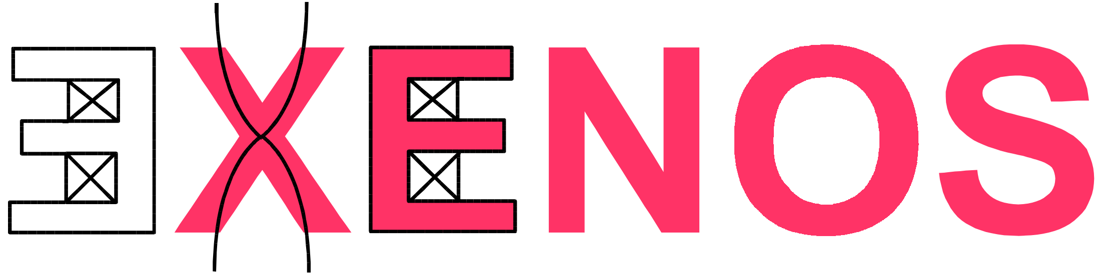


ECP Instruction Manual

XENOS Semiconductor Technologies GmbH

September 8, 2010

Contents

1 Introduction 9
- 1.1 Overview . . . 9
- 1.2 Contact Information . . . 9

2 Quick Start 11
- 2.1 Preparing a Sample . . . 11
- 2.2 Starting the System . . . 11
- 2.3 Creating a Pattern File . . . 11
- 2.4 Writing a Control File . . . 13
- 2.5 Installing the Sample . . . 13
- 2.6 Exposing the Sample . . . 14
- 2.7 Shutdown . . . 15
- 2.8 Developing the Sample . . . 15

3 Menus and Toolbars 17
- 3.1 Invoking Menus . . . 17
- 3.2 The Menubar . . . 17
  - 3.2.1 File menu . . . 17
  - 3.2.2 Edit menu . . . 18
  - 3.2.3 View menu . . . 19
  - 3.2.4 Tools menu . . . 21
  - 3.2.5 Mark menu . . . 23
  - 3.2.6 Calibration menu . . . 24
  - 3.2.7 EBL menu . . . 25
  - 3.2.8 System menu . . . 25
  - 3.2.9 Windows menu . . . 26
  - 3.2.10 Help menu . . . 27
- 3.3 Context Menus . . . 27
- 3.4 Toolbars . . . 27
  - 3.4.1 Mode Selection . . . 28
  - 3.4.2 Toolbar Buttons for GDSII Files . . . 28

4 Pattern Definition 29
- 4.1 Writing a Pattern . . . 29
- 4.2 Structure Name . . . 29
- 4.3 Increment . . . 29
- 4.4 Dwell Time . . . 30
- 4.5 Special Commands . . . 30
  - 4.5.1 PROCESS . . . 30
  - 4.5.2 FSIZE . . . 30
  - 4.5.3 UNIT . . . 31
  - 4.5.4 OFFSET . . . 31

4.6 Shapes . . . 32
4.6.1 RECT . . . 32
4.6.2 LINE . . . 32
4.6.3 XPOLY . . . 32
4.6.4 YPOLY . . . 33
4.6.5 DOT . . . 34
4.6.6 CIRCLE . . . 34
4.6.7 RING . . . 34
4.6.8 RDOT . . . 34
4.6.9 MSDOT . . . 35
4.6.10 SQUARENUT . . . 35
4.6.11 HEXNUT . . . 35
4.6.12 HEXZUT . . . 36
4.7 END . . . 37
4.8 Repetition of Shapes . . . 37
4.8.1 Rectangular Array . . . 37
4.8.2 Repetition of Structures . . . 38
4.8.3 Hexagonal Arrays . . . 39
4.9 Psl Script . . . 40
4.9.1 The Psl Console . . . 40
4.9.2 Structure of a Psl Script . . . 42
4.9.3 Defining Variables . . . 42
4.9.4 createStructure . . . 42
4.9.5 Creating Shapes . . . 43
4.9.6 Setting Exposure Parameters . . . 43
4.9.7 for/next Loop . . . 43
4.9.8 Arithmetic Commands . . . 43
4.9.9 Example PSL Script . . . 44

5 Control Files 45
5.1 Introduction . . . 45
5.2 Commands . . . 45
5.2.1 adraw . . . 45
5.2.2 align . . . 46
5.2.3 autorot . . . 46
5.2.4 bid_off_time . . . 46
5.2.5 cmark . . . 46
5.2.6 cmcoord . . . 47
5.2.7 cmfile . . . 47
5.2.8 cmfilepos . . . 47
5.2.9 cmark_fpr . . . 47
5.2.10 current . . . 47
5.2.11 dose . . . 47
5.2.12 dose_table . . . 48
5.2.13 draw . . . 49
5.2.14 dscale . . . 49
5.2.15 end . . . 50
5.2.16 fcup . . . 50
5.2.17 fcup_in . . . 52

5.2.18 fcup_out . . . 52
5.2.19 field_rot, +field_rot, -field_rot . . . 52
5.2.20 for - loops . . . 53
5.2.21 fscale . . . 53
5.2.22 fsize . . . 53
5.2.23 gdraw . . . 54
5.2.24 ginc . . . 54
5.2.25 glength . . . 54
5.2.26 gline . . . 54
5.2.27 gtime . . . 54
5.2.28 gwidth . . . 54
5.2.29 gmark (MANUAL / AUTO) . . . 55
5.2.30 gmcoord . . . 55
5.2.31 gmfile = filename . . . 56
5.2.32 gmfilepos . . . 56
5.2.33 gmark_fpr . . . 56
5.2.34 ht_off . . . 56
5.2.35 ht_on . . . 56
5.2.36 idraw . . . 57
5.2.37 image_draw . . . 57
5.2.38 image_pixsize . . . 58
5.2.39 mode . . . 58
5.2.40 next . . . 59
5.2.41 origin . . . 59
5.2.42 recipe . . . 60
5.2.43 rmark . . . 60
5.2.44 rot . . . 60
5.2.45 +rot,-rot . . . 60
5.2.46 setfocus (INIT, AUTO, OFF, NOW) . . . 60
5.2.47 sfile . . . 60
5.2.48 sii_timeout . . . 61
5.2.49 stage . . . 61
5.2.50 x,y,z . . . 61
5.2.51 +x,+y,-x,-y . . . 61
5.2.52 wait . . . 61
5.2.53 pause . . . 61
5.3 Recipes . . . 61
5.4 Proximity Correction . . . 63
5.5 Overlay files . . . 63

