the terminal is shifted or copied to the solder side;
  if the second graphic has still to be modified, you first have to shift the terminal to the solders side, you may then make the changes with the function "mod. graphic";
- if "Y" is specified in "2 sided" but a second graphic name is not specified, the 2nd graphic name is automatically generated by the system: first graphic name + apostrophe; the name of the second graphic only appears in the menu (in the second i/o field), if the second graphic has been activated, i.e., after the terminal (or in the modification part: the component to which the terminal belongs) has been copied or shifted to the solder side;

3. after <CR> or <LMB> the field "create new graphic" appears in the status window;
   > if you answer "N", you get back to the previous menu and may enter your modifications.
   > if you answer "Y", you may then place the terminal

4. place the terminal:
   * with <CR>, if the placement coordinate has been specified in the status window;
   * or with the mouse
   At this point, the terminal only consists of the placement point marked with a cross;

5. activate the function "terminal/get" and get the terminal
   * with the mouse
   * via entering the terminal name in "pin" in the status window and <CR>;

6. activate the function "mod. graphic" and "digitize";
   Note:
   - when working with the function "mod. graphic" the placement point (=rotation point) of the activated terminal is the (temporary) coordinate origin.

- to make the work easier, open a window around the terminal with the "window" function, and you might as well activate the "grid on" function for the grid output;

7. Digitizing the pad graphic for the signal layers

   a. for SMD pads
      * specify in the status window
        - in "layer" the number "2", since the track segment has to be digitized to the terminal on the component side;
        - in "trk.-type": the number of the track type (from the track definiton of the appropriate layer technology) with which the segment is to be digitized;

      * set the starting point of the track segment with the function "OFFSET" and with the coordinate specification (via the keyboard in the status window or via the mouse);
      * activate the function "LINE"
      * digitize the segment for the representation of the SMD pad on the component side

   b. for drilled-through pads
      * set the placement coordinate for the via with the "Offset" function and with the coordinate specification (via the keyboard in the status window or via the mouse)
      * activate function "via"
      * specify in the status window
        - in "via type": the number of the via type (from the via definition of the layer technology) with which the via is to be digitized on the layer specified in "via-lay."
        - in "via lay." the number(s) of the layer(s) on which the via is to be stored as pad graphic to the terminal (e.g. "*" for "all layers" or "3,4" for two inner signal layers);
      * place the via
      * if different pad graphics are to be assigned to the terminal on different signal layers or signal layer pairs, repeat the procedure for the other layers (the via type used has to be defined in the assigned layer technologies in the desired size).

8. Digitizing the pad graphic for the solder resist plan

   a. for SMD pads (for placement on the component side)
      * specify in the status window
        - in "layer": the number of the RESIST layer belonging to the component side (mostly 20), on which the track segment is to be digitized as a terminal graphic element;
        - in "trk-type" the number of the track type (from the track definition of the SOLD-RESIST layer technology) with which the segment is to be digitized;

* if applicable, use the function "OFFSET" and the coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point of the track segment onto the starting point of the track segment of the signal layer pad graphic;

* activate function "LINE"
* digitize the segment for the solder resist plan representation of the SMD pad

b. for drilled through pads
  * use the function "OFFSET" and the coordinate specification (via the keyboard in the status window or via the mouse) to set the placement coordinate onto the one of the signal layer pad graphic;

  * activate function "via"
  * specify in the status window
    - in "via type": the number of the via type (from the via definition of the SOLD-RESIST layer technology) with which the via is to be digitized;
    - in "via lay.": the number(s) of the RESIST layer (in most cases 19 and/or 20, e.g. "19,20"), where the via is to be stored as a terminal graphic element
  * place the via;

9. Digitizing the heat trap graphic for the plane connection

   * specify in the status window
     - in "layer" the number of the PLANE layer defined in the layer plan where the heat traps are to be digitized as terminal graphic elements; (this layer has also to be specified in the CAM menu "planes" in "layers" in order to generate the postprocessing data for this plane)
     - in "trk.type": the number of the track type (from the track definition of the layer technology which is assigned to the appropriate docu layer) with which the segment is to be digitized;
     - "Y" in "heat trap"
     - "Y" in "show heat trap"
   * set the starting point for the first track segment with the "offset" function and the appropriate coordinate specification (via the keyboard in the status window or via the mouse);
   * activate function "LINE"
   * digitize the first track segment of the heat trap representation
   * set the starting point of the second track segment with the "offset" function, digitize the segment, etc.

Note:
when transferring the reference terminal graphic to existing or new terminals, the heat trap graphic is transferred only to those terminals which are connected to the plane (the system automatically "recognizes" and considers this).

10. Digitizing the paste graphic for reflow solder plan (for placement on the component side)
   * specify in the status window:
     - in "layer" the number of the paste docu layer where the track segment is to be digitized as terminal graphic element;
     - in "trk-type" the number of the track type (from the track definition of the respective PASTE layer technology) with which the segment is to be digitized;
   * set the starting point for the paste track segment with the function "OFFSET" and the appropriate coordinate specification (via the keyboard in the status window or via the mouse)
   * activate function "LINE"
   * digitize the segment for the representation of the solder paste area over the signal layer representation of the SMD pad

11. if applicable, digitize via blocks around the terminal
   * activate the functions "digitize" and "block"
   * click the positions to be blocked for router and digitized vias with <LMB> - the blocked positions are marked with a cross.

12. Store the data with <ESC><ESC> after finishing all graphic elements for the terminal;

   result:
   the terminal graphic now exists with the name(s) specified under 2. (c and d) as part of the respective reference terminal as well as a 'global' pool graphic and may be transferred to other terminals in the PCB modification part as well as in the reference component part.

   note:
   Remember or better note down the name of the generated pool graphics, since at present there is no list function for the graphics of the 'pool'.

Transfer of an existing graphic when generating a new reference terminal

Existing graphics may be used for the generation of new reference terminals.
Even if the user has not yet defined any reference terminals, graphics may already exist in the pool: if vias have already been used for digitizing tracks, the graphics of these via types exist in the pool (graphic name: VIAGRF1, VIAGRF2, etc.).
Unlike above, enter the name(s) of the desired existing graphic(s) in the status window under "graph." (ref. point 2c/d) instead of the new graphic names, if existing graphics are to be used for the generation of new reference terminals.
Then you may place the terminal together with its assigned graphic.
Notes:
- If "Y" has been entered in "2 sided" and if the name of an existing graphic has been specified for the first graphic and no name has been specified for the second graphic (ref. point 2c/d), the system presumes that a second graphic exists with a respective name (name of the first graphic supplemented with an apostrophe); if such a graphic does not exist, the message "terminal graphic not found" is output;

- The graphic used for the generation of the reference terminal may be modified with the function "mod. graphic" (by moving or deleting activated elements or by digitizing further elements).
  Attention:
  Since reference terminal graphics are basically of a "global" nature, graphic changes always apply to all terminals with that graphic (to the reference component and to the PCB).

Transfer of the generated terminal graphic onto terminals which are to be newly created

1. after the generation of the component frame (see function "component/create") get the component and call the function "terminal/create";

2. specify in the status window
   - the number of the first terminal in "pin"
   - if applicable the placement coordinate for the first terminal in "coord.:" (if the placement is to be done via the mouse, this specification is not necessary)
   - if a terminal row is to be generated:
     -> in "dist." the distance between the terminals in the appropriate measurement (e.g. in mm:"2.54/0.00" for a row to be generated in the positive x-direction; "0.00/-2.54" for a row to be generated in the negative y-direction)

-> the number of the terminals to be generated in "number"
-> in "series" e.g. "1" for a continuous numbering of the pins (1, 2, 3, etc) or "2" for a series like 1, 3, 5, etc.
- the name(s) of the reference terminal graphic(s) under "graphic:"

3. place the first terminal with the mouse or press <CR>, if the coordinates have been specified in the status window;

Transfer of the generated terminal graphic onto existing terminals:

this is done by exchanging the existing terminals against a reference terminal:

1. activate the function "terminal/get"
2. specify in the status window:
   - the pin name of the reference terminal in "pin:"
   - the reference component name "**!PCBREFGRP**" in "component"

3. <CR> automatically activates the function "exch. shape";
4. click the terminals to be exchanged with <LMB>; the terminals clicked are exchanged against the reference terminal, i.e., get its graphic.

Notes:

a. When deleting a reference terminal the graphic remains existing in the pool.

b. Since at present there is no list function for the graphics in the pool, it is advisable to remember or to note down the names of the pool graphics, in order to be in a position to activate the graphics at any time.

c. You should generate the necessary reference terminals and all the other technology definitions before beginning with the actual layout generation.
The most effective way is to directly modify the standard technology files; this saves you defining the same elements in different work files;

d. Terminal graphics which are used seldomly or only for generating single board terminals should not be defined in the standard technology files, but in the actual work files (as reference terminal, if it is used several times on that board, or 'locally' if it is only one single terminal).

C H A N G E   F I L E

This function provides the possibility to exit the current .adb file with or without saving it and to read another .adb file - without leaving CIS.

Way of procedure:

1. Click the function "Change file" with <LMB>.

2. The menu function "save file" is now automatically activated, since the system first expects that the current .adb file is saved.
   The name of the current .adb file appears in the status window in "jobname:" and may be altered according to the user's requirements.

3.
   a. If you would like to save the file, enter <CR>.
      Note:
      - If the data are to be compressed, "Y" has to be entered in "compress data:" before pressing <CR>.
        That the .adb file has been compressed is marked by the extension "_Z" in the file name.
      - For the modification with other PRISMA programs than CIS (i.e. for running the router or the postprocessing) a compressed .adb file has first to be uncompressed with the "uncompress" command.
        (go to the adb directory; enter the "uncompress" command, a blank and the complete name of the adb-file, e.g. "uncompress myfile.adb_Z".)

   b. If you do not want to save the file,
      - enter <ESC><ESC>,
      - answer the query "exit?" with "Y"
      - and enter <CR>.
      Then only the function "change file" is active (is highlighted).

4. The name of the .adb file to be read in can now be entered in "jobname" in the status window.
   If it is a standard technology file stored on the directory **projects/std/tec**, you also have to enter "Y" in "read standard technology:". After pressing <CR> the file is read in.

Note:
If you enter <ESC><ESC> instead of <CR>, the message "abort programm?" is displayed.
If you answer with "Y", the CIS program will be aborted.
If you answer with "N" instead, the menu for file-loading appears. You may now load a .adb-file for the modification with CIS.

S A V E F I L E

With this function you save the current .adb file (=storing without leaving the program).

Way of procedure:

1. click the function "Save File"

2. The name of the file which is being modified appears in the status window in "jobname:" and may be changed.

3.
   a. If you want to save the file, enter <CR>
   Note:
      - If for reducing the necessary space, the data are to be compressed, you have to enter "Y" in "compress data" before pressing <CR>.
      The compressed .adb file is marked by the extension "_Z".
      - For processing the file with other PRISMA programs than CIS (i.e., e.g. for running the router or the post-processing), a compressed .adb file has first to be uncompressed with the "uncompress" command.
      (go to the adb-directory; enter the "uncompress" command, a blank and the complete adb-file name, e.g. "uncompress myfile.adb_Z".)

   b. If you do not want to save the file, enter <ESC><ESC> instead of <CR>.

   Then you may continue your work at the .adb file.

E X I T

With this function you may exit the program CIS - with or without saving the current .adb-file.

Way of proceding:

1. activate the function "exit"

2. Before the program is left, the system expects that the current .adb-file is saved.
   The name of the file appears in "jobname:" in the status window; if you want to save the file under a different name, replace the name by the desired new name.

3.
   a. If you want to save the file, enter <CR>
      Note:
      - If for reducing the necessary space the data are to be compressed, "Y" has to be entered in "compress data" before pressing <CR>.
        The compressed .adb-file is marked by the extension "_Z".
      - For processing the file with other PRISMA programs than CIS (i.e., e.g. for running the router or the post-processing) a compressed .adb-file has first to be uncompressed with the "uncompress" command.
        (go to the adb directory; enter the "uncompress" command, a blank and the complete adb-file name, e.g. "uncompress myfile.adb_Z".)

   b. If you do not want to save the current file, enter <ESC><ESC> and answer the query "interrupt saving" in the status window with "Y".

4.
   a. If you then enter "Y" in "exit program?" and enter <CR>, the program CIS is left.

   b. If you answer the query "exit program?" with "N" plus <CR>, you get back to the main menu of the appropriate CIS part.

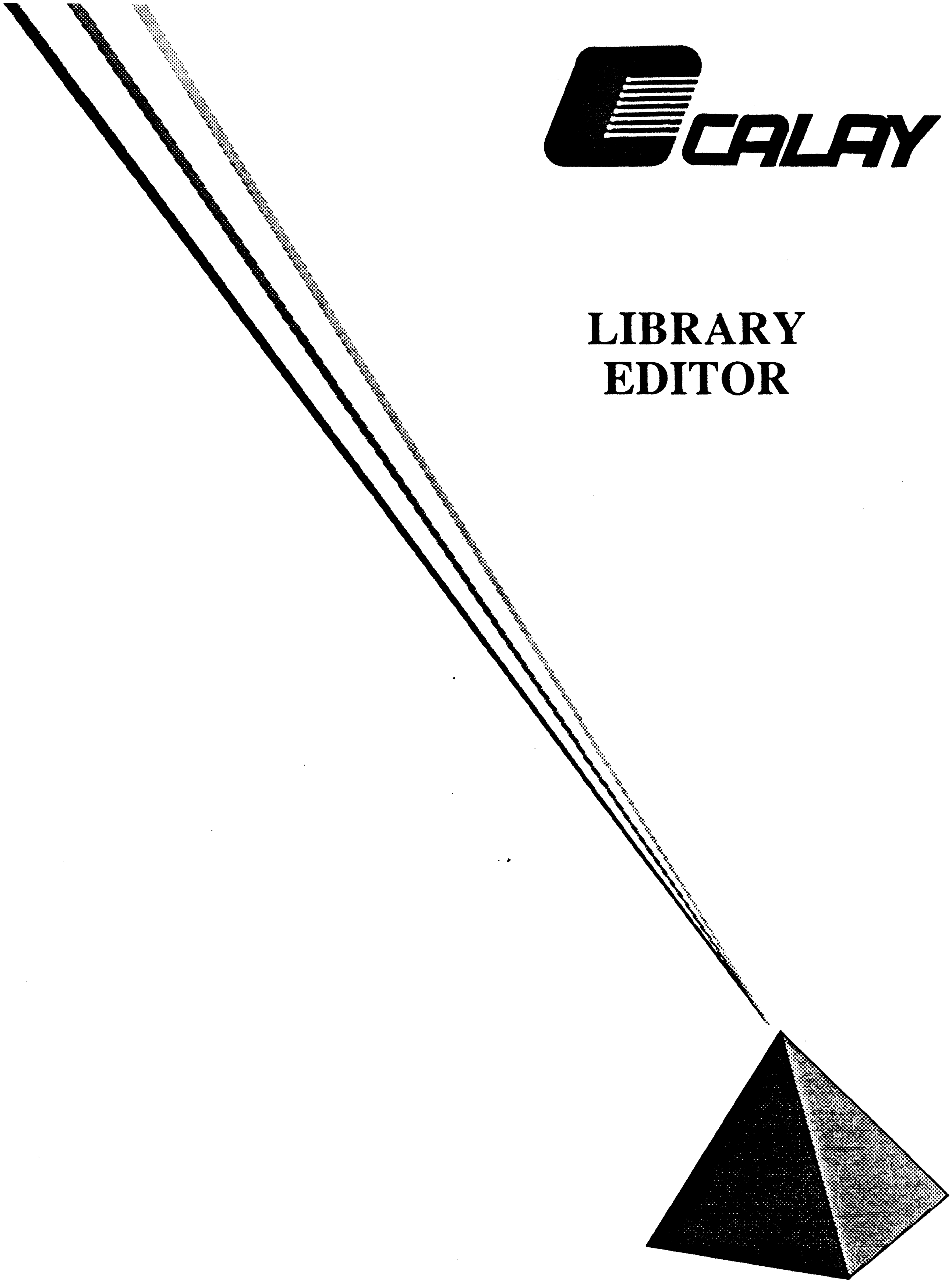
S T A T U S / M A N U A L

This function shows the current version number of the CIS software in the status window.
The information contains the number of the release, the version number of the program and the version number of the menus in the status window.

By marking the function "manual" with a cross plus pressing <CR>, the online manual can be called from there.

The menu "status" is left with <ESC><ESC> or by calling another menu function.

LIBRARY EDITOR



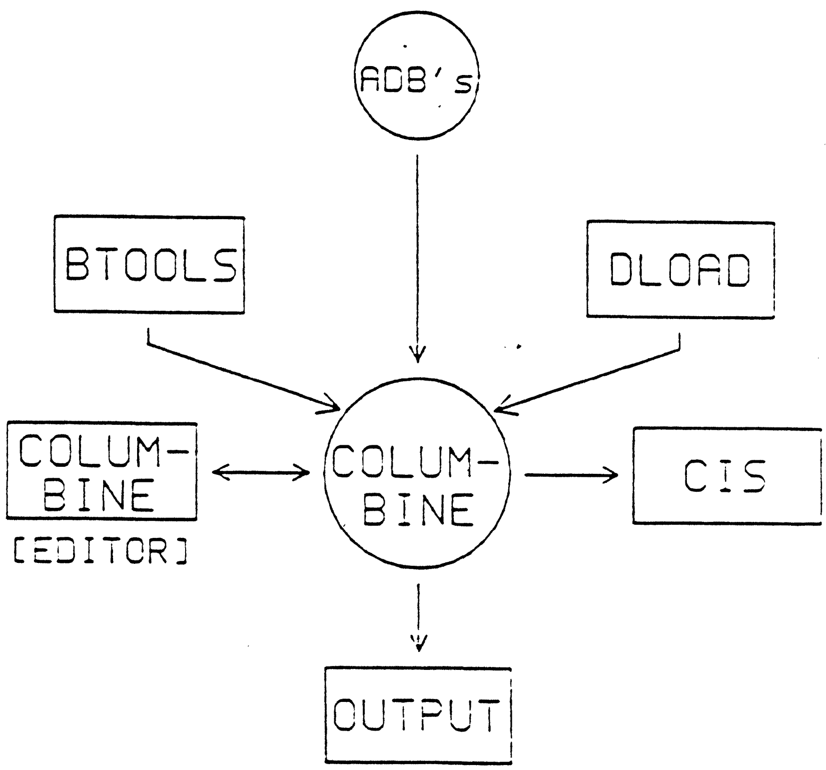
L I B R A R Y

T A B L E   O F   C O N T E N T S

LIBRARY

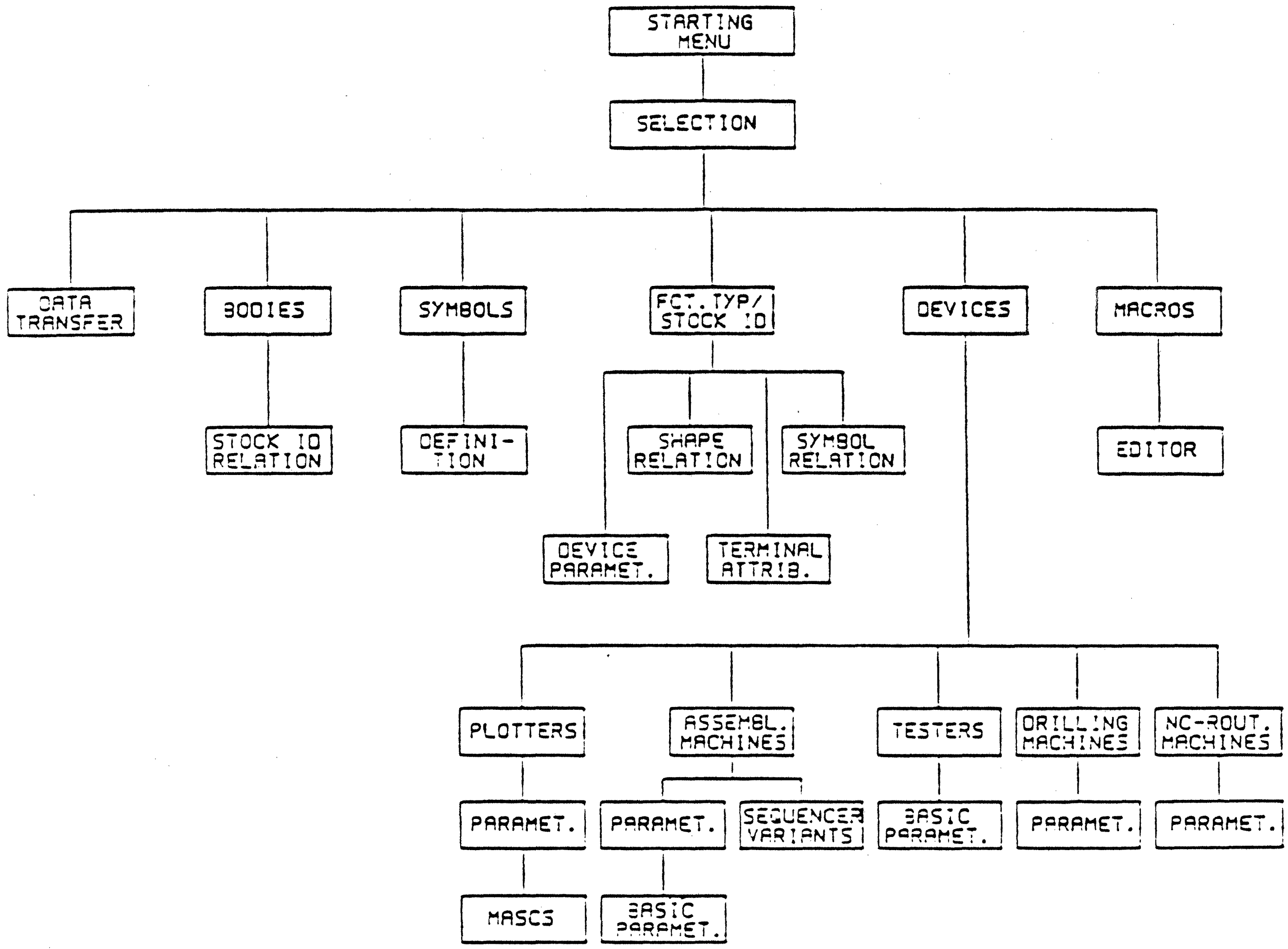
1. Introduction

COLUMBINE is the central library of the CALAY-PRISMA-system. It contains physical and logical component data, information about peripheral devices (insertion, tester, NC-machine, plotter --> together with information to pens/masks and macros for the board elements).
The program **COLUMBINE** belonging to the library is used for the data acquisition and change (=library editor).
As the programs CIS and OUTPUT automatically take access to COLUMBINE, the library to be used must always have this standard name or must have a link to that name.
The following illustration shows the connections of COLUMBINE to other PRISMA programs and files.



The program BTOOLS is used for the creation of the library and the definition of the access rights for the user.
With the program DLOAD, external component data in the ASCII format is taken into the library.

The following diagram gives you an overview of the menus and their connections existing in the program COLUMBINE.

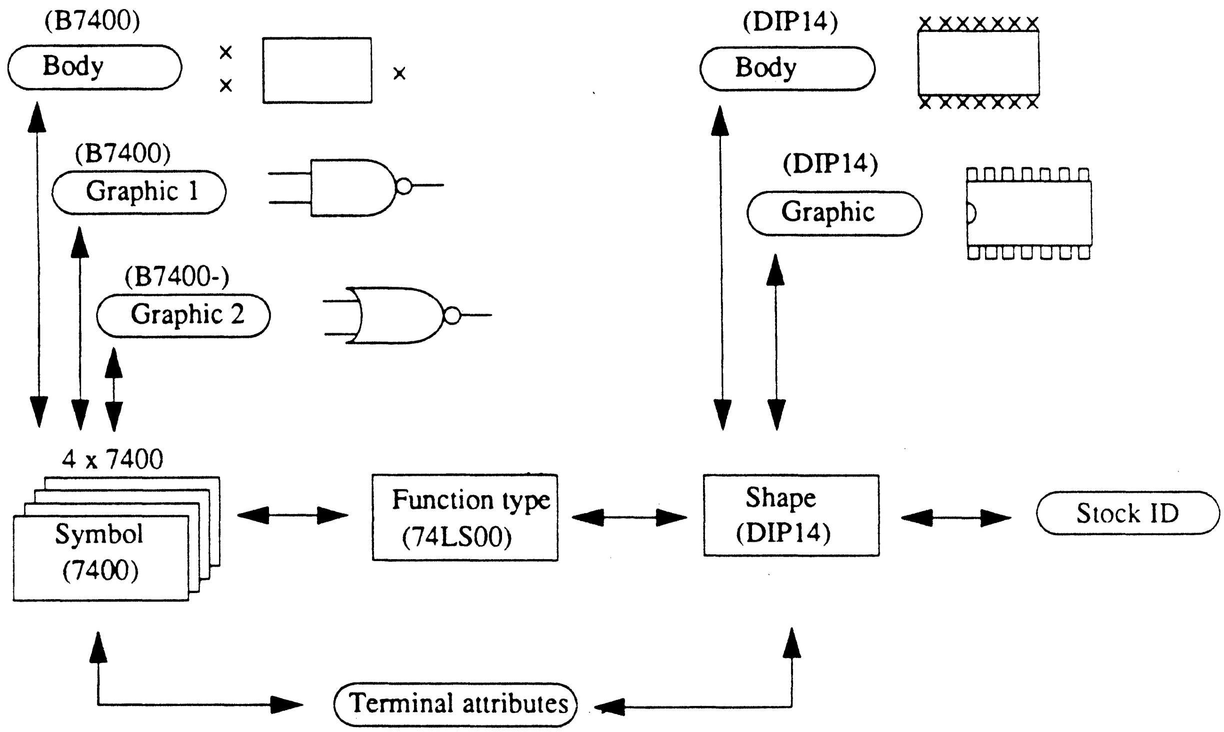


The library COLUMBINE is organized in 5 categories (areas) in which information is stored or can be stored:

Data transfer	(read in physical and logical component information from .adb-files)
Body	(information to shape and symbol bodies)
Logic	(information to logical symbols for electrical function elements)
Function types	(logical and physical component information)
Devices	(device definition for plotter, insertion, tester, and NC-machines)

Load Symbols (macros for the representation of PCB elements on the plotter.)

The following diagram shows how COLUMBINE is structured as regards function types, shapes, symbols and stock IDs.



Attention:

The terms used in the above graphics correspond to those used in the library editor; These terms do not always correspond to those used in CIS (e.g. for symbols: LIBRARY="Graphic 2" --> CIS="alt. graphic name").
If there are differences between the library and CIS, the following term definitions contain the term used in CIS in parenthesis.

Note:

Relating symbols and shapes to function types executable in CIS can also be done directly in the library with the editor functions "function type" and "logic".

Term definitions:

1. **Component**

   Stands for the element existing on the PC board (not relevant to COLUMBINE).

2. **PCB name**

   Placement name of a component on the PC board (e.g. IC1 -> not relevant to COLUMBINE).

3. **Shape**

   Stands for the physical form of a component and consists of the following elements:

   a. **body** (there is no parameter for the body in CIS)
   Consists of the **terminal positions** (referred to the component rotation point) and the **shape extension** (the area that the shape needs on the PC board).

   b. **graphic** (=parameter "graphic" in CIS/PCB)
   Consists of the graphic representation ofthe shape frame and text fields.

   Notes:

   - When new shape graphics are generated in CIS/PCB, the graphic representation of the frame is created as a rectangle --> corresponds to the shape extension.
     The rectangle can then be modified with the function "mod. graphic".

   - If a new shape is created in CIS/PCB, the component rotation point is always the lower left corner of the frame; it can be moved with the function "mod. graphic".

   - A component terminal is composed of its position referred to the rotation point and of the graphic representation of the terminal on the different layers.

4. **Symbol**

   Stands for the logical side of a component and represents an electrical function element (e.g. 7400 or NAND). A symbol is defined with the following elements:

a. **body** (= parameter "symbol body" in CIS/PCB)
Consists of the **terminal positions** and the **symbol extension** (the area needed by the symbol).

note:
In CIS, the extension of the symbol is visible only when the parameter "symbol frames" of the SCS function "modes" is marked with a cross.

b. **graphic 1** (= parameter "act. graphic name" in CIS/SCS)
Consists of the graphic symbol representation generated in CIS (lines, circle arcs and text fields).

note:
When symbols are created in CIS/SCS, the extension of the symbol is automatically represented with a 'rectangle frame'. This rectangle can then be modified with the function "mod. graphic".

c. **Symbol graphic 2** (=parameter "alt. graphic name") in CIS/SCS)
Signifies the 'inverse' graphic to graphic 1.

5. **SCS name**

This is the placement name of a symbol in the schematics (not relevant to COLUMBINE).

6. **Function type** (=parameter "fct. type" in CIS/PCB)

Stands for a function specific standard component (e.g. the 4 NAND component 74LS00) that may exist in one or several 'variants'. According to the number of variants, a function type contains one or several relations between shape(s) (e.g. DIP14, DIP14F, DIP14X, SO14F) and symbol(s) (e.g. 4 NAND symbols 7400) and their terminals, as well as the assignment of a stock ID to each relation. If several relations exist for a function type, the definite selection of one 'variant' (a concrete component) can only be effected by the stock ID.

7. **Stock ID**

Definite identification criterion for a function type 'variant', i.e. for a concrete component (see item 6). The assignment of device parameters for insertion machines and testers is made with the stock ID.

2. Starting the program

After the **LIBRARY** icon has been clicked, a selection menu appears where you select through entering an appropriate number in "your choice? _" either
- the library editor COLUMBINE (in the 'normal' or the 'graphic' mode) or
- the 'library administration program' BTOOLS or
- the 'library report generator' REPORT or
- the program DLOAD for 'importing' external data in ascii format into the library or
- the program DLOADI for 'exporting' data in ascii format from the library into an output file, or
- a consistency check of the component data existing in the library.

The following describes the work with the library editor COLUMBINE.

After the number "1" or "2" has been entered in the LIBRARY selection menu, the editor program COLUMBINE is started either without ("1") or with ("2") a graphic function.
The graphic function is necessary only if load symbols are generated or edited (library section 6).

The editor for the library COLUMBINE is menu-driven.

After you have entered a function in the editor main menu with <CR>, you get to subordinate menus.

The function **"0"** (=end) finishes your work in a menu and stores the data. You get back to the superior menu or finish the program.

With <ESC><ESC>, you leave the menu without storing your entries and get to the superior menu.

After the program has been started, the following menu appears:

COLUMBINE Library editor Version:	2

User	: ___________
Password	: ___________

notes:

- In order to avoid an inconsiderate or unauthorized alteration of the library, you have to identify yourself as authorized person with your name and your password to change the library contents.
  The program only accepts specifications that are identical with the ones defined in BTOOLS.
  It distinguishes between uppercase and lowercase!
  After you have entered the user name, you have to press the "Cursor down" key to enter your password in the next line.
  The password will not be displayed on the screen.
  The access rights are determined in the program BTOOLS.
  They can be restricted to different library sections and functions.

- After you have stored your entries with <CR>, you get to menu 14.

3. Library sections

In menu 14, you select the library section to be modified by entering the respective number.
On the right side of the menu, a library statistics appears from which you can determine the free library capacity.

COLUMBINE Library editor Version:	**14**

creation date	**:** ________	**free blocks**	**:** ___	**(blocks)**
last change	**:** ________	**deleted blocks**	**:** ___	**(blocks)**
		number of entries	**:** ___	
version	**:** ________	**deleted entries**	**:** ____	

Functions

(1) Data transfer	**(3) Bodies**	**(5) Devices**
(2) Function types	**(4) Symbols**	**(6) Macro editor (plotter)**
(0) End		

Function : >0<

3.1 Data transfer

With this function, physical and/or logical component data can be taken into the library from an .adb-file.

3.1.1 Conditions for the data transfer

In order to execute the data transfer, the release of the physical and/or logical component or symbol data for the library must have been effected in CIS in the menus "Change component attributes" and/or "Change group attributes". Physical <u>and</u> logical component data can only be transferred together if a respective 'relation' has been executed in CIS before and a function type has been assigned.

Procedure for enabling a component for the transfer to the library

1. Enabling a shape (layout) for the transfer to the library

   a. Get the respective component in the layout submenu "Modification" and activate "mod. param.";

   b. Enter -Y- in "to lib." and in "-shp" in the menu "Change component attributes".

      note:
      A shape name must be assigned to the component in "Shape" --> when the data is transferred, the shape information will be stored in COLUMBINE with this name!

   c. press <CR>.

2. Enabling a symbol (schematics) for the transfer to the library

   a. Get the respective symbol in the schematics submenu "Modification" and activate "mod. param.";

   b. Enter in the menu "change symbol attributes":
      - -Y- in "transfer symbol" and in the fields "symbol:" and "symbol body" (in parenthesis)
      - in "symbol body" the name under which the symbol body is to be stored into the library (e.g. BNOR) and in "symbol:" a symbol name (e.g. NAND)

   c. press <CR>.

3. Enabling a related function type (schematics and layout) for the transfer into the library

   a. Shape and symbol must be enabled in the way described in item 1 and 2;

   b. Between the terminals of the symbol and the terminals of the component, a 'relation' must be executed (see next page);

   c. On the layout side, you have to assign a function type specification to the component in "Function type" in the menu "Change component attributes".

   d. On the layout side, you have to enter -Y- in "-rel." and in "-typ" in the menu "Change component attributes".

   e. On the schematics side, you have to enter -Y- in "transfer relation" in the menu "Change symbol attributes".

Relation shape <--> symbol to a function type

If a component has been generated on the layout side and a respective symbol on the schematics side, the relation can be executed in the schematics part. In order to do so, proceed as follows:

a. Activate the function "relation" in the modification menu of the schematics part; subsequently, the menu "enter relations"

b. In "PCB name", you enter the name of the component to be related from the layout side and press the <CR> key or the left mouse button (called **<LMB>** in this description);
   the component is now 'attached' to the digitizer and can be released with <LMB> (sensibly near the respective symbol);

c. Move the crosshair to a terminal of the component to be related and press <LMB>, then you go to the respective terminal of the symbol and press <LMB>; proceed in the same way with all other terminals to be related.

   note:

   When 'clicking' a terminal, the name and the placement coordinate of this terminal appear in the menu;

d. If you want to relate another component, activate the function "get", enter the name of the desired component in "PCB name" and relate the terminals of the component with those of the symbol with the same procedure;

e. After the relations have been finished, you can continue your work with the functions of the schematics menu;

   note:

   When picking up the symbol after the relations have been executed, the respective component name (PCB name) appears in brackets behind the symbol name (SCS name).

3.1.2 Execution of the transfer

After you have activated the function (1) "Data transfer" in menu 14, the following menu appears:

COLUMBINE Library editor Version:	47

DATA TRANSFER

file name : ______________.adb (0) query (1) always (2) never
name : _____________ > < overwrite mode

(1) transfer shapes

(2) transfer symbols

(3) transfer related function types

(0) End

Function : > <

File name

Specify the name of the .adb-file from which the data is to be transferred. After the input of the file name and <CR>, the file is read into the memory.

Name

Depending on the specification in "function" ("1", "2" or "3") and on the name specification in "name:" ("*" or specification like "DIP*" or complete name), the shapes, symbols or related function types are transferred to the library with the appropriate information.

Overwrite-Mode

Based on this selection, you determine whether or not and in which cases the elements existing in the library are to be overwritten by elements from the .adb which have the same name and are of the same kind.

- **(0) query**
  For every element from the .adb-file which already exists in the library, the system queries "transfer?". "Y" or "N" decides whether the existing element is overwritten by the one from the .adb-file or whether it remains unchanged.

- **(1) always**
  Elements existing in the library are always overwritten without any preceding query; all elements from the .adb-file are transferred (if they have been enabled for the transfer to the library in CIS).

- **(2) never**
  Existing elements are not overwritten (in no case); only elements which do not yet exist in the library are transferred from the .adb-file.

Function selection

By specifying the appropriate selection number the data tranfer is activated for the appropriate element type (shape or symbol or related function type).
"0" finishes the work in this menu and leads back to menu no. 14.

A. Read in shapes

a. In "Name", you specify the shape body name(s) (e.g. DIP*) entered in CIS in the menu "Change component attributes" in "Shape".
b. Enter the appropriate selection number in "overwrite mode"
c. Enter "1" in "function";

result:

Of the appropriate shapes from the .adb-file (e.g. the shape body names of which begin with "DIP") the following elements are transferred into the library (see page 4):

- the shape body (name and extension of the shape body as well as terminal information);
- the graphic (shape frame)
  -> at the moment, the 'graphic' has the same name like the shape body;
- the function type specification, if it has been assigned to the components in CIS in "Function type" in the menu "Change component attributes".

note:

If a shape with the same name already exists in COLUMBINE and if "0" has been selected in "overwrite mode", the query "transfer?" appears for each existing element of the shape (body name, terminals, graphic of the frame, function type specification)
--> if you enter "Y", the respective shape element in COLUMBINE will be replaced by the 'new' one from the .adb-file (overwritten);
--> when entering "N", the element from the .adb-file will not be overtaken.

B. Read in symbols

a. In "Name", you specify the name(s) of the symbol body (bodies) (e.g. B74*) entered in CIS in the menu "Change symbol attributes" in "symbol body";
b. Enter the appropriate selection number in "overwrite mode"
c. Enter "2" in "function";

result:

For all symbols of the specified .adb-file (e.g. the symbol shape name of which begins with "B74"), the following elements are transferred into the library (--> ref. to page 5):

- the symbol body (name and extension of the body and terminal information);
- the graphic display of the symbols
  -> the 'graphic' name starts with "G" and from the second character on it is identical with the name of the symbol body.
- the specification assigned to the symbols in CIS in the menu "Change symbol attributes" in "Symbol".

note:

If a symbol with the same name already exists in COLUMBINE and if in "overwrite" "0" has been selected, the query "transfer?" appears for each existing element of the symbol (body name, terminals, graphic and symbol name)
--> if you enter "Y", the respective element in COLUMBINE will be replaced by the 'new' one from the .adb-file (overwritten);
--> when entering "N", the element from the .adb-file will not be overtaken.

C. Read in related function types

a. In "Name", you specify the name(s) of the function type(s) (e.g. 74*) entered in CIS in the menu "Change component attributes" in "Function type";
b. Enter the appropriate selection number in "overwrite mode"
c. Enter "3" in "Function".

result:

For the function types of the .adb-file (e.g. the names of which start with "74"), the following elements are transferred into the library (--> ref. to page 5):

a. from the layout part

- the shape body (name and extension of the shape and terminal information);
- the shape graphic (shape frame)
  -> at the moment, the 'graphic' has the same name as the symbol body.
- the function type specification assigned to the components in CIS in the menu "Change component attributes" in "Function type".

b. from the schematics

- the symbol body (name and extension of the body and terminal information);
- the graphic display of the symbols
  -> the 'graphic' name starts with "G" and from the second character on it is identical with the name of the symbol body.
- the symbol specification assigned to the symbols in CIS in the menu "Change symbol attributes" in "Symbol".

note: see notes to points A and B.

3.2 Body

With the functions of the following menu, you can duplicate, rename, list and delete shape and symbol bodies existing in the library. In addition, device parameters and stock IDs, previously defined in menu 3, can be assigned to a shape for its additional elements (e.g. socket, screws, etc).

COLUMBINE Library editor Version:	45

SHAPE BODIES or SYMBOL BODIES

Body name : ____________
> < (1) shapes (2) symbols

(1) Names **(5) Device parameters (only shapes)**

(2) Copy **(6) Stock ID assignment (only shapes)**

(3) Kill

(4) Rename

(0) End

Function : 0

(1) Names

When you have entered "1" for "Shapes" or "2" for "Symbols", this function lists all (*) or the specified (..*) names of the shape or symbol bodies existing in COLUMBINE.
By 'clicking' the function "next" with the left mouse button (called <LMB> in this description), you get to the next page, with "previous", you get back and with "quit", you finish the listing.

(2) Copy

Duplicates the specified shape body (selection "1") or the specified symbol body (selection "2"). You are asked for the name of the new shape or symbol body. The new element contains the information of the old shape or symbol body (no 'graphic'!).

(3) Rename

Renames the specified shape or symbol body. You are asked for the new name of the shape or symbol body. The new element still contains the assigned information (no graphics!). The old name of the shape or symbol body does no longer exist.

(4) Kill

Deletes the specified name of the shape or symbol body together with the assigned information from the library.

(5) Device parameters

For the shape type specified in "Body name", 'mechanical' device parameters for assember and/or tester can be created or existing device parameters can be changed, listed, copied or deleted.

note:

The 'mechanical' device parameters 'belong' to the shape from now on, i.e. when assigning this shape to a function type, the shape specific device information will automatically be overtaken and need not be entered separately with the stock ID. (Only the function type specific ('electrical') device information must be defined with the stock ID --> ref. to page 24).

The following menu appears:

COLUMBINE Library editor Version:	162

DEVICE PARAMETER Mode :

(E) Edit, (K) Kill, (L) List, (C) Copy, select: >E<

device : # : type : (2 = Ass., 3 = Test)

shape :

(L) List

Lists the device parameters assigned to the specified shape. First the parameters of one device will be listed.

After pressing any button, the parameters (if defined) of further devices will be listed.
With <CR>, you finish the listing and with <ESC><ESC>, you get back to the upper part of the menu.

(E) Edit

With this function, you can assign assembling or tester data to the specified shape or change the data. In "Device name", you enter the name of the assembling automat or tester for which the parameters shall be created or changed (the device must be defined in the library area "Device"). In "Type", you enter the device type (2 = assembling automat, 3 = tester), <CR>.
Now you can enter the shape-specific parameters for the respective machine.
You must only use the kind of characters determined during the device definition in "Type"; the number of characters must not be greater than the number entered in "Len" during the device definition.

With <CR>, the current parameters are stored and with <ESC> <ESC>, you get back to the upper part of the menu.

(K) Kill

Deletes the device parameters of the specified shape for the device, whose name and type you have entered in "Device name" and "Type", <CR>.
With <ESC><ESC>, you get back to the upper part of the menu.

(C) Copy

The device parameters of the respective shape will be assigned to another device (tester or assembling automat). First you enter the device for which the parameters already exist, <CR>, then you enter the device which shall get the same parameters.

With <ESC><ESC>, you get back to the upper part of the menu.

(6) Stock ID assignment

The stock IDs that may be assigned to a shape in this menu do not stand for the shape itself but for the **'additional elements'** like socket, screws, etc.

You can only use stock IDs created in menu 3 to which device parameters for the assembling were assigned in menu 162 (ref. to page 22 and page 24).

The following menu appears:

COLUMBINE Library editor Version:	**140**

Shape : ______
Index : >0< Stock ID : __________________

0
1
2
3
4
5
6
7
8
9

By specifying an index number (0 to 9), you can assign up to 10 stock IDs to a shape. Each stock ID stands for one of the 'additional elements' of the shape.
After entering the index number, <CR>, you get to the field for the input of the stock ID.

After entering the stock ID, <CR>, the assignment appears below the menu and the next assignment can be made.

With <ESC><ESC>, you leave the menu.

3.3 Logic

With the functions of the following menu, you can duplicate, rename, list and delete symbols existing in the library. In addition, new symbols can be defined (with the assignment of symbol bodies, 'graphics', 'inverse graphics' and function types) and the assignments of existing symbols can be changed.

COLUMBINE Library editor Version:	46

LOGIC

Symbol name : ____________

(1) Names	**(5) Definition**
(2) Copy	
(3) Kill	
(4) Rename	
(0) End	

Function : 0

(1) Names

This function lists all (*) or the specified (..*) symbol names of the library; by 'clicking' the function "next" with <LMB>, you get to the next page, with "previous", you get back and with "quit", you finish the listing.

(2) Copy

Duplicates the specified symbol. You are asked for the name of the new element. The new symbol contains the information (symbol body, graphics and function types) of the old one.

(3) Rename

Renames the specified symbol. You are asked for the new name of the symbol. It still contains the assigned information. The old symbol name does no longer exist.

(4) Kill

Deletes the specified symbol together with the assigned information from the library, provided that only **one** assignment exists for this symbol. If more assignments exist, they must be deleted one by one in menu 171.

(5) Definition

In the menu to this function, you can assign (existing!) symbol bodies, graphics and function types to a symbol name, as well as delete or change existing assignments.

```
COLUMBINE      Library editor      Version:                 | 171

Symbol name:                       (E) Edit       (N) New
__________                         (L) Kill       (+-) Step
                                   >E< Select

E   Function type       Body     Graphic 1     Graphic 2
```

The assignments are listed below the menu. The symbol name automatically appears in the menu.

>N< New

An additional menu part appears, in which you can assign a body, a graphic ("graphic 1"), an 'inverse' graphic ("graphic 2") and a function type to the specified symbol:

```
Function   : _____________
Body       : _____________        Alias name : ___________
Graphic 1  : _____________                 2 : ___________

C0 :              <          >  > <     B0 : LS          (_)
C1 : voltage      <123456___>  > <     B1 : S           (_)
C2 :              <          >  > <     B2 : HC          (_)
C3 :              <          >  > <     B3 : HCT         (_)
C4 :              <          >  > <     B4 : F           (_)
C5 :              <          >  > <     B5 :             ( )
C6 :              <          >  > <     B6 :             ( )
                                        B7 : advanced    (_)
```

--> In "Body" and "Graphic 1", specifications **must** be made;
--> In "Function type" and "Graphic 2", specifications **may** be made.

Make your entries in the respective fields and press <CR>. Then you get to the lower part of the menu where you can additionally specify 'dynamic parameters' (ref. to BTOOLS, function "Define user") for the symbol. The 'dynamic parameters' represent electrical selection criteria.

In Cn (n=0-6), electrical characteristics defined in BTOOLS appear that are connected with area specifications (e.g. voltage check, etc). In the brackets, you specify the respective area variants by entering a letter or a number (e.g. "123456" in "C1"); In "> <", you can specify the desired area by entering one of the characters.

In Bn (n=0-7), electrical characteristics defined in BTOOLS appear that may exist or not (e.g. "LS", "HCT", "F", S", "advanced" etc). If the characteristic shall exist for the current symbol, put an X in the respective parenthesis.

After pressing <CR>, you get back to the menu part where you may enter the symbol body, graphic and function type for further assignments.
Press <ESC><ESC> to finish the procedure. The symbol name and up to ten assignments will be displayed in the lower part of the screen.
With <ESC><ESC>, you get back to the upper part of the menu.

>E< Edit

The specifications assigned to a symbol can be changed. Therefore, you have to mark the respective assignment line in the lower part of the screen. If more than ten assignments exist for a symbol and if the respective assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.

>L< Kill

With this function, specifications assigned to a symbol can be deleted. After entering "3" in "Select", you have to mark the respective assignment line in the lower part of the screen and press <CR>. If more than ten assignments exist for a symbol and if the desired assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.
If all assignments to a symbol have been deleted, the respective symbol name does not appear when listing the names (function "1") in menu 46.

>+<

If more than ten assignments exist, you can turn to the next page with "+", i.e. up to ten further assignments will be listed in the lower part of the screen.

>-<

You can turn to the previous page (ref. also to >+<).

3.4 Function types

With the functions of the following menu, you can define, duplicate, rename, list and delete function types and stock IDs. Device parameters for assembler and tester may be assigned to the stock IDs.

In order to **define a function type**, shapes (body + graphic 1) and symbols (body + graphic 1) that shall be assigned to the function type must exist in COLUMBINE!
The definition is made by specifying a function type name to which first one or several shapes (possibly with the defined stock ID), then one or several symbols, and subsequently the respective terminal attributes may be assigned. **(The complete definition of a function type corresponds to the relation symbol <--> component in CIS.)**

In this menu, the **definition of stock IDs for additional elements to shapes** (socket, screws, etc.) is also effected (ref. to page 17). Therefore, you have to enter the stock ID specification in "Function type or stock ID", put an X in ">_< stock ID?", enter "5" in "Function" and define the device parameters (see below).

After you have assigned a stock ID to a function type, you can also process it with the help of the assigned stock ID instead of the function type specification. Therefore you enter the stock ID in "Function type or stock ID" and put an X in the field ">_< stock ID?".

COLUMBINE Library editor Version:	**3**

FUNCTION TYPES

Function type or stock ID : ________________ >_< stock ID ?

(1) Names	**(5) Device parameters (only with stock IDs)**
(2) Copy	**(6) Shape assignment**
(3) Kill	**(7) Symbol assignment**
(4) Rename	**(8) Terminal attributes**
(0) End	

Function : 0

(1) Names

lists all (*) or specified (..*) function types of the library (by 'clicking' the function "next" with <LMB>, you get to the next page, with "previous", you get back and with "quit", you finish the listing).
By entering a respective specification and putting an X in the field ">_< stock ID ?", the desired stock IDs and the names of the function types to which they are assigned will be listed.

(2) Copy

transfers the shape and symbol assignment, as well as the terminal attributes of the specified function type to a new one, for which you have to enter a name.
The information of the old function type concerning tester and assembling automat and stock ID assignment will **not** be assigned to the new one!

(3) Kill

With this function, you can only delete function type names to which a shape, but **no** stock ID is assigned.
If a shape **and** a stock ID are assigned to a function type, you have to delete the stock ID (possible in this menu) or the complete assignment (possible in menu 151) first.
In order to delete a function type to which several stock IDs (and shapes) are assigned, you have to delete all assignments in the submenu to the function "Shape assignment" first.

(4) Rename

renames the specified function type. You are asked for the new name of the element. The function type still contains the shape and symbol assignment and the terminal attributes of the specified function type.
The information concerning tester and assembling automat, as well as stock ID assignment is still assigned to the function type.
The old name does no longer exist.

(5) Device parameters

With this function, parameters for assembling automats and tester can be created, changed, copied, deleted or listed. This function can only be called when specifying the respective stock ID and putting an X in the field ">_< stock ID ?".

notes:

a) The stock ID can stand for a function type with an assigned shape or for an 'additional element' of a shape:

- The stock ID assignment for a function type with a specified shape is made in connection with the shape assignment --> ref. to function "6".

- The stock IDs for the 'additional elements' to the shapes are defined here in menu 3 (ref. to page 22) The stock ID assignment for the 'additional elements' to the shapes (in menu 140) can only be effected after the respective stock IDs have been defined in menu 3 and device parameters have been assigned to them with function (5).

b) If device parameters have been assigned to a shape in menu 45, the shape specific device information will be overtaken, i.e. it need not be entered here (ref. to page 15)!

Therefore, the following menu appears:

COLUMBINE Library editor Version:	**162**

DEVICE PARAMETER   Mode :

(E) Edit, (K) Kill, (L) List, (C) Copy,   select: >E<

device   :   # :   type :   (2 = Ass., 3 = Test)

shape   :

(E) Edit

With this function, you can assign assembling or tester data to the specified shape or change the data. In "Device name", you enter the name of the assembling automat or tester for which the parameters shall be created or changed (the device must be defined in the library area "Device"). In "Type", you enter the device type (2 = assembling automat, 3 = tester), <CR>.
Now you can enter the machine-specific parameters for the respective device (you may only use the kind of characters determined during the device definition in "Type"); the number of characters must not be greater than the number entered in "Len" during the device definition.

With <CR>, the current parameters are stored and with <ESC> <ESC>, you get back to the upper part of the menu.

(K) Kill

For the specified shape, the assigned device parameters of the tester or assembling automat whose name and type you have entered in "Device name" and "Type" will be deleted, <CR>.
With <ESC><ESC>, you get back to the upper menu parts.

(L) List

First the parameters of one device will be listed for the shape specified with the stock ID. After pressing any button, the parameters (if defined) of further devices will be listed.
With <CR>, you finish the listing and with <ESC><ESC> you get back to the upper part of the menu.

(C) Copy

The defined device parameters of the specified shape are copied to another device (tester or assembling automat). First you enter the device for which the parameters already exist, <CR>, then you enter the device that shall get the same parameters.
With <ESC><ESC>, you get back to the upper menu parts.

(6) Shape assignment

Here, you can assign one or several shape body (bodies) together with their graphic display to the specified function type and change or delete an existing assignment.
For each shape, a stock ID can be specified --> in this way, the stock ID is defined.
Subsequently, additional specifications to the 'dynamic parameters' (physical criteria for the component selection) can be made.
The already existing assignments will be listed below the menu.

```
        COLUMBINE     Library editor     Version:          | 151

  Function type                   (E) Edit         (N) New
  _____                           (L) Kill        (+-) Step
                                  >1< Select

E   Stock ID                Body      Graphic 1       Graphic 2
```

>N< New

An additional menu part appears, in which you can assign a shape body, its graphic display(s) and a stock ID to the specified function type (shape body and graphic must exist in COLUMBINE with the name entered in the fields "Body" and "Graphic 1" (or "Graphic 2"); the stock ID is defined here and must not exist in COLUMBINE).

```
  Function type   : ____________
  Stock ID        : ____________      Body       : ___________
  Graphic 1       : ____________      Graphic 2  : ___________

  C0 : temperat.    <1234      >  >_<     B0 : SO          (_)
  C1 : tolerance    <ABCDEFGHI>   >_<     B1 : PLCC        (_)
  C2 :              <          >  > <     B2 : standard    (_)
  C3 : voltage      <3456789   >  >_<     B3 : socket      (_)
  C4 :              <          >  > <     B4 :             ( )
  C5 :              <          >  > <     B5 :             ( )
  C6 :              <          >  > <     B6 :             ( )
                                          B7 :             ( )
```

Make your entries in the respective fields and press <CR>. Then you get to the lower part of the menu where you can additionally specify 'dynamic parameters' (ref. to BTOOLS, function "Define user") for the component selection.
In Cn (n=0-6), physical component characteristics defined in BTOOLS appear that are connected with area specifications (e.g. voltage check, tolerance, temperature, etc). In the brackets, you specify the respective area variants by entering a letter or a number (e.g. "C1: TOLERANCE <1234____>" means that the tolerance areas 1,2,3 and 4 are valid). In ">_<", you can specify the desired area by entering one of the characters.

In Bn (n=0-7), component characteristics defined in BTOOLS appear that may exist or not (e.g. "PLCC", "SO", "socket", "standard", etc). If the characteristic shall exist for the current symbol, put an X in the respective parenthesis.

After pressing <CR>, you get back to the menu part where you may enter the symbol body, graphic(s) and stock ID for further assignments.
Press <ESC><ESC> to finish the procedure. The symbol name and up to ten assignments will be displayed in the lower part of the screen.
With <ESC><ESC>, you get back to the upper part of the menu.

>E< Edit

The specifications assigned to a symbol can be changed. Therefore, you have to mark the respective shape assignment in the lower part of the screen. If more than ten assignments exist for a function type and if the respective assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.

>L< Kill

With this function, specifications assigned to a function type can be deleted. Therefore, you have to mark the respective assignment in the lower part of the screen. If more than ten assignments exist for a function type and if the desired assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.
If all assignments to a function type have been deleted, the respective function type name does not appear when listing the names (function "1") in menu 3.

>+<

If more than ten assignments exist, you can turn to the next page with "+", i.e. up to ten further assignments will be listed in the lower part of the screen.

>-<

You can turn to the previous page (ref. also to >+<).

(7) Symbol assignment

With this function, you can assign symbol numbers to a function type and symbols to the symbol numbers --> if several logical function elements exist in a component, each function element must have a number (e.g. 4 NAND symbols with no. 1 to 4 for the function type 74LS00). The symbols must exist in COLUMBINE with the specified names (ref. to the function "Logic", page 18).

```
COLUMBINE      Library editor      Version:      | 153

Function type:                 (E) Edit
____                           (L) Kill        (+-) Step
                               >E< Select

# : __          Symbol : ________________
```

The existing assignments are listed below the menu. The function type name automatically appears in the menu. After you have selected the respective function, you get to the input field for the number. With <ESC><ESC>, you leave the menu.

>E< Edit

To create a new assignment, enter the symbol number, <CR>, and the name of the desired (existing!) symbol. After pressing <CR>, the symbol number and name appear below the menu.
In order to change an existing assignment, enter the respective symbol specification and press <CR>. Subsequently, the changes can be executed.
If more than ten assignments exist for a function type and if the respective assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.

>L< Kill

With this function, specifications assigned to a function type can be deleted. Therefore, you have to enter the respective number in "Symbol no." and press <CR>.
If more than ten assignments exist for a function type and if the desired assignment is not displayed on the screen, you have to use the functions "+" or "-" to get to the next or the previous page.

>+<

If more than ten assignments exist, you can turn to the next page with "+", i.e. up to ten further assignments (symbol numbers and symbol names) will be listed in the lower part of the screen.

>-<

You can turn to the previous page (ref. also to >+<).

(8) Terminal attributes

With this function, you can assign a name for the layout (PCB) to the single terminals of a function type, a name for the schematics (CAE), and the desired connection width for the tracks to be digitized or routed. In this way, you define a terminal relation between shape and symbol terminals that corresponds to a respective relation in CIS. Already existing relations can be changed. The specifications of this menu will then automatically be taken into the netlist in CIS.

COLUMBINE	**Library editor**	**Version:**	**160**

```
Terminal           :                            Symbol #     :
PCB - name         :                            Symbol name  :
CAE - name         :
Connection width   :           in (mm)
Signal             :
Function type :                                      Shape:
Ed  Shape        Stock ID
_
_
```

If several shapes are assigned to the function type, move the cursor to the shape whose terminal(s) shall be assigned or whose assignment shall be changed and press <CR>.
In "Terminal", you enter the specification of the terminal to be assigned. If the terminal shall belong to a signal, enter the desired number in "Symbol number" (a respective assignment to the function type must have been created with the function "Symbol assignment" first).
After pressing <CR>, the symbol name is automatically entered and you can edit the following fields:

PCB name: The specification of this terminal for the layout (e.g. IN1).

CAE name: The logical terminal specification for the schematics (e.g. I1). If no symbol number is specified (i.e. symbol no. = 0), the terminal cannot get a CAE name.

Connection width: If a signal name has been entered in "Signal", you can specify the width of the track type here with which the terminal is to be connected.

Signal: Here, you can specify the name of the net (signal name) to which the entered terminal is to be connected.
This function only makes sense for the specification of plane connections. In this case, you need not enter a symbol number (i.e. "0" in "Symbol no.") and CAE name.

With <CR>, the entered terminal 'attributes' are stored and other terminals of this function type can be related.
With <ESC><ESC>, you leave the menu.

3.5 Devices

In this part of the library, you can define all devices (plotter, tester, assembling automat, NC-routing and drilling machines), i.e. machine specific parameters, control files, drivers, etc. will be assigned. Parameters may also be changed or deleted and the defined devices can be listed.

The following menu appears:

COLUMBINE Library editor Version:	88

DEVICES

Device name :______ Type:_ (1) Plot, (2) Assemble, (3) Test
(4) Drill, (5) NC-routing, (6) List

(1) Names	**(5) Edit parameters**
(2) Copy	**(6) Variants of assembly machine**
(3) Kill	
(4) Rename	
(0) End	

Function : 0

(1) Names

Lists the names of all ("Device name:" *, enter a respective number in "Type") or specified devices. By 'clicking' the function "next" with <LMB>, you get to the next page, with "previous", you get back and with "quit", you finish the listing.

(2) Copy

Duplicates the specified device. You are asked for the name of the new device. The information of the old device will also be assigned to the new one.

(3) Kill

Deletes the specified device from the library. The specification is made with a respective entry in "Device name" and "Type".

(4) Rename

Renames the specified device. You are asked for the new name of the device. The device still contains the assigned information.

(5) Edit parameters

With this function, you can set or change parameters for the device entered in "Device name" and specified by the number entered in "Type".

Different menus appear for the different types of devices.

3.5.1 Device parameters for plotter

After specifying the plotter name, the selection number (1) in "Type" and "5" in "Function" in menu 88, you get to menu 89 for the parameter definition for this device:

```
        COLUMBINE     Library editor     Version:              | 89

DEVICE PARAMETERS for plotter: ____                        # :__

   Driver                       :   ________
   Plotter Unit                 :   0          (0=mm, 1=inch)
   Step Width                   :   0.00000    (plotter units)
   Area                         : X ________     Y _______      in (mm)
   maximum mask #               :   ___
   Plotter type                 :   1 (1=pen,2=photo,3=grid,4=laser
   Interface                    :   0 (0=online,1=offline,2-4=offline
   Interface type               :   0 (0=binary,1=ASCII)
   Comment                      :   _____________________________

   Pre-commands                 :             Program ID    : 0

Step   (+-)   > <
```

- **Driver:**
  Here, you enter the name of the driver file (without extension!) for the specified plotter.

- **Plotter Unit:**
  The number determines the plotter unit.

- **Step Width:**
  Specify the step width - in the plotter unit determined in the previous line - for the specified plotter (according to the data in your plotter manual).

- **Area:**
  Here, you define the maximum size of the plotting area (i.e. the maximum sheet or foil size).

- **maximum mask #:**
  Here, you enter the maximum number of the masks/pens available in the plotter.

- **Plotter type:**
  Choose the desired plotter type by entering the respective number: 1 = pen, 2 = photo, 3 = grid, 4 = laser plotter; the selection numbers 3 and 4 are ineffective - only the numbers for pen and photo plotter are considered; if "1" is specified, flash masks cannot be generated.

- **Interface:**
  This field is not yet evaluated.

- **Interface type:**
  This field is not yet evaluated.

- **Comment:**
  Here you have the possibility to make notes, etc.

- **Program ID:**
  This field is not yet evaluated.

When you press the <CR> key, three lines appear for the input of pre-commands.

- **Pre-commands:**
  With these commands, you can directly contact the plotter interface (e.g. for the presetting of the light value for a photo plotter). These fields should only be changed or extended in consultation with the distributor.

- **Step (+-)   > <**
  If "+" is entered, further pre-commands exist (with <CR> you turn to the next page).
  If you want to edit further pre-commands, but there is no free line any more, you can generate further pre-command lines by entering "+", <CR>.

With <ESC><ESC>, you get from the above mentioned menu to a menu where you can assign masks and symbols (see next page). Here, you newly define the parameters of the plotter masks/pens (number, size, form and kind of employment), or you change the parameters of masks already defined.
These specifications are needed by OUTPUT for the execution of a correct automatic mask selection for the representation of the board elements.

COLUMBINE Library editor Version:	98

MASK DEFINITIONS

Mask # : ___

inner size	**: X 0.000**	**Y 0.000**	**in (mm)**
outer size	**: X 0.000**	**Y 0.000**	**in (mm)**

Mask type : _ **flash**
x plot

Mask shape : _ **rectangle**
_ **circle**
_ **octagon**
_ **square**
_ **special**
x symmetric

If masks have already been defined for the current device, these may be called by pressing <CR> (incl. the defined sizes). If the user enters the number of an already defined mask plus <CR>, the defined parameters of the mask are entered into the menu and may be changed.

- **Mask #:**
  For new definitions the user assigns numbers to the masks/pens of the plotter.
  For predefined masks applies: the entry of the number plus <CR> results in the automatic entry of the mask parameters.

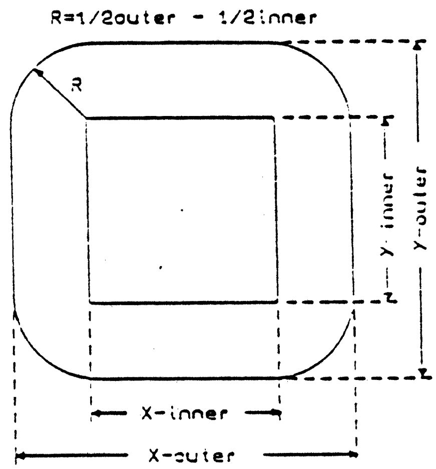
The entry of <u>two</u> values for a mask is necessary to provide the data not only for the size of the mask but also for the 'shape' (ref. to definition of the track and via shape --> CIS description).

- **Outer size:**
  The outer diameter is the real diameter of the mask; at present only symmetric masks (same values for X and Y) may be defined.

- **Inner size:**
  With this size and the specification in "Outer size", the geometrical shape of a mask is determined (ref. to the illustration on the next page).
  At present, only symmetrical masks (same value in X and Y) can be defined.

Temporarily, the following mask shapes are possible:

Inner size X=Y=0, outer size X=Y > 0	-->	round
Inner size X=Y = outer size X=Y	-->	square
Inner size X=Y < outer size X=Y	-->	square with rounded corners



- **Mask type:**
  By marking this field with a cross, you specify whether the mask is to be used for plotting and/or for flashing. E.g. a mask can only be defined as a flash mask for OUTPUT if "flash" has been marked with a cross in this menu.

- **Mask shape:**
  Based on the specification in "inner size" and "outer size" the system recognizes the mask shape and automatically marks it with a cross (at present only "circle" or "square"). The parameter "symmetric" is always marked, since at present, only symmetric masks can be defined.

  The parameter **special** may be marked with a cross by the user for marking special masks. Masks for which this parameter is marked, may only be specified manually, i.e. --> during the automatic mask selection these 'special' masks are not considered.

3.5.2 **Parameters for assembly machines** (not yet implemented)

After entering the name of the assembly machine, "2" in "Type" and "5" in "Function" in menu 88, the following menu appears for the definition of machine-specific parameters:

COLUMBINE Library editor Version:	142

DEVICE PARAMETERS for assembly machines : __ # :__

Driver : ________

System : 1 (0=abs, 1=rel)
Code : _ (0=EIA, 1=ISO, 2=ASCII, 3=ASCII2, 4=BSH,5=EBCDIC)

Zero point : X __________ Y __________ in (mm)

Step length : _________ in (mm) / 10000

Comment : ___________________________________

- **Driver:**
  Here, you enter the name of the driver for the specified assembling automat, <CR>. The data contained in the driver is automatically entered in the respective fields of this menu.

  Attention:
  For changing the entries, please refer to the manual of this machine (if you have doubts or problems, contact the distributor).

- **System:**
  Here, you enter if the coordinates for the automat are to be specified absolutely or relatively.

- **Zero point:**
  A new zero point for the machine relative to the absolute zero point can be defined (X=0, Y=0 as a rule).

- **Step length:**
  Defines the smallest step length of the specified assembling automat (ref. to the manual of the automat).
  The specification is made in 1/10,000 mm!

- **Code:**
  Defines the data format (ref. to the manual of the machine).

- **Comment:**
  Here you have the possibility to make notes, etc.

After you have saved your entries with <CR>, the following menu appears:

COLUMBINE	Library editor	Version:	5

Device name : _____ **# :** __ **Type : 2**

specification	basic parameters	Un	Len	Type
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___
____________	______________________	__	__	___

The specifications in this menu are entered automatically according to the information contained in the driver. For changing the entries, please refer to the manual of the assembling automat (if you have doubts or problems, contact the distributor).

- **Un (Unit)**
  defines the text data unit

  0 = no unit is defined
  1 = 0.0001 inch
  2 = 0.001 inch
  3 = 0.01 inch
  4 = 0.1 inch
  5 = 0.0001 mm
  6 = 0.001 mm
  7 = 0.01 mm
  8 = 0.1 mm

- **Len (Length)**
  Here, you enter the number of characters specified in "Basic parameters".

- **Type**
  defines the kind of character of the "Basic parameters". It is entered as the sum of the below mentioned character numbers (e.g. "7" means, all capital and small letters as well as blanks are allowed).

```
  1 = capital letters
  2 = small letters
  4 = blanks
 10 = numbers
 20 = sign characters
 40 = period
100 = special characters
```

3.5.3 Variants of assembly machine

After entering the name of an already defined insertion machine, "2" in "Type" and "6" in "Function" in menu 88, the following menu appears in which different 'variants' for the magazine assignment of the specified assembly machine can be defined, duplicated, renamed and deleted.

COLUMBINE Library editor Version:	91

DEVICES

Assembly machine : _____   **Variant # :** ___
                                      New # : ___

(1) Names	**(5) Define variant**
(2) Copy	
(3) Kill	
(4) Rename	
(0) End	

Function : >0<

After entering the desired number in "Variant #" and "5" in "Function", a magazine assignment variant for this assembly machine can be defined in menu 92 (see next page).

The definition is made by assigning magazine numbers (in "M #") to the stock IDs of the respective element (in "Stock ID").

COLUMBINE Library editor Version:	92

VARIANT DEFINITION

```
Assembly machine: ____   Variant # : ___
Function        : >E<  (N) New, (E) Edit, (K) Kill, (+/-) Step

M # : ______             St ID : ________________
```

-->
-->
-->

- By entering "N", you define a new assignment that will be listed below the menu.
- With "E" (Edit), you get to the listing below the menu where you can change the selected assignment.
- By entering "K", the selected assignment will be deleted.

3.5.4 **Parameters for tester** (not yet implemented)

After entering the name of the tester, "3" in "Type" and "5" in "Function" in menu 88, you get to menu 5 for the definition of the specific parameters for this device.
The device parameters to be entered in this menu can be taken from the manual of the test automat (if you have doubts or problems, contact the distributor).

COLUMBINE **Library editor** **Version:**	**5**

Device name : _____ **# :** __ **Type : 3**

specification	**basic parameters**	**Un**	**Len**	**Type**
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__
________	________________	__	__	__

- **Un (Unit)**
  Defines the text data unit
  - 0 = no unit is defined
  - 1 = 0.0001 inch
  - 2 = 0.001 inch
  - 3 = 0.01 inch
  - 4 = 0.1 inch
  - 5 = 0.0001 mm
  - 6 = 0.001 mm
  - 7 = 0.01 mm
  - 8 = 0.1 mm

- **Len (Length)**
  Here, you enter the number of characters specified in "Basic parameters".

- **Type**
  Defines the kind of character of the "Basic parameters". It is entered as the sum of the below mentioned character numbers (e.g. "7" means, all capital and small letters as well as blanks are allowed).

```
  1 = capital letters
  2 = small letters
  4 = blanks
 10 = numbers
 20 = sign characters
 40 = period
100 = special characters
```

3.5.5 Parameters for drilling machines

After entering the name of the drilling machine, the selection number (4) in "Type" and "5" in "Function" in menu 88, you get to menu 143. After entering the control file name and <CR> the parameters defined in this control file are automatically entered into the menu.
The values are only for information, they may not be changed.
User entries are possible only in "control file name" and in "comment".

Note:
If you want to read in a changed or new control file for an already defined drill machine, that device has first to be deleted and has then to be newly defined under the same name. Only then you may enter the name of the control file in the menu 143.

COLUMBINE Library editor Version:	143

```
DEVICE PARAMETER for drill/NC-routing : ______        # :__

Control file name: _______

System         : 1 (0=abs, 1=rel)
Code           : _ (0=EIA, 1=ISO,2=ASCII,3=ASCII2,4=BSH,5=EBCDIC)

Zero point     : X _________    Y _________     in (mm)

Area in        : X _________    Y _________     in (mm)

Step length    :   ________                     in (mm) / 100000

Maxtools       :   ___

Drill diameter :   ________                      in (mm) / 100000

Comment        : ________________________________
```

- **Control file name:**
  Here, you enter the name of the control file convenient for the specified drilling machine.
  After pressing <CR> the parameter values of the control file appear in the appropriate fields.

- **Step length:**
  Defines the smallest step length of the specified drilling machine in 1/100,000 units (ref. to the manual of the machine).

- **Area in:**
  Here, you define the maximum size of the drilling area.

- **Maxtools:**
  Here, you enter the maximum number of drills which fit in the magazine.

- **Code:**
  Defines the data format for this machine (ref. to the manual of the machine).

- **System:**
  Here, you enter if the coordinates for the machine are to be specified absolutely or relatively.

- **Zero point:**
  A new zero point for the machine relative to the absolute zero point can be defined (X=0, Y=0 as a rule).

- **Unit: Drill diameter**
  Defines the size unit (= the smallest possible difference of two drills), in which drill diameters are specified (in 1/100,000 units!).

3.5.6 Parameters for NC-routing machines

After entering the name of the NC-routing machine, the selection number (5) in "Type" and "5" in "Function" in menu 88, you get to menu 142.
After entering the control file name and <CR> the parameters defined in this control file are automatically entered into the menu.
The values are only for information, they may not be changed.
<u>User entries are possible only in "control file name" and in "comment".</u>

COLUMBINE Library editor Version:	143

```
DEVICE PARAMETER for drill/NC-routing : ______        # :__

Control file name :_______

System          : 1 (0=abs, 1=rel)
Code            : _ (0=EIA, 1=ISO,2=ASCII,3=ASCII2,4=BSH,5=EBCDIC)

Zero point      : X _________     Y _________     in (mm)

Area in         : X _________     Y _________     in (mm)

Step length     :   ________                      in (mm) / 100000

Maxtools        :   ___

Drill diameter  :   ________                       in (mm) / 100000

Comment         : ______________________________
```

- **Control file name:**
  Here, you enter the name of the control file convenient for the specified NC-routing machine.
  After pressing <CR> the parameter values appear in the appropriate menu fields.

- **Step length:**
  Defines the smallest step length of the specified NC-routing machine (ref. to the manual of the machine).

- **Area in:**
  Here, you define the maximum size of the NC-routing area.

- **Maxtools:**

Here, you enter the maximum number of NC-routing heads which fit in the magazine.

- **Code:**
  Defines the data format for this machine (ref. to the manual of the machine).

- **System:**
  Here, you enter if the coordinates for the machine are to be specified absolutely or relatively.

- **Zero point:**
  A new zero point for the machine relative to the absolute zero point can be defined (X=0, Y=0 as a rule).

- **Unit: Drill diameter**
  Defines the size unit (= the smallest possible difference of two NC-routing heads), in which NC-routing heads are specified.

3.6 Graphic macros (for plotter)

A precondition for working with this function is that the library editor has been called with "2" = "library editor (graphic window)" of the LIBRARY selection menu.

With this function, you can define symbols (macros) for the display of layout elements on the plotter.
The following menu appears:

COLUMBINE Library editor Version:	40

GRAPHIC MACROS

Macro - name : ________________________

(1) Names **(5) Edit macros**

(2) Copy

(3) Kill

(4) Rename

(0) End

Function : 0

Macro - name:

Here, you enter the name of the macro to be defined or changed.

(1) Names

Lists the names of all (*) or the specified (..*) macros that have already been defined.

(2) Copy

Duplicates the macro specified in "Macro - name". You are asked for the name with which the new macro shall be saved in the library.

(3) Kill

Deletes the specified macro from the library.

(4) Rename

Renames the specified macro. You are asked for the new name of this macro.

(5) Edit macros

Calls the graphic macro editor.
Subsequently, graphic screen and mouse are activated for the digitizing of the macros.
The following menu appears:

COLUMBINE Library editor Version:	**20**

MACRO EDITOR **Macro name :** ________________	**Mouse - buttons :** **0 : Position** **1 : Draw line** **2 : Delete line** **<CR>: End**
Window definition : **lower left corner X: -10__ Y: -10__** **upper right corner X: 10___ Y: 10___**	**Current position :** **X : Y :**

First, only the upper part of the menu appears. After you have entered the symbol name, <CR>, the lower menu part is called.

- **Symbol name:**
  Here, you enter the desired name. If you have already entered a symbol name in the previous menu, this name will be taken here. The lower part of the menu appears after the entry of the symbol name and <CR>.

- **Window definition**
  Here, you enter the coordinates of the lower left and the upper right corner of the window in which the symbol is to be edited.

As by definition the origin of the coordinates lies in the centre of the rectangular window, the coordinates of the lower left corner must have negative signs and the same number values as the coordinates of the upper right corner.

- **Current position:**
  Specifies the coordinates of the current mouse position, referring to the centre of the window (=zero point).

- **Mouse-buttons:**
  Shows which functions are assigned to the mouse buttons for the edition process.

  0: Position (= left button)
  By pressing the left mouse button, you determine the current mouse position as starting position of a line.

  1: Draw line (= middle button)
  By pressing the middle mouse button, a line is drawn between the starting position and the current mouse position (=end of the rubber thread). The end of this line automatically becomes the starting point of the next line. If you want another starting position, move the rubber thread to the desired point and press the left button.

  2: Delete line (= right button)
  By pressing the right mouse button, the last digitized line will be deleted. By pressing this button again, the line before will be deleted, etc.

  <CR>: End
  By pressing <CR>, the edition process for this load symbol is finished and the macro is stored. When entering a new symbol name, another symbol can be edited.

  <ESC><ESC>:
  With this entry, you interrupt the current edition process (without storing it!) and get back to menu 40.

BTOOLS

1. Introduction

The program BTOOLS is used for the generation (reservation of the necessary space) of the central PRISMA library **'columbine'** and the procedure library **'output.sys'**, as well as the administration of the library 'columbine' (access definition, etc.). This program can only be called by the system administrator.

2. Handling

The program is started by clicking on the **LIBRARY** icon and activating number 3 ('library maintenance') in the selection menu.
After pressing <CR>, the following menu appears:

B T O O L S - LIBRARY ADMINISTRATION Version : _______	3

Library name : : columbine_____.___

(1) Generate
(2) Squeeze
(3) Define user
(4) Delete user
(5) List users

(0) End

Function : >0<

First only the upper part of the menu (heading) appears (as menu 2).
You can now enter the name of the library file to be created or processed (the standard name appears as the default).
If you want to create an additional library file, you have to specify another name.

Attention:

The central library file used for the programs CIS and OUTPUT must either have the standard name or a link to the standard name must exist, otherwise the programs cannot use it.
The procedure library must be named 'output.sys' or must be linked to it so that it is found by the program OUTPUT.

Functions:

In order to activate the desired function, you have to enter the respective number. Do not press <CR> afterwards!

(1) Generate

With this function, you create a new library file and reserve the necessary space on the device specified in the menu header. After entering "1" in "Function" (no <CR>!), the following menu appears:

B T O O L S - LIBRARY ADMINISTRATION Version : _______	**4**

Library name : : _________ . ___

Size of the data area : 100__ blocks (min.: 100 max.: ____)

Size of the key chart : _____ blocks (min.: ____ max.: ____)

In 'library name', the name specified in menu 3 appears.
The number of blocks for the new library file, which you have to enter in "Size of the data area", must be within the limits that appear in parentheses (in 'min.:', the minimum number of necessary free blocks and in 'max.:', the available number of free data blocks on the directory 'lib/sys' appears.
After the entry and <CR>, you get to the next line.

The size of the key area lies between 4% and 20% of the blocks specified in "Size of the data area". The limits of the key area are automatically calculated by the system and appear in parentheses. The number of blocks which you have to specify for the key chart must be within these limits.

(2) Squeeze

If you want to store new data in an existing library file but there is not enough space in the file, you get the following message:

"Not enough space. Please squeeze!"

With the "Squeeze" function, the different free blocks are combined to one free space.
After entering "2" in "Function" (no <CR>!), the inquiry

"Please confirm "SQUEEZE" (Y/N)"

appears. For the execution of the function, you have to enter -Y-.

Note:

If the message "File inconsistency ...." appears on the screen while working with the COLUMBINE editor, you have to leave the program and squeeze the 'columbine' library. Then you can restart COLUMBINE.

Attention:

The squeezing must not be interrupted, otherwise the library data might be lost.

(3) Define user

With this function, maximal 255 users can get the access to the library file entered in the menu heading by specifying a user name and a password.

After entering "3" in "Function" (no <CR>!), the following menu appears:

B T O O L S - LIBRARY ADMINISTRATION Version : _______	**5**

Library name : : columbine_____.___

User name : _________

The name which you have to specify here can have up to 9 alphanumeric characters.
After entering the user name and <CR>, an additional menu part for the determination of the access rights is called:

```
ACCESS RIGHTS        Password :  ________
                                 1 2 3 4 5 6 7 8
             Group assignment :  _ _ _ _ _ _ _ _
-----------------------------------------------------
             Components       :       (X = Write)
             Shape            :          ...
             Technology       :          ...
             Device           :          ...
             Assignments      :          ...
             Graphics         :          ...
             Others           :          ...
             Others           :          ...
```

The password may consist of up to 9 alphanumeric characters.

Note:
The password you have to enter when calling the COLUMBINE program must exactly correspond to the one you enter here, also in regard to upper and lower case. Otherwise you will get the message "User not in the list".

The assignment of the users to user groups with the same access rights ("Group assignment") and the specification of the access rights for the library areas listed in the lower part of the menu has not yet been implemented.

After pressing <CR>, another additional menu is called in which you can determine further selection criteria for the access rights of the specified user:

```
DYNAMIC COMPONENT SELECTION:        >P<  (P = PCB, C = CAE)

     C0 : _________   <_________>    B0 : _________
     C1 : _________   <_________>    B1 : _________
     C2 : _________   <_________>    B2 : _________
     C3 : _________   <_________>    B3 : _________
     C4 : _________   <_________>    B4 : _________
     C5 : _________   <_________>    B5 : _________
     C6 : _________   <_________>    B6 : _________
                                     B7 : _________
```

By entering the respective selection letter (without pressing <CR>!), you determine the access area (PCB or CAE data) of the library.
In "Cn" (n=1-6) you enter component characteristics (max. 9 alphanumeric characters) that are connected with area specifications (e.g. voltage check, tolerance, amplifier gain, etc.).

In the brackets, you specify the respective area variants by entering a letter or a number (e.g. C1: TOLERANCE <1234____> means that the tolerance areas 1,2,3,4 are valid).

In "Bn" (n=1-7) you enter component characteristics (max. 9 alphanumeric characters) that may exist or not (e.g. "on socket", "in pairs", etc). If the characteristic shall exist, put an X in the respective parenthesis.

After pressing <CR>, the cursor moves back to the field for the input of the data area selection and you can execute the same procedure for the second data area.

With <ESC><ESC>, you get back to the part of the menu in which you enter the user name. You may now define a new user and his access rights.
When leaving this menu part with <ESC><ESC>, you get back to menu 3.

(4) Delete user

With this function, you can delete existing user definitions. When entering "3" (no <CR>!), menu 5 appears.

After entering the defined user name and <CR>, the user will be deleted from the list of the accessors.

With <ESC><ESC>, you get back to menu 3.

(5) List users

If you enter "5" in "Function" in menu 3, you get a list of all users and their passwords.
By entering a respective number in "USRID", the succession of the defined users is shown.
The group assignment specified in menu 5 will be listed in "GROUP".

REPORT

section1
1. Introduction

With this program, you can create lists with information about the function types, shape and symbol bodies, symbols, stock ids, load symbols, devices contained in the library COLUMBINE.

The program is started by clicking on the **LIBRARY** icon and activating number 4 in the selection menu.
sectionend

section2
2. User specification

After the program has been started, the following menu appears:

REPORT	**Library report generator**	**2**

Library report generator **Version :**

User : ___________

Here, you enter the name of the user and press <CR>.

Note:
The specified user must be defined in the program BTOOLS.

sectionend

section3
3. Area selection

Subsequently, menu 11 appears, where you can determine the library area from which the information shall be taken, specify the elements of this area, and enter name and extension of the output file in which the information is to be stored (see next page).
In the upper part of the menu, general information to the library appears (creation date, free memory capacity, number of entries, etc).

REPORT **Library report generator**	**11**

Creation date	**:** ________	**Free data blocks**	**:** ___	**(blocks)**
Last change	**:** ________	**Deleted blocks**	**:** ___	**(blocks)**
		Number of entries	**:** ___	
Version	**:** ________	**Deleted entries**	**:** ___	

**What is to be output ? ***______________________

F_Function type	**K_Body**	**P_Devices**
L_Macros	**I_Stock ids**	**S_Logic**
E_End		

Function : >E< Output : :OUT__________________

With most of the functions you get to subordinate menus (<u>without</u> pressing <CR>). With "E" or <ESC><ESC>, you return to the main menu.

When you activate a function in the submenus, you get back to the main menu.
To work with another function in the same submenu, you have to activate the respective library area in the main menu again.

Notes:

- The data for each function activated in a submenu is stored in the same output file OUT (Attention: uppercase!) on the **lib/sys** directory if you leave the predefined file specification unchanged. When creating a new report output file, the old file will be overwritten with the new data contents if the name of the output file (= OUT) is not changed.

- If you want to store the data of single functions in separate output files, enter different names for the output files in menu 11 (you may add different extensions).

  Attention:
  You only get to the input field for the file specification <u>before entering the function</u> (i.e. press the "arrow down" key twice)!

- You can edit or print the report output files if you want to read them (when using the standard name for the output file, you have to enter the file name OUT in uppercase!).

sectionend

section3.1
3.1 Function types

After specifying the desired function types in "What is to be output?" (e.g. "74*") and entering "F" in "Function", the following submenu appears:

```
Function type :

A Contents
B Body assignment
C Symbol assignment
```

Function : >A<   Output :   :OUT____________________

Here, you enter the desired function and the name of the output file.

A Contents

A list of the specified function types from COLUMBINE is created and written into the output file.

B Body assignment

A list of the specified function types and the shape bodies and stock ids assigned to them is created and written into the output file.

C Symbol assignment

A list of the specified function types, together with the assigned stock ids and symbols (with terminal relation and connections) is created and written into the output file.
sectionend

section3.2
3.2 Body

```
Body :

A Contents
B Device parameters
```

Function : >A<   Output : OUT:_______________________

A Contents

A list of the specified shape types from COLUMBINE is created and written into the output file.

B Device parameters

A list of the device parameters (for assemblers and testers) assigned to the specified shapes in COLUMBINE is written into the output file.
sectionend

section3.3
3.3 Devices

Devices :

A Contents
B Plotter descriptions
C CAM - device descriptions
D Lists

Function : >A<   Output : OUT:________________________

A Contents

A list of the specified devices from COLUMBINE is created and written into the output file.

B Plotter descriptions

The parameters (driver, type, interface, step width, size of the plotting area, masks and pre-commands) assigned to the plotters defined in COLUMBINE are output.