6 GDS-II Files **65**
6.1 Editing GDSII Files . . . 65
6.2 Exposing Data from GDSII Files . . . 66

7 Alignment Marks **67**
7.1 Editing Marks . . . 67
7.1.1 Cross Marks . . . 67
7.1.2 Image Mark . . . 68
7.1.3 Geometry Mark . . . 72

7.2 Test Mark . . . 72
7.2.1 Cross Mark . . . 72
7.2.2 Image Mark . . . 73
7.3 Mark Detection . . . 73
7.3.1 Cross Mark . . . 73
7.3.2 Image Mark . . . 73

8 Stage Control **77**
8.1 Moving Around . . . 77
8.1.1 Entering Coordinates . . . 77
8.1.2 Using the Stage Drawing . . . 77
8.1.3 Manual Mode . . . 77
8.2 Memory Buttons . . . 79
8.2.1 Sample corners . . . 79
8.2.2 Position Memory . . . 79
8.2.3 Sample Rotation . . . 79
8.3 Configuration files . . . 79

9 XeBaSe **81**
9.1 Stage . . . 81
9.1.1 Handling the Stage . . . 81
9.1.2 Using a JEOL Airlock . . . 81
9.2 Blanker . . . 84
9.3 XeBaSe Central Controller . . . 85
9.4 Blanker Tension Supply . . . 85
9.5 Saving Parameters . . . 85
9.6 Display and Control Unit . . . 85
9.6.1 Mode Switching . . . 85
9.7 Controlling the Stage . . . 85
9.7.1 Parameter Discovery . . . 85
9.7.2 Hard Limits . . . 86
9.7.3 Stage Reset . . . 86
9.7.4 Velocity Warnings . . . 86
9.8 Controlling the Blanker . . . 86
9.9 Setting Up the Blanker . . . 86
9.9.1 Out Position . . . 87
9.9.2 Blanking Position . . . 87
9.9.3 BCD Position . . . 87
9.9.4 Resetting Position Memory . . . 88
9.9.5 Setting Blanker Tension . . . 88
9.9.6 Leaving Adjust Mode . . . 88

10 General Controls **89**
10.1 SEM Image Scan . . . 89
10.2 Beam Control . . . 90
10.3 Dose Calculator . . . 90

11 Calibration **93**
11.1 Field Calibration . . . 93

11.2 Working Distance Correction . . . 94
11.3 Global Marks . . . 94
11.3.1 Setting Up Global Marks . . . 94
11.3.2 Aligning the Sample . . . 95
11.3.3 Repeating Alignment . . . 95
11.4 Using a Reference Mark . . . 96
11.5 Chip Marks . . . 96

12 Settings **99**
12.1 Calibration . . . 99
12.2 Exposure Field Settings . . . 100
12.3 File Formats . . . 100
12.4 Logging . . . 101
12.5 Stage . . . 101
12.5.1 Stage Coordinates . . . 101
12.5.2 Laser Stage . . . 102
12.6 SEM Interface . . . 102
12.7 Startup . . . 102

A Key Shortcuts **107**
A.1 Global shortcuts . . . 107
A.2 Text Editor . . . 108
A.3 Stage Control . . . 109

B Pattern reference **111**
B.1 Numerical shape codes . . . 111

C CTL Reference **113**

D Step-by-Step Startup Lists **115**
D.1 Startup List . . . 115
D.2 Restart List . . . 115

E Notes on Third Party Software **117**
E.1 Kiss FFT . . . 117

1 Introduction

The Exposure Control Program (ECP) is a software for pattern design and exposure with the XENOS pattern generator. The software is very flexible and can be used for completely automatic control of E-beam lithography systems as well as for manual exposure of single exposure fields with a SEM. Therefore some of the options and commands described in this manual may not be available in your version of the ECP software. Please refer to the data sheet of your pattern generator for a detailed description of your system configuration.

1.1 Overview

ECP supports several possibilities to define and edit layouts. The simplest way to define a pattern is to create a text file, which is called pattern or pat file. A pat file contains a list of pattern names and coordinates of shape primitives. An additional file, which is called control or CTL file, is needed to expose the patterns. The commands in a Ctl file move the stage, expose patterns, perform alignment to marks, etc. Multiple Ctl files can be viewed simultaneously as different layers by using overlay (ovl) files. Furthermore, ECP supports a number of additional file formats (such as GDSII and DXF) and conversion of those files from and to CTL and PAT files.

This manual is organized as follows: chapter 2 gives a brief glance at pattern and control files using a simple example. Chapter 3 gives an overview over the menus and control buttons. Chapter 4 provides a description of all available shapes and the syntax of pattern files. Chapter 5 describes the command language used in ctl files and ovl files. How to prepare data from a GDS-II file is explainde in chapter 6. The definition and use of alignment marks is explained in chapter 7. Chapter 8 gives a description of the stage control menu. Chapter 11 describes the filed calibration procedures of the ECP.

1.2 Contact Information

XENOS Semiconductor Technologies GmbH
Lustnauer Straße 78 72127 Kusterdingen
Germany

phone: +49 7071 32416
fax: +49 7071 32436

email: info@xenos-semi.com
http://www.xenos-semi.com

2 Quick Start

This chapter gives a quick description on how an exposure run with ECP and a XENOS pattern generator works. A run consists of the steps

- preparing a sample,
- starting up the system,
- creating pattern or image files,
- creating a control file
- installing the sample in the e-beam or SEM,
- exposing the pattern onto the sample,
- shutting down the system and
- developing the resist and using the sample

The steps that do not involve the ECP or the pattern generator are only superficially described in this manual.

The detailed description on creating patterns with the pattern editor is provided in chapter 4. A complete list of commands available in control files can be found in chapter 5.

2.1 Preparing a Sample

Coat a sample with an electron sensitive resist and mount it on an sample holder

2.2 Starting the System

Power on the pattern generator. Then start the ECP program on the PC. Next select “Load XeDraw2 Kernel...” from the system menu and click on “open” to use the standard file to load the kernel into the pattern generator. See also D.1.