C CAM - device descriptions

The parameters (control file, code, step length, etc.) assigned to the assemblers and testers defined in COLUMBINE are output.

D Lists

sectionend

section3.4

3.4 Macros

For this function, there is no submenu. The names of all macros (=load symbols) defined in COLUMBINE are directly written into the specified output file.

sectionend

section3.5

3.5 Stock ids

Stock ids :

A Contents
B Function type assignment
C Symbol assignment
D Device parameters

Function : >A<   Output : OUT:________________________

A Contents

A list of the specified stock ids from COLUMBINE is created and written into the output file.

B Function type assignment

A list of the specified stock ids and the shapes and function types assigned to them is created and written into the output file.

C Symbol assignment

A list of the specified stock ids and the shapes and function types assigned to them is created and written into the output file.

D Device parameters

A list of the device parameters (for assemblers and testers) assigned to the shapes in COLUMBINE is written into the output file.

sectionend

section3.6

3.6 Logic

Logic :

A Contents
B Element assignment

Function : >A<   Output : OUT:________________

A Contents

A list of the specified symbols from COLUMBINE is created and written into the output file.

B Element assignment

A list of the symbols existing in COLUMBINE and the symbol bodies, symbol graphics and function types assigned to them is written into the output file.

C O N T E N T S

C A M

C A M

I N T R O D U C T I O N

The CAM-functions serve for optimizing the PCB layout as regards production and/or aesthetical aspects (e.g. "slanting of corners", "Center", "Generation of teardrops", etc.). It also serves for generating additional layout data, which are necessary for the generation of certain plots in the postprocessing ("solder resist plan", "drilling plan", "silk screen" and "layers").

CAM Main Menu

The main menu of the CAM function is invoked either through activating "CAM" in the main menu or through activating "CAM" in the PCB main menu.

Apart from functions the names and effects of which are similar to the layout main menu (e.g. "modes", "technology", etc.) the following functions can be found in the CAM main menu:

MODIFICATION

This function calls the submenu with the actual CAM functions. Within the CAM main menu the single functions can be called

- by clicking the function with <LMB>

or

- by calling the function in the pop-up menu (callable with <RMB>

or

- by entering the respective function short form in the status window in "fct."

PCB

This function switches back to the main menu of the PCB part.

SCS

With this function you get back into the main menu of the SCS part.
Note:
If you activate the function "LAYOUT" in the SCS part again, you will get into the main menu of the CAM part, not into the PCB part!

The functions of the CAM part the names of which are identical to the PCB part also have the same functionality.

Note:

* As a rule, the CAM functions should be used <u>after</u> finishing the layout (incl. routing), as a last preparation of the file for the postprocessing, since
  - the offgrid segments and vectors generated during some optimizing functions make subsequent editing very difficult in CIS/PCB.
  - as a rule the execution of the functions "create resist plan", "drill plan", "silk screen" are necessary for the postprocessing.

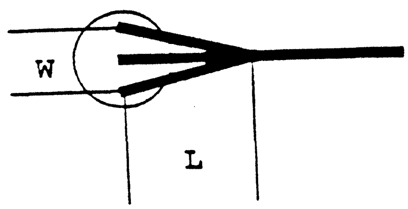
M O D I F I C A T I O N

CREATE/REMOVE TEARDROPS

CREATE TEARDROPS

Creates teardrops for specified pads and vias.

Teardrops may only be generated for pads and vias, not for SMD's.



After invoking this function a menu appears in the status window where the following parameters can be set:

1.	SIGNALS:	Specifies the signal or signals to be evaluated (* is used as a wildcard for all).
2.	EXCEPT:	Used when a wildcard has been specified in the signal menu. Signals entered here will be ignored.
3.	LAYERS:	Specifies the layers to be evaluated.
4.	TRACK TYPE:	Specifies the track type to be considered, i.e. teardrops are generated only for those vias and pads belonging to the respective signals which are connected with this track type.

this track type is then also used for the generation of the teardrops.

5. L: Specifies the "length" of the teardrops (=the distance between the center of the via and the point of intersection of the two teardrop segments).

6. W: Specifies the width of the teardrops (= distance between the ends of the two teardrop segments at the via). As a rule, this value should be:
width of the via type minus width of the track type specified in "trk.type";
the value should be smaller/equal to the width of the via type.

After all data is entered with <CR> the functions will be executed for the pads and vias concerned.

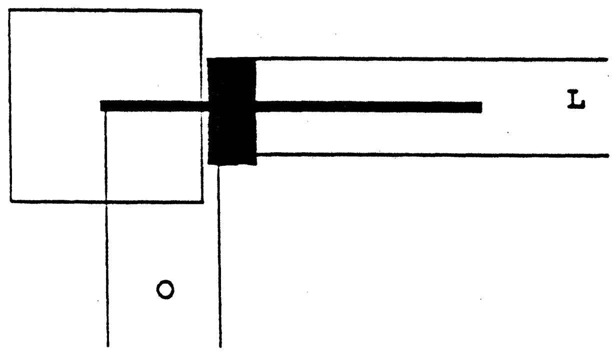
REMOVE TEARDROPS

After activating this function and after specifying the signal names, layers and via type, the teardrops are removed from the pads and vias concerned.

CREATE / REMOVE SUBLANDS

CREATE SUBLANDS

The "subland" function enables the user to generate additional copper (in the form of a segment horizontal to a track) at the point of intersection of track and pad or track and via. This function is only carried out for tracks which enter the vias orthogonally.



After activating this function, a new menu is displayed and the following parameters can be set:

1. SIGNALS: Specifies the signal or signals to be evaluated (* is used as wildcard).
2. EXCEPT: Used when a wildcard was used in the signals field. Signal names entered here will be ignored.
3. LAYERS: Specifies the layers to be evaluated.
4. VIA TYPE: Specifies the via type to be evaluated (no wildcards).
5. TRACK TYPE: Specifies the track type to be used for the generation of the sublands.
6. L (LENGTH): Specifies the length of the track to be added (as a rule, smaller than/equal to the via size).
7. O (OFFSET): Specifies the offset in reference to the point of intersection.

After all data is entered with <CR> the function is executed.

REMOVE SUBLANDS

This function serves for deleting existing sublands. Specify the signal name, layers and via type of the sublands to be deleted and enter <CR>.

REROUTING

This function serves for automatically rerouting very small, invisible airlines to vias with a respective track type. As a rule, the length of these airlines is smaller than a routing grid, and they may occur if big vias connected to a track are subsequently made smaller.

The function is automatically carried out when calling the CAM functions "Create Teardrops", "45° connections" and "Centered". This function should be used after reducing the via size in the technology.

After calling this function, a menu appears where you may specify the following parameters:

1. SIGNALS: contains the signal(s) to be evaluated ("*" may be used as a wildcard).
2. EXCEPT: If wildcards have been used for the signal specification, you may enter the signals for which this function is not to be executed.
3. MAXIMUM LENGTH: contains the maximum length of the airlines which are to be replaced by a segment (default length is the routing grid size). Airlines which are longer are not considered.

DELETE SEG. -> TERM.

This function serves for automatically deleting segments which are routed into big terminal vias.

It is automatically executed when calling the CAM functions "Create Teardrops", "45° connections" and "Centered". This function should be executed after routing an .adb-file.

After calling the function a menu appears where you may specify the signals for which the segment elimination is to be executed.

1. SIGNALS: contains the signal(s) to be evaluated ("*" may be used as a wildcard)

2. EXCEPT: if wildcards have been used for the signal specification, you may specify here the signals for which this function is not to be executed.

DELETE VIA -> TERM.

This function serves for deleting vias which the router sometimes places into terminal elements.

The function is automatically executed when calling the CAM functions "Create Teardrops", "45° connections" and "Centered". It should be executed after routing an .adb-file.

After calling this function a menu appears where you may specify the signals to which the segment elimination is to be applied.

1. SIGNALS: contains the signal(s) to be evaluated ("*" may be used as a wildcard)

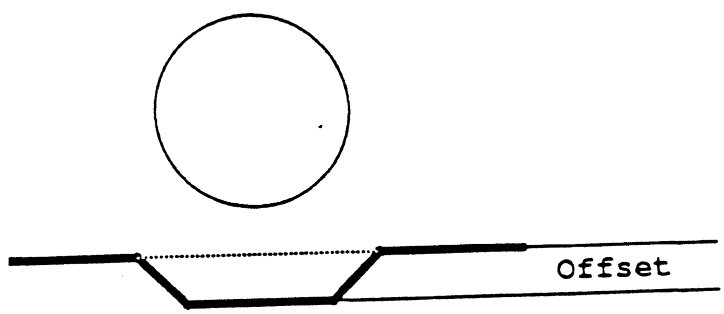
2. EXCEPT: if wildcards have been used for the signal specification, you may specify here the signals for which this function is not to be executed.

VIA DISTANCE

The via distance function allows the user to either globally or selectively generate a greater clearance for routed vias. The shift will only affect tracks which are within one grid point of the specified via type. Therefore, the function is only sensibly applicable for boards with 'micro' technology (via size .015 - .020 and correspondingly 'thin' tracks).

Note:

- Since this function is grid-bound, only on-grid elements are considered.
- The clearance around vias may be increased by a maximum of 1/2 routing grid.



The following parameters can be set in the menu.

1. SIGNALS: Specifies the signal or signals to be evaluated (* is used as a wildcard).

2. EXCEPT: Used when a wildcard has been specified in the signals menu. Signals entered here will be ignored.

3. LAYERS: Specifies the layers to be evaluated.

4. DISPLACE: Specifies the distance by which the track is to be shifted from the via. (NOTE: The max. distance allowed is 1/2 routing grid).

5. V-TYPE: Specifies the via type to be evaluated.

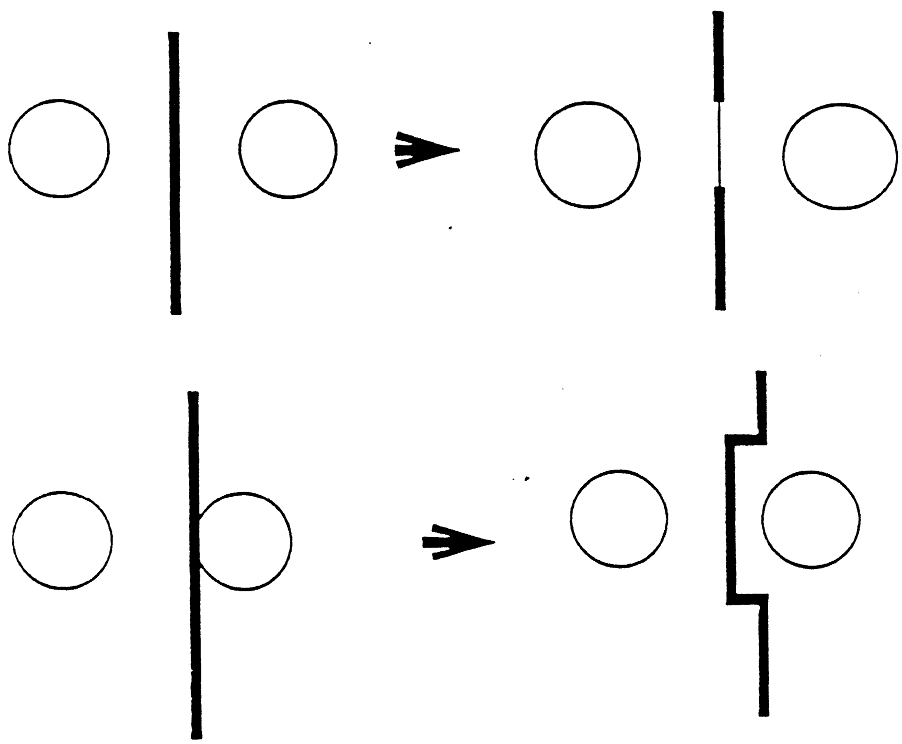
6. "ALL LAYER":

VIAS ONLY Specifies that only those vias with the attribute "D" in the technology list will be evaluated. No buried vias.

After all data is entered with <CR> the function will be executed.

NECKING

Used to correct technology violations (if the clearance between segments and pads/vias is not adhered to).
In a first step, the tracks violating the clearance are cut into technologically right segments (which consider the clearance) and into technologically wrong segments (which violate the clearance).
In a second step the technologically wrong segments are replaced by a smaller track type (if such a track type has been specified under "to trk type" and/or are shifted if required (if possible and if "shifted" has been marked with a cross).



In the status window a menu appears in which the necessary parameters can be set:

1. SIGNALS: Specifies the signal or signals to be evaluated (* is used as a wildcard).

2. EXCEPT: Used when a wildcard has been specified in the signals menu. Signal names entered here will be ignored.

3. LAYERS: Specifies the layers to be evaluated.

4. TRK.-TYPE: Specifies the track type to be evaluated.

5. TO TRK.-TYPE: Specifies the track type to be used for the technologically wrong segments.

6. SHIFT: If this field is selected, the respective tracks - if required and if possible - will be shifted after the replacement by smaller tracks in order to adhere to the clearance.

After the specifications and <CR> the function is executed.

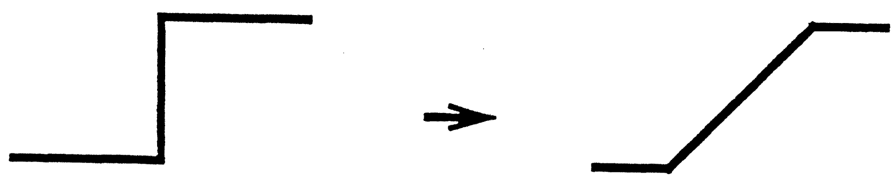
TECHN. CORRECTIONS

This function is used for correcting technology violations through shifting track segments.
The function has not yet been completely implemented. The clearance violation is only checked for terminals; a multiple shift function is yet to be implemented.

BEND SLANT

This function serves for changing corners of 90 degrees (= orthogonal track segments) into corners of 45 degrees - incl. the steps generated during routing (for the representation of slants).

Slanting is carried out under design rule check, i.e., where the action would lead to technology violations, it is not carried out.



The following parameters can be set in the status window:

1. SIGNALS: Specifies the signal or signals to be evaluated (* is used as a wildcard).

2. EXCEPT: Used when a wildcard has been specified in the signals menu. Signal names entered here will be ignored.

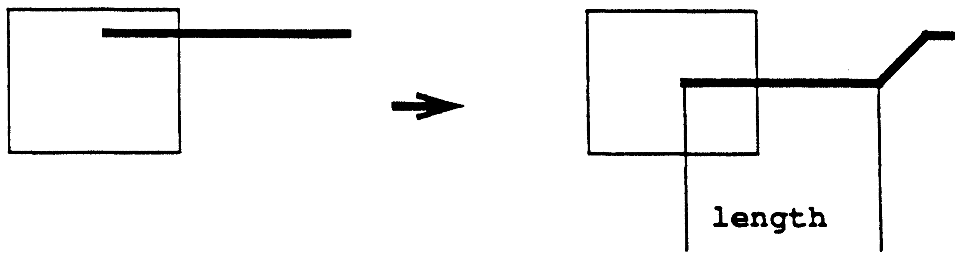
3. LAYERS: Specifies the layers to be evaluated.

4. LENGTH: Specifies - in units of 1/2 routing grid - the maximum length of corner segments (in x or y direction) up to which slants may be gener-ted.

5. SHOW GRAPHIC: If you would like to see the execution of the function on the screen, confirm the default with "Y".
Otherwise enter "N" and make a "refresh" after the function has been finished.
Note:
With big PCBs the graphic output should not be activated, since the time consumed would considerably increase.

CENTERED

The center function allows the user to either globally or selectively center "on-grid" tracks connected to "off-grid" pads, vias or SMD-pins (the centered segments are then "off-grid").



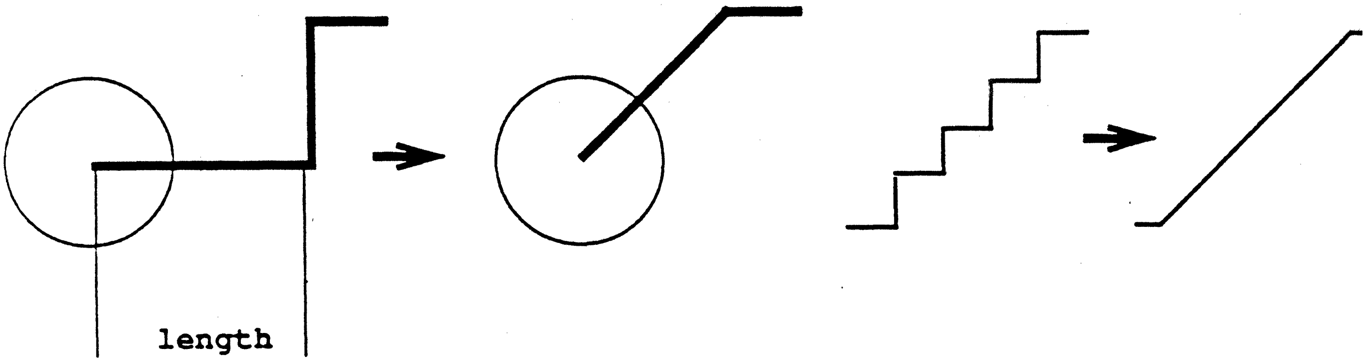
After calling the "centered" function a menu appears in the status window, where the following parameters may be set:

1. SIGNALS: Specifies the signal or signals to be evaluated. (* is used as a wildcard).
2. EXCEPT: Used when a wildcard has been specified in the signals menu. Signals entered here will be ignored.
3. LAYERS: Specifies the layers to be evaluated.
4. LENGTH: Specifies the length of the segment to be centered.
5. 45° Allowed: If this field is selected, the connection segments between the segments shifted and those not shifted are generated as 45 ° slants (instead of orthogonal segments) - if possible.

After all data is entered and <CR> the selected signals will be centered.

45° CONNECTIONS

Converts tracks which enter pads or vias at 0 or 90 degrees to 45 degrees.



The track segments may be changed globally or selectively to 45° slants.
After invoking this function, a new menu is displayed in the status window and the following parameters can be set.

1. SIGNALS: Specifies the signal or signals to be evaluated (* is used as a wildcard)

2. EXCEPT: Used when a wildcard has been used in the signals menu. Signal names entered will be ignored.

3. LAYERS: Specifies the layers to be evaluated.

4. LENGTH: Specifies the maximum length of the segment to be slanted (from the center of the pad or via). Connection segments which are longer than this value are not considered for this procedure.

5. V-TYPE: Specifies the via type to be evaluated (no wildcards).

6. "ALL LAYER": VIAS ONLY If this field is selected, only those via types that are specified with "D" in the technology table will be evaluated.

7. SMOOTH STAIRSTEPS: If you select this function, stairsteps generated by the router (above all during optimizing) are changed to 45-degree-vectors.

After all data is entered with <CR> the slanting is done for all selected connections.

CREATE RESIST PLAN

This function only serves for the data generation for the **via** representation in the solder resist plan; the respective graphic is represented on the assigned RESIST docu layer.

The solder resist plan representation of the **pads** has to exist as a terminal graphic element on the RESIST layer which is assigned to the appropriate outer layer
This means:

a. for newly generated components the user has to generate the pad representation for the solder resist plan on the respective RESIST layer when defining the terminals (see function "TERMINAL/create" in the PCB modification part or function "reference component" in the PCB technology part)
b. If a component has been imported from the library, and if its terminals do not contain the solder resist graphic, solder resist graphics have to be assigned to the terminal graphics; this is done within the function "reference component" of the PCB technology with the help of the function "mod. graphic" (see point 6: "Subsequent generation of a solder resist graphic")

1. Preconditions:

In order to execute the CAM function "Create Resist Plan", the following preconditions have to be fulfilled:

a. In the docu part of the layer plan a respective RESIST layer with type mark **RES** has to exist and has to be assigned to the appropriate outer layer.

b. A layer technology has to be assigned to this RESIST layer in the layer plan, in the via technology of this layer technology the widths (plus radius, if applicable) of the enlarged vias have to be defined in the via types (e.g. SR1 in the startup technology).

   Note:

   In the standard technologies the via sizes of the via technology belonging to the SR1/SR2 layer technology represent the default sizes for the via representation in the solder resist plan (SR stands for "Solder Resist").
   The default sizes (in "width" or in "radius") may be changed according to the user's requirements by editing this via technology; it is also possible to insert additional definitions. Of course, you may as well generate new RESIST layer technologies with own via definitions for the representation of the vias in the solder resist plan and assign these to the appropriate RESIST layer.

2. Menu Description

After activating the function "create resist plan", a menu appears in the status window, where the following specifications can be made:

LAYER FOR RESIST SCREEN: specify - according to the layer plan definition - the number or the name of the RESIST layer (=target layer) where the solder resist plan is to be generated for the outer layer (=source layer) to which this RESIST layer has been assigned in the layer plan.

A subsequent <CR> generates the via solder resist plan for the specified source layer on the specified target layer.

3. Output of solder resist plan on the graphic screen

In order to display the solder resist plan on the screen, the respective layer has to be activated for the graphic output. For this purpose, call the "graphic" function in the CAM or PCB menu, enter the RESIST layer in "layer", mark this layer with a cross, enter <CR> and activate the function "refresh".

Note:
Since airlines are independent of any layers, they are always output with a graphic output, if "airlines" has been marked with a cross in the graphic menu; when generating the layer plots in the postprocessing, the airlines are not transferred.

If you would like to change the output colour (default = colour no 80 = white), activate the function "colours" and specify the appropriate colour number for the DRILL layer in "colours".

With the appropriate entries in "top layers:" and with the 'toggle' function on the F5-key you may move the DRILL layer into the foreground (see function "colours" in chapter "graphic setup" of the PCB part).

4. Supplementary Digitizing

If you activate the function "LAYERS" in the PCB menu "Modification", you may digitize additional graphic on the appropriate RESIST layer.

This is done with the following steps:

1. enter
   - the number of the RESIST layer in "layer:" (according to the layer plan
   - "F" in "lay.type" ("F" stands for "free digitizing")

   in the status window

2. then activate the function "digitize";
   You may now digitize the desired graphic with the functions "OFFSET", "LINE", "CIRCLE" or "TEXT".

Note:

With every new generation of a solder resist plan on the same RESIST layer the "old" solder resist plan is replaced by the new via solder resist plan; an existing layer-specific graphic (="lay.type:F") remains unchanged when newly generating the solder resist plan on the same RESIST layer.

5. Plotting the Solder Resist Plan

After generating the solder resist plan (and digitizing graphic supplements) the RESIST layer may be processed with an existing or a new "ordinary" layer plot procedure and may then be plotted.

6. Subsequent generation of the terminal graphic for the solder resist plan

1. call the function "TERMINAL/get" in the PCB modification part;

2. get a terminal of the component concerned; remember or better note down the name appearing under "graph.:";

3. activate the functions "main menu", "technology", then "reference component";

4. call the function "TERMINAL/create", enter the graphic name (see pt. 2.) in the status window in "graph.:" in the first input field;

5. move the crosshair to a free position and place the new reference terminal with <LMB>;

6. activate the terminal just generated with the function "terminal/get" and call the function "mod. graphic";

7. Digitizing the pad graphic for the solder resist plan

a. for SMD pads (for a placement on the component side)
   * specify in the status window
     - in "layer" the number of the RESIST layer belonging to the component side (as a rule 20), where the track segment is to be digitized as a terminal graphic element
     - in "trk.-type": the number of the track type (from the track definiton of the appropriate SOLDER-RESIST layer technology) with which the segment is to be digitized;
   * use the function "offset" and the coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point of the track segment onto the one for the track segment of the signal layer pad graphic;
   * activate the function "LINE"
   * digitize the segment for the solder resist plan representation of the SMD pad

   Note:
   the solder resist graphic for the placement on the solder side may be generated in the PCB modification part (with the system-internal generation of the "second" graphics for the component and its terminals) when moving or copying the component onto the solder side.
   How to procede:
   During the first attempt to place the component onto the solder side, the system outputs the message: "second component/pin graphic does not exist".
   By pressing <CR> or <LMB> a second time you can make the system generate the second graphics (for frame and textfield) by mirroring the first graphics onto the opposite layer(s); thus the second solder resist graphic is also generated (by mirroring the first solder resist graphic from RESIST-1-layer onto RESIST-0-layer).

b. for drilled-through pads
   * use the "offset" function and the coordinate specification (via the keyboard in the status window or via the mouse) to set the placement coordinate onto the one of the signal layer pad graphic;
   * activate the function "via"
   * specify in the status window
     - in "via type": the number of the via type (from the via definition of the SOLDER-RESIST layer technology) with which the via is to be digitized;
     - in "via lay." the number(s) of the RESIST layer(s) (as a rule 19 and/or 20, e.g. "19,20") on which the via is to be stored as a terminal graphic element;
   * place the via

9. Store the data with <ESC><ESC> after finishing the solder resist graphic; since the modification is of a "global" nature, it is transferred to all terminals with this graphic.

DRILL PLAN

With this function the data for drill plans are generated.

This function serves for transferring the via data relevant for the drill plan (via type, placement coordinate, drill type and assigned macro) from the source layer onto the DRILL layer (= the docu layer defined in the layer plan which has to bear the key **DRL**).
Based on these data the system then generates the graphic representation of the drills with the help of the macros assigned in the drill technology.

1. Preconditions

In order to execute the function "Drill Plan", the following preconditions have to be fulfilled:

a. In the documentation part of the layer plan an appropriate DRILL-layer with type mark **DRL** has to exist and has to be assigned to the respective outer layer.

b. The drills for the vias may only be represented in the drill plan, if existing drill types have been assigned to the via types in the via technology and if existing macros have been assigned to these drill types (see point 6).

Note:
Every drill type defined in the drill technology has to have an assigned macro - independent of whether or not it is used.

c. If the drill plan is to be generated for an 'old' .adb-file which has been converted into the new layer technology format with the program **ioconv**, you first have to generate the necessary drill macros and assign these to the drill types (see point 6).

2. Menu Specifications

After activating the function "drill plan", a menu appears in the status window where the following specifications can be made:

1.	LAYER FOR DRILL PLAN	Specify the number or the name of the drill layer where the drill plan is to be generated (=target layer) - according to the layer plan definition.
2.	VIAS FROM:	Specify - according to the layer plan definition - the number or the name of the starting layer;

"starting layer" means: the first layer of the layer pair or the layer pair package or the board (depending on what the drill plan refers to).

3. TO — Specify - according to the layer plan definition - the number or the name of the final layer; "final layer" means: the last layer of the layer pair, the layer pair package or the board (depending on what the drill plan refers to).

4. CREATE LEGEND? — Specify whether the drill legend is to be generated together with the drill plan on the DRILL layer; the legend contains:
   a. the macro graphics used for the representation of the drill types
   b. the drill types
   c. the drill type diameters
   d. the number of drills for each type

Note:

- the direction for the "from" "to" command is of no importance; you may begin with the component side towards the solder side or vice versa.
- In the drill plan only those vias are considered which go from the first to the last specified layer.

Example:

a) from 2 to 3 (or vice-versa)	b) from 4 to 3 or (vice versa)

In case (a) only V1-vias are considered.
In case (b) only V3-vias are considered.

After the entries and <CR> the drill plan - and the drill legend, if specified - is generated on the specified target layer.

3. Output of the Drill Plan on the Graphic Screen

In order to display the drill plan after its generation on the graphic screen, the respective layer has to be activated for the graphic output. For this purpose, call the function "graphic" from the CAM or PCB menu, enter the respective DRILL layer in "layer:" of the status window, mark this layer with a cross, enter <CR> and activate the graphic function "refresh".

Note:
Since airlines are independent of layers, they are always output, if "airlines" has been marked with a cross in the graphic menu; when generating the layer plots in the postprocessing, the airlines are not transferred.

If you would like to change the colour (default = colour 80 = white), activate the function "colours" and enter the appropriate colour number for the DRILL layer in the status window under "colour".

You may switch the DRILL layer into the foreground with the appropriate specifications under "top layers:" and the 'toggle' function on the F5 function key (see function "colours" in chapter "graphic setup" of the PCB part).

4. Supplementary Digitizing

If you activate the function "LAYER" in the menu "Modification" of the PCB part, you may digitize additional layer-specific or only legend-specific graphic on the DRILL layer.

How to procede:

1. Specify in the status window:
   - the DRILL layer number in "layer:" (according to the layer plan)
   - in "lay.type"
     "F" if the graphic is to belong to the layer
     or
     "L" if the graphic is only to belong to the drill legend
2. Activate the function "digitize";
   Now you may digitize the desired graphic with the functions "OFFSET", "LINE", "CIRCLE" or "TEXT".

Note:
Layer specific graphic remains unchanged after a new generation of the drill plan on the same DRILL layer, whereas the legend-specific graphic disappears when newly generating the drill plan, since the legend is newly generated together with the 'new' drill plan.

5. Plotting the Drill Plan

After generating the drill plan (and digitizing additional graphic) the respective DRILL layer may be processed with an existing or a new "ordinary" layer plot procedure and may then be plotted.

6. Generation and Assignment of the Drill Macros

If macros have still to be generated and assigned, procede as described in the chapter "drills" of the PCB technology description.

A. Generation of the macros for the representation of the drills

How to procede for generating a macro:

1. call the function "component/create" in the PCB work menu;
2. Enter the macro name in "graphic" and "Y" in "create new graphic:";
3. open a component frame with the mouse at a free position in the graphic window and place it (is done like creating a component);
4. Get the component, activate the function "mod. graphic"; enter "Y" under "glob." in the status window!
5. Modify the component graphic so that it gets the form desired for the macro (digitize the macro and delete the original component frame);
6. Get and delete the component text field;
7. leave the function "mod. graphic" with <ESC><ESC>;
8. get and delete the component;

Now the graphic exists in the "graphic pool" under the name specified during the generation, and with this name it may be specified in the drill technology as a macro.

Note:
- Before leaving the function "mod. graphic" you should make sure that the macro exactly corresponds to your requirements!
- Do not forget to enter "Y" in "glob." in the status window, otherwise the graphic automatically gets a new name by the system ("STDN..")

- For digitizing the graphic, the coordinate origin is the lower left corner (=rotation point) of the original component.

- You have to remember or better to note down the names of the generated macro graphics, since at present there is no list function for the graphic names of the 'pool'.

B. How to enter macro names into the drill technology

1. call: main menu/technology/drills;

2. enter the graphic names for the representation of the single drill types in "macro";

3. activate "Save File" in the pop-up menu;

4. leave the menu with <ESC><ESC>

After this preparatory work call the CAM-function "drill plan", make the appropriate specifications in the status window and then execute the drill plan generation with <CR>.

SILK SCREEN

This function serves for generating the data for silk screens.

In contrast to the representation on the SHAPE layers, the component outlines are left free where they collide with vias.

After invoking this function a new menu appears in the status window, where the following parameters can be set:

LAYER FOR SILK SCREEN: Specify - according to the layer plan definition - the number or the name of the silk screen layer (=target layer) where the silk screen is to be generated for the outer layer (=source layer) to which this silk screen layer has been assigned in the layer plan.

After <CR> the silk screen is generated on the target layer. The component outlines and texts are transferred from the SHAPE layer (a docu layer assigned to solder or component side, which contains the graphic information of the silk screen plan) to the target layer (which has to bear the key **SSC**). Based on placement and size of the vias, the components (frame and text) colliding with vias are highlighted; the colliding outline segments are automatically removed.

With colliding texts the user has to go into the PCB part, interactively modify or delete and newly digitize the texts (see below point "supplementary digitizing").

1. Silk Screen output on the graphic screen

In order to display the silk screen on the graphic after its generation, the respective layer has to be activated for the graphic output. For this purpose, call the function "graphic" in the CAM or PCB part, enter the SILK layer in "layer:" of the graphic window, mark this layer with a cross, enter <CR> and activate the graphic function "refresh".

Note:

Since airlines are independent of layers, they are always output, if "airlines" has been marked with a cross in the graphic menu; when generating the layer plots in the post-processing the airlines are not transferred.

If you would like to change the colour (default = colour 80 = white), activate the function "colours" and enter the appropriate colour number for the SILK layer in the status window under "colour".

You may switch the SILK layer into the foreground with the appropriate specifications under "top layers:" and the 'toggle' function on the F5 function key (see function "colours" in chapter "graphic setup" of the PCB part).

2. Supplementary Digitizing

If you activate the function "LAYER" in the menu "Modification" of the PCB part, you may digitize additional layer-specific graphic on the SILK layer as well as move component texts or delete and newly digitize them.

If in the silk screen texts collide with component outlines, the components are highlighted. If the function "layer" is active, you may get the texts and delete them.
For this purpose, enter "S" in the status window (for "silk screen") in "lay.type", activate the function "text/get", get the text with the mouse and place it at a convenient position; if moving is not possible, you may delete the text by activating the function "delete".
Then activate the function "digitize/text" and digitize the text at the required position; when generating the text with the mouse, open a frame that is smaller than the original text frame.

How to procede for the digitizing:

1. Enter in the status window:

   - the SILK layer number in "layer:" (according to the layer plan)

   - either
     "F" in "lay.type" ("F" stands for "free digitizing")
     or
     "S" (stands for "Silk screen-specific digitizing")

2. Activate the function "digitize";
   Now you may digitize the desired graphic with the functions "OFFSET", "LINE", "CIRCLE" or "TEXT".

Note:

- The layer-specific graphic (="lay.type:F") remains unchanged after a new silk screen generation on the same SILK layer, whereas the silk screen-specific graphic (="lay.type:S") disappears.
- With a new silk screen generation on the same SILK layer the 'old' silk screen is replaced by the new one.

3. Plotting the Silk Screen

After generating the silk screen (and changing text and/or digitizing additional graphic) the respective silk screen may be processed with an existing or a new "ordinary" layer plot procedure and may then be plotted.

PLANES

1. General

This function serves for

- dividing the plane into separated copper areas (so-called **split plane**) by digitizing the dividing lines (for storing several power supply signals on one plane layer)
- generating the **positive** representation of the planes or of the part planes as real copper area(s) on the plane layer
- generating the **negative** representation of the planes or split planes on the docu layer

Note:

- After executing the "plane" function you get the negative plane layout by plotting the respective docu layer in the post-processing.
- The positive plane layout is generated - after executing the function "planes" - by plotting the plane layer in the postprocessing.
  Note:
  In order to plot the copper of the plane, the copper area (or for split planes the single copper areas) have to be represented **cross-hatched** before storing the file. (Copper areas which are not cross-hatched cannot be handled by OUTPUT).

2. Preconditions

In order to execute the function "Planes" the following precon-ditions have to be fulfilled:

a. The plane layer as well as the docu layer for the negative representation have to be defined in the layer plan.
   The plane layer should be marked with "PLN" and should be given a convenient name (e.g. "GND", "VCC" or "GND+VCC").
   The docu layer has to be marked with "DOC" and should also be given a convenient name (e.g. "NEG").
   An appropriate layer technology has to be assigned to the layers for the layouts.

b. The terminals belonging to plane signals have to have an appropriate graphic for the representation of the heat traps or that layer where the plane is entered in the layer plan. If this is not the case, you have to generate the heat trap graphic before executing the "planes" function (see point 6).

3. How to procede

1. If all preconditions are fulfilled, call the function "planes" in the CAM work menu.

2. Make the necessary specifications in the status window:

SIGNALS: If split plane dividing lines are to be digitized, enter the names of those plane signals, the connections of which you would like to highlight through airlines for a better identification (you may use wildcards);

EXCEPT: If wildcards have been used in the field "SIGNALS", enter those signals which are not to be considered for the function; you may also use wildcards

LAYERS: enter the name or number of the plane layer specified in the layer plan

D-LAYERS: enter the number of the docu layer where the negative representation of the plane is to be generated and where the dividing lines for the split planes are to be digitized;
After this procedure, this layer contains
- the dividing lines for the split plane
- the heat trap graphic for the vias/pads of the plane connection
- the representation of the vias/pads enlarged by the clearance for vias/pads which do not belong to the plane signals);

3. After a <CR> the digitize mode is activated; the menu "digitize graphic" appears in the status window.

   a. If nothing is to be digitized, enter <ESC><ESC>

   b. If the split plane dividing lines are to be digitized, procede as follows:

> specify the names of the signals to be highlighted in the field "signals";
these two features (airlines and highlighting) provide for the possibility to differentiate between the single plane signals - this makes digitizing the dividing line considerably easier.

> activate the function "digitize/LINE" and generate the necessary dividing lines;
Note:
The dividing line and the board outline must at least have one point in common, i.e., the area limited by the dividing line may not exist as an "island" within the board area; otherwise the system cannot differentiate between the two areas!

> after digitizing the dividing lines activate the STOP function and finish the procedure with <ESC><ESC>.
Note:
If you had used the window function for digitizing, it is <u>absolutely</u> necessary to display the <u>overview</u> with the "origin" function before entering <ESC><ESC>.

4. The plane - or, for split planes, the first part - is now highlighted and you have to enter in the status window the name of the signal to which the part is to be assigned;
for split planes the next part is highlighted with a <CR> and you may enter the respective signal name in the status window.

5. finish the procedure with <CR>;
then the plane/part planes is/are transferred onto the plane layer in a positive representation; now the plane layer corresponds to an "ordinary" signal layer; the plane/part planes is/are output as copper areas; the complete negative representation of the split plane layer is now automatically generated on the docu layer used for digitizing the part plane contours.
(this is somewhat time-consuming)

6. If you would like to plot the positive representation of the plane layer (= the copper), it is necessary to represent all copper areas cross-hatched.

Notes:

- If more than 510 segments are used for the generation of the outlines and dividing lines, the part planes cannot be output as cross-hatched or filled areas, but only with their contours. This may occur above all when slanted segments are used; non-orthogonal segments are internally represented as a sequence of grid-sized orthogonal steps -> this increases the number of segments considerably.
- When digitizing the split plane contours, please see to it that they are perfectly closed.
- In order to make a plot of the heat traps in the negative representation of the plane layer possible (for the above example, signal layer 4), the heat traps have to be stored on the respective plane layer in CIS. That means the respective terminals have to have a heat trap graphic on the respective plane layer (see chapter "reference component" of the PCB technology description).
  If a heat trap graphic does not exist, you have to generate it (see point 4), before executing the "plane" function.

4. Subsequent Generation of the Heat Trap Graphic for Plane Terminals

If the heat trap graphic is still to be generated, this has to be done prior to executing the CAM function "Planes".
A precondition is that all the planes have the same pool graphic; if this is not the case, the following has to be done for each graphic separately.

1. Activate the function "TERMINAL/get" in the PCB modification menu

2. get a plane terminal; remember or better note down the name appearing under "graph.";

3. Activate the functions "main menu", "technology" and "reference component"

4. call the function "TERMINAL/create", enter the graphic name noted down in the first input field under "graph." in the status window;

5. move the crosshair to a free position and place the new reference terminal with <LMB>;

6. activate the terminal just created with the function "TERMINAL/get" and activate the function "mod. graphic"

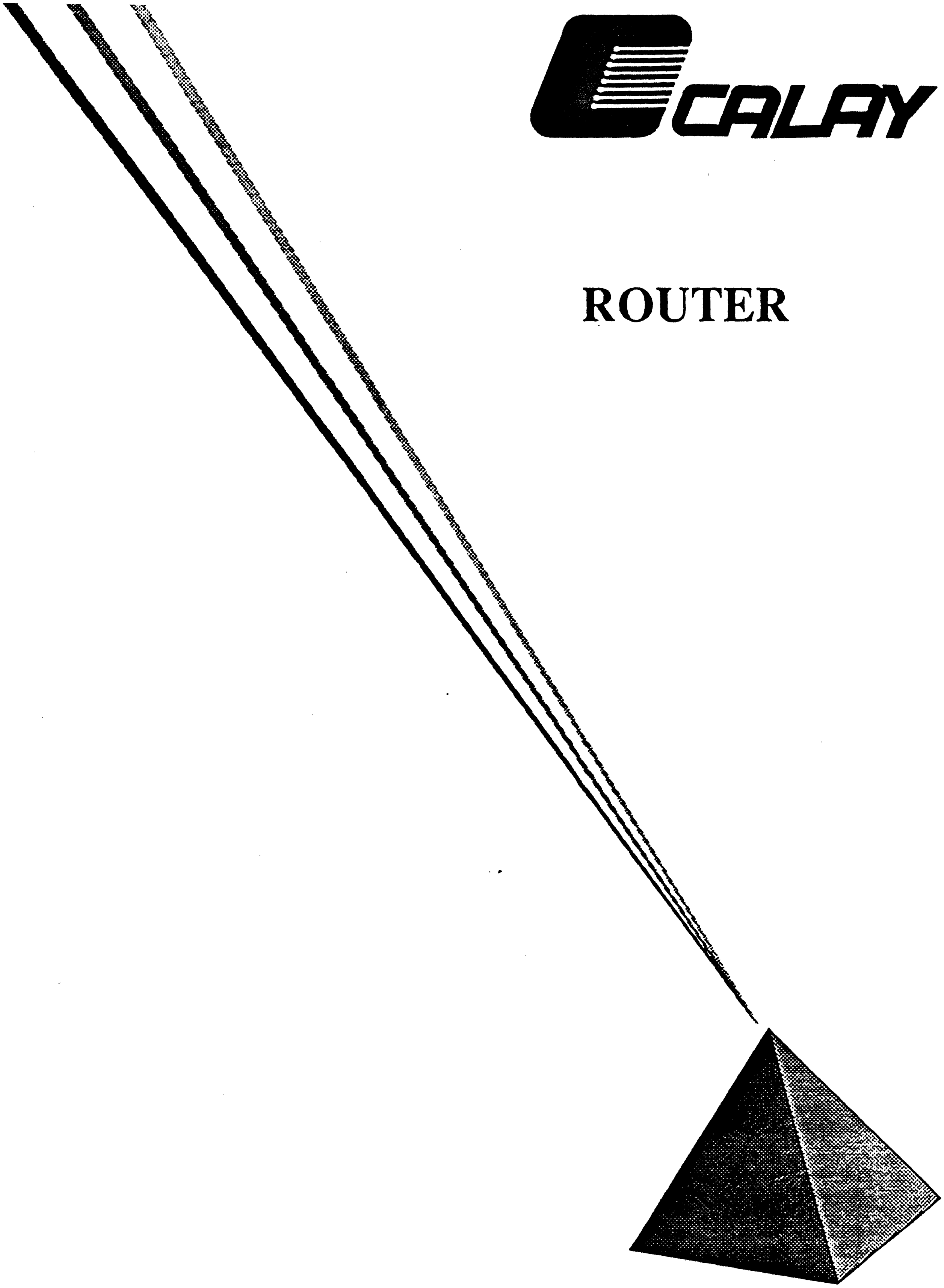
7. Digitizing the Heat Trap Graphic

   * specify in the status window
     in "layer" the number of the plane layer defined in the layer plan where the track segments for the heat traps are to be digitized as terminal graphic elements;
     (the layer has then also to be specified in the CAM menu "planes" in "layers", if you want to generate the postprocessing data for this plane)
     - in "trk.-type" the number of the track type with which the segment is to be digitized (from the track definition of the layer technology which is assigned to the respective docu layer);
     - "Y" in "heat traps"
     - "Y" in "show heat traps"
   * use the "offset" function and the respective coordinate specification (via the keyboard in the status window or via the mouse) to set the starting point for the first track segment
   * use the function "LINE"
   * digitize the first track segment for the heat trap representation
   * use the "offset" function to set the second starting point for the heat trap representation, digitize the track, etc.

8. After finishing the heat trap graphic store the data with <ESC><ESC>; since the modification is of a "global" nature, it is transferred to all terminals with that graphic.

Note:
the heat trap graphic is only output for the terminals connected to the plane (the system automatically "recognizes" and considers this);
if terminals not connected to any planes have the same graphic, the heat trap graphic is not output for these terminals.

ROUTER



L D R O U T

T A B L E O F C O N T E N T S

ROUTER LDROUT

I. INTRODUCTION

After the .adb-file has been created in CIS, it is transferred to the router for an automatic routing of the connections.
The **input data** relevant to the router is contained in the PCB technology as well as in the signal and terminal definition of the **.adb-file.**

The PCB technology of the .adb-file contains the following information for the router:

- routing grid (to which LDROUT is bound for routing)
- routing area (as a limitation for routing)
- board outline (as an additional limitation for routing)
- assignment of the signal layers to production layer pairs
- assignment of signal layers to routing layer pairs
- layer technologies (clearance, track widths and assigned via types, via width)

The signal definitions of the .adb-file contain the netlist together with the routing relevant signal parameters (e.g. connection layer, track type to be used, etc.).

The terminal definitions contain the information on position and extension of the terminals, on their connection widths, on their assignments to subsignals, etc.

The router processes the routing layer pairs from the component towards the solder side. As a rule, the tracks routed are stored in the .adb-file after each layer pair.

The **output** of the router is the .adb-file extended by the tracks routed and a routing report file.
The report file contains the protocol output during the routing process on the screen, as well as additional information (see chapter ".ldr-file"). It is stored onto the adb-directory and has the name of the .adb-file with the extension **.ldr.**

Prior to the actual routing process the input data is sorted for the router (sorting of the nets, sorting of the terminals within the nets) and pre-information is output (routing grid, number of missing connections and of the layers, etc.).
Then the actual routing process is performed.

After all connections have been found, glossing iterations (cost optimization of segments and minimization of the vias) are automatically performed.

An additional optimization may either be invoked directly after the routing or as a separate process. This is to fully automatically optimize the routed tracks by moving or changing segments or parts of segments and vias. The optimization is oriented at cost, production, electrical and aesthetical criteria.

II. HANDLING

A. STARTING THE PROGRAM

Under the ROUTER icon, all programs can be found that are necessary for:

- the software routing
- the hardware routing
- displaying the routing reports (*.ldr)
- calling the help function

At first the following menu appears for the function selection:

```
*********************************************************************
   PRISMA    ROUTER    SELECTION                v .....
*********************************************************************
                                 ***   default selection: >> 5 <<   ***

1.  Run Softwarerouter (ldrout)
2.  Run Hardwarerouter (ld400)
3.  Run 'gph > adb' conversion
4.  View 'list files' (router and conversion)

5.  Help information

E.  Exit
---------------------------------------------------------------------
your choice ?
```

Handling instructions:

- Every input has to be confirmed with <CR>.

- After entering the selection number and pressing <CR>, a respective submenu for a further function selection or a menu for the file specification appears.

- Selection menus contain a default selection in the upper right corner as a rule; the default selection may be activated by pressing <CR>.

- The selection menus can only be left by entering "0" (you get back to the superior menu) or "E" (Exit).

- <CTRL C><CR> is to be used for interrupting a process or for leaving a file inquiry menu (<u>not</u> for leaving selection menus).

- Hardware routing is, of course, only possible with the RPR-400.

After calling the software or hardware router, you are asked for the specification of the name of the file to be routed (without extension!).

```
enter the filename [without ext.] :
```

After the specification of the file name with <CR>, a selection menu appears for the specification of options.
After specifying the desired option(s) and <CR> the routing data is extracted from the .adb-file.
Then the routing process is automatically started.

B. EXIT and RESTART

Exit:

When routing without graphic output (i.e. without option "-f"), the routing may be aborted with <CTRL C>. The routing result up to the last finished layer pair iteration is automatically stored.
Note:
When using the option **-c:ld1**, the routing should not be interrupted before the 14th iteration, as in this special case the routing result is stored automatically only after 14 iterations (see page 8).

When routing with graphic output, the routing can only be aborted by placing the mouse cursor on the edge of the LDROUT graphic window and pressing the L10 key.

Restart:

"Restart" means the repeated routing of an already routed file **without** the option **"-n"**.
The router starts with the result of the last finished layer pair iteration.

If a file was routed (partly), then changed interactively in CIS, the router considers the changes made in CIS when restarting the routing process.

C. OPTIONS

You may enter several options (switches) before starting the routing process which directly influence the working mode of the router.

The specification may contain several options in a row (without blanks; the "-" character belongs to the option!). You have to take care that the options do not exclude one another.

```
**********************************************************************
* PRISMA   R O U T I N G S W I T C H   S E L E C T I O N   v...     *
**********************************************************************

Router Options :

-f       - graphic output (slows down routing speed)
-g       - perform only 4 glossing iterations per layerpair
-i:n     - perform a maximum of n iterations
-k       - routing without restrictions (use with '-n'only)
-l:n     - route only on layerpair n
-n       - new routing

---------- usable control files: ----------------------------------

-c:ld1   - busrouting - save after 14 cycles through pcb
-c:ld2   - no busrouting
-c:ld3   - perform only 2 final glossing iter. per layerpair
-c:ld4   - perform only optimizing router
-c:ld5   - standard control file + optimizing router
-c:ld6   - perform only optimizing router without 'L-opt.'
-c:ld7   - like 'ld5'  but optimizing router without 'L-opt.'
         (without -c option a standard file will be processed)
--------------------------------------------------------------------

enter [option][option]

option  :
```

1. Router options

-f (graphic output)

This option effects the graphic output of the routing process on the screen. As the graphic output appears in the foreground, the report output is not visible.

Notes:

- The effect of the graphic output is a slow down in the routing speed.
- In the graphic output mode, the routing can only be aborted by placing the mouse cursor on the edge of the LDROUT graphic window and pressing the L10 key!

- White terminals cannot be connected by the router (-> check whether they lie within a restricted area or if false connection restrictions have been defined).

- Blue areas indicate clearance violations.

-g (perform only 4 glossing iterations per layerpair)

When specifying switch "-g", only 4 **glossing iterations** will be executed per layerpair, regardless of whether the board has already been routed to 100%.

Note:
As a reduction of the segments and vias is effected during the glossing, and further routing attempts are executed if the board is not completed, the routing result can be improved by interrupting the router and by optimizing the current interim result with pure glossing routing (restart with option "-g").
But the glossing may also result in channel blockings since the via reduction is more important for the router than the 'wrong-way'-routing.

-i:n (perform a maximum of n iterations)

Independent of the number of missing connections, the routing will be interrupted after n iterations (possible glossing iterations included!);
(n = 1, 2, 3, etc.).
Without the option '-i:n', the router does not stop before routing the board 100%.

-k (routing without restrictions)

This option should only be used for boards that are easy to route (with much free space) for an acceleration of the routing. This option switches off 'space-saving' restrictions that are effective in the first five iterations during the 'ordinary' routing (limitation of the allowed number of vias per net; limitation of the track course to the shortest possible way).
The -k option should only be used for a restart (with option '-n').

-l:n (route only on layer pair n)

For multilayers, you may specify by entering a respective number for "n" that the router may only route on this layer pair.

-n (new routing)

This switch causes the router to **restart**. If the file has already been routed or if variable tracks have been digitized interactively in CIS after the routing, all variable tracks and vias will be deleted and newly routed when starting the router with this function.

Note:
As a rule, the first iteration of a restart is the so-called **busrouting** (ref. point 'D').

2. Selection of the control file

The basic way of proceding for the router is determined by the .CTL-control file. The .CTL-files are stored on the directory **lib/sys.**
A **.CTL**-control file contains:

- parameter sets that control the routing with regard to low production costs ('cost parameter sets')

- control commands that activate or suppress special functions during the routing (e.g. execution of the busrouting, setting of 'stubs' for SMD routing, activation of the suitable parameter sets, calling the optimizing router, etc.).

The selection of the .CTL-file is done with the option '-c:'. If no specification is made, the standard control file ldrout.CTL will be used automatically.

-c:ld1 (save after 14 cycles through pcb)

When specifying this option, the router uses the control file ld1.CTL for the current routing. The only difference between this control file and the standard control file ldrout.CTL is that

--> the routing result is not saved after every board iteration (iterations necessary to process all layer pairs once), but only after every 14th iteration.

This effects a considerable acceleration of the routing. But in this case, the routing can only be interrupted after every 14th iteration, otherwise iteration results will be lost.

Note:
You should use this option only for two layer boards.

-c:ld2 (no busrouting)

When specifying this option (= using the control file ld2.CTL), the busrouting is only executed on the inner layers.

--> no busrouting is effected on the outer layers, i.e. the router may already place through-connections in the first iteration on the outer layers;

--> in the first and second run, only one iteration per layer pair is executed;

--> from the third run onwards, 2 iterations per layer pair are executed.

-c:ld3 (perform only 2 final glossing iter. per layer pair)

With this option (= using the control file ld3.CTL), only two final glossing iterations per layer pair are executed.
In the final glossing iterations, the router does not lay much stress upon the 'wrong-way'-routing. Therefore, this option should only be used for an additional glossing of 100% solved boards.

-c:ld4 (perform only optimizing router)

When using this option, only optimizing routing is executed (see point "E"); before the optimizing iterations, the router executes a glossing iteration per layer pair.

-c:ld5 (standard control file + optimizing router)

This file controls the execution of standard routing with subsequent optimizing routing.

-c:ld6 (perform only optimizing router without 'L-opt.')

Corresponds to the option '-c:ld4', but the L-optimization is not carried out (see point 'E'). This option is sensible if the CAM function "bend slant" is to be used instead of the router-L-optimization after the routing process.

-c:ld7 (like 'ld5' but optimizing router without 'L-opt.')

Corresponds to option 'c:ld5', but the L-optimization is not carried out. This option is sensible if the CAM function "bend slant" is to be used after the routing process.

D. STANDARD CONTROL FILE ldrout.CTL

If the user does not specify a special .CTL-file with the option "-c:", the standard control file **ldrout.CTL** is always automatically used. This file controls the routing as regards cost parameter sets and effects the execution of the following functions:

1. Busrouting

For the first routing or for a restart applies that only direct connections without vias (bus connections) will be routed in the first layer pair iteration on every layer pair.
The routing of the tracks is done with strict adherence to the priority directions (if the assignment of the routing layer pairs corresponds to the assignment of the production layer pairs):

- on the solder side and the respective inner layers in the X direction
- on the component side and the respective inner layers layers in the Y direction.

2. SMD connections

If a board contains SMD components with pins to be connected, the router creates connections to the other layers (called **'stubs'**) in the first iteration on the outer layer pair by routing short track segments to the respective SMD pins and by placing through-connected vias there (ref. also to chapter "Routing with SMD technology"). Subsequently, the bus routing is performed in a second iteration on the outer layers.
If the board is a multilayer, the busrouting is executed on the inner layers afterwards. The 'stubs' may be used as bus connections.

In the output of the connections not found, the stub connections are marked with **".{ }"** [e.g. "SMD1.{3}"].
The unused stubs are deleted during the glossing.

3. Glossing

After the iteration in which the PC board is solved (i.e. all connections are found), 2 x 2 glossing iterations are performed per layer pair.

During the glossing iterations, special corrections are executed: the number of vias is minimized and the course of the tracks is optimized (see option "-g").

E. OPTIMIZING ROUTER

The optimizing router LDOPT that is called with the option '-c:ld4' or '-c:ld5' performs the following optimizations in the order described below:

VIA: Vias are shifted to less dense areas. Vias are shifted on-grid in the positive and negative X or Y direction to get more space for the segments.

AVERAGE: Segments are averaged, i.e. they are shifted to the vertical direction as far as possible (x segments into the negative and positive Y direction, Y segments into the negative and positive X direction).
As a result, distances may become too large and U-shaped segment topologies may be generated. In this case, averaging is only reached with a subsequent U-optimization.

U: U-shaped segment topologies are minimized or, if possible, deleted.

VIA CORNER: Vias belonging to L-shaped segment constellations and lying within a segment are shifted to the corner where the segments meet.

T: T-shaped segment constellations are treated like double L-structures and become almost V-shaped by slanting them.

L: Slanting of corners. The L-structure can be changed into several slants that are separated by steps (= staggered).

L-Move: Slants are shifted until all steps are deleted.

Notes:

- AVERAGE and the subsequent U-optimization will first be performed for X segments and then for Y segments.
- The 45° slants created during the T and L optimization will not be generated as vectors but as steps (like on the graphic screen). The number of segments is increased respectively after this optimization.
  The conversion of these 45° slants into real 45° vectors can be performed in CIS with the CAM function "45° connections"; after invoking this function, "smooth stairsteps" has to be marked with a cross in the appropriate menu.
- Vias will only be shifted and segments will only be changed if free space exists, i.e. obstacles cannot be leaped.
- The optimizing router is bound to the grid. Parameters are for example the clearance to be adhered to (1 grid unit for L and T optimization, 2 grid units for all other optimization methods) or the length of the slants (e.g. at most 15 grid units).

III. OUTPUT FORMAT

A. BASIC INFORMATION

After the read in of the file, the router first determines the router technology and checks the technology definition for clearance violations or other technological errors (if there are any, the router displays warnings or error messages on the terminal).

After the check, the following information is output:

- the version number of the router,
- the number of layers (".. layers"),
- the routing grid ("grid = 1/..."),
- the number of missing connections ("... missing connections"),
- the number of missing connections that belong to 'fixed nets' (i.e. to signals that were marked as non routable in CIS) ("... of them in fixed nets")

  Attention: Although the router outputs these connections as missing connections, the routing is finished when all other connections have been found; a subsequent technology check in CIS shows the respective number of interruptions!

- the number of offgrid terminals ("... terminals offgrid")

Then, the main router starts the first iteration. During each iteration, a line is output that contains the following basic information:

- iteration specification: b = bus or stub iteration
  i = standard iteration
  g = glossing iteration
- current number of the iteration,
- number of calculated connections,
- number of missing connections,
- the connection points that were 'touched' by the router but could not be connected.

```
i  1( 1p  1s  1)   321 / f   152   nnnn   nnnn   nnnn
```

non-connected point
number of missing connections
number of calculated connections
current iteration number

The number of calculated connections is newly displayed after every tenth connection 'touched' by the router. This automatic feature is interrupted if a connection is found or a missing connection is written into the output line.

During the first 9 iterations on a layer pair, not all of the connections already found are calculated anew, as a rule. This results in a considerable saving of time.

If the specification of a connection not found contains ".( )" between the component name and the terminal number, then it is an **SMD**-pin to which a via for changing layers (= '**stub**') has been connected by the router in the first iteration.

B. ADDITIONAL INFORMATION

Behind the current iteration number, the following **additional information** is output in parentheses:

- total number of bus/stub, standard or glossing iterations performed on this layer pair (the three iteration types are counted separately),
- the cost parameters from the .CTL-file and
- the track type routed in this iteration.

```
i  135 ( 52p  16s  3)
```

track type
cost parameters
no. of standard iterations executed on this layer pair

After every completed iteration, the total time consumed by this iteration is displayed in a separate line, as well as the time consumed for the system internal preparation of the iteration:

```
CPU      0:01:54.04      (37% system)
```

preparation part
1/100 seconds
seconds
minutes
hours

IV. ROUTER INTERNAL REPRESENTATION OF THE CIS/PCB VALUES

A. ROUTER SYMBOLS

For the internal representation of the layout elements defined in the CIS/PCB technology (pins, tracks, vias), the router uses 16 symbols the diameters of which are calculated in terms of the widths of the routable track types and the routing grid.

min. symbol size = 0.000
max. symbol size = 2 x routing grid - clearance

This limitation does not restrict the usable technology sizes for tracks and vias/pins. If an element is larger than the maximum symbol size, it is composed of several symbols. For the representation of unmoved elements an uneven number of the same symbol is used (see fig. 1); for moved elements an even number of symbols is used.
Zero symbols may also be used - they do not have an extension but occupy grid cells.

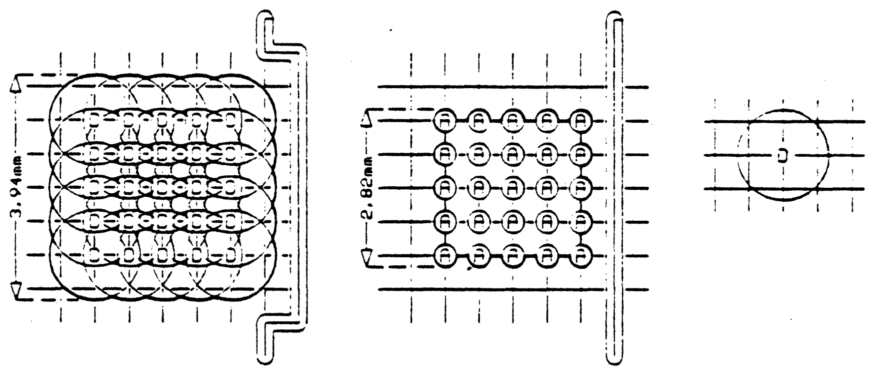


Figure 1

B. REPRESENTATION OF ROUTABLE TRACKS

As the number of router symbols is limited to 16, only 16 free routable tracks are available.

For the router a track type is "freely routable", if
- in the track technology it is specified as "A" (active) and "R" (routable).
- it has been specified for at least one terminal to be connected (i.e., in the PCB menu "terminal/mod. parameter" this track type has to be specified under "-type").

  But also the signal with which this terminal is to be connected has to be routable (i.e., in the PCB menu "signal/mod. param" "Y" has to be specified in "routable:").

To every freely routable track type a via type has to be assigned in the track technology; this via type is then used for routing the track. It has to be defined as "routable" in the via technology (attribute "R").

Freely routable tracks are always exactly represented in the routing grid. This also applies to routable tracks the width of which is larger than the maximum symbol size.

Note:

Tracks and vias shifted by 1/2 grid into negative x or y direction are always represented by an even number of grid cells. This results in a router-internal minimum limit of one grid cell for the diameters of shifted elements.

- If a track type defined in the technology as "active" and "routable" is not used in the layout, the router may change the width of this type defined in the CIS/PCB technology for its internal calculation in order to be able to exactly represent a special via type.
  In this case, the width used by the router appears in the technology chart of the routing report file, instead of the real CIS width.

C. REPRESENTATION OF THE VIAS AND NON-ROUTABLE TRACKS

Vias (pads and through-connections) as well as non-routable tracks the widths of which exceed the maximum symbol size are represented router-internally with several symbols of the same kind. For unmoved elements the router uses the symbol that - composed of an uneven number of symbols - comes next to an almost exact representation of the real CIS-size.
For moved elements the router uses that symbol which - composed of an even number of symbols - comes next to the CIS-size.

Attention:
Routable vias and non-routable tracks can be exactly represented in the routing grid if they contain the same parameters (width, radius, additional clearance) like a routable track.

The default parameter sizes entered in the CALAY standard technologies are optimized and guarantee an exact representation in the router.

If the standard parameter values in the technology charts in CI_ are changed by the user, it might occur that the widths and diameters of vias and non-routable tracks lie between the values that can be calculated by the router. In this case they are realized by the next higher value that can be represented (ref. to the chapter ".ldr-file"). The difference between the value with which the router calculates and the value specified in CIS (= inaccuracy) becomes apparent in the layout in a respective increase in the clearance.

When changing the standard values of the CIS technologies, we advise you to check the accuracy of the router-internal representation of the CIS technology sizes with the .ldr routing report file (ref. to ".ldr-file").

V. LDR-FILE

The routing report file is automatically created by the router. It is stored on the **adb**-directory and has the name of the routed file with the extension "ldr".
Beside the information that is output during the routing (iteration report, error messages, etc.), the .ldr-file contains a chart with the CIS/PCB sizes of the layer technology assigned to the outer layers and their router-internal realization.

The following example shows an .ldr-chart which results from routing a two-layer board with 1/40" technology (widths and diameters in mm):

```
       a    b     c  d e   f      g h     a    b     c   d e   f     i

track  1:  0.290     1 4=  0.290  r 1  via  1: 0.980 <   2 5=  0.980
track  2:  0.420     1 7=  0.457 *     via  2: 1.270     3 0=  1.270
track  3:  0.980 <   2 5=  0.980       via  3: 0.000     1 0=  0.000
track  4:  1.400     3 2=  1.400       via  4: 0.000     1 0=  0.000
track 17:  0.000     1 0=  0.000       via 17: 1.400     3 2=  1.400
track 18:  0.000     1 0=  0.000       via 18: 1.700     3 7=  1.727 *
track 19:  0.000     1 0=  0.000       via 19: 2.500     5 0=  2.540 *


       j  k

symbol 0: 0.000
symbol 1: 0.066
symbol 2: 0.130
symbol 3: 0.197
symbol 4: 0.290
symbol 5: 0.345
symbol 6: 0.392
symbol 7: 0.457
symbol 8: 0.522
symbol 9: 0.587
symbol a: 0.652
symbol b: 0.717
symbol c: 0.782
symbol d: 0.847
symbol e: 0.912
symbol f: 0.980
```

The columns have the following meaning:

a: consecutive track and pin type number
b: track or via diameter defined in CIS (nominal value)
c: "<" indicates the shifted types
d: number of symbols used for the display
e: mark of the used symbol
f: track or via width calculated by the router (actual value)
g: with "r" only the actual <u>routable</u> track types are marked
h: the number specifies the via type that is used as 'stub' when digitizing the track type marked with "r"
i: the widths calculated by the router which are marked with "*" differ from the nominal value
j: mark of the symbol
k: diameter of the symbol

CIS - Initial situation:

- in the PCB technology the track types 1-4 and the via types 1, 2, 17, 18, 19 are defined unequal 0 and with the attributes "A" ("active") and "R" ("routable"); track type 3 and via type 1 are defined as shifted (attribute "S");
- for the generation of routable signals only track and via type 1 have been used; only they are "routable" for the router.
- the board contains through-connected pins (realized with via type 17, 18, 19) and SMD pins (realized with track type 4).

With this chart, the defined sizes for track and via diameters may be compared to the sizes internally calculated by the router. If the difference between defined and calculated size is too large, you can check by means of the chart if it is possible to assign a more suitable size to the respective track or via type.

Notes:

- Apart from the technology chart and the information that is displayed on the screen during the routing, the .ldr-file contains some additional information (e.g. on the layer pair combination, on the layer technology used, on the use of 'buried' vias, etc.) that does not appear on the screen.

- Tracks and vias defined in the CIS/PCB technology that are not used on the board (for digitizing variable and fixed tracks or for creating SMD or through-connected pins) are not considered by the router. In the report chart they either do not appear at all or they appear with zero values.

VI. WORKING METHOD OF THE ROUTER

A. BASIC INFORMATION

Before a restart, the router sorts the nets and connections of the netlist.

Within a net, the router only connects **one** pin with one **target structure**. This target structure may be the next pin of that net or a track segment which has already been partially realized (see figure 1 and 2).

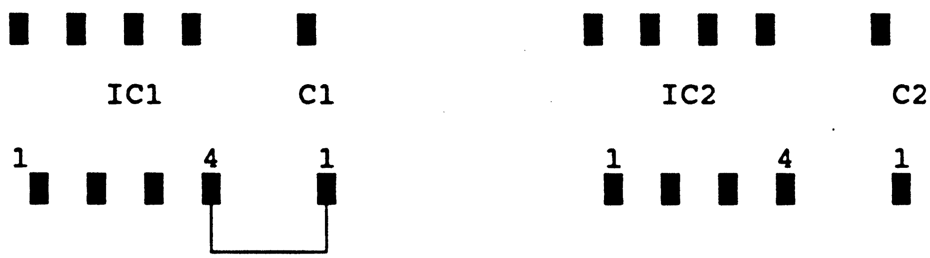


figure 1

Figure 1 shows a constellation in which IC1 pin 4 is the target structure.
The net GND has i.a. the following connection points:
/GND: IC1/4, IC2/4, C1/1, C2/1, ...;

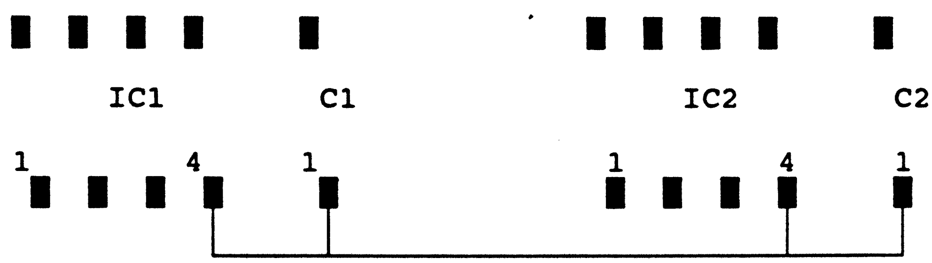


figure 2

Figure 2 shows the target structure (bold type) as a partially realized track (IC 1 pin 4 to C1 pin 1) of the same net.

The two steps illustrated in figure 1 and 2 will ordinarily be performed within one working step.

When defining fixed tracks, take care that the tracks of one net are connected to each other.

Figure 3 illustrates a wrong constellation of fixed tracks. In the net GND, IC1 pin 4 has been connected to C1 pin 1 and IC2 pin 4 to C2 pin 1; respectively by a fixed track. With such a defined track course, the router considers all pins of the net as being connected. The two digitized tracks are not connected to each other; this results in an interrupt of the net.
The track course in figure 2 is acceptable for the router. The non-connected pins of IC2 and C2 will be connected to the existing track without any problems.

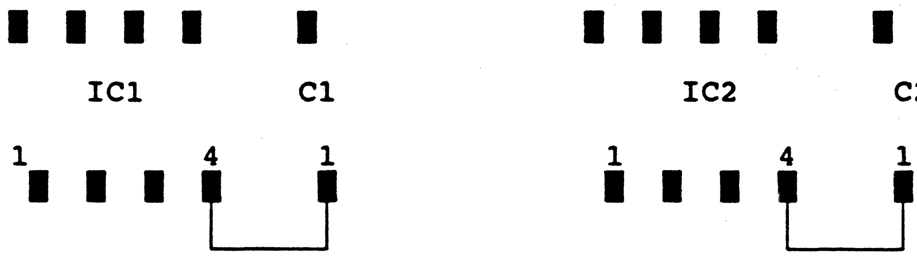


figure 3

When all connections have been calculated once, a complete iteration is performed.

The single subsequent iterations are controlled by varying parameter sets (cost functions) in order to achieve a 100% routing by means of different strategies.

The router operates according to the "**rip up and re-route**" principle. I.e. tracks which have already been routed but prevent the router from further distributions will be routed anew.

During the first iteration (starting the router with "-n"), a **busrouting** is performed. This means only track courses are routed which can be realized without any through connections. To get an optimal busrouting for **memory structures**, take care that the distance between pins of neighbouring components which are to be connected is not larger than 0.3 inch or 7.62 mm.

The **priority direction** for the routing is as follows:
- on the solder side and the respective inner layers it is the x-direction
- on the component side and the respective inner layers it is the y-direction.

The **track types** are routed in a descending order of their width (i.e. the widest first, the smallest last).

Minimize the number of **"off-grid" pins** as far as possible since the router blocks additional grid points when connecting these pins. Besides, "off-grid" pins are not connected at the center.

The router treats **text fields** on signal layers as blocking areas, i.e. no segments or vias may be placed in the text fields during the routing.
For text fields on documentation layers, a via blocking may be defined in CIS.

Glue pads of SMD components are also blocking areas for vias and tracks.

Plane connections

For **SMD pins**, the router connects track segments to the pin and generates a via to make a connection to the plane possible. If there is not enough space for a via, the connection can also be realized by connecting such a pin to an already existing via.
If a net contains through-connected pins defined as plane connections, the router may also realize a plane connection via these pins.
Through-connected pins defined for a plane connection are not considered by the router since these pins go through all layers and are automatically connected to the plane.

B. MULTILAYER ROUTING

For the router the number of layers of a multilayer is limited to a maximum of 32 layers, or 16 layer pairs.

When routing multi layers, part of the connections is routed on the additional inner layers. As a rule, the parameter control is the same for every layer, i.e. the first iteration on the inner layer pairs is the same like the first iteration on the outer layers; this also applies to the following iterations.

1. Proceding of the router (for a new start = option -n)

a. outer layers

The router always starts the multilayer routing on the outer layers, independent of the number of layers (if this is not prevented with option "l:n").

First the router tries to route all connections on the outer layer that, according to the netlist, are allowed to be routed there.

When using the standard control file ldrout.CTL, the router executes a **busrouting** in the first iteration on the outer layers and changes then to the inner layers.

b. Inner layers

After the completion of the iterations on the outer layers, the router tries to distribute the tracks of the still missing connections on the inner layers. The first iteration on the inner layers is also a bus-routing iteration.

On the inner layers, the procedure is the same as on the outer layers; the router tries to route the connections not yet found on the current layer pair provided that these connection layers have been defined in the netlist.

The number of connections not found on the outer layers does not always have to correspond to the number of calculated connections on the inner layers.
The reason is that connections that have been specified for certain layers are not calculated on the other layer pairs.

There is the general possibility that the number of missing connections does not correspond to the number of non-connected connection points, since the number of missing connections refers to the whole board (= all layers), whereas the number of non-connected connection points refers to the calculated connections of the current layer pair.

With multi layers with SMD components the output differ in so far as the generated stubs are considered as being additional connection points - as long as they are not connected or deleted by the router.

2. Important notes for multilayer routing

a. With the respective layer specification at the signal parameters in CIS, the router can be 'forced' to distribute **nets on special layers** (default for the "layers" is "*", i.e. all layers).

By specifying the connection layer in the terminal parameters in CIS, you can define on which layer a terminal is to be connected (default is "*", i.e. all layers).
If the specified terminal connection layers do not include the layer(s) on which the pin really exists (SMD pins only exist on the solder or component side), the router ignores the layer specification and tries to connect the pin where it really exists.

b. Vias go either through all layers (drilled through vias) or only through one layer pair (buried vias). The router can only place **buried vias** on the inner layers if a via type that is defined as not drilled through has been assigned to the respective track type in CIS (attribute "B" has to be specified).

Note:

In the following cases, the router uses **drilled through vias** even if a via type not drilled through has been specified in the technology for the track type to be routed:
a. for SMD plane connections,
b. when creating 'stubs' for SMD connections,
c. if layers are routed together that are not in the same production layer pair (e.g. if the layer plan contains empty layers).

c. Multilayer routing may be aborted on any layer pair. When restarting the program without the option "-l:n", the router starts again on the outer layers.

C. ROUTING WITH SMD TECHNOLOGY

When routing with SMD technology, the proceding of the router differs from the description above.

1. Stub and busrouting

When using the standard control file ldrout.CTL, the router procedes as follows:

- In the first iteration on the outer layers, the necessary stubs are created.
- In the second iteration, the busrouting is executed on the outer layers; existing stubs can be used as connections.
- Then the router changes to the inner layers and performs the busrouting there; existing stubs can be used as connections.

2. Stub creation

One or several track segments of the SMD pins to be connected are routed and a through connected via is placed at the end. The result is that the connection of the SMD pin with this via is now possible on all layers.

Figure 1 shows a possible placement constellation of SMD-components. Pins 2 of the SMD components placed on the component and the solder side are to be connected. This connection can only be realized with a stub.

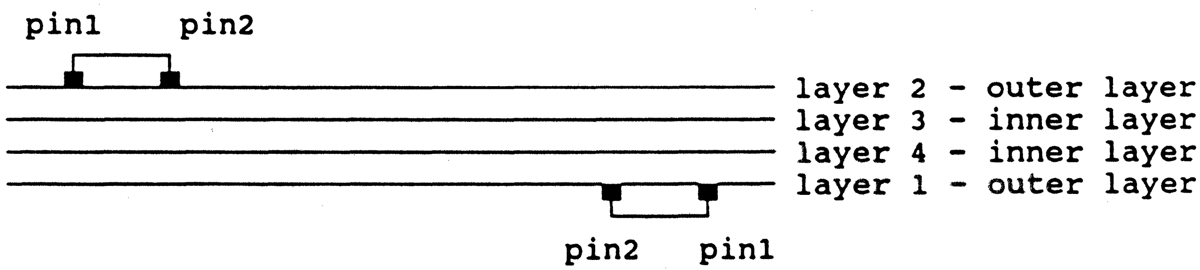


figure 1

In the first iteration, the router generates a track segment with a via at both SMD pins, see fig. 2.

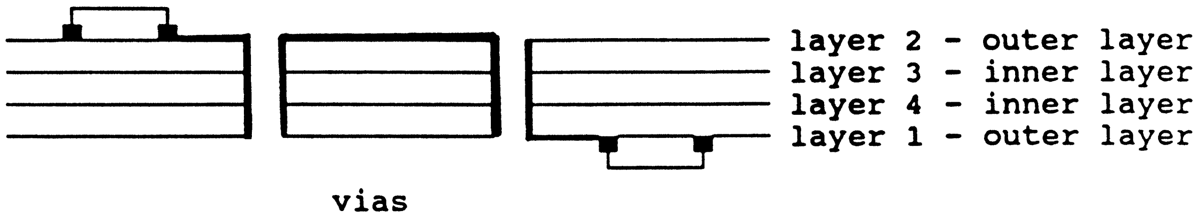


figure 2

Note:
- The stubs created by the router will be kept until the connection has been realized. If a stub is not necessary for the routing, it will be deleted automatically.

- The router considers the stubs as additional connection points (apart from the terminals) until they are either connected or deleted by the router.

VII. GENERAL INFORMATION

1. Before starting the router, a better power supply can be achieved by digitizing fixed tracks in a comb structure.

2. In order to achieve optimum routing of a memory structure, the distance between two component pins should not be larger than 0.3 inch or 7.62 mm.

3. The component placement in the layout also determines the routing success of LDROUT. The user should therefore critically observe the first iterations. The reason for an early stagnation may be a placement problem.
   The placement should be done in a 1/10" grid, because this grid is a multiple of the standard routing grids (1/20", 1/40", 1/50", 1/60", 1/70", 1/80") and offgrid placements are avoided when changing the routing grid.

4. If possible, offgrid pins should not be used.

5. Only after the glossing, the layout is ready for production.
   Repeated glossings ('restart' with option "-g") increase the quality and minimize the number of through connections.
   The provisional results that are created when the routing is aborted should also be smoothened for further processing by restarting the router with the option "-g".

VIII. ROUTER MESSAGES

We differentiate between error messages, warnings and comments. Errors abort the program.
Warnings inform the user about problematic constellations; the program will not be aborted. But warnings may also indicate difficulties of the router.

At present, not all of the existing router messages are described here.

A. ERROR MESSAGES

1. 'element outside area'

One or several components are outside the routing area.

Elimination:
Get the component(s) in CIS/PCB and place them within the routing area.

2. 'board too large'

The work sheet for this board is too large.

Elimination:
Change the sizes in the technology (CIS/PCB).
The maximum size in the X and Y direction that can be processed by the router is:
routing base x 32767 (mm)

3. 'too many routable seg-types'

More than 16 track types are marked as routable in the track technology, and more than 16 track types are used for the signal generation.

Elimination:
Change the track assignments for signals so that a maximum of 16 track types are used (in the menu "signals/mod. param").

4. 'illegal subtree-assignment'

The .adb-file contains unallowed subtree assignments.

5. **'multiple track-widths in subnet'**

   Subnets defined in the .adb-file contain several track widths.
   Each subnet, except for subnet 0, may only contain one track type.

6. **'track-width in higher subnet is larger'**

   The assignment of the track types to the subnets is incorrect.
   Higher subnets must contain track types of smaller or of the same track width of lower subnets, i.e. the track width for subsignal n+1 must be the same or smaller than the track width for subnet n.

7. **'subnets not continuously defined'**

   The subnet numbers have not been assigned correctly. The numbers have to be defined continuously (there may not be any 'gaps' in the number sequence!).

8. **'no subnet 0 in signal'**

   During the definition of subnets, the subnet number "0" has not been assigned.
   If a net contains subnets, there has to be a subnet 0!

9. **'no license to run program'**

   The license for LDROUT does not exist.

B. WARNINGS

1. **'... pins off-grid'**

   The number of pins that do not lie on the grid is output.

C. COMMENTS (also see chapter "output format")

1. **'... missing connections'**

   The number of missing connections is output.

2. '... of them in fixed nets'

The number of connections belonging to 'fixed' nets (i.e. to signals specified as "not routable" in CIS) is output.

Note:

The router displays these connections as missing connections, but finishes the routing when all other (= the 'non-fixed') connections have been found.

A subsequent technology check in CIS shows the appropriate number of interrupts!

CDBL - Customer Distributed Bugs List

April 1990
Issue 1990-06

Central Support Services

CALAY

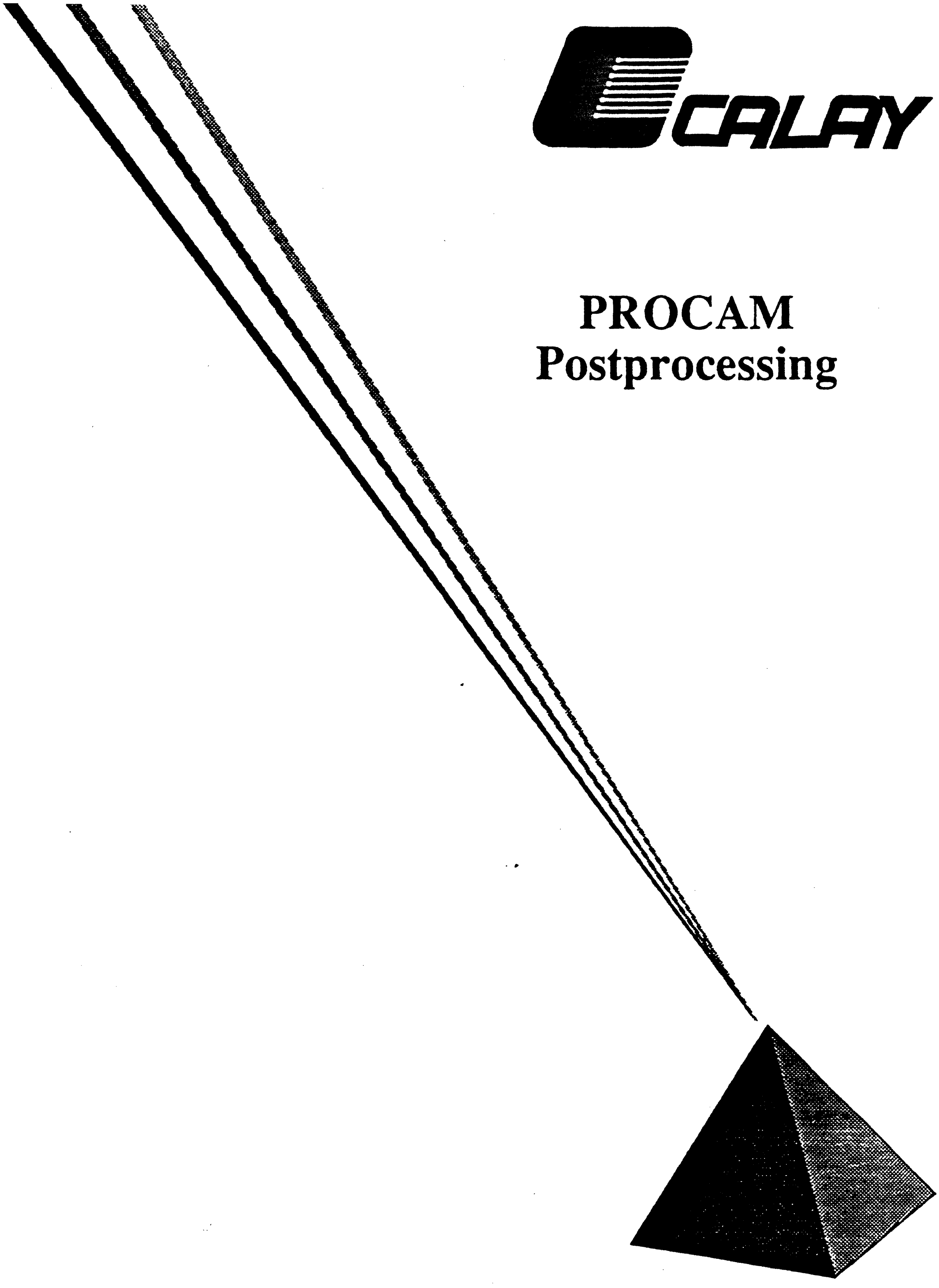
Solved Bugs

Known Bugs

Solved Bugs

CALAY

PROCAM Postprocessing



T A B L E   O F   C O N T E N T S

P O S T P R O C E S S I N G - P R I S M A

I. INTRODUCTION

1. General

The superior postprocessing function (POSTPROCESS.-icon) comprises all data output possibilities of the system that refer to the PC-board production.

The data basis for the postprocessing is the .adb-file generated in CIS and processed by the router. For the post-processing the data base of the .adb-file which refers to the whole PCB, has to be "splitted up" into layer-specific data bases. This is done by the program **APHCOM** which generates a separate aph-file for every layer.
In addition to the layer-.aph's the program generates an .aph with the netlist, if applicable an .aph with the NC-routing information and, if applicable, one or more .aph's with the drill information.

2. Function selection

After clicking the POSTPROCESS.-icon, the main menu for selecting the postprocessing functions appears:

```
******************************************************************
*   PRISMA  POSTPROCESSING                          V...         *
******************************************************************
                                    ** default selection: >>2<< ***

    1.   Generate or modify output procedures

    2    Execute 'output-procedures' (automatic mode)
    3    Execute 'output-procedures' (selection mode)

    4.   Generate plotter data (online/offline)

    5.   List or delete output-procedures
    6.   Run list-generator
    7.   View list-files

    8.   Transfer data to a serial port

    9.   Generate output for 'genrad tester'

    10.   Delete postprocessing-data-files

    E.   Exit
-----------------------------------------------------------------
    your choice ? _
```

The main part of the postprocessing is the program package **OUTPUT** (function 1 to 3). From the data bases of the aph-files OUTPUT generates output data for the creation of layout plots, drill and NC-routing tapes and insertion files.