2.3 Creating a Pattern File

Select “New” in the File menu or press the 🗋 (“New File”) button on the file toolbar. The program will ask for the type of file to be created. Check the box “PAT” for a pattern description file and press the “Apply” button. The main window will change and is now divided into three empty areas. The area in the lower left is a text editor which is used to edit the coordinates of the patterns in pat files. Click in this area and a cursor will appear. Now enter the following text:

```
D Q-Start
I 2
C 200
RECT 10000, 20000, 30000, 40000
END
```

The `D` gives the pattern a name, "`Q-Start`". `I` sets the increment. "`I 2`" means that the beam is moved to only every second point. `C 200` sets the dwell time, that is the time a single point is exposed, to 200 ns. `RECT` is the code for a rectangle, 10000 and 20000 define its lower left corner and 30000 and 40000 its upper right corner.

> **N. B.** Most commands in ECP are case sensitive. You must input them exactly as shown; in pat files the letters must usually be uppercase, in most other cases they must be lowercase. An exception are ctl files, where case doesn't matter in the commands.

Press F5 or the ("Redraw") button and then the ("View All") button.
The screen should now look like figure 2.1.

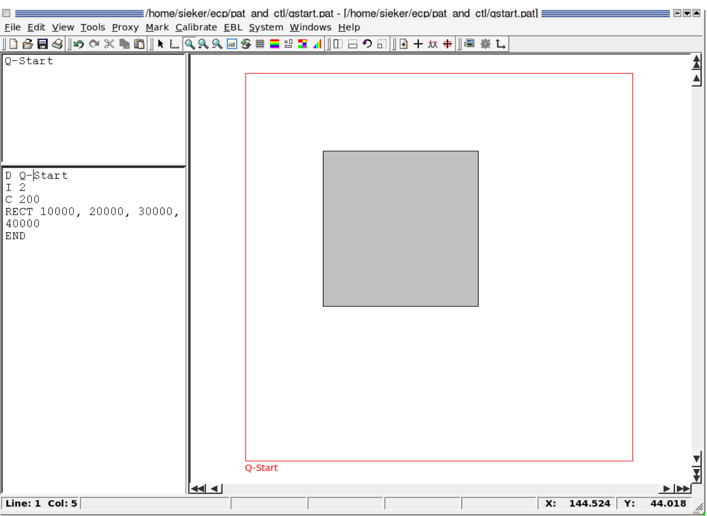


Figure 2.1: Editing a pattern file

The pattern is displayed in the display area on the right side. The box on the upper left side contains a list of all patterns in the current file and allows quick access to them. Click on the name of a pattern and the cursor in the text editor will jump to the place of the pattern definition in the file and only this pattern is shown on the right. A movable splitter separates the display area from the two widgets on the left side and can be used to adjust their relative sizes.

The red square surrounding the rectangle represents the exposure field. Depending on DAC configuration of your pattern generator the number of available pixels will be different. Therefore, the pattern will fill more or less of the exposure field. The name of the pattern is displayed in red text at the bottom of the field.

Now save the file as "qstart.pat" by selecting "Save As" from the file menu.

2.4 Writing a Control File

Among many other things a control file defines which pattern should be written where on the sample.

To create a basic ctl file chose “New File” again, but now select “CTL”. The screen is again divided into the display area on the right and a text editor on the left side.

Type the following text into the editor:

```
sfile = qstart

current = 500
origin = 0, 0

x = 2500
y = 2500
stage

draw (Q-Start)

+x = 200
stage
draw (Q-Start)

end
```

`sfile = qstart` defines which pattern file to use—the one we just wrote. `current` defines the writing current in pico ampere (pA).[1] `origin` defines at what stage coordinates the origin of the coordinates used in the rest of the ctl file lies. `x = 2500`, `y = 2500` and `stage` move the stage so that the writing starts 2500 µm to the right and up from the origin. `draw (Q-Start)` draws the pattern named “Q-Start” from the pat file. `+x = 200` and `stage` move the stage by 200 µm to the right relative to the current position. `draw (Q-Start)` draws the same pattern again and `end` tells ECP that the exposure is finished.

Press F5 or the button and then the button. The screen should now look like figure 2.2.

You can write these two files with a standalone version of ECP, that is one without a pattern generator.

2.5 Installing the Sample

Next, you should install the prepared sample in the sample chamber. Follow the appropriate instructions. When the chamber has been evacuated start taking images of the sample either the usual way with the SEM or use the “EBL/SEM Image…” menu in ECP, see section 10.1. Start with a rather low magnification to get an overview. Select the “EBL/Stage Control” menu (see chapter 8) in ECP. In the dialog that appears click on “Manual Mode”. Now you can move around with the number keys of the keypad. Select large or small steps from the “Move by:” area, the numbers are step length in micrometers. Move to the lower left corner of the sample. Try not to move over the interior of the sample, as every scan will lead to visible artifacts. Focus the beam on the sample and measure the current. Set up the SEM so

[1] This sets the current ECP uses for calculations. The current can be set on the SEM or e-beam only.

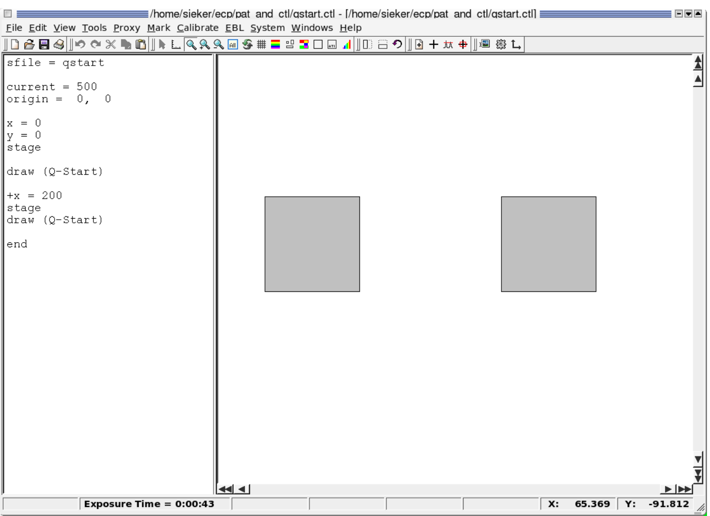


Figure 2.2: Editing a control file

that it accepts external scan input. Consult the site specific instructions on how to do this. It can often be done with the SEM software or with a special switch or the SEM may be ready without any further setup.

2.6 Exposing the Sample

Select the menu item "Tools/Map exposure origin". This replaces the origin value in the control file with the current stage position. This position can be seen in the upper right corner of the stage control dialog. . The fourth line of the ctl file should now look like `origin =`

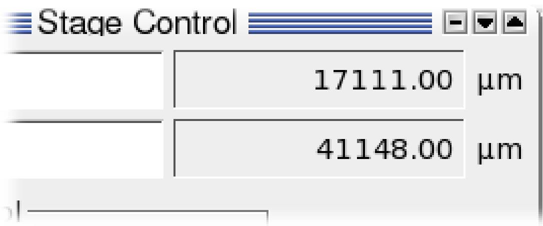


Figure 2.3: The current stage position as starting point for the exposure

`17110, 41148` if you take figure 2.3 as an example.

Now everything is ready for the first exposure. Press the ("Expose") button or select the "EBL/Expose" menu item. Click "Yes" to the "Do you really want to expose?" question and the exposure will start. During the exposure the button will be replaced with a ("Stop Exposure") button and everything else is deactivated. When the button reappears and the other controls are active again the exposure has finished.

2.7 Shutdown

If you don't want to do another exposure quit the ECP program and switch of the pattern generator. Please note D.2 for restarting ECP.

2.8 Developing the Sample

Unmount the sample and develop it. It should now contain two squares 20 µm to a side 2.5 mm to the right and to the top of the area used for focusing. Squares this small may be hard to find under a microscope.

3 Menus and Toolbars

This chapter explains each menu option of the ECP program and its function. If the program is used in a stand-alone configuration (no pattern generator attached), the menus for hardware control will either be hidden or provide only dummy functionality. The menus will be slightly different depending on the file type (GDSII, pattern, . . .). Many menu options lead to dialog boxes that provide additional options and functionality. The most commonly used menu options also have corresponding toolbar buttons. For menu options that invoke a dialog box or which have a corresponding toolbar button, there is a cross-reference to the detailed explanation that appears in the relevant chapter.

3.1 Invoking Menus

There are several ways to invoke a specific action. As an example, you can open a file by:

- Open the “File” menu and then select “Open. . . ”
- Click on the “Open” button in the file toolbar
- Use the key shortcut Ctrl+O
- Open the “File”menu with the key Alt+F[1], then select “Open. . . ” by moving the selection with the cursor keys and pressing “Enter”.
- Open the “File” menu, then select “Open. . . ” by pressing O[1].

3.2 The Menubar

Most functions of ECP can be invoked through a menus item from the menubar.

3.2.1 File menu

Figure 3.1: The **File** menu

The **File** menu provides options known from many other computer programs and some more special ones. They are in detail:

New. . . Click this menu option (or press Ctrl+N) to create a new layout. ECP will ask which type (PAT, CTL, GDS or OVL) of file should be created. The name of the new layout will be “new-file” plus a file extension corresponding to the file type (e.g. new-file.pat for a pattern file).

[1]Use the letters that are underlined in ECP for this kind of control

Open. . . Click this menu option (or press Ctrl+O) to open existing layouts. The File Open Dialog is invoked through which an existing file can be selected.

Open Recent This sub-menu provides shortcuts to the last files opened.

Save Click this menu option (or press Ctrl+S) to save the layout. When ECP detects that the file on disk has been modified after it has been loaded or saved, a message box appears, offering three possible actions: overwrite the file on disk with the one in ECP, revert to the version on disk or cancel and do nothing. To keep both versions use "Save As. . . " and select another filename.

Save As. . . Saves the layout under a different name. ECP will automatically add the correct file extension.

Close Click this menu option to close the active file. If the file has been modified since the last save, a dialog appears, asking if the file should be saved ("Save Now"), not be closed after all ("Cancel") or the changes be discarded ("Close Anyway").

Close All This closes all open files, asking what to do when a file contains unsaved changes.

Properties. . . Display some properties of the current file. What is displayed depends on the type of file.

Export Converts the layout to a different format and saves it. Exporting as an image file is always possible. If you select "Image", the Export Image dialog will be opened and ask for a filename and an image format. Exporting it to a ctl and pat file with this menu item is the usual thing to with a gds file, see chapter 6.

Print. . . Opens the Print dialog. First choose the printer, orientation etc., then select if the text or the drawing should be printed.

Quit Closes the ECP program. This also closes all open layouts, asking what to do with unsaved changes.

3.2.2 Edit menu

The **Edit** menu is only visible when a file is open. It provides the following options:

Undo Undoes the last change to the layout.

Redo Re-does the last undone change to the layout.

Cut Cuts the selected text (PAT, CTL or OVL files) or the selected elements (GDS files) and puts them on the clipboard.

Copy Copies the selected text (PAT, CTL or OVL files) or the selected elements (GDS files) to the clipboard.

Paste Inserts the contents of the clipboard at the current cursor position.

Delete Deletes the selected text (PAT, CTL or OVL files) or the selected elements (GDS files).

Select All Selects the whole text in the editor.

Find. . . Opens a find dialog to find a string in a text file.

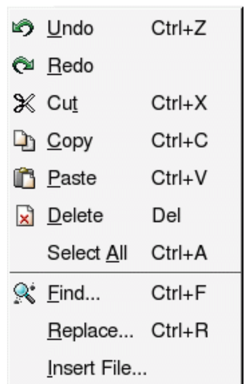


Figure 3.2: The **Edit** menu

Replace. . . Opens a replace dialog that lets you replace one string with another. “Replace all” switches off prompting and “Replace” asks for confirmation every time it finds the search string.

Rename Structure. . . Renames the selected structure in a GDSII file

Find Structure. . . Finds a structure in a GDSII file with the exact name given.

Insert File. . . Inserts a file at the current position. This works only in text files like .pat or .ctl.

N. B. Only changes that have been made manually in the editor can be undone with “Undo”. Operations from, for example, the Tools menu can not be undone this way.

3.2.3 View menu

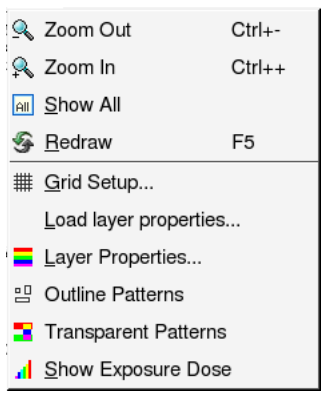


Figure 3.3: The **View** menu

The **View** menu is only visible when a file is open. It provides the following options:

Zoom Out Zooms out by a factor of 2.

Zoom In Zooms in by a factor of 2.

View all Changes the scale of the display so the complete layout is shown.