In addition, the "POSTPROCESS." icon contains:

- the program **APHCOM** for an access to the postprocessing data for the single layers of the .adb-file;
- the program **PLOT** for creating the plotter control data (from the output data of OUTPUT and the respective drivers);
- the program **PROCED** for processing the procedure library;
- the program **CALPRI** for creating different lists from adb-files (netlists, component statistics, etc.);
- a 'view' function for the output of the list files generated during the work with the different postprocessing modules (postprocessing error file, CALPRI lists, drill protocol, NC-routing protocol)
- a 'transfer' function for sending data (ascii/binary) to one of the serial ports (ttya/ttyb) of your system;
- a program for the generation of control files for 'genrad' test systems

3. Notes to the documentation

The OUTPUT functions 1 to 5 are explained in the following chapters of this manual.
In addition, function 6 (=CALPRI) is described.
The rest of the functions is self-explanatory.

4. OUTPUT structure

The program OUTPUT consists of several modules (**userpp, readpp, sortpp, plotpp, drilpp, millpp, ercopp**) that are necessary for the creation of 'procedures', the read in and sorting of the input data and the creation of the output data.

'Procedure' means all functions and parameters to be executed when processing an aph-file.

The 'procedures' are defined in a form series of "userpp" in which you determine the output devices and their parameters, as well as the kind and form of the output (e.g. for plots: the form and means certain PC board elements are to be plotted with).

This means a procedure is the 'tool' with which an .aph-input file is processed to get the desired output data.

The 'procedures' are stored in the library **output.sys** and may be called at any time for creating output data for a new input file.
(The procedure library may be generated with the program BTOOLS - under the LIBRARY icon - and has to be stored on the directory lib/sys).

With the OUTPUT functions, output data are generated for:

a. all kinds of repro-fit layout plots for production (output files with the default extension **.gnn**)

b. the generation of NC-routing and drill tapes (output files with the default extension **.bnn** or **.fnn**)

(n is a real number between 0 and 9)

Figure 1 shows the output possibilities of the program OUTPUT and the interdependences with other programs of the PRISMA system (see page 1-4).

5. Menus for the procedure definition

The menu-driven OUTPUT module **userpp** serves for the generation and editing of procedures.

The menus contain functions and data fields for entries.
As a rule, function are selected by marking them with a cross.
Marked functions are activated by pressing <CR> and entries are stored by pressing <CR>.
<ESC><ESC> leads back to the previous menu (or to the preceeding menu part). The entries made in the menu you leave are then not considered.

Figure 2 on page 1-5 is an overview over all menus of the program.
The diagram illustrates the hierarchical organization of the functions with superior and subordinate menus and shows the possibilities for a movement within the menus.

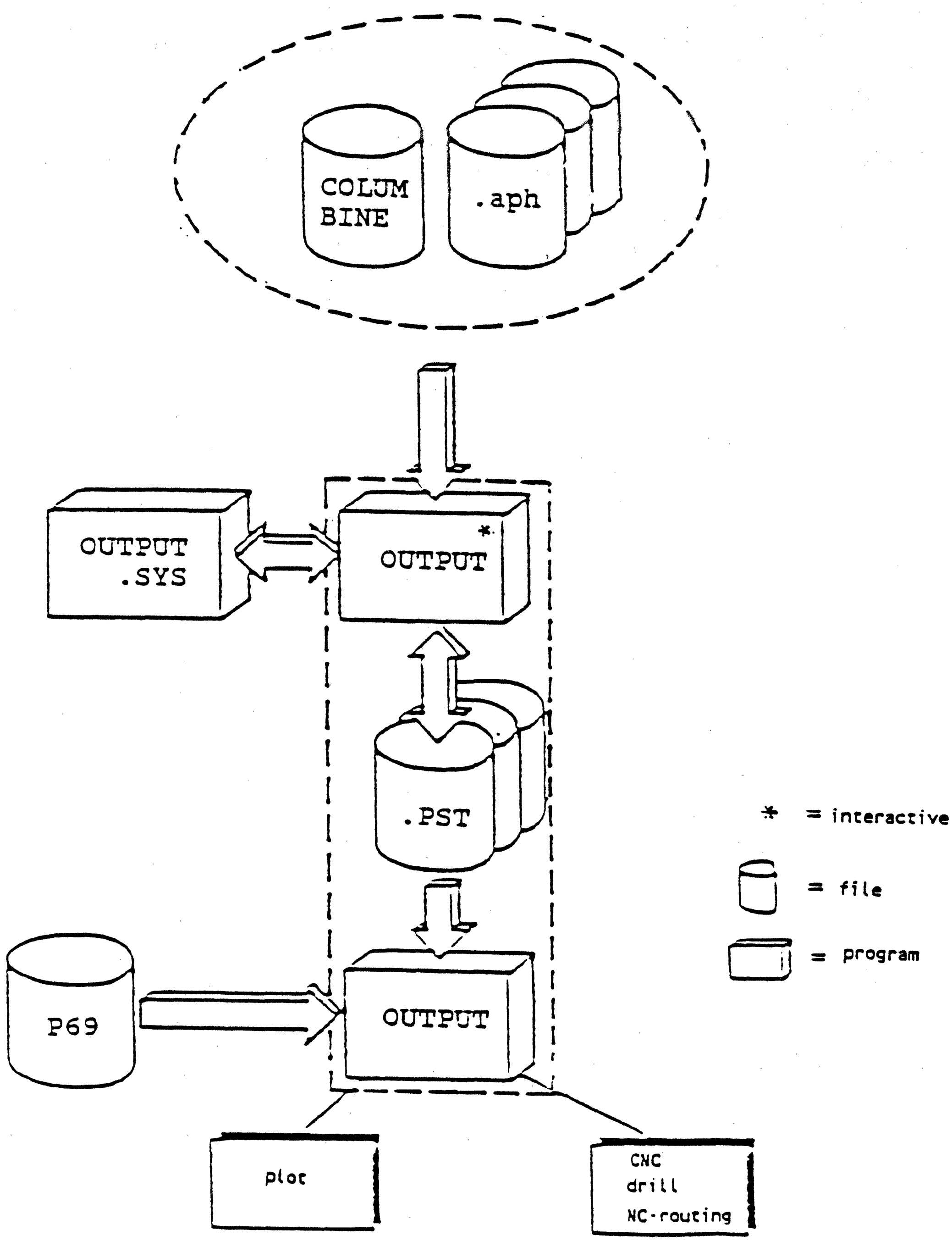
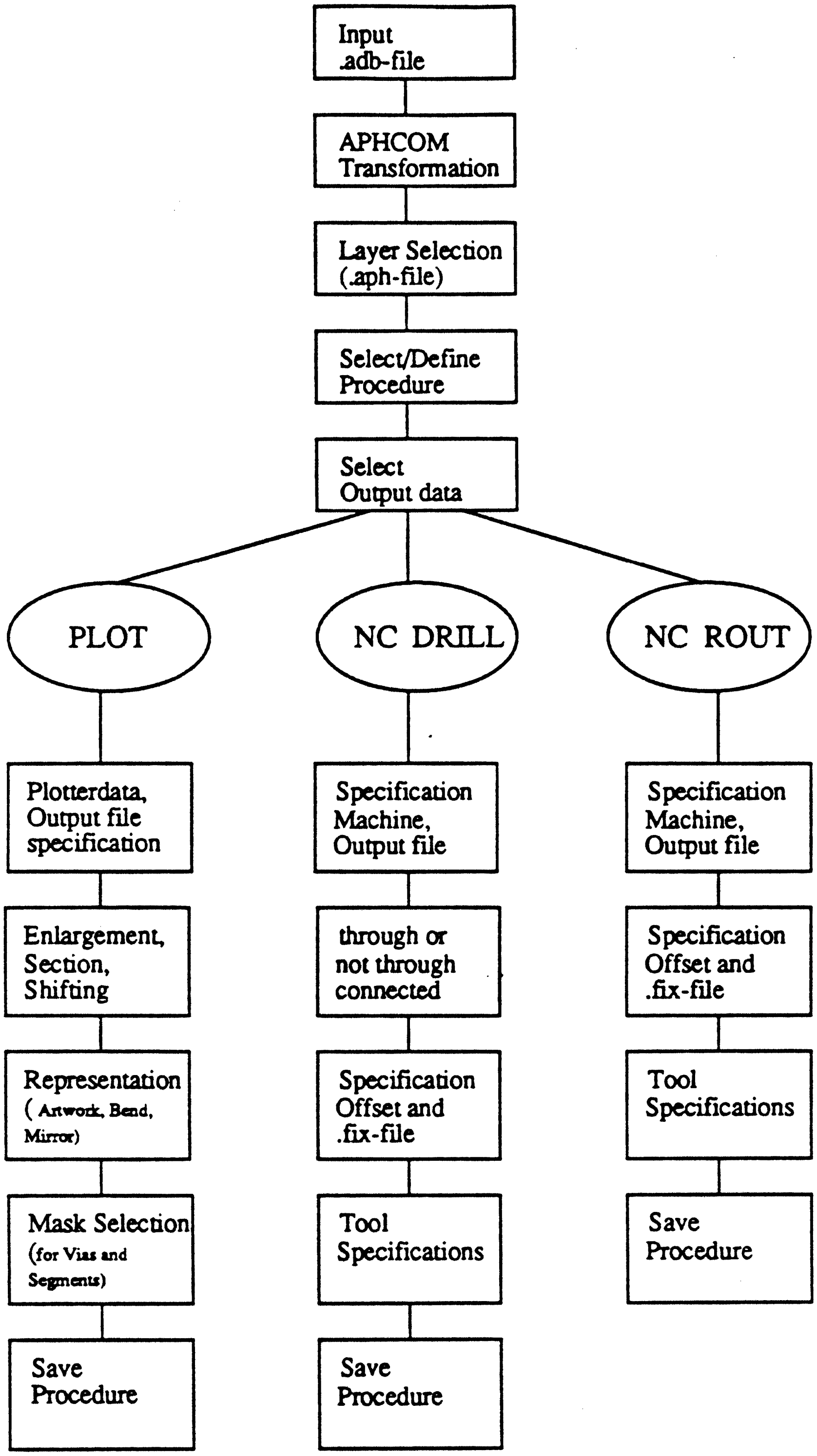

COLUM
BINE
.aph
OUTPUT
.SYS
OUTPUT *
.PST
P69
OUTPUT
plot
CNC
drill
NC-routing
* = interactive
= file
= program

fig. 1


Input
.adb-file
APHCOM
Transformation
Layer Selection
(.aph-file)
Select/Define
Procedure
Select
Output data
PLOT
NC DRILL
NC ROUT
Plotterdata,
Output file
specification
Enlargement,
Section,
Shifting
Representation
(Artwork, Bend,
Mirror)
Mask Selection
(for Vias and
Segments)
Save
Procedure
Specification
Machine,
Output file
through or
not through
connected
Specification
Offset and
.fix-file
Tool
Specifications
Save
Procedure
Specification
Machine,
Output file
Specification
Offset and
.fix-file
Tool
Specifications
Save
Procedure

6. Generation of the Postprocessing Data Base

The program **APHCOM** which controls the transformation of .adb-files into .aph-files is invoked with function "1" of the postprocessing selection menu.
When activating function 1/2/3 of that menu, APHCOM is automatically offered first.

From an .adb input file the following **.aph-files** are generated:

a. for every layer defined in the layer plan of the .adb-file a separate .aph-file is generated containing the layer-specific postprocessing data base;

name:
name_layer name.aph or **name_LAYnumber.aph**

i.e., the adb-name is supplemented with a mark containing either the layer name specified in the layer plan or - if it does not exist - the layer number defined in the layer plan.

b. a file containing the netlist as well as all other information of the .adb-file necessary for the list generation with CALPRI.

name:
name_NET.aph

i.e., "_NET" is attached to the name of the .adb-file;

c. if the .adb-file contains NC-routing, an .aph-file with the NC-routing information is generated;

name:
name_mill.aph

d. one or more "via-.aph's" with the information for the generation of drills;

name: (for all possibilities)

name_DALL.aph	vias going through all layers (phys.+doc.)
name_DBUR_x_y.aph	buried vias for layer pair x/y
name_D1_x.aph	one layer (x)
name_D2_x_y.aph	2 layers (x and y)
name_D3_x_y_z.aph	3 layers (x, y and z)
name_DFU_x_y.aph	"from-to" drill info (from x to y)
name.DPHY.aph	going through all physical layers
name_DDOC.aph	docu layers
name_DCBL_x.aph	"blind" vias from comp.side to x
name_DSBL_x.aph	"blind" vias from solderside to x

The messages output during the aphcom process are all stored in a file with the name of the .adb-file and the extension .lst.

In addition to that the program generates a contents file containing a list of all files generated during the processing of the .adb-file; it has the name **contents.lst.**

For the files to be generated from an .adb-file, aphcom generates an own subdirectory on the directory aph; the subdirectory has the name of the respective .adb-file.

After the start the program first asks for the name of the .adb-file to be processed (without extension):

enter .adb-file [without ext.] or ? for directory: ...

With "?" and <CR> you may list the files of the .adb directory. After the specification of the .adb-file and <CR> aphcom generates the subdirectory for the files to be generated; then the layer specific postprocessing data bases are taken from the .adb-file.

If a subdirectory of the .adb-file already exists on the aph directory, the program asks:

postprocessing directory for file ... exists, create new [y/n]?

With "N" and <CR> the transformation is not carried out; the .aph files already exisiting are used for the postprocessing.

With "Y" and <CR> the transformation is carried out; the "old" .aph-files are deleted, the new ones are stored.

7. Starting the Procedure Generator

The procedure generator **userpp** is started by clicking the POSTPROCESS.-icon and selecting function 1 in the appearing main menu.

After the program has been started, the program **aphcom** is invoked for the generation of the .aph-files from the .adb data base (see point 6.).
Then, menu 0 appears for the procedure selection.

PROCEDURES AND OUTPUT

1. Define/change procedures

After activating the POSTPROCESS.-icon and entering the selection number "1" APHCOM is invoked for the generation of the layer specific .aph files (see chapter "Introduction", point 6.).
After the APHCOM process has finished, the menu no. 0 appears in which the name of the current .adb-file has already been entered automatically. <CR> lists all .aph-files generated from the .adb-file.

```
***************************************************************
*              O U T P U T     Version ________          *  0  *
***************************************************************
*                                                             *
* name of the .adb-file: test______.aph                       *
*                                                             *
***************************************************************
*                                                             *
*   -   test_SOLD                                             *
*   -   test_COMP                                             *
*   -   test_SIG-2                                            *
*   -   test_SIG-3                                            *
*   -   test_SHAPE-1                                          *
*   -   test_SHAPE-0                                          *
*   -   test_RESIST-0                                         *
*   -   test_RESIST-1                                         *
*   -   test_MILL                                             *
*   -   test_DRILL                                            *
*   -   test_DPHY                                             *
*   -   test_DBUR_2_3                                         *
*                                                             *
***************************************************************
```

By marking <u>one</u> of the files with a cross and <CR> you choose the file for further processing.

Note:
If you would like to process another .adb-file, enter the name of that .adb-file instead of the one entered automatically; after <CR> the .aph-files generated from this .adb-file are listed.
Precondition: the new .adb-file has already to have been transformed into .aph-files with aphcom.

After selecting the .aph-file to be processed the list of the procedures existing in the library **output.sys** are read into the menu no. 0.

You may now either

-> generate a new procedure based on the .aph-file activated

or

-> use a procedure - as a rule generated from another .aph-file - for the current .aph-file and change it if necessary.

```
*****************************************************************
*              O U T P U T     Version ________          *  0  *
*****************************************************************
*                                                               *
* filename : test_SOLD                                          *
*****************************************************************
*   (1) Generate new procedure                                  *
*   (2) Change existing procedure      Choice  (1,2) ==> (2)    *
*****************************************************************
*   Existing procedures :                                       *
*                                                               *
*   SOLD    LAYER 1                                             *
*   COMP    LAYER 0                                             *
*                 .                                             *
*                 .                                             *
*                 .                                             *
*****************************************************************
```

By entering the respective number, you choose whether a new procedure is to be defined or whether an existing procedure is to be read in.

With <ESC><ESC>, you get back to the file specification, if the cursor is on the input field of the file name or of the selection number.

1.1 Menu description

- **name of the .adb-file:**

  Here appears the name of the current .adb-file for which the output data are to be generated with the specified procedure.

- **Create new procedure** = selection (1)

  The 'constellation' of functions, sizes and other data generated for the current .aph-file is to be stored into the 'procedure library' **output.sys** under the name specified under
  - procedure name:__________
  (max. 16 characters). After the definition this name automatically appears in the list of existing procedures in menu no. 0.
  A comment may be assigned to this new procedure in
  comment :__________________________
  (max. 39 characters incl. blanks).

  The menu part for the specification of this data appears after activating selection (1) and <CR>. With <ESC><ESC> the entry is aborted and the cursor is back on the field for entering the selection number.

  Note:
  When specifying the name of a new procedure please see to it that the procedure name is identical to the layer-specific name supplement of the .aph-file, on the basis of which it is generated (e.g. COMP for the file name_COMP.aph or RESIST-0 for the file name_RESIST_0.aph).
  This is the precondition for the "automatic" mode of generating procedures (see point 2.).

- **Existing procedures:**

  In the lower part of the menu, the already defined output procedures and their comments are listed.
  You get to this part of the menu by entering the selection number "2" (without <CR>!).
  If further procedures exist, you may page down with <CR>, as long as no procedure has been selected.
  With <ESC><ESC>, you get back to the field for the selection number.

- **Change existing procedure** = selection (2)

  The selected procedure is loaded from the library into the program and may now be tailored to the processing of the specified .adb-file.

Attention:

> If with an already existing procedure another .adb-file than the one used during the definition of the procedure is to be processed, the user has to go through all of the procedure menus relevant to the generation of the desired output data (incl. the menus for mask or tool selection!).
> Otherwise, instead of the technology of the current .aph-file, the technology of the .aph-file originally used for the procedure generation will be used for the generation of the output data.

If the changes are not to be considered, you have to finish the procedure editing in menu 80 without storing the procedure.

1.2 Storing the procedures

If a new procedure is created, you get directly back to menu 0 after finishing the procedure menus; name and comment of the procedure appear in the list of existing procedures.

If you edit an existing procedure for changing it and leave at least one of the menus with the command "S" (= Store), the inquiry menu 80 appears before you get back to menu 0.

```
***************************************************************
*              O U T P U T      Version ________       * 80  *
***************************************************************

***************************************************************
*                                                             *
*  changed procedure ...                                      *
*    (0) .. don't store procedure in library                  *
*    (1) .. store procedure with old name in library          *
*    (2) .. store procedure with new name in library          *
*    >0<                                                      *
***************************************************************
```

By entering the selection number and pressing <CR>, you specify if the changed procedure is to be stored and, if so, under which name.

Attention:

> If you store the changed procedure with its old name, the old procedure will be overwritten!

Afterwards, you get back to menu 0 where you may edit or create a procedure or leave the menu with <ESC><ESC> and return to the postprocessing selection menu.

2. Execute procedures

Procedures are always generated or changed on the data basis of an .aph input file. For the first execution of a newly generated or changed procedure the default is that the output data is generated from the .aph-file used for the generation or for the change.

A defined procedure may also be used for .aph-files which have been generated from other .adb-files, as a 'tool' for generating output data.

Precondition:
In the .adb-files from which the layer.aph's are generated, the <u>layer technologies</u> have to <u>be exactly the same.</u>

2.1 'automatic mode'

With the 'automatic' mode it is presumed that the layer-specific name supplements of the .aph-files generated from .adb-files are identical to the procedure names with which they are to be processed (e.g. procedure name "SHAPE-1" for the file "name_SHAPE-1.aph")

If you select funtion "2" of the postprocessing selection menu for the automatic procedure, at first the query appears:

enter .adb-file [without ext.] or '?' for directory: ......

With "?" and <CR> you may list the files existing on the .adb directory.

Since the subdirectory with the .aph-files from the .adb-file already exists, the following query appears after the specification of the .adb-filename with <CR>:

postprocessing directory for file ... exists, create new [y/n]?

As a rule, this is to be answered with "N" and <CR>, since the .aph-files already existing in the subdirectory are to be used for the postprocessing.

If the "old" .adb-file has been changed, and these changes are now to be considered by OUTPUT, you have to answer "Y" and <CR>, under the condition that the technology of the "old" .adb-file has not been changed.

If this applies, APHCOM first generates the "new" .aph-files; the "new" data are immediately used for the postprocessing.

If OUTPUT does not find any procedures for one or more .aph-files belonging to the .adb-file specified, the following message appears:

The following ... procedures do not exist:

together with a list of the missing procedures.
With the corresponding entry in

continue [y/n]

the procedure is either aborted or continued with the procedures belonging to the .aph-file.

Before actually carrying out the procedures, a list shows which procedures are used for which .aph-files. Then the system asks whether or not the execution is to be continued.

With "Y" the executable processes are listed ('the following layers will be generated:....). The system prompts again:

continue [y/n]

With "Y" the listed processes are executed ('batch operation'). For every process the PCB data contained in the .aph input file are transformed into output data comprising the functions defined in the procedure.

2.2 Selection mode

With this mode, <u>the layer-specific name supplement</u> (e.g. "COMP") of the .aph-file and the name of the procedure are entered via the keyboard for every process.
ATTENTION: specifications must be made in capital letters!

After activating the function with "3" and <CR>, the system asks for the specification of the .adb-file name.

enter .adb-file [without ext.] or ? for directory:

After the specification of the .adb filename and <CR> the message appears:

postprocessing directory for file ... exists, create new [y/n]?

For the proceding, see point 2.1

Then you are asked for the layer specific name supplement of the first .aph-file to be processed:

enter the layer name or ? for directory:

With "?" and <CR> the .aph-files of the aph directory are listed.

After the specification of the 'layer name' (e.g. "COMP") and <CR> you are asked for the name of the procedure to be used:

enter the procedure name, ? or <return> for '...':

With "?" and <CR> the procedure names available in the library 'output.sys' are listed.
If there is a procedure name identical to the 'layer name', it automatically appears as default (in "...for '...'") and may be confirmed with <CR>; otherwise specify the procedure name at the cursor position and press <CR>.

Then the system asks for 'layer name' and the procedure name of the next process.

Enter <CR> to finish the query.

OUTPUT carries out the single processes and generates the respective output data.

2.3 Output Files

The output files generated by OUTPUT always have the name of the respective .aph-file.
The default extensions of the output files are:

- **.g01** for plot output files
- **.b01** for drill output files
- **.f01** for NC-routing output files

The default extensions of the output files may be changed by the user within the appropriate procedure definiton.

Note:

By assigning different extensions to output files that have been created with different procedures from the same .aph-file, you may avoid that the output files will be overwritten.

In addition, OUTPUT creates the following report files which are all stored on the appropriate .aph subdirectory (name of subdirectory = name of the .adb file).

- an **error report** in the file **post.log**; here you will find the errors and warnings that occured when executing the procedure(s) with the input-.adb-file.

  Attention:
  Every time when executing the postprocessing selection function "2" or "3" with the same .adb file, the old error file is overwritten by the new one. If you want to prevent this, the old post.log-file has to be renamed at first.

- a **drill report** in a file with the name of the input file and the extension **.drp** (if the creation of drill tapes was also defined in the procedure);

- an **NC-routing report** in a file with the name of the input file and the extension **.mrp** (if the creation of NC-routing tapes was also defined in the procedure);

The report files are displayed with the 'view' function (enter "7" in the postprocessing selection menu).

3. Output of the plot data on the plotter (online) or in a file (offline)

The data of the .gnn-files is prepared for the plotter by activating function "5" in the postprocessing selection menu. The following menu appears:

```
*****************************************************************
*    P R I S M A   P L O T   S E L E C T I O N        v....   *
*****************************************************************
                                   *** default selection: >> 0 << ***

  1.      : create 'hp plot'
  2.      : create 'gerber plot'
  3.      : create selected [....] file
  4.      : select plotterdriver
  5.      : show plot directory

  0.      : Upper level
  E.      : Exit
 _____________________________________________________________
  your choice?  _
```

You can directly select the output for an HP plotter or a GERBER plotter.
For the creation of offline files with another driver, you have to select the driver first (selection number "4").

```
*****************************************************************
*    P R I S M A   P L O T T E R S E L E C T I O N    v...   *
*****************************************************************
  1.          Aristo
  2.          Awiota
  3.          Benson
  4.          Cal-Comp
  5.          Cal-Comp 960
  6.          DMP
  7.          Marconi-Emma
  8.          Glaser
  9.          Secrol
 10.          Watanabe-wx463x
 11.          Postscript

  0.         Upper level
 _____________________________________________________________
  your choice ? _
```

Of course you can only specify the drivers for which you have a licence.

After the driver specification, you return to the menu for the plot selection; the name of the selected driver automatically appears in brackets under point "3".

After entering 1/2/3 in the menu for the plot selection and pressing <CR>, the system prompts with:

create offline plot [y/n] ?

Online Plot Generation

If you have a plotter and an appropriate plotter connection, i.e., if you answer the query "create offline plot?" with "N", the system asks for the specification of the plotter connection port:

select plotter port [ttya, ttyb]:

After specifying the port and <CR> the system asks whether several plots are to be generated on the same sheet (or film):

Are there multiple plots on one sheet [y/n] ?

If you answer with "N" and <CR>, the system asks for the specification of the .gnn file(s) for the plot (plot #1) (see below).

If you answer "Y", i.e. if multiple plots are to be generated on the sheet, and press then <CR>, the system first asks for the offset in x or y direction between the single plots:

Enter the offset [0-99] between 2 plots [mm]:

After the offset has been specified and <CR> has been pressed, the system asks for the selection of the .gnn-file(s) to be used for plot #1:

Please enter 1st plot file name for plot #1 :

In the upper part of the shell window the **gdi** directory with all plot files is listed. Now, you can specify the name of the .gnn-file from which the first plot is to be generated on the sheet - either by entering the name of by using the "stuff" procedure with the mouse.
After a <CR>, the system asks for the name of hte second .gnn-file for the second plot, etc. - if several layers are to be plotted one on another (=control plot).

If only one layer is to be plotted (production plot), press <CR> twice after entering the file name.

After the .gnn-files have been specified for plot #1, the position of plot #2 referred to plot #1 is queried. At first, specify the offset as an integer multiple of the x-value of plot #1 and then specify the offset as an integer multiple of the y-value of plot #1:

--> enter the X location [0-99] for plot #2 :
--> enter the Y location [0-99] for plot #2 :

If you specify "1" for the X position and "0" for the y position, plot #2 is placed left of plot #1. The two plots have the distance specified in "enter the offset [0-99] between 2 plots [mm] :".

After the positions are specified and <CR> is pressed, the system asks for the selection of the .gnn-file(s) for plot #2:

Please enter 1st plot file name for plot #2:

After the files to be used for plot #2 are specified, the position of plot #3 is to be entered, if existing. At first, specify the offset as an integer multiple of the x-value of plot #1 and then specify the offset as an integer multiple of the y-value of plot #1. If plot #2 is bigger in x or y direction than plot #1, the system automatically adds this value, so that an overlapping of plot #2 and plot #3 is ruled out.

After the positions are specified and <CR> is pressed, the system asks for the selection of the .gnn-file(s) for plot #3, etc.

After all specifications have been made, the system is ready for executing the plots:

Prepare plotter and press <return> to start online plot

Generation of an Offline Plot File

If an offline plot is to be generated, and if you answer the query "create offline plot?" with "Y" and <CR>, the system asks for the name of the offline plot file to be generated

enter the file name [without extension] for the output plot file :

After specifying the file name with <CR>, the system asks hether a multiple plot is to be generated on one sheet (or film), etc.
The further proceding corresponds to the "Online Plot Generation".

4. Processing the procedure library output.sys

The procedure library can be processed with the program **PROCED.**

With this program,
- procedures may be deleted from the library;
- the contents of the different procedures may be listed in an ASCII-file;
- the procedure library may be squeezed by moving the free blocks together to get more space.

When selecting "6" in the postprocessing main menu, the following menu appears:

```
************************************************************************
*         PRISMA POSTPROCESSING  list procedures               v....  *
************************************************************************
                                          *** default: >> 0 << ***
  1. Delete procedures or create procedure listfiles
  2. View an existing procedure-list-file

  0. Upper level

  E. Exit
 ______________________________________________________________________
  your choice ? _
```

With "2", you can view a procedure list file created with function "1" on the screen. After selecting this function, you are asked for the name of the list file (without extension).

If you enter "1", the program PROCED is started. The following menu appears:

```
***  *******************************************************************
*                          PROCED                                      *
************************************************************************

     number of procedures          :
     number of deleted procedures  :
     number of deleted blocks      :
     number of free blocks         :

************************************************************************
*  _     COMP             layer plan 1                                 *
*  _     RESIST-0         solder resist 0                              *
*  _     SHAPE-1          insertion pl. 1                              *
*  _     DRILL            drill plan                                   *
************************************************************************
```

SELECTION OUTPUT DATA

1. Introduction

If you have defined a new procedure in selection menu 0, or if you have selected an existing procedure with <CR>, menu 2 appears where you specify the purpose of the generated output data by selecting the respective function.

2. Menu 2

O U T P U T	2

Plot	**:**	_
Drill	**:**	_
NC-routing	**:**	_

3. Menu description

- **Plot:**

  Data for the control of pen, photo or laser plotters are generated.

- **Drill:**

  Data for the control of NC-drilling machines are generated.

- **NC-routing:**

  Data for the control of NC-routing machines are generated.

Only one function may be selected.

PROCEDURE DEFINITION PLOT

1. Introduction

If you activated the function "Plot" in the menu for the selection of the output data, you get to several menus for the detailed definition of the plot(s).

A procedure serves for generating a plot output file from a layer-.aph-file. The output file gets the name of the .aph-input file and the default extension .g01.

The upper part of each menu belonging to the "plot" function contains information on the following parameters:

procedure: is the name of the current procedure

plotter: is the name of the plotter selected

output file: is the name of the plot output file

When newly generating a procedure, the plotter specifications appear only after the specification of the plotter name in menu no. 3.

2. DEVICE SPECIFICATION

After the activation of the function "Plot", menu 3 appears where you determine the name of the output device and where you may change the extension of the output file if applicable.

2.1 Menu 3

```
*****************************************************************
* PROCEDURE      :  COMP                                 ***** *
* PLOTTER        :  HP                                   * 3 * *
* PICTURE NO.    :  test_COMP.g01                        ***** *
*****************************************************************

*****************************************************************
* output file    : test_COMP                 .g01               *
* output device  : HP                                           *
*****************************************************************
*                                                               *
*                      PLOTTER DATA                             *
*                                                               *
* driver name : HP              device type : 1                 *
*                                                               *
* Routing area   in X : 999.998 mm     in Y      : 999.998 mm   *
*                                                               *
* Correction     in X : 1.0000         in Y      : 1.0000       *
*     factor                                                    *
* step 1/100000 mm  : 2500           masks : 8                  *
*                                                               *
*****************************************************************
```

2.2 Menu description

Output file:

Name and extension of the output file are automatically entered.
As a rule, the plot output files are stored on the directory **gdi** and have the name of the .aph-file and the default extension .g01.
The extension may be changed, if necessary, e.g. if an output file with the same name is already existing (because this .aph-file has already been processed with another procedure).

Output device:

The name of the output device (=plotter) for which the output data is determined, is entered here. The output device has to be defined in the library COLUMBINE.

Notes:

a. "*" and <CR> instead of the name of the output device list all names of the plotter defined in COLUMBINE After finishing the listing with <CR>, the first device name is automatically entered in "Output device".

b. In order to get the desired kind of output data, you have to make sure that the entered output device is defined in COLUMBINE with the respective parameters (e.g. masks) (ref.to the COLUMBINE description, chapter 3.5.1).

After the output device has been entered with <CR>, part of the parameters of the specified plotter defined in COLUMBINE are automatically entered into the menu.

If an already defined procedure has been loaded, the respective plotter name automatically appears in "Output device".

Attention:

If you use an already defined procedure and change the name of the original output device (i.e. you want to use another plotter and other masks), you have to go through all mask menus and check the mask assignment for the defined plots.

PLOTTER DATA:

After entering the output device and <CR> the data from COLUMBINE is entered into "PLOTTER DATA". This only serves for information and cannot be changed.

3. PLOT SPECIFICATIONS

After you have defined the output device, you get to the menu for the specification of the section, the enlargement and the shift of the plot.

3.1 Menu 4

```
******************************************************************
* PROCEDURE    :  COMP                                     ***** *
* PLOTTER      :  HP                                       * 4 * *
* OUTPUT FILE  :  test_COMP.g01                            ***** *
******************************************************************

******************************************************************
*                                                                *
* ENLARGEMENT of the plot       : 1.0000                         *
*                                                                *
******************************************************************
*                                                                *
* SECTION       (1) $$PCB,  (2) $$BOARD,  (3) WINDOW   ==> (1)   *
*                                                                *
*       if   WINDOW : Koordinaten  X1: 0.0___   X2: 160.0003     *
*                                                                *
*                              mm  Y1: 0.0___   X2: 110.0001     *
*                                                                *
******************************************************************
*                                                                *
* SHIFTING of the plot        :  in X: 0.0___  in Y: 0.0___  mm  *
*                                                                *
* ****************************************************************
```

3.2 Menu description

Enlargement of the plot:

Here, you may enter an enlargement factor for the output of the current plot. Real numbers between 0,1000 and 100,00 may be used as enlargement factor. The default is 1.

Section:

By entering the respective selection number and the coordinates if necessary, you determine the section of the working sheet defined in CIS/PCB that is to be represented in the current plot.

The starting point (origin) of the plotter is the lower left corner of the defined section.

$$PCB:

If you specify this section, the entire working sheet is represented.

$$BOARD:

If you specify this section, only the PC board area is represented.

WINDOW:

Only the section (window) specified in "Coordinates" is considered for the plot.

X1, Y1 define the lower left corner
X2, Y2 define the upper right corner

of the board section to be plotted.

The reference point for these specifications is the origin of the working sheet.

SHIFTING the plot:

Here, you determine whether and if so, how the current plot is to be shifted with reference to the starting position on the sheet or film (see chapter "Procedures and Output", point 3, page 2-11 ff).
For the X and the Y coordinate the shifting is entered separately (in the measurement defined in **calay.usr**)

Note:
If the shift refers to a plot already existing on the sheet, a possible enlargement or reduction of the existing plot has to be considered for the calculation of the coordiantes.

4. REPRESENTATION OF THE PLOT

After you have saved your entries with <CR> in menu 4, you get to menu 5 to determine the representation.
Here you may specify the parameters "Artwork" and "Bend" for the representation of the PC board elements, the numbers of the special pens to be used, etc.

4.1 Menu 5

```
*******************************************************************
* PROCEDURE    :  COMP                                      ***** *
* PLOTTER      :  HP                                        * 5 * *
* OUTPUT FILE  :  test_COMP.g01                             ***** *
*******************************************************************

*******************************************************************
* ARTWORK :         (3)              *        BEND :          (2)    *
*                                    *                               *
*   Scheme          (1)              *          Angular       (1)    *
*   Contour         (2)              *          Slanted       (2)    *
*   Repro           (3)              *          Rounded       (3)    *
*******************************************************************
*              Exactness     -0.00 __/+0.00__ (MM)                  *
*******************************************************************
* MIRROR    :                                                     *
*         mirror everything: _, DIN-texts: _  mirror $$BOARD: _   *
*******************************************************************
* Special pen for texts            : 1__                          *
*******************************************************************
```

4.2 Menu description

ARTWORK:

By entering the respective selection number, you may define the general representation of the board elements here.

Scheme (1):

This representation applies to control plots.

- tracks of type 0 are displayed with a line;
- track type 1 is displayed with a line;
- all other track types are displayed in their contours (=not filled) and only with the core (= track width minus 2 x radius).
- vias are displayed unfilled.

Contour (2):

This kind is used for survey plots as a rule.

- tracks of type 0 are not plotted;
- track type 1 is represented with a line;
- all other track types are represented in their contours (=not filled) and only with the core (= track width minus 2 x radius).
- vias are represented unfilled.

Repro (3):

This kind is used for repro-fit creations of layouts.

- tracks of type 0 are not plotted;
- all other track types are displayed filled.
- vias are displayed filled.

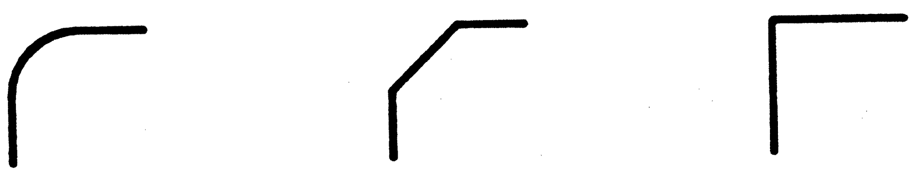
BEND:

With the help of this parameter, the shape of the track angles is determined by entering the respective selection number. The slant or rounding is achieved with a grid cell of the grid defined in the .adb-file.

Rectangular (1):
rectangular angles;

Slanted (2):
slanted angles (45 degrees);

Rounded (3):
rounded angles



3 = rounded 2 = slanted 1 = rectangular

EXACTNESS:

If round or square vias are to be flashed and if the mask is automatically chosen, you may define how far the width of the flash mask may maximally diverge from the width of the vias to be plotted defined in the technology, so that it can still be used for the creation of these vias.

Attention:

This parameter has no effect on the automatic selection of

- the flash masks for square vias with rounded corners;
- all plot masks.

MIRROR:

By marking the respective field with a cross, you may determine the mirroring and/or adjustment of the texts.
It is possible to select several functions.

mirror eyerything:

The total work sheet is mirrored; i.e. all elements (segments, vias, frame, text fields and texts) are mirrored.
Mirror axis is the Y-axis in the middle of the work sheet.

mirror $$BOARD:

The board is mirrored, i.e. all elements (segments, vias, frame, text fields) belonging to the special component $BOARD are mirrored.
Attention: the **texts** in the text fields are **not mirrored.**
Mirror axis is the Y-axis in the middle of $$BOARD.

DIN-texts:

All texts are adjusted (rotated and/or shifted) that they can be read in the DIN-direction (i.e. horizontally from the left to the right and vertically from the bottom to the top). If "mirror everything" and/or "mirror $$BOARD" have also been selected, the texts are first mirrored and then adjusted.

Notes:

- If "mirror everything" as well as "mirror $$BOARD" have been selected, all elements belonging to $BOARD - except for the texts! - are mirrored twice! In this way, the **texts** are only **mirrored once**. But you have to take care that the board ($BOARD) is placed symmetrically on the 'work sheet' so that the two mirror axes are congruent!

- In the appendix to the POSTPROCESSING description (manual), you find an example that illustrates the 'MIRROR' function with all possible variations.

Special pens for texts:

All text fields of the current plot are plotted with the specified plot mask.

5. MASK SELECTION

5.1 Mask selection for the via representation

As a principle, the mask selection is effected **automatically** with the priority on **flash masks**.

If necessary, the masks and load symbols for the representation of the different via types in the current plot can be **changed manually**, if the automatic mask selection is not successful (i.e. if technology violations are found).

```
*****************************************************************
* PROCEDURE     :  COMP                                  ****** *
* PLOTTER       :  HP                                    * 27 * *
* OUTPUT FILE   :  test_COMP.g01                         ****** *
*****************************************************************

*****************************************************************
*VIA* WIDTH   * AUTOMAT.* MASK    MASK * DIAMETER * LOAD-   * FLASH *
*TYP* (mm)    * MASKSEL.*  A        B  * (mm)     * SYMBOL  * (Y/N) *
*****************************************************************
* 0 *         *         *              * A:       *         *       *
*   *         *         *              * B:       *         *       *
*   *         *         *              *          *         *       *
* 1 * 0.198   *         *              * A:0.198  *         *       *
*   * 1.199   *    J    *   1          * B:       *         *   N   *
*   *         *         *              *          *         *       *
* 2 * 0.0__   *         *              * A:0.198  *         *       *
*   * 1.270   *    J    *   1          * B:       *         *   N   *
*   *         *         *              *          *         *       *
* 3 *         *         *              * A:       *         *       *
*   *         *         *              * B:       *         *       *
*****************************************************************
*                                       * Paging   (+/-) >+<*
*****************************************************************
```

Menu description

VIA TYPE:

This field contains the via types defined in the via technology of the adb-file.

WIDTH:

This field contains the via sizes defined in the CIS/PCB technology.
The **first value** stands for the radius. "0.00" implies square vias.

The **second value** stands for the via width.

For round vias the first value has exactly to be half of the second value (radius = 1/2 width).

For square vias with rounded corners the value for the radius has to be bigger than 0 and smaller than half of the width (Ref. to CIS/PCB technology).

FLASH (Y/N):

Y:

If automatic mask selection has been selected and OUTPUT finds a suitable flash mask or, for square vias with rounded corners, a mask for flashing the 'inner circle' of the via, "Y" is automatically entered here.

If the user enters "Y" for a manual mask selection, OUTPUT uses the specified mask regardless of whether it is suitable or not, provided that it has been defined in COLUMBINE (if it has not been defined as flash mask, a respective message appears).

N:

If OUTPUT does not find any suitable flash mask during the automatic mask selection, "N" is automatically entered here.
For the manual mask selection, the user has to enter "N" if mask A (existing as plot and flash mask) is to be used.

AUTOM. MASK SELECTION:

Y:

When calling this menu for the first time, "Y" is entered in "AUTOMATIC MASK SELECTION" for every via type the width of which is > 0 if the automatic mask selection has been successful for this type.

The **proceding** during the automatic mask selection depends on the shape of the via type.

1. **For round vias** (radius is 1/2 width)

a. If a suitable round flash mask (corresponds to the technology ± exactness) is found, it is used for the via generation.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; "Y" is entered in "FLASH"; for mask B no entry is made.

b. If no suitable flash mask is found, a round plot mask is searched in COLUMBINE the width of which corresponds to half of the via width.
If this search is not successful, the next smaller round plot mask defined in COLUMBINE for this plotter is used for the creation of this via type.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; for mask B no entry is made; "N" is entered in "FLASH".

If the smallest plot mask is bigger than half the via width, the message "Technology violation" appears.
Now "N" is entered in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects respective manual changes.

2. **For square vias** (radius =0)

a. If a suitable square flash mask (corresponds to the technology ± exactness) is found, it is used for the via generation.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; for mask B no entry is made; "Y" is entered in "FLASH".

b. If no suitable flash mask is found, OUTPUT searches a square plot mask the width of which corresponds to half of the via width. With this mask, the via is plotted, starting from the centre.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; for mask B no entry is made; "N" is entered in "FLASH".

c. If this mask is not found, a round plot mask is searched the width of which corresponds to half of the via width.
If the search is not successful, the next smaller round plot mask is used.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; "N" is entered in "FLASH".

In addition, the smallest available round plot mask is searched in order to trace the via contours and fill the remaining gaps (= via corners).
The number of the selected mask is entered in "MASK B", its diameter appears in "B:"

If the smallest plot mask is bigger than half of the via width, the message "Technology violation" appears.
Now "N" is entered in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects a respective manual change.

3. **For square vias with rounded corners**
(0 < radius < 1/2 width)

a. If a suitable flash mask (corresponding to the technology) is found, it is used for the via generation.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; for mask B no entry is made; "Y" is entered in "FLASH".

b. If no suitable flash mask is found, OUTPUT tries to find a round flash mask the diameter of which corresponds to the via width. This is used to flash the 'inner circle' of the via.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; "Y" is entered in "FLASH".