Redraw Refreshes the display to update any changes.

Grid Setup... Set if the origin, a frame representing the sample size and a grid with variable pitch are shown. The "Display Sample frame" works correctly only when sample corners have been stored in the stage control memory buttons (see section 8.2.1).

Load layer properties... Loads a description how the different layers of a layout should be drawn from a file.

Layer Properties... Shows the properties (style, fill and outline color, fill and show) of the layers in the current layout. For gds files, only the layers that are checked in the "Show" column of this control are shown.

Outline Patterns or **Fill Patterns** Toggles if a pattern is drawn only as an outline or if it is filled.

Transparent Patterns or **Opaque Patterns** Toggles how areas are drawn where patterns overlap: in opaque mode the patterns are drawn one after the other, possibly obscuring each other. In transparent mode overlapping areas become visible. How the pattern

```
D intersecting
RECT 10000, 10000, 30000, 30000
RECT 20000, 20000, 35000 ,25000
RECT 15000, 15000, 40000, 40000
END
```

is rendered in these modes can be seen in figures 3.4 and 3.5.

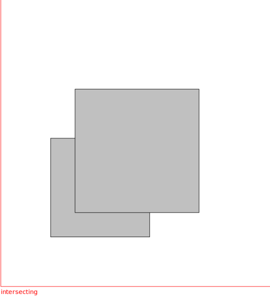


Figure 3.4: Three overlapping rectangles, drawn in opaque mode, one is completely obscured

No Frame or **Frame Field** Hides or draws a frame around every pattern drawn in a ctl file.

Display Pattern Names or **Hide Pattern Names** Shows or hides the names of patterns below their frame in a ctl file. In "Display Pattern Names" mode patterns are always drawn with a frame, even if "No Frame" is selected above.

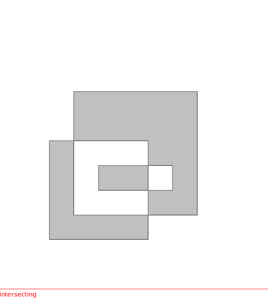


Figure 3.5: The same three rectangles as in figure 3.4, drawn in transparent mode

Show Exposure Dose or **Show Layout** "Show Exposure Dose " paints every structure with a color corresponding to the dose the area will receive during exposure. This also opens a small "Dose Scale" window showing what color represents what dose. If selected in a pat file a dialog asks for the writing current that should be used for the calculation. (The current is usually set in the ctl file.) "Show Layout" switches back to a view defined by the switches above.

Increase Recursion In a GDS file the view of a structure the sub-structures referenced (included) are shown only as frames. Selecting "Increase Recursion" shows the content of these frames, sub-structures below this are still drawn as frames until "Increase Recursion" is selected again.

Decrease Recursion Opposite of "Increase Recursion". Hides the lowest visible hierarchy layer of sub-structures in frames.

3.2.4 Tools menu

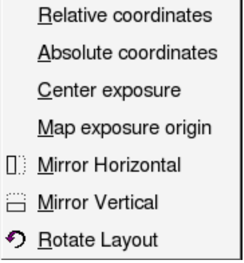


Figure 3.6: The **Tools** menu for a ctl file.

How the tools menu looks depends on the type of the active file

Control files

For ctl files the tools menu holds these special items:

Relative coordinates This function replaces all but the first appearance of absolute coordinates like `x = X` or `y = Y` with the appropriate relative coordinates like `+x = XX` or

`+y = YY`. This can be useful when overlaying different ctl files (see section 5.5), as all shapes can be moved by changing just one set of coordinates.

Absolute coordinates Opposite of the above. Replaces relative coordinates like `+x = XX` outside of `for` loops with absolute values `x = X`.

Center exposure Uses corner positions of the sample stored in the stage control dialog (see section 8.2.1) to place the exposure in the middle of the sample using the `origin` command. When at least two opposing corners positions of the sample are stored in the memory buttons, Center exposure calculates the size of the sample and of the exposure area. It then replaces the coordinates of an `origin` command with positions that place the center of the exposure area in the center of the sample.

Map exposure origin Sets the origin of the exposure to the current stage coordinates.

Pattern and Control Files

These items apply to pat and ctl files:

Mirror Horizontal Flips the layout, that is the file is modified so that patterns are mirrored on a vertical axis (moved horizontally).

Mirror Vertical Flops the layout, mirrors on a horizontal axis, moving patterns vertically.

Rotate Layout/Pattern The layout is rotated 90° counterclockwise (by $\pi/2$).

N. B. These menu items, when called for a ctl file, modify the pat files used in that file on disk, calling the appropriate mirror or rotate functions on them as well as changing the stage move commands in the ctl file itself. It is best not to have this pat file open in ECP when using these actions on a ctl file. **Caution:** Saving a pat file after ECP has changed it with these functions can make a ctl file unusable. Using one pat file with different ctl files can be dangerous as well. On rotating one ctl file, the other is no longer usable.

Pattern Files

For a pat file you get these special items

Scale Pattern... Multiplies all positions in the pattern file by the number entered in the dialog.

Scale Dose... This multiplies the dwell time and consequently the exposure dose of the pattern by the factor entered in the dialog.

PSL Console... Opens the Psl Console where patterns can be generated using psl, a simple scripting language. See section 4.9.

N. B. The pattern may become invalid if it is scaled beyond the borders of the exposure field. The scaled pattern is rounded to integer pixel values, for example scaling by 0.1 and then by 10 may slightly change a pattern. Scaling a Pattern or the dose by a factor of 0 is allowed but destroys information.

N. B. Most of these tools modify the pattern file irreversibly (un-undo-ably). They also remove comment lines from the files.

GDSII Files

GDSII files have a tool menu of their own. These tools are

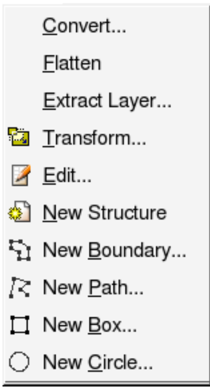


Figure 3.7: The **Tools** menu for a GDSII file.

Flatten Produces a new top level structure in the GDSII file that contains a copy of the shapes of the selected structure, but without the hierarchy.

Extract Layer... Produces a new top level structure with a copy of the layer selected in the extract layer dialog. Works only on flat(tened) structures.

Transform... Opens the Transform Menu dialog. With this dialog a structure can be mirrored, scaled, rotated and moved. Select a structure from the structure tree, put the desired values in the appropriate fields (or check the "Mirror" boxes) and click "Apply". Rotation is counterclockwise and the "Move" unit micrometers.

Edit... Opens the "Edit Reference" dialog. This is described in section 6.1

New Structure... Creates a new structure at top level.

New Boundary... Creates a new boundary in the active structure. Input a number of x- and y-values in the dialog and select a layer.

New Path... Like "New Boundary", only that the new shape is defined as a path.

New Box... Creates a new box in the selected structure. Input the coordinates of the lower left and upper right corners and the layer.

3.2.5 Mark menu

The **Mark** menu provides the following options:

New Mark... Creates a new mark in the mark editor.

Edit Mark... Opens an existing mark definition in the mark editor.

Test Signal... Performs a line scan across the mark and displays the scan signal.

Figure 3.8: The **Mark** menu

Detect Mark... Performs a mark detection using a specified mark definition.

Make Global... Copies a mark file from the mark directory (`~/.ecp/marks` or `C:\Documents and Settings\`*user name*`\ecp\marks`) to the working directory.

A detailed description of the marks and the editor is given in chapter 7.

3.2.6 Calibration menu

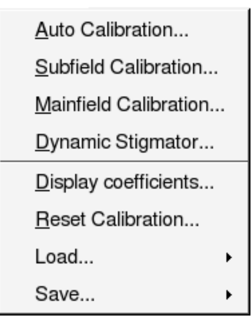


Figure 3.9: The **Calibrate** menu

These menu items contain functions used to set up calibrations.

Field Calibration This opens the "Field Calibration" dialog described in section 11.1. For a quick calibration center a mark in the picture using the stage control, then select "Field Calibration", pick a field size and a fitting mark description file and click on "Calibrate" and then "Save".

Working Distance Correction... This opens the "Working Distance Correction" dialog. This can be used to automatically correct the focus when the sample is tilted slightly. See section 11.2 for details.

Display coefficients... This displays the results of the last field calibration.

Reset calibration This resets the calibration parameters to Scale x = Scale y = 1 and Rot x = Rot y = 0.

Load/Field Calibration. . . Opens a dialog to load data from a previous calibration. This is usually a good starting point for a precise calibration.

Save/Field Calibration. . . Opens a dialog to save the current calibration coefficients.

See chapter 11 for more details.

3.2.7 EBL menu

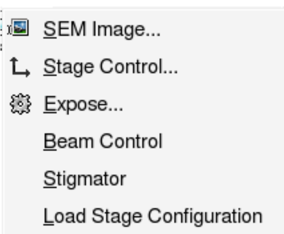


Figure 3.10: The **EBL** menu

This menu provides some general functions.

SEM Image. . . Acquires and displays SEM images, either as a continuous scan or as single scans. See section 10.1.

Stage Control. . . Activates the manual stage control (see chapter 8), that allows you to position the sample with respect to the e-beam by entering desired x and y values for the stage position, by pointing to a desired position of the sample image or by manually moving the stage.

Expose. . . This important menu item starts an exposure with the currently loaded ctl file.

Beam Control. . . Allows you to turn the electron beam on or off.

Stigmator. . . Check and set the SEM's x and y stigmators.

Load Stage Configuration. . . Loads a configuration file describing the stage used. See section 8.3.

3.2.8 System menu

Load XeDraw2 kernel. . . Loads the selected file as the kernel into the dsp of the XeDraw2.

Reset XeDraw2 Resets the XeDraw2.

Set XeDraw2 Clock. . . Sets the clock of the XeDraw2 in kHz. The Maximum clock frequency of the pattern generator is 10 MHz, but a lower limit can be set in the settings. The clock can be set in 1 kHz steps.

Settings. . . This opens the settings dialog described in chapter 12.

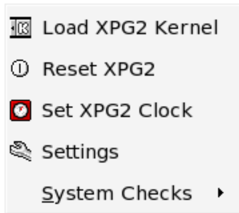


Figure 3.11: The **System** menu

System Checks/DAC Test... Opens a dialog where you can enter arbitrary values and send them through the DACs to test them.

System Checks/Focus Test... Opens a dialog that can be used to set up the focus. To set the focus use a picture with a high contrast, press "start" and change the focus setting until the steps in the line scans shown do not get any steeper.

3.2.9 Windows menu

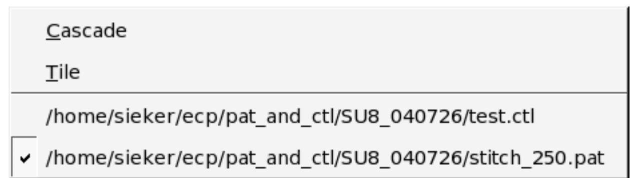


Figure 3.12: The **Windows** menu with two example files open.

ECP can have many files open at the same time. Every open file is associated with a window. How these windows are arranged is controlled with the windows menu.

Cascade Puts all windows in the upper left corner, cascaded down and to the right.

Tile Shows all windows side by side.

Window List Below the menu separator is a list of all open files. Selecting one makes it the active window.

The main window has two main modes. Either one window is maximized, that means that only this file is visible. This is usually the most convenient mode. When you select "Cascade" or "Tile" from the "Windows" menu or "Restore" or "Minimize" from the window menu (left of the "File" menu) ECP changes to the multi window mode. Here you can move around and resize the windows. This can be useful to compare to files or to copy parts of one file into another. To go from the many windows mode to the single window view click the maximize button of one window.

Figure 3.13: The **Help** menu

3.2.10 Help menu

The **Help** menu provides the following options:

Online Help Starts a browser for the online help system.

About ECP Displays general information about the program, including the version number and the date and time of compilation.

About Qt Displays information about Qt, the cross platform toolkit which is used for the user interface of ECP.

What's this This is part of the online help system. After selecting "What's this", click a tool-button to get more detailed information about the corresponding action.

3.3 Context Menus

In many situations a context menu can be activated right under the mouse by using a right mouse click. A context menu offers a range of functions that are appropriate for the current situation. For example the context menu for the text edit area of ctl or pat files is very similar to the "Edit" menu. The context menus for the structure tree of a GDSII file or the pattern selection area (top left) of a pat file are very similar to the "Tools" menus. For a pat file the operations like "mirror" or "rotate" apply only to the selected structure.

The view (right) area has no context menu. A right click here zooms out.