In order to trace the via contours and fill the corners, a round plot mask is searched the diameter of which fulfills the following condition

D = 2 x via radius

If such a mask cannot be found, the next smaller round plot mask is used.
The number of the selected mask is entered in "MASK B", its diameter appears in "B:"

c. If such a round flash mask could not be found, a round plot mask is searched the width of which corresponds to half of the via width. This mask is then used for 'filling' the via starting from the centre.
If it cannot be found, the next smaller available mask is used.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; "N" is entered in "FLASH".

As regards the creation of the remaining part (corners and frame) see "b."

If the smallest plot mask is bigger than half of the via width and/or bigger than 2 x the via radius, the message "Technology violation" appears. Now "N" is entered in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects a respective manual change.

N: (= manual mask selection)

a. If during the automatic mask selection OUTPUT has to use the smallest plot mask and this is larger than half of the via width (or for square via types with rounded corners: larger than 2 x the via radius) the message "Technology violation" appears and "N" is entered in "AUTOMATIC MASK SELECTION". The cursor automatically moves to the respective line and you may specify a mask of your choice. Now OUTPUT accepts any mask specification (even the mask that has not been accepted during the automatic selection), provided that the mask has been defined in COLUMBINE.

b. If you want to change mask specifications or if you want to use a load symbol for the via representation in spite of a successful automatic mask selection, you have to enter "N" in "AUTOMATIC MASK SELECTION".
Subsequently, you may change the mask and load symbol specifications in the respective line. Then OUTPUT uses the specified flash or plot masks (provided that they have been defined in COLUMBINE for the respective device).

c. If you enter the masks and the load symbol manually and then decide to better use the automatic mask selection, just press <ESC><ESC> to effect an automatic mask selection for the respective via type.

MASK A :

1. If the automatic mask selection has been successful, the number of the chosen mask appears here automatically.

2. <u>Non-automatic mask selection</u> **without** <u>load symbol specification</u>

   By entering the respective mask number in "MASK A", you specify:

a. for flash masks:

   - with which mask this via type is to be flashed (requires the entry "Y" in "FLASH").

   Attention:
   If you have square vias with rounded corners and if only the 'inner circle' is to be flashed with this mask, you have to specify a pen in "MASK B" with which the remaining part of the vias is then plotted.

b. for round plot masks:

   - with which round pen the 'inner circle' of square vias

   or

   - with which round pen the 'inner circle' of square vias with rounded corners

   is to be plotted.

Attention:
In both cases, a mask specification has to be made in "MASK B", otherwise the via corners are not plotted and the result is round vias in the plot.

c. For rectangular plot masks:

- with which rectangular pen the whole square via is to be plotted.

Note:
A specification in "MASK B" is not necessary in this case.

3. <u>Non-automatic mask selection</u> **with** <u>load symbol specification</u>

By entering the respective mask number in "MASK A" you specify:

- with which pen the <u>contours</u> of the load symbol are to be plotted.

MASK B :

1. If the automatic mask selection has been successful, the number of the chosen mask appears here automatically.

2. <u>Non-automatic mask selection</u> **without** <u>load symbol specification</u>

By entering the respective mask number in "MASK B" you specify:

- with which (round!) plot mask the remaining parts (corners and contours) of the vias shall be plotted.
Refer also to "MASK A", point a. and b.

3. <u>Non-automatic mask selection</u> **with** <u>load symbol specification</u>

By entering the respective mask number in "MASK B", you specify:

- with which (round!) plot mask the load symbol (the contours of which have been plotted with mask A) shall be filled.

Attention:

If no specification is made in "MASK B", the load symbol is not filled!
The mask has to be defined as a round plot mask in COLUMBINE.

DIAMETER:

The diameters for mask A (and for mask B), defined in COLUMBINE appear here.

LOAD SYMBOL:

For the 'manual' mask specification, you may determine the kind of representation for each via type by specifying a load symbol (defined in COLUMBINE, section Graphics/"Macros").

If you enter "*" instead of a load symbol, all macros defined in COLUMBINE are listed. By pressing <CR>, the name of the last macro is entered into the respective field.

Paging (+/-):

In the CIS/PCB technology, up to 255 via types can be defined. Accordingly, the menu for the mask selection of the vias also contains more via types than can be listed on one page.
For that reason, you have to page in this menu.
In order to page down, enter "+" in the respective input field. You get to the next page and the cursor remains on the input field so that you can enter "+" again for paging down or "-" for paging up.
If the cursor is already located on the input field and "+" or "-" is already entered, paging is also possible by just pressing <CR>. In this case, the cursor moves to the first input field for the mask selection on the respective page and not on the input field for the paging!

Handling instructions

- If the cursor is not on the field >+< or >-< and you press <CR>, the entered data is saved and you leave the menu if there are no technology violations during the automatic mask selection.

  Therefore, you should only use the <CR> key for paging if you want to make entries or check the mask selection for further via types.

- With <ESC><ESC> in this menu, you do not leave the menu but activate another automatic mask selection.

- If you want to edit a defined procedure, for which the automatic mask selection has already been effected, and specify another device name in menu 3, you have to go through all submenus of this procedure (incl. the menus for the mask selection) in order to effect that the masks of the current device are accepted.
  When you get to the menu described here, you have to press <ESC><ESC> to effect that the automatic mask selection is executed with the masks of the new device.

5.2 Segment representation in production plots

If the current .aph-file also contains segments, the last menu to be called is the menu for the mask selection for the representation of segments.

As a principle, the mask selection is effected **automatically**.

If necessary, or where the autoamtic mask selection has been unsuccessful (i.e. where technology violations have been found), the masks for the representation of the different segment types in the current plot may be **changed manually**.

```
*********************************************************************
* PROCEDURE     :  COMP                                     ****** *
* PLOTTER       :  HP                                       * 26 * *
* OUTPUT FILE   :  test_COMP.g01                            ****** *
*********************************************************************

     ******************************************************
     * SEG * WIDTH *  AUTOMAT. * MASK     MASK  *  DIAMETER*
     * TYPE* (mm)  *  MASKSEL. *  A        B    *  (mm)    *
     ******************************************************
     * 0   *       *           *                * A:       *
     *     *       *           *                * B:       *
     *     *       *           *                *          *
     *  1  * 0.000 *           *                * A:0.198  *
     *     * 0.290 *    J      *  1             * B:       *
     *     *       *           *                *          *
     *  2  * 0.000 *           *                * A:0.198  *
     *     * 0.422 *    J      *  1             * B:       *
     *     *       *           *                *          *
     *  3  * 0.000 *           *                * A:0.198  *
     *     * 0.989 *    J      *  1             * B:       *
     ******************************************************
     *                                 * Paging   (+/-) >_<*
     ******************************************************
```

Menu description

SEG.TYPE:

Here, the segment types defined in the layer technology (CIS/PCB technology) of the .adb-file are automatically entered.

WIDTH:

The widths assigned to the segment types in the CIS/PCB technology are automatically entered. The first value stands for the parameter **'radius'**. The value entered below stands for the parameter **'width'**.

If the first value is "0.00", you defined a track type with rectangular ends (for rectangle SMD-pins).

If the first value is exactly half of the second value, you defined a track type with round (semi-circular) ends (standard SMD-pin; radius = 1/2 width).

If the radius value is > 0 and smaller than 1/2 width, you defined a track type with semi-circular (=rounded) ends. (Ref. to CIS/PCB technology).

AUTOM. MASK SELECTION:

Y: (automatic mask selection)

OUTPUT automatically searches for suitable plot masks in the library COLUMBINE and enters "Y" in "AUTOMATIC MASK SELECTION" for all track types with a width > 0, if the automatic mask selection is successful for these types. The search for the plot mask depends on the shape of the track types (defined in the CIS/PCB technology).

1. **Track types with round ends (width = 2 x radius)**

   A round plot mask is searched the width of which is equal to the track width. If this search is not successful, the next smaller mask is used.

   The number of the selected mask is entered in "MASK A", its diameter appears in "A:"; for mask B no entry is made.

   If the smallest plot mask defined for this plotter is bigger than half the track width, the message "Technology violation" appears.
   Now "N" is entered in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects a respective manual change.

2. **Track types with rectangular ends**
(radius = 0)

Two plot masks (MASK A and MASK B) are searched:

a. For plotting the track, a round mask (mask A) is searched the width of which is equal to the track width.
If this search is not successful, the next smaller mask is used.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:".

b. For plotting the rectangular ends, the smallest round plot mask defined in COLUMBINE for this plotter is used.
The number of the selected mask is entered in "MASK B", its diameter appears in "B:".

If the smallest plot mask defined for this plotter is bigger than half of the track width, the message "Technology violation" appears.
Now "N" is shown in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects a respective manual change.

3. **Track types with semi-circular ends** (rounded corners)
(0 < radius < 1/2· width)

Two plot masks (MASK A and MASK B) are searched:

a. For plotting the track, a round mask (mask A) is searched the width of which corresponds to the track width.
If this search is not successful, the next smaller mask is used.
The number of the selected mask is entered in "MASK A", its diameter appears in "A:".

b. For plotting the corners at the segment ends, a plot mask with the diameter
D = 2 x radius of the track
is searched for.
If this search is not successful, the next smaller mask is used.
The number of the selected mask is entered in "MASK B", its diameter appears in "B:".

If the smallest plot mask defined for this plotter is bigger than half of the track width, the message "Technology violation" appears.

Now "N" is entered in "AUTOMATIC MASK SELECTION"; the cursor automatically moves to the field where this entry appears and expects a respective manual change.

N: (= manual mask selection)

a. If during the automatic mask selection OUTPUT has to use the smallest plot mask and this is larger than half of the track width (or for segment types with rounded corners: larger than 2 x the radius) the message "Technology violation" appears and "N" is entered in "AUTOMATIC MASK SELECTION". The cursor automatically moves to the respective line and you may specify a mask of your choice. Now OUTPUT accepts any mask specification (even the mask that has not been accepted during the automatic selection), provided that the mask has been defined as a plot mask in COLUMBINE.

b. If you want to change mask specifications in spite of a successful automatic mask selection, you have to enter "N" in "AUTOMATIC MASK SELECTION". Subsequently, you may change the mask specifications in the respective line. Then OUTPUT uses the specified plot masks (provided that they have been defined in COLUMBINE for the respective device).

c. If you enter the masks manually and then decide to better use the automatic mask selection, just press <ESC><ESC> to effect an automatic mask selection for the respective track type.

MASK A :

For the non-automatic mask selection, you specify with which pen the track segments are to be represented by entering the respective number (independent of the shape of the ends) (ref. to "AUTOMATIC MASK SELECTION", point 1, 2 and 3).

MASK B:

For the non-automatic mask selection, you specify the number of the plot mask with which the ends of the segments are to be plotted for track types with rectangular or 'semi-circular' ends (ref. to "AUTOMATIC MASK SELECTION", point 2 and 3).

DIAMETER:

The diameters for mask A (and for mask B), which are defined in COLUMBINE, appear here.

Paging (+/-):

In the CIS/PCB technology, up to 255 track types can be defined. Accordingly, the menu for the mask selection for the segment representation may also contain more track types than can be listed on one page.
For that reason, you have to page in this menu.
In order to page down, enter "+" in the respective input field. You get to the next page and the cursor remains on the input field so that you can enter "+" again or "-" if you want to page up.
If the cursor is already located on the input field and "+" or "-" is already entered, paging is also possible by just pressing <CR>. In this case, the cursor moves to the first input field for the mask selection on the respective page and not on the input field for the paging!

Handling instructions

- If the cursor is not on the field >+< or >-< and you press <CR>, the entered data is saved and you leave the menu if no technology violations could be found during the automatic mask selection.
  Therefore, you should only use the <CR> key for paging if you want to make entries or check the mask selection for further track types.

- With <ESC><ESC> in this menu, you do <u>not</u> leave the menu but activate another automatic mask selection.

- If you want to edit a defined procedure, for which the automatic mask selection has already been effected, and specify another device name in menu 3, you have to go through all submenus of this procedure (incl. the menus for the mask selection) to effect that the masks of the current device are accepted.
  When you get to the menu described here, you have to press <ESC><ESC> to execute the automatic mask selection with the masks of the new device.

PROCEDURE DEFINITION DRILL

1. Introduction

If you activated the function "Drill" in menu 2, you get to several menus for the detailed determination of the drill data.

In one procedure, only one **drill tape** can be defined. Whether it is the drill tape of the drills going through all layers or the drill tape for the 'buried vias' of a certain layer pair depends on the .aph-file activated before (the .aph standard name for the drills going through all layers is "name_DPHY"; the standard name for the layer pair drills is "name_BUR_x_y", with x and y being the numbers of the respective layer; -> see "introduction", point 6 and chapter "define/change procedures"

The data for the drill generation are stored in an output file, the file name consists of the name of the input-aph-file with the default extension .b01. This file is created during the execution of the procedure (selection number "2" or "3" in the postprocessing main menu) for the appropriate .aph file and is stored on the directory **gdi**.

Note:
If the creation of a drill tape has been defined in the procedure, OUTPUT creates a **drill report** in addition to the respective output file. It is stored in a file with the name of the input file and the extension **.drp** on the sub-directory of the aph-directory which has the same name like the input-adb-file.
With the help of the 'view' function of the postprocessing main menu (selection "8") the report file may be displayed.

The heading of the menus belonging to the "Drill" function contains information on the following parameters:

PROCEDURE: This field contains the name of the current procedure.

DEVICE: This field contains the name of the selected drilling machine.

OUTPUT FILE: This field contains the name of the current output file for the drill tape.

The specifications to the single parameters first appear after the menu in which they have been defined.

2. Drill tape definition

After the activation of the function "Drill" in menu 2 for the output selection, menu 66 is called.
Here, you enter the name of the drilling machine to be used, the name of the output file and - if desired - a text for the drill tape header. The specified drilling machine must be defined in COLUMBINE, section "Device". "DRILL" is entered into the menu header.

Notes:

- If you enter "*" in "Name of the drilling machine to be used" and press <CR>, the names of the drilling machines defined in COLUMBINE will be listed. When pressing <CR> again, you return to menu 66; the first device name automatically appears in the menu.

- The extension of the output file can be changed, if necessary; e.g. if an output file with the same name is already existing and shall not be overwritten.

2.1 Menu 66

```
*****************************************************************
* PROCEDURE   : TEST                                    ***** *
* DEVICE      :                                         *66 * *
* OUTPUT FILE : test_DPHY.b01                           ***** *
*****************************************************************

*****************************************************************
*  Name of the drilling machine to be used: SM25                *
*                                                               *
*  Output file for drill tapes            : test_DPHY    .b01   *
*                                                               *
*  User text for drill tape header        :                     *
*                                                               *
*****************************************************************
```

After entering the specifications with <CR>, an additional menu part appears where you specify whether only the through-connected or only the not through-connected or whether both kinds of vias are to be considered:

```
*****************************************************************
*   through connected : X      not through connected   : X     *
*****************************************************************
```

3. Specifications to offset, FIX-file, etc.

After finishing menu no. 66, menu no. 69 is called.

3.1 Menu 69

```
*******************************************************************
* PROCEDURE   : TEST                                        ***** *
* DEVICE      : SM25                                        *69 * *
* OUTPUT FILE : test_DPHY.b01                               ***** *
*******************************************************************

    ***********************************************************
    *                                                         *
    * Standard values for drill tools Y/N : N                 *
    *                                                         *
    * Offset for machine zero point in x   : 0.000___(mm)     *
    *                                                         *
    * Offset for machine zero point in y   : 0.000___(mm)     *
    *                                                         *
    * If tooling holes name of fixed file  : __________.fix   *
    *                                                         *
    * Drill only fixed file Y/N            : N                *
    *                                                         *
    * Mirror   Y/N : N                                        *
    ***********************************************************
```

3.2 Handling instructions

By entering the respective coordinates, you may determine the **offset** to the machine zero point.
Coordinate origin for the drilling is the production origin (defined in CIS/PCB/Sizes).

Tooling holes are holes which may be used for several PC-boards. To avoid a new definition of these fixed 'standard drills' for each board-file, the respective information can be stored in a special file (ASCII-format) with the extension .FIX. It has to be stored on the directory lib/sys.

Note:

> Documentation about the creation of .FIX-files will be delivered as soon as possible.

If tooling holes are to be generated, you have to enter the name of the .FIX-file in the above menu.

If only tooling holes shall be generated, the default entry -N- in "Drill only fixed file" has to be replaced by -Y-.

The drill coordinates are **mirrored** when -Y- is entered in "Mirror Y/N".
Mirror axis is the vertical to the x-axis that goes through the x-center of the board.

OUTPUT automatically enters the **drill tools** to be used (defined by their number), their **tool life** and the "x" selections for the **execution.**
To each drill type, a tool number is assigned that is identical to the number of the drill type. The default value for the tool life is "32767" (see menu on next page).

When you answer the inquiry "Standard values for drill tools Y/N" with -Y-, you get directly to menu 80 after leaving menu 69.
If you confirm the default entry -N-, menu 70 is called where you can see and change the default values for the drill tools.

4. Changing the standard values for drill tools

If "N" has been entered in "standard values for drill tools Y/N" and if the specifications have been stored with <CR>, the following menu appears for changing the default values:

4.1 Menu 70

```
*******************************************************************
* PROCEDURE   : TEST                                       ***** *
* DEVICE      : SM25                                       *70 * *
* OUTPUT FILE : test_DPHY.b01                              ***** *
*******************************************************************

 **************************************************************
 * DRILL *  DIAMETER *NUMBER*    EXECUTE   *DRILL TOOL*TOOL LIFE*
 * TYPE  *    (mm)   *      *              *          *         *
 **************************************************************
 *   0   *           *      *              *          *         *
 *   1   *  0.599    *  24  *      X       *   1___   *  32767  *
 *   2   *  0.798    *  11  *      _       *   2___   *  32767  *
 *   3   *           *      *              *          *         *
 *   4   *           *      *              *          *         *
 *   5   *           *      *              *          *         *
 *   6   *           *      *              *          *         *
 *   7   *           *      *              *          *         *
 **************************************************************
 *                                         *Paging    (+/-) >+< *
 **************************************************************
```

4.2 Menu description

Since up to 255 drill types may be defined in an .adb-file, 'paging' may be necessary in this menu. This is done by entering '+' (page down) or '-' (page up) in 'paging' or by confirming the default entry with <CR>.

Attention:

If the cursor is not on the input field for the paging, the entries are stored with <CR>.

DRILL TYPE:

This field contains the numbers of the drill types defined in CIS.

DIAMETER:

This field contains the drill diameters which have been defined in the CIS technology and which are used for the vias of the activated .aph-file.

NUMBER:

This field contains the number of drills needed of this drill type.

EXECUTE:

By marking this column with a cross, you specify that the drills of this drill type are to be executed. If an existing cross is deleted, the respective drill type is not drilled.

DRILL TOOL NUMBER:

Here, you may replace the drill tool number entered automatically by a number of your choice. This drill type will then be executed with the specified drill tool.

TOOL LIFE:

Here, you may replace the tool life entered automatically by a tool life of your choice. The figures specify the number of allowed drills; then the tool is considered as being used up and has to be replaced.

PROCEDURE DEFINITION NC-ROUTING

1. Introduction

When you have activated the function "NC-routing" in menu 2, you get to several menus for the detailed specification of the NC-routing data.

In order to define a procedure for the generation of an NC-routing tape, the current .adb-file has to contain NC-routing information; the NC-routing information has to be stored in CIS/PCB on layer 1 (for more details, see point 5). APHCOM then generates the .aph-file "name_MILL.aph"; this file has to be activated prior to the definition of the NC-routing procedure (-> see "Introduction", point 6 and chapter "define/change procedures").

The data for the generation of the NC-routing tape is generated for the appropriate .aph-file during the execution of the procedure (selection number "2" or "3" in the postprocessing main menu) and is stored on the directory **gdi** in an output file which has the name of the input-aph-file with the default extension .f01.

Note:

> If the creation of an NC-routing tape has been defined in a procedure, OUTPUT creates an **NC-routing report** in addition to the respective output file. It is stored in a file with the name of the input file and the extension **.mrp** on the subdirectory of the aph-directory which has the name of the input-adb-file.
> With the help of the 'view' function selectable in the postprocessing main menu (selection "8") the report file may be displayed.

The heading of each menu belonging to the "NC-routing" function contains information on the following parameters:

PROCEDURE: This field contains the name of the current procedure.

DEVICE: This field contains the name of the selected NC-routing machine.

OUTPUT FILE: This field contains the name of the output file for the current tape.

The specifications to the single parameters first appear after the menu in which they have been defined.

2. Tape definition

After activating the function "NC-routing" in menu 2 for the output selection, menu 72 is called.
Here, you enter the name of the NC-routing machine to be used, the name of the output file and - if desired - a text for the tape header. The specified NC-routing machine has to be defined in COLUMBINE, section "Device". In the upper part of the menu, "NC-ROUTING" is entered.

Notes:

- If you enter "*" in "Name of the NC-routing machine to be used" and press <CR>, the names of the NC-routing machines defined in COLUMBINE are listed. When pressing <CR> again, you return to menu 72; the first device name automatically appears in the menu.

- The extension of the output file can be changed, if necessary; e.g. if an output file with the same name is already existing.

2.1 Menu 72

```
*****************************************************************
* PROCEDURE   : TEST                                    ***** *
* DEVICE      :                                         *72 * *
* OUTPUT FILE : test_MILL.f01                           ***** *
*****************************************************************

*****************************************************************
*  Name of the NC-routing machine to be used : SM25             *
*                                                               *
*  Output file for NC-routing tapes          : test_MILL.f01    *
*                                                               *
*  User text for tape header                 :                  *
*                                                               *
*****************************************************************
```

3. Specifications to offset and FIX-file

When menu 72 has been finished and you have saved your entries, menu 73 is called.

3.1 Menu 73

```
**************************************************************************
* PROCEDURE   : TEST                                              ***** *
* DEVICE      : SM25                                              *73 * *
* OUTPUT FILE : test_MILL.f01                                     ***** *
**************************************************************************

    ************************************************************
    * Offset for machine zero point in X   : 0.000___(mm)     *
    *                                                          *
    * Offset for machine zero point in Y   : 0.000___(mm)     *
    *                                                          *
    * If auxiliary NC-routing,                                 *
    * name of fixed file                   : __________.fix   *
    *                                                          *
    * Mirror   Y/N : N                                         *
    ************************************************************
```

3.2 Handling instructions

By entering the respective coordinates, you can determine the **offset** to the machine zero point.
coordinate origin for the NC-routing is the production origin (defined in CIS/PCB/Sizes).

Auxiliary NC-routing are NC-routings that may be used for several PC-boards. To avoid a new definition of these fixed 'standard NC-routings' for each board-file, the respective information can be stored in a special file (ASCII-format) with the extension .FIX. It has to be stored on the directory lib/sys.

Note:

> Documentation about the creation of .FIX-files will be delivered as soon as possible.

If auxiliary NC-routings are to be generated, you have to enter a name for the .FIX-file in the above menu.

The NC-routing coordinates are **mirrored** when entering -Y-in "Mirror Y/N".
Mirror axis is the vertical to the x-axis that goes through the x-center of the board.

4. Specifications for NC-routing tools

After storing the entries with <CR>, menu 74 appears for the specifications to the NC-routing tools.

4.1 Menu 74

```
*****************************************************************
* PROCEDURE   : TEST                                      ***** *
* DEVOCE      : SM25                                      *74 * *
* OUTPUT FILE : test_MILL.f01                              ******
*****************************************************************

*****************************************************************
*  DRILL *  DIAMETER   *    EXECUTE    * TOOL NUMBER* TOOL LIFE *
*  TYPE  *    (mm)     *               *            *           *
*****************************************************************
*   0    *             *               *            *           *
*   1    *  1.499      *      X        *  1___      *  32767    *
*   2    *  0.00       *      X        *  2___      *  32767    *
*   3    *             *               *            *           *
*   4    *             *               *            *           *
*   5    *             *               *            *           *
*   6    *             *               *            *           *
*   7    *             *               *            *           *
*****************************************************************
*                                      * Paging   (+/-) >+<     *
*****************************************************************
```

In CIS up to 32 NC-routing diameters can be defined for one .adb-file - on the basis of the drill type diameters 224 - 255 definable in the drill technology; these diameters are assigned to the NC-routing reference terminal graphics NC-GRAPHIC0 - NC-GRAPHIC31 which are available for the generation of the NC-routing terminals (for more details see point 5 or the CIS/PCB description, chap. "Terminals /create").

Contour NC-routing has to be assigned to the NC-routing diameter 0.00 so that OUTPUT 'recognizes' it as contour NC-routing.

OUTPUT automatically assigns NC-routing types to the NC-routing diameters defined in the input file. If for the definition of two NC-routing segments in CIS two different reference terminal graphics have been used for the representation of the two starting points, but if the drill types assigned to these graphics have been defined with the same diameter, then OUTPUT considers the two segments belonging to the same NC-routing type. For contour NC-routings the value 0.00 appears in the menu for the diameter of the appropriate NC-routing type (here: NC-routing type 2).

Since up to 32 NC-routing types may be defined, paging may be necessary. This is done by entering '+' (page down) or '-' (page up) in "Paging". If the cursor is already located on the input field, paging is also possible by just pressing <CR>.
Attention:
If the cursor is not on the input field for the paging, a <CR> stores the entries.

OUTPUT automatically enters the **NC-routing tools** to be used (defined by their number), their **tool life** and the "x" selections for the **execution.**
To each NC-routing type, a tool number is assigned that is identical to the number of the NC-routing type. The default value of the tool life is "32767".

NC-ROUTING TYPE:

contains the numbers of the NC-routing types automatically assigned by OUTPUT

DIAMETER:

contains the diameter of the NC-routing defined in CIS (corresponds to the diameter of the drill type that has been assigned to the NC-routing reference terminal graphic with which the NC-routing terminal has been generated).

EXECUTE:

By marking this column with a cross, you specify that the NC-routing of this type is to be executed. If an existing cross is deleted, it will not be executed.

NC-ROUTING TOOL NUMBER:

Here, you may replace the automatic entry by a number of your choice (according to the NC-routing point diameter). This NC-routing type will then be executed with the specified tool.

TOOL LIFE:

Here, you may change the automatically entered tool life, if necessary. The number specifies the maximum tool life in mm.

After you have stored your entries with <CR>, you get to the menu no. 80 for storing the procedure.
If you leave the menu with <ESC><ESC>, the entries will not be stored and you get back to menu no. 0.

5. Generation of the NC-routing information in CIS/PCB

(The following descripton is identical to the chapter "Generation of the NC-routing terminals and NC-routing contours" in the chapter "Terminals/create" of the CIS/PCB description.)

The information needed for the NC-routing of the PCB is created by generating so-called "NC-routing terminals" and by digitizing the appropriate NC-routing contours to these terminals.
The NC-routing terminals represent the starting points for the NC-routing tools.
The NC-routing contour starting from the NC-routing terminal represents the NC-routing line with the appropriate tool.

For the generation of the NC-routing terminals 32 predefined reference terminal graphics are available in the graphic pool. To each of these graphics one drill type between no. 255 and 224 from the drill technology has been assigned:

graphic name	drill type
NC-GRAPHIC0	255
NC-GRAPHIC1	254
NC-GRAPHIC2	253
NC-GRAPHIC3	252
.	.
.	.
.	.
NC-GRAPHIC29	226
NC-GRAPHIC30	225
NC-GRAPHIC31	224

The diameter which is assigned to the appropriate drill type in the drill technology represents the diameter of the NC-routing tool which is to be used for the NC-routing contour, i.e. the NC-routing width.
In order to define an NC-routing contour with a certain width, this diameter has to be assigned to one of the drill types between 255 - 224; the reference terminal graphic assigned to this drill type has then to be used for the generation of the appropriate terminal.

How to procede

In order to define an NC-routing contour with a starting point and with a certain width, procede as follows:

1. Assign a diameter to the drill type for determining the NC-routing width:
   * call the function "drills" in the technology menu
   * call a drill type (224-255) and assign the value for the width in "diameter" (e.g. drill type 254 with diameter 1.00mm);

* store the definition with the (pop-up) function "save";

2. Generate the starting point

   * generate a special component with an appropriate name (for easy identification) in the "modification" menu
   * generate a terminal ("NC-routing terminal") to that component, by making the following entries in the status window:
     - an appropriate terminal name for marking the special function (e.g. NCROUT1)
     - "F" in "t-typ"
     - in "graph:" the name of the reference terminal graphic which has been assigned to the drill type defined under (1) (e.g. NC-GRAPHIC1);
     - place the NC-routing terminal at the desired position;

   Note:
   After the generation of the necessary NC-routing terminals you may delete the component frame and the text field with the component name with the help of the function "Component/mod. graphic".

3. Digitize the NC-routing contour

   * activate the function "mod. graphic" and the subfunction "line"
   * digitize the NC-routing contour

Notes:

- An NC-routing contour must always consist of connected track segments!

- The NC-routing terminal must always be placed at the beginning of a track segment. It must neither be placed on a segment nor at a T-shaped intersection of two segments.

- For each NC-routing contour, a separate terminal must be defined!

- For generating several NC-routing contours with different widths, you have to assign the reference terminal graphic to each of the terminals; the drill type assigned to the terminal graphic has to have the diameter of the desired NC-routing width.

- For creating perforation NC-routings, a separate terminal must be defined for each NC-routing segment (specify the same reference terminal graphic for all terminals).

- To ensure that NC-routing of board contours is recognized by OUTPUT, the diameter of the drill type assigned via the reference terminal graphic (see point 1) has to be defined with the value 0.00.

Only if that precondition is fulfilled, OUTPUT definitely executes this NC-routing as the last NC-routing procedure. In the 'starup'technology file the drill type 255 is defined with diameter 0.00 - this drill type is assigned to the reference terminal graphic NC-GRAPHIC0.

- In an .adb file a maximum of 32 different NC-routing diameters may be used, since the number of the drill types which may be used is limited to 32.

- During the digitizing, the NC-routing contours are automatically placed on the solder side (=layer 1). The default for the digitizing is track type 1.

- If an NC-routing contour in the signal layer plot on the solder side is to be represented in its real width, you have to enter a track type with the respective width in "Trk-type" in the status window in CIS.

- In order to avoid that the tracks cannot be routed or digitized in NC-routing areas, the NC-routing contours must be within restricted areas - on all layers! The area <u>outside</u> the board outline (created with the function 'Sizes'of the technology menu) is a restricted area on all layers. (If necessary, you can activate this restricted area and change it with the function 'mod. graphics'.) If you create additional NC-routing contours <u>within</u> the board area, they must be surrounded with 'normal' restricted areas <u>on all layers</u> (to be generated with the function 'restricted areas' of the modification menu).

NC-routing direction:

The NC-routing direction depends on the course of the track segments from the starting point (= NC-routing terminal). At a joint with more than two segments, the direction is determined as follows:

1. In the X direction if an X segment exists (regardless of whether Y or non-orthogonal segments exist).

2. In the Y direction if a Y segment but no X segment exists.

3. In the direction of the non-orthogonal segment if a non-orthogonal segment but no X or Y segment exists.

The preferred direction is always the X direction.

The knowledge of the NC-routing direction rules is of importance where the determination of the right NC-routing direction is a precondition for a correct use of the self-compensation.

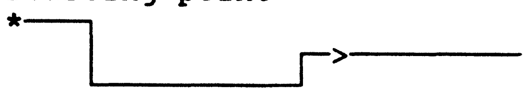
Example: inner NC-routing

End of a NC-routing process

The NC-routing process is finished if

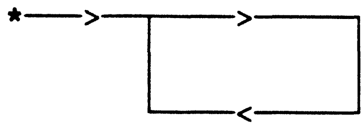
1. a track ends (or is interrupted)

starting point

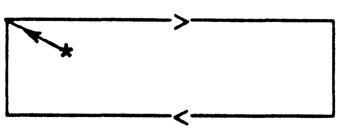


end

2. a track ends in a T-connection



3. an already processed point is touched again (inner NC-routing)



Sequence of the NC-routing operations:

The sequence depends on whether or not the 'contour mark' exists.

1. The NC-routing starts with the tracks the NC-routing terminals of which were created with reference terminal graphics, the assigned drill types of which are defined with a diameter unequal 0 (i.e. have no 'contour mark')

2. Next, the tracks are considered that have a contour mark (i.e. that were assigned to a drill type with diameter=0).

C A L P R I

1. Introduction

By selecting function "7" (= 'Run list-generator') in the postprocessing main menu and by entering the name and the extension of the .adb-file to be processed, the program CALPRI is started.

With this program, lists and statistics can be created from the data of the specified file.

Note:

> If the .adb-file has not yet been divided into .aph-files by the program APHCOM, this is done after calling function "7". The reason is that CALPRI needs the file "name_NET.aph" because it contains the information necessary for the lists.

After the specification of the file name to be processed and after the transformation with APHCOM (if necessary) the menu for the selection and output modes of the lists to be generated appears.

2. List selection

```
*****************************************************************
*          Lists from the .APH-file      Version                *
*****************************************************************

*****************************************************************
* Name of the .aph-file      : __________                       *
*                                                               *
*   output to terminal (0) or file (1)                  1       *
*      file on device APH with extension .lis                   *
*                                                               *
*   netlist                      _   list type (0,1)     _      *
*   component list               _   list type (0,1)     _      *
*      only solderside (0), only comp.side (1), both (2)  2     *
*                                                               *
*   non connected terminals      _     technology        _      *
*   cross-Reference-list         _     statistics        _      *
*   collision of component       _                              *
*                                                               *
*   change printer controlling   _                              *
*                                                               *
*   <ESC><ESC> terminates program                               *
*                                                               *
*****************************************************************
```

Handling instructions

- "Name of the .GPH-file" contains the name of the specified file.
  If you enter another file name (without extension!) in this field, you may generate lists from the newly specified file (provided that the file name_NET.aph exists on the appropriate subdirectory of the **aph** directory).

- The **list selection** is done by marking the respective field with a cross.
  You may select as many lists as required.

- For the netlist and the component list, also the **list type** has to be specified. This specification determines the form and the complexity of the information output from these lists (see "Output form"). In one program run, only one type of each list can be created!
  For the component list, you have to specify which side(s) are to be considered (default entry = "2", i.e. solder and component side).

- If the list shall be output on the **terminal**, change the default number to "0".

- Exit the program with <ESC><ESC>.

- The files generated by CALPRI are written onto the **aph** directory, i.e. onto the directory with the name of the .adb-file. The list file has the name of the adb or aph-file with the extension **.lis**.

3. List output in a file

If the lists are output in a file, this file can be printed or displayed with the "more" command or can be edited (e.g. with the vi-editor).

Change printer controlling

The default for the file output generated by CALPRI is a page format of 65 lines per page and 80 characters per line.
But list type 1 needs more lines than the default line length, and your printer may require another setting of the line number per page.

With the help of the function "Change printer controlling", you may enter commands for switching the printer to another page and line length.
When selecting this function, the following additional menu appears (see next page):

```
**************************************************************************
*   lines per page                                          65          *
*                                                                        *
*   sequence for printer at 132 characters                  ESCQ___     *
*   sequence for printer at  80 characters                  ESCN___     *
**************************************************************************
```

- In the input field of the first line, you may specify how many **lines** the pages of the lists shall contain when printing the file.
  The number of lines depends on the printer you use.
  The default is the number of lines optimized for the line printer BINDER 8510 B. This is the information for the printer to continue with the headline on the following page after 65 lines.

- The maximum line length of the 'normal' operation mode of the printer varies from type to type (e.g. for BINDER 8510 B line printer it is 80 characters).
  To get a correct output of list type 1 on these devices, it is often necessary to compress the characters.

  This is achieved by switching the printer with a certain command at the beginning of the file.
  The default of the input field of the second menu line is the command sequence for automatically setting printer BINDER 8510 B to 132 characters per line - for printing list type 1.

  The default of the input field of the last menu line is the command sequence for the automatic set back of printer BINDER 8510 B from 132 to 80 characters - after printing list type 1 -.

If you use another printer, the default command sequences have to be replaced by those convenient for your printer. For the respective command, please refer to your printer manual.

The command sequences for switching the printer become active only when activating list type 1 and when outputting it in a .lis-file.

4. List output on the terminal

The output on the printer (.lis-file) should be preferred to the direct output on the terminal!

Lists of type 1 cannot be represented correctly in the shelltool standard window because their lines are too long. This applies also to the editing of the CALPRI output files (.lis-files) with the vi-editor.

5. Output form of the lists

5.1 Netlist

List type 0

```
***********************************************************************
netlist alphab. sorted             ** test **  01-SEP-89 page : 1
***********************************************************************

NETNAME                  COMMENT                            TT
     COMPNAME                 TERMINALNAME

GND
     $0                       14                             1
     C1                       1                              1
     IC3                      5                              1
     ST1                      11                             2

SIG 1
     C4                       2                              3
     IC2                      5                              1
     IC4                      7                              2
     IC9                      3                              3

   etc.
```

Notes:

a. The headline contains the name of the .adb-file from which the list has been generated, the date of generation and the page number.

b. Every line of this list contains the following information from the netlist:

- names of nets (NETNAME),
- connection points in net, specified by the component name (COMPNAME) and the terminal name (TERM),
- track type for connecting this terminal to the net

List type 1

```
*************************************************************************
netlist  alphab. sorted              ** test **  01-SEP-89  page : 1
*************************************************************************

NETNAME         COMMENT
   COMPNAME        TERMINALNAME          TT   SUBSIGNAL     K   CAS
SIG1
   C1              1                     1                      NNU
   IC2             5                     1                      NNU
SIG 2
   C4              2                     3    0-1               NYA
   IC5             7                     3    2-3               NYA
   IC9             3                     3    1-3               NYA
SIG 3
   C1              2                     2                  D   YYH
   IC1             12                    2                  C   YYH
   IC4             6                     2                  D   YYH

   etc.
```

Note:

Apart from the information in list type 0, list type 1 contains the following signal parameter information from the netlist:

- if the connection point belongs to one or more subsignals, it contains the respective subsignal number(s) (SUBSIGNAL),
- if the connection point belongs to a plane or to a chain (daisy chain), it contains an according mark in K (P for plane; D for daisy chain, C for connection point).
- additional signal parameters (CAS); CAS has the following meaning:
  - -> C = critical signal
  - -> A = autoroutable
  - -> S = signal type (analogue, digital, hybrid or not defined)

  Under these parameters the following abbreviations may appear:
  C -> Y/N for Yes/No
  A -> Y/N for Yes/No
  S -> A (=analogue), D (=digital), H (=hybrid),
  U (=undefined, i.e. without specification)

5.2 Component list

List type 0

```
**********************************************************************
component list alphab. sorted    ** test **  01-SEP-89    page : 2
**********************************************************************

   FLAGS:  O=N SOCKET, F=IXED, A=SSEMBLY, V=IABLOCK, D=UMMY COMP., P=LOTNAME
           S=OLDERSIDE, L=IBRARY IDENT., I=N-CIRCUIT-TEST

COMPNAME             SHAPETYPE                  ROT        FLAGS

C1                   DCRU2A                     90.0       FA
IC1                  DIP14F                     270.0      OAV
R1                   WID2L                      180.0      A
SM1                  DCC16                      0.00       S

etc.
```

Notes:

a. The headline contains the name of the .adb or .aph-file, from which the list has been generated, the date of generation and the page number.

b. This list contains the following component information:

- names of components (COMPNAME),
- shape type (SHAPETYPE),
- component rotation for placement (ROT),
- additional component parameters (see FLAGS); the abbreviations have the following meaning:
  - -> O = on socket
  - -> F = fixed
  - -> A = assembly
  - -> V = via blocking
  - -> D = dummy component
  - -> P = output of component name when plotting
  - -> S = placed on solder side
  - -> L = identical to library COLUMBINE
  - -> I = in-circuit test

Attention: The flags D, I and L are not correctly evaluated at present!

List type 1

```
****************************************************************************************************
component list alphab. sorted          ** test **          01-SEP-89          page : 2
****************************************************************************************************

FLAGS:  O=N SOCKET, F=IXED, A=SSEMBLY, V=IABLOCK, D=UMMY COMP., P=LOTNAME
        S=OLDERSIDE, L=IBRARY IDENT., I=N-CIRCUIT-TEST

COMPNAME                 SHAPETYPE                     ROT      FLAGS          COMPONENTHEIGHT (MM)

        COMPONENTTYPE             STOCK IDENTIFIER              COMMENT

C1      KO               DCRU2A     K/00037            90.0     F              0.00
IC1     74LS06           DIPP14F    LNR74LS00          270.0    OV             0.00
R1      WID              WID2L      W/0003             180.0                   0.00
SM1     SMD/7400         DCC16      XX-8769            0.00     S              0.00

etc.
```

Note:

Apart from the information in list type 0, list type 1 contains the following information:
- function type specification (COMPONENTTYPE).
- the stock ID number (STOCK IDENTIFIER)
- comments assigned to the components (COMMENT)
- the component height in mm (COMPONENTHEIGHT)

5.3 Non connected terminals

This list contains the terminals of the components from the .adb or .aph-file which are not contained in the netlist (=not connected).
The component names are output in an alpha-numeric order.
The column COMPONENTTYPE contains the function type specification.

```
******************************************************************
list of non connected terminals      **test** 01-SEP  page : 4
******************************************************************

COMPNAME        TERMINALNAME      SHAPETYPE    COMPONENTTYPE

IC3             3                 DIP14        74LS00
IC3             5                 DIP14        74LS00
IC3             7                 DIP14        74LS00
IC3             9                 DIP14        74LS00
D12             1                 D035         1N4148

etc.
```

5.4 Cross-reference-list (= component connections)

This list contains all components (incl. shape, function type specification and stock ID), their terminals and the signals to which the terminals are connected.
The list is output in an alpha-numeric order.

```
************************************************************
cross-reference-list    ** test1 **  01-SEP-89   page : 5
************************************************************

OUTPUTFORMAT:
COMPONENTNAME   COMPONENTTYPE
SHAPETYPE       STOCK ID
TRMINALNAME(x)  SIGNALNAME(x)

IC1            74LS00
DIP6           ABC123
1              SIG1
2              SIG12
3              NC.
4              GND
5              SIG17
6              NC.
  etc.
```

(NC. indicates that the terminals are not connected.)