3.4 Toolbars

The toolbars offer fast access to often used actions. Toolbar buttons are grouped. These groups are called docks and have handles at the left side. You can drag a dock by its handle and move it to a different position in the toolbar area or to a different window border. Docks can also be dragged out of the toolbar area entirely and made into a stand-alone tool dock windows. To hide or restore a tool dock window, right click in an empty part of the tool area or on a dock handle, then click the name of the tool dock window you wish to hide or restore.

If you hold the mouse cursor over a toolbutton for a certain time, a yellow box with a the name of the button will pop up. Additional information is displayed in the status bar at the bottom of the main window. A more detailed description of the action corresponding to the button can be obtained by selecting "What's This" from the help menu and then clicking on the button.

Some tools can be reached only via a tool button.

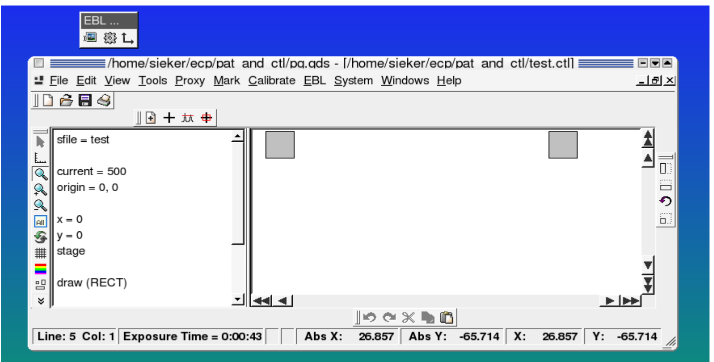


Figure 3.14: The tool docks can be arranged around the main window area or torn off.

3.4.1 Mode Selection

The View Settings dock has three buttons on the left (or on top), the “Select Mode“, “Measure Mode” and “Zoom Mode” buttons. Exactly one of these modes is always active.

In select mode you can click on an exposure field in a pat file and the pattern is selected in the editor window.

In measure mode a small measurement window appears and when you drag open a frame with the mouse button pressed the measure window shows the frame size.

Zoom mode is the default setting. When you drag open a frame in the view area ECP zooms into that area.[2]

3.4.2 Toolbar Buttons for GDSII Files

The “View Settings” dock shows a number with up and down arrows beside it. This is the number of lower hierarchy levels shown. Changing this number hides or shows details described in lower hierarchy levels. Increasing this number by one is identical to selecting “Increase Recursion” from the “View” menu; decreasing it by one identical to selecting “Decrease Recursion”.

The “Transform” for GDSII files has some special buttons:

- Align Grid left
- Align Grid right
- Align Grid top
- Align Grid bottom
- Snap

[2]More precisely ecp shows the smallest area that completely contains the selected area while keeping the aspect ratio.

4 Pattern Definition

ECP uses its own description language for exposure patterns. This language is human readable and writable and the patterns are stored in ASCII format. This means that you can use your favorite text editor or an external scripting languages to create patterns.

4.1 Writing a Pattern

The content of one pattern file is called a pattern. Each pattern can contain many structures, each with a unique name, one structure can be composed of many shapes. Patterns are defined by simple text commands. .

`D` *structure_name*
`I` *increment*
`C` *dwell time*
shape 1
...
shape n
`END`

`D` *next_name* ...
`END`

You can use comments in a pat file, a comment line begins with a semicolon ("`;`") or a hash sign ("`#`").

> **N. B.** Comments can be added at the end of a line containing a shape, but this comment will be deleted by the commands from the "Tools"-menu.

Note that the origin is at the bottom left corner.

4.2 Structure Name

The structure name follows the initial "`D`" at the beginning of the structure definition, for example `D my_structure`. The structure name must consists only of letters, numbers, a hyphen ("-") and an underscore ("_"). Structure names are case sensitive, "example" and "Example" are both valid structure names specifying *different* structures. Just numbers like "42" are also valid. It is best to use descriptive structure names and to use uppercase and lowercase letters in a consistent way.

4.3 Increment

The increment defines the step size of the e-beam in units of pixels. This increment can be changed several times in one structure. The increment is specified by the letter `I` (uppercase i), followed by the parameter. For example: `I 2`

Usually accepted values are between 1 and 128.

The increment defines how many pixels are written, an increment of n means that only every n^{th} dot of a line is written and the pattern generator steps over $n-1$ positions and only every n^{th} line is drawn. When writing an area with increment n, $1/n^2$ as many points are written as with an increment of 1, for a line, $1/n$ as many points are written. For the different types of dots the increment value has no meaning.

4.4 Dwell Time

The dwell time is the time spent by the the e-beam at every pixel of the structure. After the dwell time has passed, the e-beam moves on to the next point with the defined increment. To specify the dwell time type "`C`" followed by the dwell time in nanoseconds. For example: `C 100`

The dwell time must be higher than the inverse clock frequency (specified in the Settings Menu—explained in chapter 12). For example if the clock is set to 5 MHz, the dwell time must be larger than 200 ns. The dwell time can be changed several times in one structure.

4.5 Special Commands

Apart from `C` and `I` there are four more special commands in pat files that don't define shapes: `PROCESS`, `FSIZE`, `UNIT` and `OFFSET`.

4.5.1 PROCESS

The `PROCESS` shape is intended for processing a sample with an ion beam. This command allows for the beam to scan over the shape more than once.

The exposure of the shape still obeys the `C` and `I` parameters. `C` should be set to low and `I` to high values if the shape should be processed over many times.

One `PROCESS` directive is valid until it is overridden by another `PROCESS` or to the end of the current structure. To reset the processing use "`PROCESS 1`".

There are three ways the length of the processing can be set: as a fixed number of scans, as a time, or as a depth or height to which the shape should be milled. Which mode should be used is determined by the unit used (for example `PROCESS 20 s` specifies a time and `PROCESS 10e-6 m` a depth.

4.5.2 FSIZE

The field size to be used can be set in a pat-file as well. Use the command `FSIZE` followed by a number and a length unit. A field size set here always overrides those set with `fsize` (5.2.22) and `fscale` (5.2.21) in the ctl-file where the structure is used.

The field size must be set inside a structure, after the `D` definition of the structure name and before the first shape.

Several length units are recognized, among them meter, millimeter, micrometer and nanometer, written out or with the common unit symbols. For example the strings "1e-4 metre"[1], "0.1

[1] The "metre" and "meter" spelling are both recogized.

millimeter", "100 μm" and "100 um"[2] all specify the same length. New units can be added on the user's request.