5.5 Component collisions

This list contains the names of the components that collide with each other.

```
************************************************************
collision of component ** test1 **  01-SEP-89   page : 6
************************************************************

R2               <-->  C8
R2               <-->   C9
SM1              <-->  R11

etc.
```

5.6 Technology

When selecting this function, you get a list with the following information on the CIS/board technology of the current file:
- placement & routing
- Sizes
- Layer plan

The list has the following output format (the data contained only serve as examples):

```
************************************************************
technology header         ** test **  01-SEP-89  page : 8
************************************************************

ROUTING
-------

grid base        1 / 960"

routing grid     x:  0.64 mm         y:  0.64
via grid         x:  0.64 mm         y:  0.64

clearance        x:  0.29 mm         y:  0.29


SIZES
-----------

sheet size         x: 210.82         y: 150.799

PC-board   start   x:  25.40         y:  25.40
            size   x: 185.40         y: 124.46

rout. area start   x:  25.40         y:  25.40
            size   x: 185.40         y: 124.46

prod.-origin       x:  25.40         y:  25.40
```

5.7 Statistics

This function generates a list that gives a statistical overview of the .adb-file.
The list has the following output form (the values entered only serve as examples):

```
*************************************************************
statistics                    ** test **  01-SEP-89  page : 9
*************************************************************


NUMBER OF DRILLS
----------------
                     type   number      diameter
====================================================
through connected     1       27        0.97
through connected     2        3        1.40
through connected    14        9        1.40
-----------------------------------------------------
total                         39

NUMBER OF PADS
--------------
type    number    diameter
===============================
 1        5       0.97
17       27       1.40
18        3       1.40
-------------------------------
total    35

NUMBER OF SMD-PADS: 20
----------------------

NUMBER OF VIAS
----------------
type    number    diameter
===============================
1       12        1.40
-------------------------------
total   12
```

NUMBER OF COMPONENTS AND SHAPE TYPES

layer	total number	insertable
solder side:	1	1
comp.side :	6	6
total	7	

number	shape type	stock ID
28		
1	POL	
3	B41588	B41588-C4107-T
3	DIP14	12345
4	DO35	062702-A273-F100

etc.

NUMBER OF SIGNALS

number	comment
6	signals with 25 terminals
30	terminals not in the netlist

NUMBER OF SPOTS

		layer	number
test points	:		0
copper points	:		0
glue points	:	0	1
NC-routing points	:	0	1

APPENDIX

POSTPROCESSING

M I R R O R

What happens if you mirror?

BOARD:

All elements of $$BOARD are mirrored.
Text fields are mirrored but their contents remains **readable.**

'$$BOARD' is the area defined in "PC-board" in the menu 'Sizes'.

BOARD, DIN:

Like BOARD, but all text fields are rotated so that they can be read from below or from the right side.

ALL:

All elements of $$PCB and $$BOARD are mirrored.
The contents of text fields is also mirrored.

'$$PCB' is the area defined in "Sheet size" in the menu 'Sizes'.

ALL, DIN:

Like ALL, but the contents of the text fields is rotated so that it can be read from below or from the left side.

DIN:

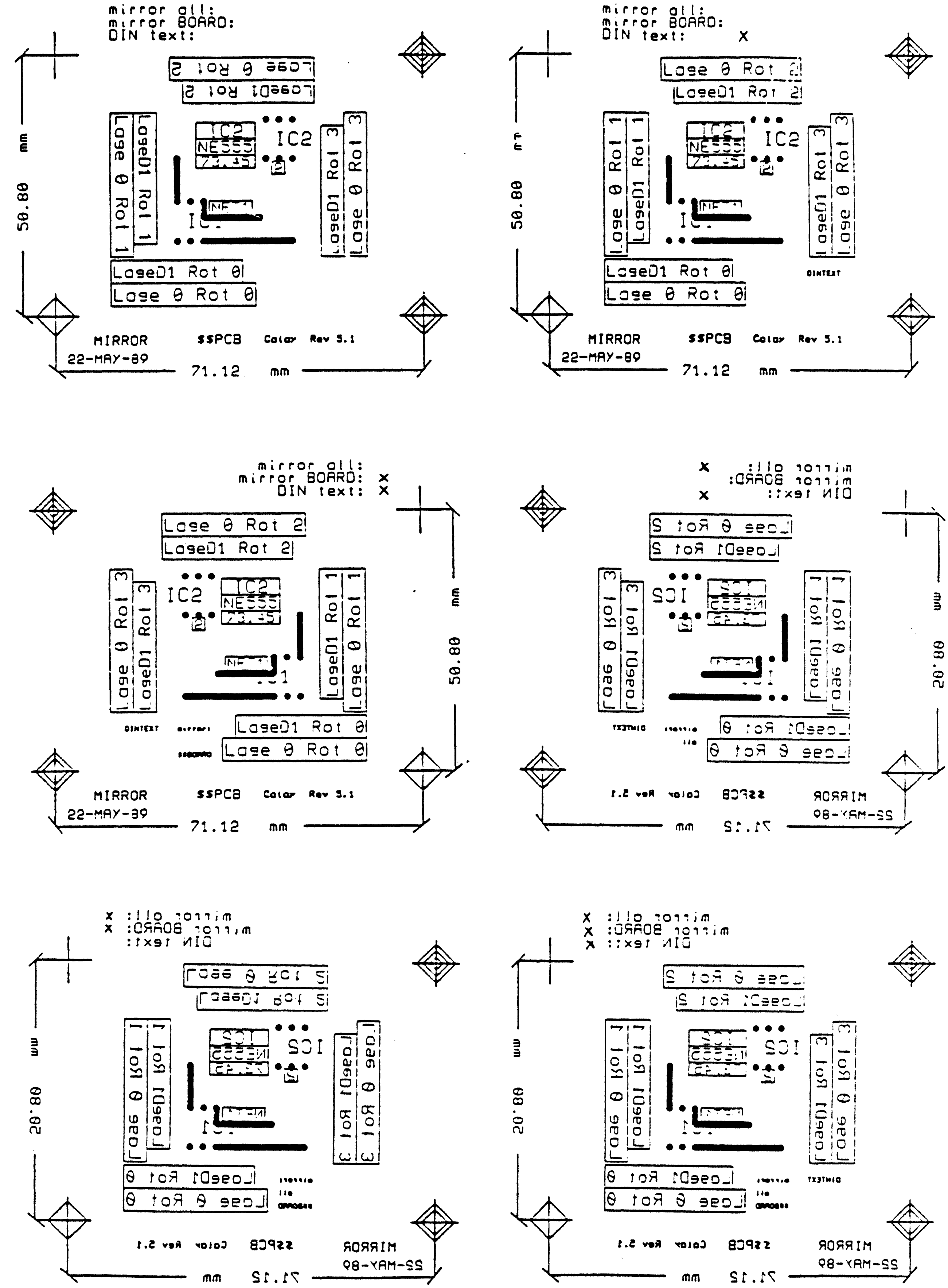
The contents of the text fields is rotated so that it can be read from below or from the right side.

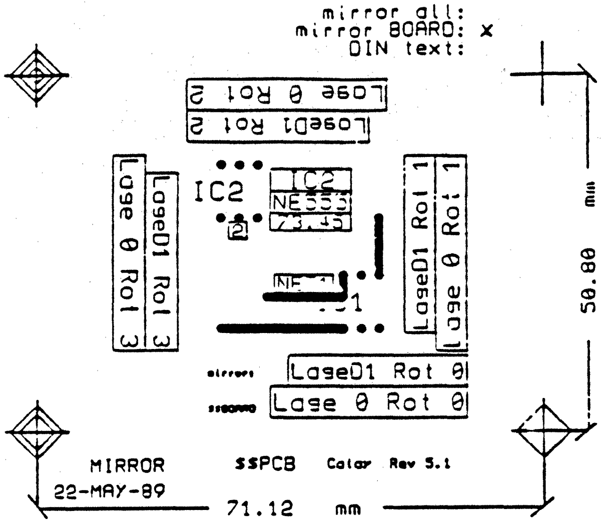
ALL, BOARD:

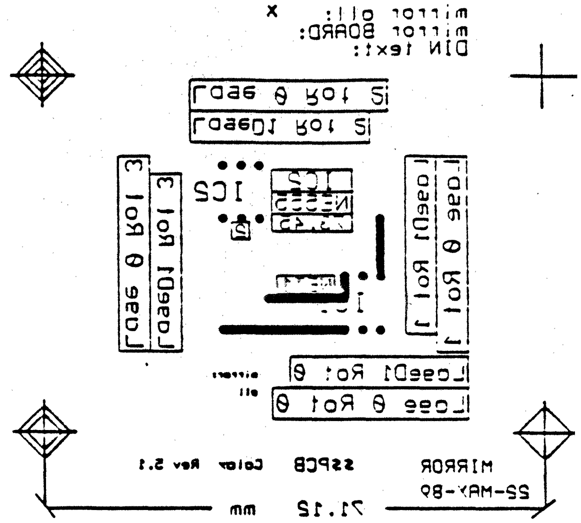
Only text fields or their contents is mirrored, the rest remains unchanged.
Text fields to $$PCB are mirrored <u>with contents</u>.
<u>Only the contents</u> of text fields to $$BOARD is mirrored.

ALL, BOARD, DIN:

Like ALL, BOARD, but the contents of the text fields is rotated so that it can be read from below or from the left side.


mirror all:
mirror BOARD:
DIN text:
IC2
NE555
LaseD1 Rot 0
Lase 0 Rot 0
MIRROR
22-MAY-89
SSPCB
Color
Rev 5.1
71.12 mm
50.80 mm
DINTEXT


mirror all:
mirror BOARD: x
DIN text:
IC2
LaseD1 Rot 1
LaseD1 Rot 3
LaseD1 Rot 0
Lase 0 Rot 0
50.80 mm
MIRROR
22-MAY-89
SSPCB
Color Rev 5.1
71.12 mm



CDBL - Customer Distributed Bugs List

April 1990
Issue 1990-06

Last Minute-Supplement

Central Support Services

CALRAY

CISPCB

CISPCB Components

REFERENCE NUMBER.: 21922
RELEASE..........: 3.11

SYNOPSIS
copy component

DESCRIPTION
When components are copied, existing signal connections are also copied, although if the comp. submenu the function "connected S+K+L" is inactive (is 'off').

REFERENCE NUMBER.: 21924
RELEASE..........: 3.11

SYNOPSIS
vias are placed on top of 'glue-pads'

DESCRIPTION
the router should not place vias on glue-pads; - this is not checked in the 'final check' - restricted areas around glue-pads are missing

---

CISPCB Miscellaneous

REFERENCE NUMBER.: 21921
RELEASE..........: 3.11

SYNOPSIS
measurement function : graphic full view error

DESCRIPTION
In the measurement function if "graphic set-up; full view;" is performed all graphics disappears.

CISPCB Text

REFERENCE NUMBER.: 21932
RELEASE..........: 3.11

SYNOPSIS
wrong coordinate

DESCRIPTION
On the drill layer when changing text in the drill legend box, you pick up a piece of text and change it, then hit return to put it down---- the text will be placed about 3inches to the right , which means outside of your working sheet.

WORKAROUND
place it with the cursor

CISSCS

CISSCS Hierarchy

REFERENCE NUMBER.: 21948
RELEASE..........: 3.11

SYNOPSIS
Move hierarchy to another sheet

DESCRIPTION
If a main group is moved to another sheet a core dump is the result

---

CISSCS Signals

REFERENCE NUMBER.: 21827
RELEASE..........:

SYNOPSIS
delete bus

DESCRIPTION
If a bus has been deleted and a new one is to be generated with the same name, CIS prompts with: "name already exists or wrong". This name can neither be activated, nor deleted, nor listed.

CALLIB

REFERENCE NUMBER.: 20023
RELEASE..........: 2.07B

SYNOPSIS
Upload components

DESCRIPTION
It is not possible to read in components into the library, which have no graphic elements on layer no. 18.

WORKAROUND
Digitize at least a small track segment, like 1 mil long to the component graphic on layer 18, before uploading the component into the library.

CIS

REFERENCE NUMBER.: 20133
RELEASE..........: 2.07B

SYNOPSIS
Adb read error RC=-13

DESCRIPTION
This .adb is a 2.02p design with many busses. with 2.07b cis returns a -13 when reading the adb, also adbclean3 can not solve the problem.

CISCAM

REFERENCE NUMBER.: 19515
RELEASE..........: 2.07B

SYNOPSIS
CAM Tools, Centering

DESCRIPTION
Centering is only done on the last segment of a track, which might be inside a pad. In such a case the centering function is not working.

REFERENCE NUMBER.: 20343
RELEASE..........: 2.07B

SYNOPSIS
Sublands/ tracktype is fixed to 255 (ok 3.11)

DESCRIPTION
The Sublands tracktype i/o field is fixed to type 255, other entries will result in both, a beep and a ?

CISPCB

CISPCB Components

REFERENCE NUMBER.: 19581
RELEASE..........: 2.07B

SYNOPSIS
Copy components

DESCRIPTION
Core when copying components. The system will core if you place the copied component within a certain distance to the master component. The range around the master component within a copy is not possible is approximately equal the size of the largest defined technology set. In example: if a via size defined with 100 mm in the technology, the system will core if the distance between the terminals of the master component and the terminals of the copied component is less 100 mm.

WORKAROUND
a. Place the copied component far away from the master and move it afterwards, as the core will only happen in the copy procedure and not while moving. b. Place the copied component onto the master with component collision. c. Do not copy components locally, but insert them out of the library.

REFERENCE NUMBER.: 19586
RELEASE..........: 2.07B

SYNOPSIS
Multiselect terminals

DESCRIPTION
You will get a core dump if you issue the commands multiselect terminals / modify graphic / modify parameters.

REFERENCE NUMBER.: 19632
RELEASE..........: 2.07B

SYNOPSIS
Exchange terminals

DESCRIPTION
Create a terminal with a textfield and use it as master with the exchange shape function. - The exchanged terminal has no textfield.

REFERENCE NUMBER.: 19681
RELEASE..........: 2.07B

SYNOPSIS
Exchange components

DESCRIPTION
Connections are lost after a component exchange and after exiting CIS. Tricky is, that a final check before saving the adb file will not find the opens as the connections will be lost only when exiting CIS.

WORKAROUND
Read the adb file again into CIS and run a final check.

REFERENCE NUMBER.: 19716
RELEASE..........: 2.07B

SYNOPSIS
Place components / restricted areas

DESCRIPTION
You will get a component collision message if you try to place components into a restricted area, which is only restricted for tracks and vias, but not for components.

REFERENCE NUMBER.: 19968
RELEASE..........: 2.07B

SYNOPSIS
Reorganize airlines

DESCRIPTION
The switch reorg. on/off in the component submenu is not working anymore if the adb file has been converted with SCSCOM from .pcb to .adb .

REFERENCE NUMBER.: 20172
RELEASE..........: 2.07B

SYNOPSIS
Gate swap

DESCRIPTION
It is possible to swap gates of different function types (i.e. 7400 and 7402). It seems the function type is not taken in consideration in the swap process.

REFERENCE NUMBER.: 20760
RELEASE..........: 2.07B

SYNOPSIS
Terminal origin layer

DESCRIPTION
When creating a terminal to a component you may specify a layer for the terminal origin. The system allows you to specify a layer number from 0 up to 32767. Also it is possible to move existing terminals completely (origin and graphic) to those invalid layers. If you have a mixed combination like origin on layer 0 and graphics on layer 1 then this terminal will not be plotted in Output !

CISPCB Tracks

REFERENCE NUMBER.: 19533
RELEASE..........: 2.07B

SYNOPSIS
Splitted Planes / Digit. tracks , vias

DESCRIPTION
Digitize Tracks -- 1) After a SPL was created, it is not possible any more to move vias on the board 2) If you switched on -select no areas- in the tracks submenu, it is also not possible any more to select vias at all.

REFERENCE NUMBER.: 19598
RELEASE..........: 2.07B

SYNOPSIS
Move vias

DESCRIPTION
In rare cases you may get a core dumped when you move vias of the same signal onto eachother.

REFERENCE NUMBER.: 19988
RELEASE..........: 2.07B

SYNOPSIS
Multiselect tracks

DESCRIPTION
multi-select all non-fixed tracks on layers 1,2 del e. ...loop

WORKAROUND
multiselect tracks of all other layers, fix them then multiselect all tracks (only variable tracks will be selected) and delete them.

REFERENCE NUMBER.: 20022
RELEASE..........: 2.07B

SYNOPSIS
Move tracks

DESCRIPTION
Tracks can be moved onto textfields on copper- layers without getting any short circuit message from the system.

REFERENCE NUMBER.: 20658
RELEASE..........: 2.07B

SYNOPSIS
Multi delete tracks

DESCRIPTION
Multi selecting tracks with a window and a successive delete will result in 'STEINER'-points left on the screen and a wrong opens count in final check. Also when getting an airline to such a 'Steiner'- point the layer I/O field in the status-window shows either layer no. 26 or no. 0 . the steinerpoints should be deleted when running function 'reorganize airlines'

WORKAROUND
Select the tracks if possible with a wildcards entry.

CISPCB Graphic

REFERENCE NUMBER.: 19597
RELEASE..........: 2.07B

SYNOPSIS
Layout : graphic error

DESCRIPTION
Graphic error occurs when exiting Technology; Layer Plan (happens only with the OS4 version of CIS).

WORKAROUND
Issue a refresh command.

CISPCB Miscellaneous

REFERENCE NUMBER.: 19131
RELEASE..........: 2.07B

SYNOPSIS
Final check

DESCRIPTION
Final Check within a given window counts also the shorts at vias outside the window.

REFERENCE NUMBER.: 20219
RELEASE..........: 2.07B

SYNOPSIS
Final check, opens within terminals

DESCRIPTION
Final check finds 48 opens and 205 isolated areas, but shows no airlines. The terminals are composed out of arcs.

WORKAROUND
Arcs within a terminal graphic are not checked right now for connectivity. If you have such terminals, please do an interactive check and ignore the opens message from the final check.

REFERENCE NUMBER.: 20581
RELEASE..........: 2.07B

SYNOPSIS
Final check after board rotation

DESCRIPTION
final check reports 7 shorts (board rotation 0) final check reports 0 shorts (board rotation 900) final check reports 16 shorts (board rotation 1800) final check reports 0 shorts (board rotation 2700) BUT THERE ARE NO SHORTS AT ALL !!

WORKAROUND
Delete your board outline with the functions /restricted area/ get/ delete/ . digitize a new board outline as usual with /technology/sizes/ digitize outline.

CISPCB Technology

REFERENCE NUMBER.: 19512
RELEASE..........: 2.07B

SYNOPSIS
Layerplan, Documentation layers

DESCRIPTION
To add a docu-layer into the layerplan is not possible if you once deleted this docu-layer and have saved the file. If you read in this file again, you can re-enter the layer into the layerplan, but if you try to save this file you will get a message : 'Internal Error RC=428'.

WORKAROUND
Configure your layerplan always within the first CIS session and consider that docu-layers may only be deleted and re-entered within the same CIS session.

REFERENCE NUMBER.: 19541
RELEASE..........: 2.07B

SYNOPSIS
Via technology

DESCRIPTION
If a via has in the technology table only entries in the attribute section and drill type section and all other entries of this via type are set to 0, the via type entry will be deleted completely when saving the adb.

CISPCB Tools

REFERENCE NUMBER.: 20557
RELEASE..........: 2.07B

SYNOPSIS
AUTOCAD / dxf interface always cores (SUN4 only)

DESCRIPTION
AUTOCAD / dxf interface always cores (SUN4 only)

CISSCS

CISSCS Components

REFERENCE NUMBER.: 19115
RELEASE..........: 2.07B

SYNOPSIS
Symbol search

DESCRIPTION
Searching symbols is only possible with the SCS-name and not with the PCB-name.

REFERENCE NUMBER.: 19591
RELEASE..........: 2.07B

SYNOPSIS
Replace symbols

DESCRIPTION
1. program will not replace the shape of a gate if all gates are not used within the package. 2. If all gates are used and a replace is performed a new component is added and the original comp. has 3 out of 4 gates used. 3. If #2 above is performed the new component name is not an increment of the original. i.e. instead of Uxx the name is the system default $xx.

WORKAROUND
Terminals must have same name, for signal allocation.

REFERENCE NUMBER.: 19633
RELEASE..........: 2.07B

SYNOPSIS
Symbol textfields

DESCRIPTION
If textfields of a symbol are located outside the symbol frame and the symbol is not located in the window, also the textfields of this symbol are not shown. Exception : terminal text type 4 is shown.

REFERENCE NUMBER.: 20127
RELEASE..........: 2.07B

SYNOPSIS
Move Symbol

DESCRIPTION
Moving,with route mode on,of even small 3 terminal gates gives you horrible results of pseudoterminals staying somewhere on the sheet, useless loops in the same signal, open segments and so on.

CISSCS Hierarchy

REFERENCE NUMBER.: 19625
RELEASE..........: 2.07B

SYNOPSIS
Rename bus

DESCRIPTION
Core when renaming a bus within a copied hierarchy.

REFERENCE NUMBER.: 20126
RELEASE..........: 2.07B

SYNOPSIS
Copy hierarchy

DESCRIPTION
If a maingroup is copied with collision, the hierarchy below is not copied, but only the empty maingroup.

WORKAROUND
Try to avoid a collision to symbols when copying a hierarchy.

REFERENCE NUMBER.: 20130
RELEASE..........: 2.07B

SYNOPSIS
Move symbol to upper level hierarchy; signals lost

DESCRIPTION
If you move a symbol, connected only within the hierarchy to the maingroup's sheet, then the terminals of this symbol are deleted out of all signals. (connected move was switched on)

REFERENCE NUMBER.: 20141
RELEASE..........: 2.07B

SYNOPSIS
Multi connected copy

DESCRIPTION
Select a group of connected symbols (also BUS connections) , copy (connected copy and move is on), -- new BUS and new signals without any connection to the original group will be created.

WORKAROUND
copy single symbols with switch 'connected copy' instead of using multi symbol copy

---

CISSCS Miscellaneous

REFERENCE NUMBER.: 19600
RELEASE..........: 2.07B

SYNOPSIS
Reports

DESCRIPTION
Graphic windowing functions will not work after a component report is executed.

WORKAROUND
Press two times the key [F2].

REFERENCE NUMBER.: 19717
RELEASE..........: 2.07B

SYNOPSIS
Sh~w relations

D :RIPTION
R· .tion is not possible if the phy. component is mirrored or rotated.

CISSCS Signals

REFERENCE NUMBER.: 20021
RELEASE..........: 2.07B

SYNOPSIS
Bus connections

DESCRIPTION
It is not possible to connect 2 signals graphically on the same bus terminal. You get only a message 'short circuit with bus terminal'.

REFERENCE NUMBER.: 20183
RELEASE..........: 2.07B

SYNOPSIS
Rename 'sheet tag'-terminals

DESCRIPTION
If length of old terminal name modulo 4 is 2 (2,6, 10....) and the new name is one character longer, the system writes a binary NULL uncontrolled into the memory. Pointer violations may happen, resulting in a core dump.

WORKAROUND
Do renaming in two steps if you have such terminals.

LDROUT

REFERENCE NUMBER.: 19969
RELEASE..........: 2.07B

SYNOPSIS
Additional clearance

DESCRIPTION
LDROUT ignores additional clearances defined for tracks or vias

WORKAROUND
Instead of using the additional clearance, add the clearance value to the respective via and track sizes before routing, rout the board and change the sizes back to their original value.

REFERENCE NUMBER.: 19970
RELEASE..........: 2.07B

SYNOPSIS
Buried vias, glossing

DESCRIPTION
LDROUT is not glossing buried vias in a multilayer design.

WORKAROUND
LDROUT 3.19 is available on request

REFERENCE NUMBER.: 20282
RELEASE..........: 2.07B

SYNOPSIS
Router miscalculates via sizes in 1/20" rgrid.

DESCRIPTION
LDROUT miscalculates via sizes if the routing grid is 1/20" . In the example via type 17 is 1.4 mm, the router calculates the size for this via type 0.157 mm and is actually routing with this size, causing shorts in the design. This error happens only on SUN3 architecture !!

WORKAROUND
For SUN3 ld400_v3.19j and ldrout_v3.19j are available on request.

REFERENCE NUMBER.: **20342**
RELEASE..........: **2.07B**

SYNOPSIS
'Inhibit vias under components' switch

DESCRIPTION
Router ignores inhibit vias under component switch

REFERENCE NUMBER.: **20652**
RELEASE..........: **2.07B**

SYNOPSIS
Plane connections

DESCRIPTION
LDROUT is not putting any stub-out vias at smd terminals which have to be connected to plane layers, if routing is done on two signal layers only.

REFERENCE NUMBER.: **20671**
RELEASE..........: **2.07B**

SYNOPSIS
Loops

DESCRIPTION
When routing multilayers with more than one signal layer pair, LDROUT randomly creates loops, which means component pads are connected twice to the next routed via on different routing layer pairs.

OUTPUT

REFERENCE NUMBER.: **19634**
RELEASE..........: **2.07B**

SYNOPSIS
File names

DESCRIPTION
userpp produces a coredump if there is a second '. ' in filename i.e. neil.tst.gph

---

REFERENCE NUMBER.: 20273
RELEASE..........: 2.07B

SYNOPSIS
Terminals are not plotted

DESCRIPTION
Terminals which are not in the netlist, but having the plane-connection bit set, are not plotted at all on negative plane layers. If you have more than one plane layer then these terminals will short all your planes. This error does not happen with the next release anymore.

WORKAROUND
Cleaner was shipped 6th OCT.

SCSCOM

REFERENCE NUMBER.: 19585
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : track technology

DESCRIPTION
When converting a .pcb file to a .adb file, SCSCOM optimizes all not used, but in the track technology defined track types away.

WORKAROUND
A patched version 2.25r is available on request.

REFERENCE NUMBER.: 20060
RELEASE..........: 2.07B

SYNOPSIS
ADB - PCB : proposals

DESCRIPTION
Proposal track segments are put always to the terminal origin layer, no matter on which layer they were defined. Proposal vias are always placed on the terminal origin, no matter with which offset they were defined. This proposal via displacement happens also when converting a .gph file back to a .adb file.

WORKAROUND
scscom_v2.25s is available on request.

REFERENCE NUMBER.: 20151
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : Function type is missing

DESCRIPTION
If a pcb-file contains no loclib, but only a stockid as part definition, scscom gets the right part out of the library. Missing is the function-type information, which is in the library but not in the adb-file after transformation.

REFERENCE NUMBER.: 20231
RELEASE..........: 2.07B

SYNOPSIS
ADB - PCB : name collisions

DESCRIPTION
Component or signal name collisions are possible, if you work with both automatically and manually given names like components '$2' and '2'. In the pcb file both are named '2' !!

WORKAROUND
Automatically given name prefixes like $, &, # are not transferred if you do a conversion from .adb to .pcb . If possible rename the elements before doing a conversion.

REFERENCE NUMBER.: 20233
RELEASE..........: 2.07B

SYNOPSIS
PCB -ADB : stockid

DESCRIPTION
Convert a pcb file including components with stockids to adb , run CIS / components / modify param./ and you will see in the stockid field -- - nothing! great ! it was optimized away to save valuable disk space.

REFERENCE NUMBER.: 20234
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : .COMMENT fields

DESCRIPTION
.COMMENT fields with type larger than 256 are optimized away if they belong to a component. If they belong to a signal then SCSCOM gives you a syntax error.

REFERENCE NUMBER.: 20237
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : .init.adb

DESCRIPTION
If you do a conversion from pcb to adb and you have no .init.adb in your lib/sys directory you will get a core dumped.

REFERENCE NUMBER.: 20274
RELEASE..........: 2.07B

SYNOPSIS
ADB - GAI : wrong x adb y segments in OUTPUT

DESCRIPTION
ADB-GAI: x and y segments have sometimes wrong coordinates if the data came initially from V05. result - tracks are plotted randomly over the whole drawing in output.

WORKAROUND
Cleaner was shipped 6th Oct.

REFERENCE NUMBER.: 20654
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : outside restricted areas

DESCRIPTION
According to the 'PCB'-syntax documentation a restricted area restricts in - or outside elements depending whether the terminal origin is located in - or outside the polygon described with the track segments. SCSCOM does not take care of that rule, it creates instead only inside restricted areas.

WORKAROUND
Get the respective areas and change the inside bit to outside with the modify parameters function within CIS.

REFERENCE NUMBER.: 20666
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : overlapping track segments

DESCRIPTION
When copper areas in the .pcb-file are not described as such, but are composed out of overlapping parallel track segments then SCSCOM is converting these parallel tracks 'sometimes' into strange vector elements.

REFERENCE NUMBER.: 20670
RELEASE..........: 2.07B

SYNOPSIS
ADB - PCB : copper areas

DESCRIPTION
When converting an .adb-file to a .pcb-file SCSCOM will give you for each copper area which is not located on the component side layer a warning - error in polygon layers - and subsequently puts all outline tracks of these areas to layer number [1] = component side.

APHCOM

REFERENCE NUMBER.: 21830
RELEASE..........:

SYNOPSIS
core while running APHCOM

DESCRIPTION
core while running APHCOM; could be caused by the layer names used; If this is the case, CIS should not allow illegal names, or APHCOM must be modified to accept them.

REFERENCE NUMBER.: 21919
RELEASE..........: 3.11

SYNOPSIS
layer does not mirror the segments

DESCRIPTION
the segments of the pcb outline are not mirrored only textfields are mirrored

WORKAROUND
use a component outline for the pcb outline

CALLIB

REFERENCE NUMBER.: 21854
RELEASE..........:

SYNOPSIS
data transfer core

DESCRIPTION
Callib editor will core during transfer shapes, overwrite mode = always

---

CISCAM

REFERENCE NUMBER.: 21703
RELEASE..........: 3.11

SYNOPSIS
function 'bend slant' -- crazy vectors

DESCRIPTION
crazy vectors will be created after executing function 'bend slant' - the tracks were created by the optimizing router.

CISPCB

CISPCB Components

REFERENCE NUMBER.: 21579
RELEASE..........: 3.11

SYNOPSIS
Component multi-select move with RMB core

DESCRIPTION
see above

REFERENCE NUMBER.: 21727
RELEASE..........: 3.11

SYNOPSIS
Segmentation fault in 'place groups - matrix'

DESCRIPTION
last action : displaying grid for function 'place groups - matrix model'

CISPCB Areas

REFERENCE NUMBER.: 21022
RELEASE..........: 3.11

SYNOPSIS
Heattraps in copper areas; extra clearance

DESCRIPTION
When creating copper areas with heattraps, the 'heattrap segment width' + the clearance (for this layer) will be calculated for the 'copper area pad connection' - so it is not possible to create 'positive' copper areas with heattrap connections.

CISPCB Tracks

REFERENCE NUMBER.: 21415
RELEASE..........: 3.11

SYNOPSIS
Core after 'multi delete' of tracks

DESCRIPTION
After the following action CIS interrupts with a core: track, multi-action, RETURN click on the board, delete = core

---

REFERENCE NUMBER.: 21437
RELEASE..........: 3.11

SYNOPSIS
Techno check : core; select via check only

DESCRIPTION
Track get; with the middle mouse button get a via; sub-menu; techno check; ...core

---

REFERENCE NUMBER.: 21532
RELEASE..........: 3.11

SYNOPSIS
digitize track function "prefer airlines" inop.

DESCRIPTION
The digitize track sub-menu function "prefer airlines" does not work properly. If the function is NOT highlighted, the system still prefers airlines.

REFERENCE NUMBER.: 21565
RELEASE..........: 3.11

SYNOPSIS
fix via-move

DESCRIPTION
if you move a fixed via and on the other side of the connection there is also a fixed via on the same layer, they are automatically optimized. (is not the case in 2.07b)

CISPCB Miscellaneous

REFERENCE NUMBER.: 21519
RELEASE..........: 3.11

SYNOPSIS
component copy: SCS symbol name is not incremented

DESCRIPTION
If a component is copied on the PCB side, and the related SCS symbol does not reside on sheet 0 of the schematic; the new symbol added from the copy will have the original symbols' name. This will only occur on the first component copied.

REFERENCE NUMBER.: 21847
RELEASE..........: 3.11

SYNOPSIS
list of all used plane-terminals

DESCRIPTION
we need a list which contains all used plane-terminals in a design, that the customer is able to modify them with a heattrap. without a list it is impossible to modify the real design in an acceptable time new report: listing of terminal graphics within signals

---

REFERENCE NUMBER.: 21905
RELEASE..........: 3.11

SYNOPSIS
measurement function : duplicate info added

DESCRIPTION
If measurement info is added to the system default component, a duplicate of the information is added . This occurs only with "global Y", if a "global N " has been performed on the default component, the error will never occur again. init problem...

REFERENCE NUMBER.: 21906
RELEASE..........: 3.11

SYNOPSIS
measurement function : info added in background

DESCRIPTION
when info is added using the measurement function to the system default component on layer 2 the information is added in background color, when you leave the measurement form it will disappear altogether. If you then return to measurement, it will magically re-appear.

CISPCB Signals

REFERENCE NUMBER.: 21491
RELEASE..........: 3.11

SYNOPSIS
Core in local technology check

DESCRIPTION
After selecting all traces and submenu technology check - the system crashes.

WORKAROUND
The local technology check work only with one signal selected. If you want to check more signals then use the final check function in the main menu.

CISPCB Tools

REFERENCE NUMBER.: 21801
RELEASE..........: 3.11

SYNOPSIS
acad - interface : board outline

DESCRIPTION
in the moment for 'acad-board outline' the restr. area defined in the tec. is used - this is wrong - with prisma version 3 the board outline will be defined with the new 'layer function' (layer graphic).

CISSCS

CISSCS Components

REFERENCE NUMBER.: 21479
RELEASE..........: 3.11

SYNOPSIS
symbol-mod. param. "block mirr. + blocked rotat."

DESCRIPTION
"blocked rotate 90:X" with 'symbol-mod. param.' works incorrect. 'X' cannot be deleted; i.e. 'blocked rot. 90:X' is always active

----

REFERENCE NUMBER.: 21751
RELEASE..........: 3.11

SYNOPSIS
cannot multi-copy

DESCRIPTION
the symbols being copied are first copied c'to sheet 0 with the same PCB name and therefore will not be placed... error message: name already used or wrong

----

CISSCS Hierarchy

REFERENCE NUMBER.: 21440
RELEASE..........: 3.11

SYNOPSIS
Sheet-tag; message: 'terminal hold no signal'

DESCRIPTION
If you delete a signal on a sheettag the message appears 'terminal hold no signal' If you stitch again 'signal autom. created'. If you want to connect with your own signalname the systems message is 'terminal hold signal'.

CISSCS Signals

REFERENCE NUMBER.: 21876
RELEASE..........: 3.11

SYNOPSIS
signal-merge on different hierarchies not possible

DESCRIPTION
signal-merge on different hierarchies sometimes not possible - error came from development (v. scholz) - example is needed to check this error.

CISSCS Tools

REFERENCE NUMBER.: 21825
RELEASE..........: 3.11

SYNOPSIS
bugs in hilo-interface

DESCRIPTION
multiple wire declarations of busses when bus is existing in several hierarchies statements for hilo too long if large component (e.g.. 68020) wrong wire declarations in dwl-frame

---

IOCONV

REFERENCE NUMBER.: 21762
RELEASE..........: 3.11

SYNOPSIS
bus hierarchies will be deleted after conversion

DESCRIPTION
bus hierarchies will be deleted after conversion - this results in corruption of the 3.11 adb data

---

3.11

CALLIB

REFERENCE NUMBER.: 21262
RELEASE..........: 2.07B

SYNOPSIS
Symbolclass of power or ground symbols

DESCRIPTION
Symbolclass of power or ground symbols is not stored into the callib.
this should be implemented in the 'new columbine'

CIS

REFERENCE NUMBER.: 20630
RELEASE..........: 3.11

SYNOPSIS
cisini.cmd

DESCRIPTION
cisini.cmd works differently on PCB and SCS sides. ... name prefixes

----

REFERENCE NUMBER.: 21667
RELEASE..........: 3.11

SYNOPSIS
Reports in SCS/PCB

DESCRIPTION
In all report-forms it is not possible to exit a window - when touching
function 'quit' the system goes back to the previous window information
try this with report form 'technology'

REFERENCE NUMBER.: 21716
RELEASE..........: 3.11

SYNOPSIS
panning

DESCRIPTION
To change the size of the panning area the user must first turn off
panning, change values, re-pan

CISCAM

REFERENCE NUMBER.: 20651
RELEASE..........: 2.07B

SYNOPSIS
Vias too close near one another

DESCRIPTION
The router puts vias of the same net too close together. This is true, especially when routing plane signals. In that case the router places vias (size 40 Mil) with a distance of 25 mil (origin to origin). Glossing doesn't help with these stub-out vias of plane signals. There should be a cam function which deletes those unnecessary vias.

REFERENCE NUMBER.: 21123
RELEASE..........: 3.11

SYNOPSIS
CAM -- TEARDROP & CORNERS

DESCRIPTION
If TEARDROP function is performed, then BEND SLANT some teardrops are lost or put in background mode. (2 layer board !) Upon return to LAYOUT, more are lost or put in background.

REFERENCE NUMBER.: 21257
RELEASE..........: 3.11

SYNOPSIS
Bend slant

DESCRIPTION
Bend slant ignores the track code 0 board outline added with the "layer" function. - shorts will be created

---

REFERENCE NUMBER.: 21301
RELEASE..........: 3.11

SYNOPSIS
Silkscreen; wrong text copied (this file only)

DESCRIPTION
When creating the silkscreen, the text on component "passkreuz" for layers: 100,102,110, 111,120,121 all are copied to the silkscreen layer 112.

REFERENCE NUMBER.: 21420
RELEASE..........: 3.11

SYNOPSIS
TECHNOLOGIES ASSIGNED TO SPLIT PLANE LAYERS.

DESCRIPTION
THE TRACK TYPE USED FOR DRAWING THE DIVIDING LINE GETS ITS TECHNOLOGY FROM WHICH EVER TECHNOLO GY IS ASSIGNED TO THE DOCU LAYER FOR THE PLANE, BUT THE DOCU PADS ARE CREATED FROM THE PADS FROM THE SIG LAYER PLUS THE MIN. CLEARENCE. BOTH THE TRACKS AND PADS SHOULD USE THE TECHNOLOGY ASSIGNED TO THE DOCU LAYER.

REFERENCE NUMBER.: 21514
RELEASE..........: 3.11

SYNOPSIS
Sublands

DESCRIPTION
Sublands are created to all trackwidthtypes which are used in the job. The entry of tracktype does not work.

REFERENCE NUMBER.: 21526
RELEASE..........: 3.11

SYNOPSIS
spider with split planes

DESCRIPTION
the spider is displayed in the wrong colour Furthermore, it cannot be output without the other layers.

REFERENCE NUMBER.: 21774
RELEASE..........: 3.11

SYNOPSIS
glue spots on SILK SCREEN

DESCRIPTION
glue spots on SILK SCREEN The glue spots of layer 22 are displayed on the SILK SCREEN.

REFERENCE NUMBER.: 21780
RELEASE..........: 3.11

<u>SYNOPSIS</u>
silk-screen -- error.

<u>DESCRIPTION</u>
After modifying the silk screen a second time, the reference layer is missing in 'silk layer' of the 'layer plan'.

CISPCB

CISPCB Components

REFERENCE NUMBER.: 19402
RELEASE..........: 2.07B

<u>SYNOPSIS</u>
Multiplacement; pickup window visibility

<u>DESCRIPTION</u>
with components/multiselect/move the pickup window should stay visible, or the components should be visible in the actual window. accurate placement not possible without this

---

REFERENCE NUMBER.: 20391
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Component insert : defaults

<u>DESCRIPTION</u>
although the default search sequence has changed to "P" the name input into the text field is still the "L" logical name. This results in a "name not in library" error if re-inserted.

REFERENCE NUMBER.: 20536
RELEASE..........: 3.11

<u>SYNOPSIS</u>
LAYER

<u>DESCRIPTION</u>
1. If text is added using the "layer" function on a layer other than the comp, or solder, the user cannot get that text without knowing which layer it was added. Perhaps you could add a "next y/n" as is done in track get. 2. Text added using the "layer" function cannot be gotten using the "text, get;" function.

REFERENCE NUMBER.: 20572
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Pool graphic names

<u>**DESCRIPTION**</u>
When creating a component, we cannot create 1 new graphic and use an existing graphic for the second graphic.

REFERENCE NUMBER.: 20701
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Insertion of components

<u>**DESCRIPTION**</u>
When parts are inserted from the library with 'opt. shape', the message 'name unknown in library - lxxx' appears, although the component is then correctly inserted.

REFERENCE NUMBER.: 20721
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Exchange shape; checks function type

<u>**DESCRIPTION**</u>
when a component shape is exchanged, do NOT check if the function type is the same. Maintain all logical parameters for the exchanged component, and just exchange the SHAPE. Provide warning and override capability. - only exchange of pins in 3.11 - then spec.

REFERENCE NUMBER.: 20753
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Component names

<u>**DESCRIPTION**</u>
When components are created, it is possible to specify more characters than are later displayed with 'component / get', in SCS there is not enough space for the PCB name.

REFERENCE NUMBER.: 21011
RELEASE..........: 3.11

SYNOPSIS
Terminal layers

DESCRIPTION
It should not be possible to move, or create, a terminal on any layers except the component side or solder side (and then with a new graphic). error 1: I moved a terminal to layer 3, then moved the component from the comp side to solder side; the terminal was then on layer 0. error 2: I moved a terminal to layer 3, the graphics of the terminal were mirrored.

REFERENCE NUMBER.: 21047
RELEASE..........: 3.11

SYNOPSIS
Graphic

DESCRIPTION
When components with additional graphic are moved on docu layers (unequal 17, 18), this additional graphic is highlighted.

REFERENCE NUMBER.: 21121
RELEASE..........: 3.11

SYNOPSIS
Terminal : modify graphic; via insert with mouse

DESCRIPTION
If while modifying a terminal graphic,.a via is inserted using the right most button of the mouse, the via added appears on the silkscreen docu layers in addition to all physical layers. If the via is added using digitize via, all layers (*), and the left button, the via appears only on all physical layers. - restrict it

---

REFERENCE NUMBER.: 21161
RELEASE..........: 3.11

SYNOPSIS
Impossible to transform one comp. twice

DESCRIPTION
Impossible to transform one component twice. E.g. transform comp. with terminal, then create new terminals to the new comp. and transform the comp. again -- doesn't work, CIS prompts with '[PCB-2109] component cannot be related' i.e. signals cannot be connected to the new terminals.

REFERENCE NUMBER.: 21240
RELEASE..........: 3.11

SYNOPSIS
Component (fix:Y) with multi-select/delete

DESCRIPTION
Component (fix:Y) with multi-select/delete; all components are deleted (even if there are no 'fixed' ones)

REFERENCE NUMBER.: 21281
RELEASE..........: 3.11

SYNOPSIS
Via restriction under components; digitize error

DESCRIPTION
When components are modified to not allow vias under components, when an attempt is made to digitize a via under that component the system will add a steiner point, switch to the opposite layer and release the cursor (return to track get)

REFERENCE NUMBER.: 21317
RELEASE..........: 3.11

SYNOPSIS
Component : modify parameters; via restrict error

DESCRIPTION
component, modify parameters, restrict vias Y; component outline (parts of it) were offset. component, modify parameters, restrict vias Y; the parts from the library worked ok, except I could not "get" them with the cursor anymore. If you get them with a list, no name will appear in the component field.

REFERENCE NUMBER.: 21332
RELEASE..........: 3.11

SYNOPSIS
CORE : move rot. point of via.

DESCRIPTION
CORE : move rot. point of via.

REFERENCE NUMBER.: 21383
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Move component with multi-select

<u>DESCRIPTION</u>
Get components with multi-select, then move them - Cis message: "[PCB-0000] collision with restr. area" , but single components can be moved (without any problems).

REFERENCE NUMBER.: 21424
RELEASE..........: 3.11

<u>SYNOPSIS</u>
COMPONENTS REMAIN HIGHLIGHTED

<u>DESCRIPTION</u>
PICK UP A COMP, MOVE AND LET GO.... IS STILL HIGHLIGHTED.

REFERENCE NUMBER.: 21489
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Global change of 'centre of rotation'

<u>DESCRIPTION</u>
The possibility to globally change the rot. points of components has to be restricted.

REFERENCE NUMBER.: 21548
RELEASE..........: 3.11

<u>SYNOPSIS</u>
opt. "NEXT" with comp./multi-select/get

<u>DESCRIPTION</u>
The option 'next' with component/multi-select/get works incorrectly. After 'next:Y' CIS prompts with '[PCB-0000] element does not match form values'

REFERENCE NUMBER.: 21551
RELEASE..........: 3.11

SYNOPSIS
Create component with 'existing' graphic

DESCRIPTION
activate component on solder side (repr.: dashes) /create /place graphic of new component - accord to the status window the new comp. is on the comp. side, but the graphic is dashed as if it were on the solder side (especially applies to SMD components)

REFERENCE NUMBER.: 21568
RELEASE..........: 3.11

SYNOPSIS
Component/move

DESCRIPTION
It is possible to place an SMD component onto a 'normal' component without any error message (e.g. 'component collision')

REFERENCE NUMBER.: 21643
RELEASE..........: 3.11

SYNOPSIS
rename function

DESCRIPTION
if you start with renaming the component the system recommends only componentside. no next flag

---

REFERENCE NUMBER.: 21659
RELEASE..........: 3.11

SYNOPSIS
Terminals : plane flag

DESCRIPTION
If you remove a planesignal from a terminal, the plane flag on the terminal is not reset.

REFERENCE NUMBER.: 21663
RELEASE..........: 3.11

SYNOPSIS
Component with multi move

DESCRIPTION
It is not possible to move a comp. with 'multi'; CIS prompts with: "[PCB-0000] component lies outside of sheet" Most of the time...

REFERENCE NUMBER.: 21668
RELEASE..........: 3.11

SYNOPSIS
Heat trap

DESCRIPTION
Must be defined on each layer to be sure that they are in use. This is a huge amount of data which we have to keep in library or in the start-up file

---

REFERENCE NUMBER.: 21748
RELEASE..........: 3.11

SYNOPSIS
Component height

DESCRIPTION
Component height field in modify parameters is not accessible. Also height is fixed to 7.465, no matter whether your units are in Inch or mm.

REFERENCE NUMBER.: 21782
RELEASE..........: 3.11

SYNOPSIS
plane bit set on all terminals of exchanged comp.

DESCRIPTION
All of the memory components in the above file were exchanged, and all the terminals of each of the components are set to plane connections.

REFERENCE NUMBER.: 21792
RELEASE..........: 3.11

SYNOPSIS
Mirrored and not mirrored terminal texts

DESCRIPTION
If a text is created to a terminal on comp.side the text will be mirrored when the component is moved to the solder side. Same text is not mirrored if created on solder side.

REFERENCE NUMBER.: 21834
RELEASE..........: 3.11

SYNOPSIS
rest. areas defined to components not mirrored

DESCRIPTION
If a restricted area is defined to a component on the comp side, and the component is mirrored to the solder side the restricted area remains on the component side.

REFERENCE NUMBER.: 21835
RELEASE..........: 3.11

SYNOPSIS
rest. area def. to comp. on SOLD side layer=NULL

DESCRIPTION
If a restricted area is defined to a component which is on the solder side, the restricted area layer is NULL.

REFERENCE NUMBER.: 21874
RELEASE..........: 3.11

SYNOPSIS
gluepads entry with inch unit

DESCRIPTION
if you define a gluepad by using inch unit the value will change automatically to a default

---

REFERENCE NUMBER.: 21888
RELEASE..........: 3.11

SYNOPSIS
cannot multi-copy and easily insert symbols in SCS

DESCRIPTION
When components are multi-copied on the PCB side the symbols on the SCS side get the same names as the ones which were copied, therefore they cannot be added to the schematic sheet without renaming first. Related to Package 101

REFERENCE NUMBER.: 21934
RELEASE..........: 3.11

SYNOPSIS
shifted component graphics with res. area addition

DESCRIPTION
When a restricted area is defined to a component the component outline is shifted. If the restrict area is then deleted, the outline returns to it's proper position (this file only)

CISPCB Areas

REFERENCE NUMBER.: 19980
RELEASE..........: 3.11

SYNOPSIS
Mod.graphic / copper area tracktype

DESCRIPTION
If the tracktype of the copper area is changed from 1 to 4, and if the copper area is then activated again, the tracktype is set back to 1.

REFERENCE NUMBER.: 20301
RELEASE..........: 3.11

SYNOPSIS
Copper areas to terminals not possible

DESCRIPTION
It is not possible to create copper areas to pads - when digitizing such elements following message always occur: 'area not closed' - copper areas have to be defined also on documentation layers - inputfield to define that layer is needed areas for terminals cannot be created on the desired layer. If a layer other than comp. is specified and if 'areas' is then clicked, the layer is set back to 'comp.'.

REFERENCE NUMBER.: 20324
RELEASE..........: 3.11

SYNOPSIS
Copper areas, change offset

DESCRIPTION
When the offset is changed in mod. parameters of copper areas, the message 'illegal offset' is always displayed.

REFERENCE NUMBER.: 20599
RELEASE..........: 3.11

SYNOPSIS
Restricted areas

DESCRIPTION
1. Copper area can be moved into a restricted area without any error message; 2. You can generate a restricted area with elements enclosed without an error message; only after a 'get' and a 'move' at the same place an errormessage will be displayed.

REFERENCE NUMBER.: 20668
RELEASE..........: 2.07B

SYNOPSIS
Cross hatched copper areas

DESCRIPTION
If you cross-hatch a copper area, then your area will get one cross-hatch grid smaller at the right and lower edge than your original area. This will happen only if foreign elements like tracks or vias are located somewhere within your area.

REFERENCE NUMBER.: 20804
RELEASE..........: 3.11

SYNOPSIS
Restr. areas/ mod. param./ status window --[320]

DESCRIPTION
restr. area/ mod. param./ status window -- [320] if 'block: comp.:' is set to 'N', the 'layer' automatically changes to 'layer:ZERO'.

REFERENCE NUMBER.: 20841
RELEASE..........: 3.11

SYNOPSIS
Plane copper area

DESCRIPTION
The heat trap segments do not correspond to the heat trap elements generated on the plane docu layer (graphic error)

REFERENCE NUMBER.: 21005
RELEASE..........: 3.11

SYNOPSIS
Copy restricted areas with layerchange

DESCRIPTION
Not possible to copy restricted areas with 'layerchange' - the copied area is always on the same layer like the 'source area'.

REFERENCE NUMBER.: 21650
RELEASE..........: 3.11

SYNOPSIS
Copper areas , modify graphic

DESCRIPTION
After modifying a crosshatched copper area you may get a warning "alien elements in area". When you leave "modify graphic" you are asked "ignore errors Y/N " (which errors ??) you type Y and get a message PCB[0000] invalid function (????). When you try to un-crosshatch such an area you get only a message 'not correctly defined area' (??).

REFERENCE NUMBER.: 21665
RELEASE..........: 3.11

SYNOPSIS
Copy components with plane flag signals

DESCRIPTION
If you copy components with plane flag signals, the new component keeps the flag information. This is absolutely dangerous.

REFERENCE NUMBER.: 21793
RELEASE..........: 3.11

SYNOPSIS
Copper area ignores textfields

DESCRIPTION
Copper area ignores copper textfields of a terminal.

CISPCB Tracks

REFERENCE NUMBER.: 21042
RELEASE..........: 3.11

SYNOPSIS
Layers / dig. outline

DESCRIPTION
It is not possible to dig. an outline; if the airline is lost, it cannot be activated anymore. If the last segment is to be activated with 'get', cis jumps back to layer 2; the last generated segment is difficult to activate, you do not know where to continue. You NEVER know what is activated and where. - 'get' must search only on 1 layer

---

REFERENCE NUMBER.: 21141
RELEASE..........: 3.11

SYNOPSIS
Track get

DESCRIPTION
When getting a track segment to digitize, the search sequence at the coordinate selected should be the following: 1. An airline 2. A track on the current layer 3. A track on any other layer. Currently when digitizing on an 8-layer board the system will toggle between each group of four which can be time consuming. And frustrating.

REFERENCE NUMBER.: 21242
RELEASE..........: 3.11

SYNOPSIS
Track-delete with multi-select.

DESCRIPTION
If some segments of a track are specified as 'fix' and some are specified as fix:"N", it is completely deleted after a 'delete with multi-select'.

REFERENCE NUMBER.: 21360
RELEASE..........: 3.11

SYNOPSIS
Error after moving a via; track cannot be modified

DESCRIPTION
After a via is moved onto a fixed via, the connected track segments cannot be modified anymore.

REFERENCE NUMBER.: 21396
RELEASE..........: 2.07B

SYNOPSIS
Setup highlight

DESCRIPTION
Setup highlight for tracks is deactivated after another function

---

REFERENCE NUMBER.: 21438
RELEASE..........: 3.11

SYNOPSIS
Teardrops

DESCRIPTION
Segments of teardrops which belong to signal graphic are not checked for shorts (when digitizing over teardrops)

REFERENCE NUMBER.: 21494
RELEASE..........: 3.11

SYNOPSIS
Restriction in restricted areas

DESCRIPTION
Restricted areas which are defined around the restricted parts are recognized with finalcheck and during digitizing. Restricted areas which are defined outside and then moved to the restricted parts are not recognized at all. Restricted area which are defined outside of pcb area are recognized from cis but not from routing

---

REFERENCE NUMBER.: 21613
RELEASE..........: 3.11

SYNOPSIS
Via block

DESCRIPTION
Viablocker blocks also tracks

REFERENCE NUMBER.: 21646
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
shifted via move

<u>**DESCRIPTION**</u>
if you move the shifted via, the segments are shifted in one axis and the vector segment in the via is a orthogonal Also, if you move in a non-orthogonal motion the vector will be non-orthogonal regardless of the sub-menu setting.

REFERENCE NUMBER.: 21658
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Sub-menu functions do not remain active

<u>**DESCRIPTION**</u>
It is inconsistent that some of the sub-menu functions remain active during a single editing session. Please, when the user sets or resets a sub-menu function, save that value for the entire editing session.

REFERENCE NUMBER.: 21763
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Copy graphic from master layer to another layer

<u>**DESCRIPTION**</u>
It is not possible to copy the master layer to another layer. The error occurs only with docu layers.

REFERENCE NUMBER.: 21765
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Copy TEXT from master layer to another layer

<u>**DESCRIPTION**</u>
When copying from a master layer to another layer TEXT is not copied.

REFERENCE NUMBER.: 21844
RELEASE..........: 3.11

SYNOPSIS
no drc for terminal segments

DESCRIPTION
circle-segments which belong to a terminal are not under drc. finalcheck does not recognize the shorts, but check this as an open

---

CISPCB Graphic

REFERENCE NUMBER.: 20258
RELEASE..........: 3.11

SYNOPSIS
LAYERS-digitize: rotated text cannot be generated

DESCRIPTION
LAYERS-digitize: rotated text cannot be generated (status-window - "rot.:____")

REFERENCE NUMBER.: 20339
RELEASE..........: 3.11

SYNOPSIS
Graphic set-up

DESCRIPTION
Frame: cannot turn outline off ($PHY) Segments: (X) - heattraps do not show up (T) - SMD terminals do not show up Restricted Areas: when cleared restricted areas still show (Y) free standing work as component ($PHY) Vias: (V) fixed vias show up

---

REFERENCE NUMBER.: 20514
RELEASE..........: 3.11

SYNOPSIS
Graphic-error, restricted areas invisible

DESCRIPTION
Graphic setup. If 'components' is switched off (components:__), all restr. areas are invisible also.

REFERENCE NUMBER.: 21140
RELEASE..........: 3.11

SYNOPSIS
"Top layers" in graphic menu doesn't work

DESCRIPTION
Can't look at only one layer at a time when "top layers" is "1". Some layers show other colors also. .

REFERENCE NUMBER.: 21291
RELEASE..........: 3.11

SYNOPSIS
Hardcopy of copper areas

DESCRIPTION
With hardcopies, the copper areas are drawn twice.

REFERENCE NUMBER.: 21397
RELEASE..........: 2.07B

SYNOPSIS
Airline and highlight same color

DESCRIPTION
If you work with track-highlight and you pick up another track, the first airline disappears completely

---

REFERENCE NUMBER.: 21497
RELEASE..........: 3.11

SYNOPSIS
Text fields disappear

DESCRIPTION
Text fields defined to routed or digitized vias disappear when the entry 'CPT' in Graphic Setup/ texts is changed. The text fields remain invisible even after saving and reading the adb again. A diff on the two example adbs shows that one bit in the PT:071 record has changed from 1 to 0.

REFERENCE NUMBER.: 21575
RELEASE..........: 3.11

SYNOPSIS
copper area

DESCRIPTION
outline of a copper area are visible after deleting if it is a trackwidth type higher than 1 workaround refresh?

REFERENCE NUMBER.: 21576
RELEASE..........: 3.11

SYNOPSIS
Copper areas

DESCRIPTION
Copper areas with real-mode on, are filled only up to the trackwidth-center of the outline which means area is smaller than defined. Workaround rasterize shows the real dimension?

REFERENCE NUMBER.: 21612
RELEASE..........: 3.11

SYNOPSIS
Via block move

DESCRIPTION
No cursor support for viablocker move

REFERENCE NUMBER.: 21940
RELEASE..........: 3.11

SYNOPSIS
grid turns off after layer change

DESCRIPTION
When toggling between the layers (using F5) the grid will turn off between each group of 4 in the graphic set-up. Sparc station

---

CISPCB Miscellaneous

REFERENCE NUMBER.: 19573
RELEASE..........: 2.07B

SYNOPSIS
Measurements

DESCRIPTION
Issue the commands measurements/popup modify graph ic/window/set a window/full view and all the visible graphic , except the display grid is gone. additionally the coordinate fields show some strange values like 200000mm.

WORKAROUND
Press two times the key [F2] (changing to schematics and back to layout) and your graphic is again visible.

REFERENCE NUMBER.: 20166
RELEASE..........: 3.11

SYNOPSIS
New form 'layer'; restrict via addition

DESCRIPTION
It must be restricted to digitize the graphic element 'via' in the form 'layer'.

REFERENCE NUMBER.: 20185
RELEASE..........: 2.07B

SYNOPSIS
Hardcopy, plotting of airlines

DESCRIPTION
It is not possible to plot only airlines or plot airlines with any other combination of elements.

REFERENCE NUMBER.: 20388
RELEASE..........: 3.11

SYNOPSIS
Pop-up

DESCRIPTION
If the 'modes' pop-up is switched on, and 'terminal/get' is activated, the pop-up is still ON, and you cannot continue your work

---

REFERENCE NUMBER.: 21013
RELEASE..........: 3.11

SYNOPSIS
Pool graphics: function 'delete' and 'list'

DESCRIPTION
The function 'delete' and 'list' for pool graphics is needed urgently.

REFERENCE NUMBER.: 21128
RELEASE..........: 3.11

SYNOPSIS
Pairwise interchange; placement outside brd area

DESCRIPTION
When doing a pairwise interchange with components placed on both component side and solder side some components are placed outside the board area.

REFERENCE NUMBER.: 21273
RELEASE..........: 3.11

SYNOPSIS
Splitplanes; highlight in two colors

DESCRIPTION
There is no way to view two signals simultaneously in two different colors. Highlight in white the active signal other signals in blue. Or, make user definable.

REFERENCE NUMBER.: 21309
RELEASE..........: 3.11

SYNOPSIS
Hardcopy; pin representation incorrect

DESCRIPTION
Made a hardcopy of my "split-plane" docu layer #122 and the comp pins were just small dots. If I make a "post-proc" plot, the pins are shown in there correct size.

REFERENCE NUMBER.: 21432
RELEASE..........: 3.11

SYNOPSIS
HP PLOT BUG

DESCRIPTION
AN "HP" PLOT DOESN'T SHOW THW "$" OF COMP NAME "$33" SO IT'S POSSIBLE TO HAVE TWO COMPS WITH THE SAME NAME ON A CHECK PLOT OR SILKSCREEN FILM. EX: $33 AND 33 ON YOUR BOARD LOOKS LIKE 33 ON FILM

REFERENCE NUMBER.: 21642
RELEASE..........: 3.11

SYNOPSIS
multi copy

DESCRIPTION
multi copy changes the comprefix to $ character.

REFERENCE NUMBER.: 21697
RELEASE..........: 3.11

SYNOPSIS
Final check

DESCRIPTION
The final check does not check for drill collision between two different signals. It only checks for collisions which may have been created by the router.

REFERENCE NUMBER.: 21779
RELEASE..........: 3.11

SYNOPSIS
reports : "not digitized nets" incorrect

DESCRIPTION
In "Misc." "reports" the information has not been updated since digitizing in the missing connections from the router. Final check has been run, which shows 0 missing connections.

REFERENCE NUMBER.: 21901
RELEASE..........: 3.11

SYNOPSIS
measurement function : units always metric

DESCRIPTION
Measurements are always in metric units. Surely we could check the "modes" setting and perform the simple arithmetic required.

REFERENCE NUMBER.: 21902
RELEASE..........: 3.11

SYNOPSIS
measurement function : shorts possible

DESCRIPTION
Final check does not detect shorts of signal tracks to measurement segments on copper layers.

REFERENCE NUMBER.: 21903
RELEASE..........: 3.11

SYNOPSIS
measurement function : adds on non-existent layers

DESCRIPTION
It is possible to create measurement information on layers which do not exist.

REFERENCE NUMBER.: 21904
RELEASE..........: 3.11

SYNOPSIS
measurement function : will delete outline info.

DESCRIPTION
The "delete segment" function will also delete component graphic information (including the text) Although a layer is specified when starting the measurement function, it is still possible to delete segments from other layers.

REFERENCE NUMBER.: 20470
RELEASE..........: 3.11

SYNOPSIS
Graphic error; heat trap not displayed correctly

DESCRIPTION
If you created a plane and open a window over the copper area of a heat trap, the heat trap is not displayed correctly.

REFERENCE NUMBER.: 21115
RELEASE..........: 3.11

SYNOPSIS
Trackmove

DESCRIPTION
If you move variable tracks against fixed tracks, the fixed tracks are also moved.

CISPCB Signals

REFERENCE NUMBER.: 20561
RELEASE..........: 3.11

SYNOPSIS
Signal / multi

DESCRIPTION
If 'multi' has been activated, you cannot leave this function, if you click e.g. components. Clicking components should entail a deactivation of the 'multi' function.

REFERENCE NUMBER.: 20777
RELEASE..........: 3.11

SYNOPSIS
Signal / insert

DESCRIPTION
If you make a writing error during the specification of a netlist, CIS prompts with 'pin cannot be connected' if you then enter the correct comp. and pin plus [CR], another pin is entered in the 'pin' field and the signal is not connected to the right pin. You have to specify everything again before the pin is correct.

CISPCB Technology

REFERENCE NUMBER.: 20535
RELEASE..........: 3.11

SYNOPSIS
PCB - CAM - PCB: Do not refresh the graphics

DESCRIPTION
Do not refresh the graphics when moving from Layout to CAM, or from CAM to Layout. On large boards this is very time consuming.

REFERENCE NUMBER.: 20714
RELEASE..........: 3.11

SYNOPSIS
Menu output; via and track sizes displayed wrong

DESCRIPTION
In the menus always the values of a technology are output which is defined under no. 0, e.g. the via diameter. The output of the field 'viadiameter' should be deleted, because it's a layer-independent graphic. The trackdiameter should be calculated correctly for each layer.

REFERENCE NUMBER.: 21249
RELEASE..........: 3.11

SYNOPSIS
technology/drills: delete drill type

DESCRIPTION
technology/drills [989] drill no. 4 cannot be deleted. CIS prompts:'[PCB-0000] change standard drill in viatype: 18, but viatype:18 is drill type:3.

REFERENCE NUMBER.: 21286
RELEASE..........: 3.11

SYNOPSIS
Drill technology; macro name changed after update

DESCRIPTION
If the drill macro is modified, global Y, the macro name in the technology will change to a STDN number.

REFERENCE NUMBER.: 21290
RELEASE..........: 3.11

SYNOPSIS
Menu type v of via tech generates erroneous adb

DESCRIPTION
If vias are generated in menu type v of the via technology, the adb cannot be stored anymore. CIS interrupts with the message: internal error [rc=1]. The MESSAGES window contains the message: VIA-TECH[28] via_drill-SEGtype = 0 ? In this example, via type 28 was generated.

REFERENCE NUMBER.: 21419
RELEASE..........: 3.11

SYNOPSIS
Layer plan bug. also is confusing.

DESCRIPTION
IF USING "SSC" TYPE LAYERS, THE SYS EXPEXTS THE"SHP" LAYER NAMES IN THE "REFERENCE LAYER" FIELD. FINE. BUG #1: YOU CAN'T TYPE IN THE LAYER NO.,YOU MUST TYPE IN THE NAME. BUG #2: AFTER GOING TO CAM AND CREATING A SILKSCREEN, THIS "REF. LAYER" FIELD IS AUTOMATICALLY CHANGED BACK TO LAYERS 1 & 2.

REFERENCE NUMBER.: 21421
RELEASE..........: 3.11

SYNOPSIS
GLOBAL VIA CHANGES

DESCRIPTION
CAN'T GLOBALLY MODIFY A VIA IN THE MAIN MENUE. GIVE ME ONE GOOD REASON. YOU CAN GLOBALLY MODIFY COMP GRAPHICS, SCHEME SYMBOL GRAPHICS AND TERMS, BUT NOT PCB VIAS. VERY UNCONSISTANT.

REFERENCE NUMBER.: 21475
RELEASE..........: 3.11

SYNOPSIS
Cannot delete layer in technology form

DESCRIPTION
Cannot delete layers 7 or 8. message: layer not empty. It is always possible to delete an ISO layer, even if elements exist. example: delete a signal layer which is loaded with tracks... no problem... re-add the layer all the segments are still there.

REFERENCE NUMBER.: 21662
RELEASE..........: 3.11

SYNOPSIS
Cannot write "D" in track technology

DESCRIPTION
If you delete in tracktech. the track with attr. 'D', you cannot create a terminal in REFPCBGRP, since there is no default tracktype. Back in track tech., you cannot specify a 'D' in 'attr.'. NOTE TO USERS: DO NOT DELETE ANY TRACK TYPES WHICH MAY HAVE HAD THE DEFAULT SPECIFICTATION IN YOUR 2.07 FILES.

REFERENCE NUMBER.: 21750
RELEASE..........: 3.11

SYNOPSIS
Rounding problems

DESCRIPTION
Working Units = mm ; try to enter a 2.6 mm round via into your technology (you won't make it). The diameter 2.6 is rounded to 2.5997 mm, and the radius 1.3 is rounded to 1.3002 mm. Write back technology - message 'invalid value'. Give the user the possibility to define a reasonable basis resolution like 1/25400" throughout the system to get rid of all of these rounding problems.

CISPCB Text

REFERENCE NUMBER.: 20595
RELEASE..........: 3.11

SYNOPSIS
Text sizes

DESCRIPTION
Submenu function 'text index' is sensible only if a global value transfer from the technology takes place.

REFERENCE NUMBER.: 21423
RELEASE..........: 3.11

SYNOPSIS
FIXED TEXT CAN'T BE CHANGED TO "NOT FIXED TEXT"

DESCRIPTION
ADD TEXT TO A COMP AS "FIXED". PICK IT UP AND CHANGE IT TO "NOT FIXED". LET GO AND PICK UP AGAIN . IS STILL FIXED.

REFERENCE NUMBER.: 21426
RELEASE..........: 3.11

SYNOPSIS
Placement of text when text is "not fixed" & LB

DESCRIPTION
IF TEXT IS NOT FIXED AND LEFT JUSTIFIED, IT IS NOT POSITIONED CORRECTLY...according to the cursor.

CISSCS

CISSCS Symbols

REFERENCE NUMBER.: 20755
RELEASE..........: 3.11

SYNOPSIS
Create terminals

DESCRIPTION
When creating terminals with the 'continue' function the terminal name is not automatically counted up.

REFERENCE NUMBER.: 20825
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Symbol / exchange

<u>DESCRIPTION</u>
Symbol / exchange with continue does not work.

REFERENCE NUMBER.: 20837
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Text fields of mirrored symbols

<u>DESCRIPTION</u>
If a text field is dig. to a mirrored symbol and is then placed, everything works all right. If you activate it again and then place it again, this text field gets a negative offset in x, with the next attempt a pos. offset in x. Furthermore, if two textfields are placed, then moved, the text fields are not refreshed.

REFERENCE NUMBER.: 20849
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Symbol create / creation of names

<u>DESCRIPTION</u>
With symbol create a name is not immediately entered in the status window; an immediate entry would be sensible above all because in PCB it works in that way.

REFERENCE NUMBER.: 20853
RELEASE..........: 3.11

<u>SYNOPSIS</u>
Scaled symbols

<u>DESCRIPTION</u>
If a symbol has been scaled, it cannot be replaced anymore with the 'replace' function. Furthermore, symbols are not correctly scaled.

REFERENCE NUMBER.: 20857
RELEASE..........: 3.11

SYNOPSIS
Rotation point of symbols

DESCRIPTION
If the rot. point is moved while working in 'dig. line' all dig. lines are deleted. The symbol reappears if the rot. point is moved directly from the 'delete' function. 1st action: if the function is not executable, it should not be visible in the form.

REFERENCE NUMBER.: 21025
RELEASE..........: 3.11

SYNOPSIS
Symbol insert

DESCRIPTION
When inserting symbols from the library, and the symbol contains global signals which do not exist in the design, only one such signal is displayed at a time as an error. This will not allow the symbol to be inserted. So, we go to signal create and create the signal then return to symbol insert and the system will flag another signal. Most frustrating! Please allow an override to automatically create all missing signals. Or, at least list ALL missing signals at one time.

REFERENCE NUMBER.: 21029
RELEASE..........: 3.11

SYNOPSIS
Terminal : modify graphics

DESCRIPTION
When the default text height for a particular text type (4: phys terminal) has been changed, the height displayed when adding the text is not changed. The text is added with the proper height. I do not know where the value which is displayed comes from???

REFERENCE NUMBER.: 21053
RELEASE..........: 3.11

SYNOPSIS
Symbol/delete ('immediate kill': off)

DESCRIPTION
Symbol/delete: if 'immediate kill' is switched off the message 'really delete' does not appear before the deletion - like on the PCB side.

REFERENCE NUMBER.: 21081
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Symbol replace

<u>**DESCRIPTION**</u>
Replacing gated symbols changes the gate used. example: replace U1(gate #1) with U2(gate #3) U1(gate #1) will change to U2(gate #4). I only wanted to replace the graphic!!!

REFERENCE NUMBER.: 21297
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Formats : library transfer

<u>**DESCRIPTION**</u>
Because text type 8 (sheet-number) cannot be added to symbols (only formats), there is no way to transfer this information into the library, on what will be a format. Therefore, when a format is added from the library the automatic sheet numbering cannot possibly work. The user still must modify every sheet of the schematic to reflect the sheet number.

REFERENCE NUMBER.: 21366
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Terminal : modify parameters

<u>**DESCRIPTION**</u>
If the "swapclass" is changed to anything except 0, the (output,io,input,don't care) fields are reset to don't care. If then the user changes the field to "output" the swapclass is reset to 0. Also, these fields are either not saved in the library (callib) or will not read in from the library. All terminals come in to Cis as "don't care".

REFERENCE NUMBER.: 21377
RELEASE..........: 3.11

<u>**SYNOPSIS**</u>
Connected copy

<u>**DESCRIPTION**</u>
When symbols are copied, (with connected copy on) the signal traces are not visible until a refresh has been performed.

REFERENCE NUMBER.: 21478
RELEASE..........: 3.11

SYNOPSIS
Symbol with 'deletable:N'

DESCRIPTION
Symbols with 'deletable:N' can be deleted without any problems.

REFERENCE NUMBER.: 21484
RELEASE..........: 3.11

SYNOPSIS
Symbol/create with 'continue'

DESCRIPTION
Create symbol (with symbol class P and G) with CONTINUE is not possible.

REFERENCE NUMBER.: 21501
RELEASE..........: 3.11

SYNOPSIS
Symbol-(PCB-name)-rename

DESCRIPTION
Symbol-(PCB-name)-rename; If the PCB name already exists, the system does not output any message and defaults to 'move'.

REFERENCE NUMBER.: 21506
RELEASE..........: 3.11

SYNOPSIS
Def. main group

DESCRIPTION
After 'def. main group' CIS prompts with '[SCS-1105] no symbol selected'.

REFERENCE NUMBER.: 21543
RELEASE..........: 3.11

SYNOPSIS
Terminal-replace is not possible in SCS

DESCRIPTION
A Terminal-replace function which replaces terminal graphics is badly needed on the SCS side. If a terminal is inadvertently modified "local" it will be local forever! This function is available on the PCB side.

REFERENCE NUMBER.: 21608
RELEASE..........: 3.11

SYNOPSIS
Symbol replace

DESCRIPTION
Symbol/replace: if the replacing symbol has no attributes, CIS prompts with: '[SCS-1105] no symbol selected'.

REFERENCE NUMBER.: 21610
RELEASE..........: 3.11

SYNOPSIS
Symbol/swap

DESCRIPTION
Symbol/swap : After executing this function, CIS prompts with '[SCS-1105] no symbol selected'.

REFERENCE NUMBER.: 21621
RELEASE..........: 3.11

SYNOPSIS
global change of schematic format not possible

DESCRIPTION
global change of schematic format not possible. Sometimes it works OK, most of the time it does not. It seems to work the first time it is attempted, and never again after that.

REFERENCE NUMBER.: 21766
RELEASE..........: 3.11

SYNOPSIS
relation

DESCRIPTION
Relations cannot be generated via entries with the keyboard; the cursor has to be used.

REFERENCE NUMBER.: 21868
RELEASE..........: 3.11

SYNOPSIS
copy a symbol segment -- error

DESCRIPTION
copy a symbol segment -- error: the segment is placed at a wrong position. (only if the symbol was rotated)

CISSCS Tracks

REFERENCE NUMBER.: 21684
RELEASE..........: 3.11

SYNOPSIS
Digitize

DESCRIPTION
If you digitize new signals on a virgin symbol and you connect during digitizing the second signal to the first one you can't activate anything on your schematic part again; workaround: press F2 twice

---

CISSCS Hierarchy

REFERENCE NUMBER.: 20920
RELEASE..........: 3.11

SYNOPSIS
Define main group

DESCRIPTION
When defining a main group, and sheet-tags are involved, please use the same type of sheet tag that was used in the upper level. Currently there is no way of exchanging the terminal graphics, and if the upper level hierarchy terminals have been moved this effort would be destroyed by re-adding sheet-tags on the inner level.

REFERENCE NUMBER.: 21150
RELEASE..........: 3.11

SYNOPSIS
No mod.signal parameters in higher hierarchy-level

DESCRIPTION
Modification of signal parameters (especially display, external stimuli) can be done in the lowest hierarchy-level only. If you do it in a higher level only the display of the user interface is toggled to y or n, but if you switch down one level, signal parameters haven't changed. (complete signal is on)

REFERENCE NUMBER.: 21277
RELEASE..........: 3.11

SYNOPSIS
Terminal textfield "bus" not copied

DESCRIPTION
When copying with hierarchy bus-textfields at sheet tags (= off page connectors going to the outside of the hierarchy) are not copied.

REFERENCE NUMBER.: 21512
RELEASE..........: 3.11

SYNOPSIS
Core in function 'copy with hierarchy'

DESCRIPTION
Core when executing function 'copy with hierarchy' with a symbol which has no hierarchy.

REFERENCE NUMBER.: 21689
RELEASE..........: 3.11

SYNOPSIS
Copy with hierarchy

DESCRIPTION
If you copy 'with hierarchy' and the function 'connected copy' is active, the copy action is done without the hierarchy; sometimes the system crashes. (no segmentation fault... bus error!!)

REFERENCE NUMBER.: 21731
RELEASE..........: 3.11

SYNOPSIS
define main group disconnects routed signals

DESCRIPTION
PLEASE when defining a main group maintain ALL connections exactly the way they were in the upper level, including all sheet-tags and busses. Currently, it does not even use the same type of sheet-tag...

REFERENCE NUMBER.: 21739
RELEASE..........: 3.11

SYNOPSIS
main groups

DESCRIPTION
when main groups are generated, the format (in this case sheet no. 4) is not always transferred, it just does not exist.

REFERENCE NUMBER.: 20860
RELEASE..........: 3.11

SYNOPSIS
Search sequence to CALLIB

DESCRIPTION
When specifying the search sequence B (body), L (logic) and the search pattern 7400, nothing is found. Searching 74* with the same search-sequence brings up all 74er I N C L U D I N G the 7400. In other words: search sequences are working only if wildcards are used. --- with other words: if a full name is specified only the 1. search sequence will be executed - in this example 'B (symbol body)'.

REFERENCE NUMBER.: 21048
RELEASE..........: 3.11

SYNOPSIS
Container

DESCRIPTION
The display of containers is incorrect. a) the status window of 'change symbol attributes' displays in the 2nd line under 'art' "0". The 5th line also contains an 'art' field, here, the contents is o.k. the error in line 2 entails an error in the container header (line 10). b) the contents of the container is not displayed. (this also applies if the error under a) is corrected manually).

REFERENCE NUMBER.: 21185
RELEASE..........: 3.11

SYNOPSIS
'General': subwindow works incorrectly

DESCRIPTION
If the subwindow is activated twice, the sheet (format) graphic disappears. The user has to give a 'refresh'.

REFERENCE NUMBER.: 21247
RELEASE..........: 3.11

SYNOPSIS
'Sheet' cannot be deleted

DESCRIPTION
'Sheet' cannot be deleted

REFERENCE NUMBER.: 21279
RELEASE..........: 3.11

SYNOPSIS
Define relation

DESCRIPTION
If a component is placed outside the board (e.g. after place group/ reset), it is not possible to define relations.

REFERENCE NUMBER.: 21744
RELEASE..........: 3.11

SYNOPSIS
format lost after file save

DESCRIPTION
The sheet 1 format is lost after a global modification and the file is saved and re-read in.

REFERENCE NUMBER.: 21745
RELEASE..........: 3.11

SYNOPSIS
symbol "LOG" rename core...

DESCRIPTION
If an attempt is made to rename the sheet 1 "LOG" (in order to change the BLOCK NAME in the format) the system will core. Note: the first attempt did not get a segmentation fault, it was a bus error.

CISSCS Signals

REFERENCE NUMBER.: 20328
RELEASE..........: 3.11

SYNOPSIS
Signal sub-menu - 'auto bus connect'

DESCRIPTION
In the signal submenu the opt. 'auto bus connect has no effect on the work in the schematic.

REFERENCE NUMBER.: 20934
RELEASE..........: 3.11

SYNOPSIS
Create bus and switch GRID on - error

DESCRIPTION
Create bus and switch GRID on - error. System defaults to bus/move.

REFERENCE NUMBER.: 21504
RELEASE..........: 3.11

SYNOPSIS
Define maingroup

DESCRIPTION
Def. maingroup works incorrectly - when executing function 'define maingroup' each signal will get a new sheet-tag terminal in the newly created maingroup.

CISSCS Text

REFERENCE NUMBER.: 20928
RELEASE..........: 3.11

SYNOPSIS
Terminal/digitize text, e.g. text type:4

DESCRIPTION
It is possible to activate 'lb:X' and 'rb:X' at the same time.

REFERENCE NUMBER.: 21824
RELEASE..........: 3.11

SYNOPSIS
TEXT/status window [790]:"save values" and "ISO"

DESCRIPTION
get text with "TEXT" and toggle the fields "save values" and "ISO" (from "Y" to "N", or from "X" to "_"): the text is automatically placed to another position.

LDROUT

REFERENCE NUMBER.: 21435
RELEASE..........: 3.11

SYNOPSIS
Daisy-chaining ECL-SMDS with stubs on multi layers

DESCRIPTION
Daisy-chaining ECL-SMDS with stubs on multi layer boards does not work properly. A 'daisy chain connection' is only possible on those layers where the smd-terminals exist - a workaround to rout 'smd-daisy-chain-conn.' On multilayer boards is to define the component terminals + stubs as a single unit.

REFERENCE NUMBER.: 21702
RELEASE..........: 3.11

SYNOPSIS
re-routing with SMD terminals

DESCRIPTION
If a change is made on a design which removes a SMD terminal pin-out (or a new SMD has been added) on a multi-layer PCB. The router will not "re-pin-out" the terminals, therefore there is no possibility to connect these opens on the inner layers. This results in very poor routing results. In the example, the PCB routed to 0 with no problem. The re-route would not complete...ever

---

OUTPUT

REFERENCE NUMBER.: 21483
RELEASE..........: 2.07B

SYNOPSIS
Gerber driver / EBCDIC conversion

DESCRIPTION
Only letters, numbers and the * character are translated when using the build-in ASCII - EBCDIC conversion. Need a full conversion when using other end-of-block char. than *. At the moment all other characters are suppressed in the gerber file.

SCSCOM

SCSCOM Miscellaneous

REFERENCE NUMBER.: 21664
RELEASE..........: 3.11

SYNOPSIS
Copper areas are ignored from SCSCOM to PCB

DESCRIPTION
Copper area ignored with message 'barrier ignored'

REFERENCE NUMBER.: 20232
RELEASE..........: 2.07B

SYNOPSIS
ADB - PCB : restricted areas defined to components

DESCRIPTION
Restricted areas defined in CIS to components have after pcb_out nothing to do anymore with the components, instead they are part of the board.

REFERENCE NUMBER.: 20653
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB: restricted areas; layer change

DESCRIPTION
In the 'PCB'-syntax restricted areas may be described with track elements located on all [*] layers. SCSCOM puts such an area only to the solder side.

REFERENCE NUMBER.: 20669
RELEASE..........: 2.07B

SYNOPSIS
PCB - ADB : area, track, via elem. in one terminal

DESCRIPTION
If you have besides a copper area and its track elements also additional tracks or vias within one terminal then SCSCOM treats the additional elements as part of the copper area's outline, causing in CIS a message 'area not closed' and in postprocessing a message 'copper area not correct defined -- ignored'.

REFERENCE NUMBER.: 21749
RELEASE..........: 3.11

SYNOPSIS
ADB - PCB : coordinate units

DESCRIPTION
Need badly a possibility to specify in which units coordinates should be output. In the moment only Calay's home made 1/960" units are available. This worked in CIS in v2.07b. Used by Japan

---

REFERENCE NUMBER.: 21936
RELEASE..........: 3.11

SYNOPSIS
comment

DESCRIPTION
in the version 2.25y we could handle 126 comments (from total comments 31000) NOW in this version we can handle NOTHING AT ALL

Index

N

O

P

R

NUMBER.: 20328 61
NUMBER.: 20339 42
NUMBER.: 20342 13
NUMBER.: 20343 2
NUMBER.: 20388 45
NUMBER.: 20391 28
NUMBER.: 20470 48
NUMBER.: 20514 42
NUMBER.: 20535 49
NUMBER.: 20536 28
NUMBER.: 20557 8
NUMBER.: 20561 49
NUMBER.: 20572 29
NUMBER.: 20581 7
NUMBER.: 20595 52
NUMBER.: 20599 37
NUMBER.: 20630 25
NUMBER.: 20651 26
NUMBER.: 20652 13
NUMBER.: 20653 64
NUMBER.: 20654 16
NUMBER.: 20658 5
NUMBER.: 20666 17
NUMBER.: 20668 37
NUMBER.: 20669 64
NUMBER.: 20670 17
NUMBER.: 20671 13
NUMBER.: 20701 29
NUMBER.: 20714 49
NUMBER.: 20721 29
NUMBER.: 20753 29
NUMBER.: 20755 52
NUMBER.: 20760 4
NUMBER.: 20777 49
NUMBER.: 20804 37
NUMBER.: 20825 53
NUMBER.: 20837 53
NUMBER.: 20841 37
NUMBER.: 20849 53
NUMBER.: 20853 53
NUMBER.: 20857 54
NUMBER.: 20860 60
NUMBER.: 20920 58
NUMBER.: 20928 62
NUMBER.: 20934 62
NUMBER.: 21005 38
NUMBER.: 21011 30
NUMBER.: 21013 46
NUMBER.: 21022 19
NUMBER.: 21025 54
NUMBER.: 21029 54
NUMBER.: 21042 39
NUMBER.: 21047 30
NUMBER.: 21048 60
NUMBER.: 21053 54
NUMBER.: 21081 55
NUMBER.: 21115 48
NUMBER.: 21121 30
NUMBER.: 21123 26
NUMBER.: 21128 46
NUMBER.: 21140 43
NUMBER.: 21141 39
NUMBER.: 21150 58
NUMBER.: 21161 30
NUMBER.: 21185 60
NUMBER.: 21240 31
NUMBER.: 21242 39
NUMBER.: 21247 60
NUMBER.: 21249 50
NUMBER.: 21257 26
NUMBER.: 21262 25
NUMBER.: 21273 46
NUMBER.: 21277 58
NUMBER.: 21279 61
NUMBER.: 21281 31
NUMBER.: 21286 50
NUMBER.: 21290 50
NUMBER.: 21291 43
NUMBER.: 21297 55
NUMBER.: 21301 26
NUMBER.: 21309 46
NUMBER.: 21317 31
NUMBER.: 21332 31
NUMBER.: 21360 39
NUMBER.: 21366 55
NUMBER.: 21377 55
NUMBER.: 21383 32
NUMBER.: 21396 40
NUMBER.: 21397 43
NUMBER.: 21415 19
NUMBER.: 21419 50
NUMBER.: 21420 27
NUMBER.: 21421 51
NUMBER.: 21423 52
NUMBER.: 21424 32
NUMBER.: 21426 52
NUMBER.: 21432 47
NUMBER.: 21435 63
NUMBER.: 21437 20
NUMBER.: 21438 40
NUMBER.: 21440 23
NUMBER.: 21475 51
NUMBER.: 21478 56
NUMBER.: 21479 23
NUMBER.: 21483 63
NUMBER.: 21484 56
NUMBER.: 21489 32
NUMBER.: 21491 22
NUMBER.: 21494 40
NUMBER.: 21497 43
NUMBER.: 21501 56
NUMBER.: 21504 62

S

T

U

V