> **N. B.** Setting the field size in the pattern file instead of in in the pattern file has a few disadvatages.

Setting the field size in the pattern file overrides any field size settings in the pattern file. Especially the field size variation described in section 5.2.14 cannot be used when the field size is set in the pattern file.

Another problem may appear when chip marks are used together with setting of the field size in the pattern file. While the scaling and rotation corrections work without problems, shift compensation may be incorrect under some circumstances. During the chip mark correction the shift is calculated in pixels. When the field size is changed from within a patern file, the length of one pixel may change and the offset may be incrorrect. To avoid this problem, the correct field size should be set in the control file before the chip mark correction, using the `fsize` command. See section 5.2.22.

4.5.3 UNIT

With the `UNIT` command shapes in a pat-file can be defined in a given length unit, not in pixels. To use the `UNIT` command the field size must also be given before the shapes.

For example after

```
FSIZE 100 micrometer
UNIT 1 micrometre
```

shapes are defined in micrometers up to the end of the structure. In this case a `RECT 0,0, 10,10` would define a square with a side length of 10 μm.

When specifying just a unit the "1" can be dropped, "`UNIT micrometer`" is the same as "`UNIT 1 micrometer`".

The unit or standard length defined with `UNIT` can be a multiple of the base units. This can be used for example when the unit is set to the field size or to one hundredth of the field size. Then shapes are defined as fractions of the field size, or in per cent of the field size. As an example

```
FSIZE 50 micrometer
UNIT 50 micrometer
RECT 0, 0, 0.5, 0.5
```

defines a square half the field size to a side.

4.5.4 OFFSET

Another command that controls how subsequent shapes are interpreted is "`OFFSET`". It defines the offset of the origin (point 0,0) for the following shapes with respect to the standard origin in the lower left corner.

When no `UNIT` is defined, `ORIGIN 25000, 25000` sets the origin to the middle of a 50000 pixel writing field. It is than easy to define shapes symmetrical to this center.

Like other shapes `UNIT` uses the length unit if one was defined earlier.

There are two special codes, `OFFSET center` and `OFFSET reset`. With a "reset" the origin is set back to the lower left corner. "Center" sets the offset to the center of the writing field.

[2] The Greek letter μ can be replaced by "u" or "mu".

4.6 Shapes

The main part of a structure definition is a number of lines defining one shape each. A shape definition starts with a mnemonic for the shape.

4.6.1 RECT

The shape "`RECT`" is a simple rectangle. The syntax is "`RECT` X1, Y1, X2, Y2".
(X1, Y1) is the bottom left corner, (X2, Y2) is the top right corner of the rectangle.

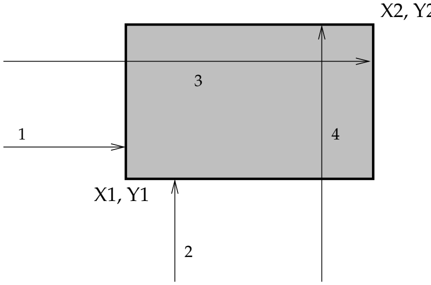


Figure 4.1: Defining a rectangle shape. The arrows indicate the order the coordinates have to be entered.

The numbered arrows in figure 4.1 are a mnemonic for the order of the parameters. A rectangle that ECP accepts must have $X1 < X2$ and $Y1 < Y2$.

4.6.2 LINE

The "`LINE`" shape is a straight line. The syntax is similar to that of a rectangle, except that a line joins the two points. The syntax is "`LINE` X1, Y1, X2, Y2".

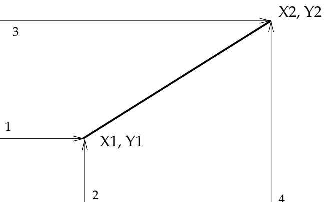


Figure 4.2: Defining a line shape

The line shape poses no constraints on its coordinates, it must only fit into the exposure field. If, for example, $X2 < X1$ and $Y2 = Y1$ the line is drawn horizontally from right to left. Only $X1 = X2$ and $Y1 = Y2$ is not allowed. Use a dot shape for this, see section 4.6.5.

4.6.3 XPOLY

Shape "`XPOLY`" is a trapezoid with two horizontal sides (parallel to the x-axis). The syntax is "`XPOLY` X1, Y1, X2, X3, X4, Y2".

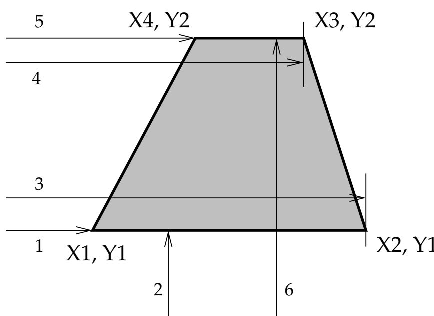


Figure 4.3: Defining a trapezoid with horizontal sides using XPOLY. The arrows indicate the order the coordinates have to be entered.

(X1, Y1) is the bottom left vertex of the trapezoid. Then, the second and third x-coordinates are for the bottom right and top right vertices, respectively. (X4, Y2) is the top left vertex of the trapezoid. The order of the coordinates is also indicated in figure 4.3.

The constraints for a XPOLY shape are $Y2 > Y1$, $X2 \geq X1$ and $X4 \leq X3$. You can use $X1 = X2$ or $X3 = X4$ to define a triangle with a horizontal sides on top or at the bottom, respectively.

4.6.4 YPOLY

Shape “YPOLY” is, in analogy to “XPOLY” (4.6.3), a trapezoid with two vertical sides (parallel to the y-axis).

The syntax is “YPOLY X1, Y1, Y2, Y3, X2, Y4”.

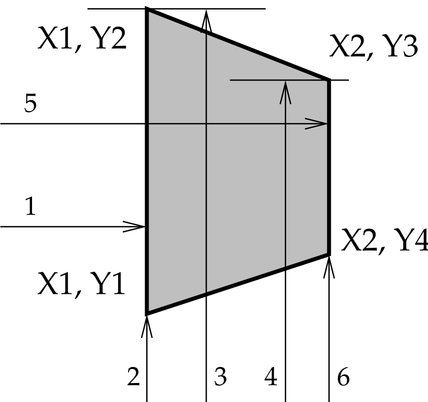


Figure 4.4: Defining a trapezoid with vertical sides using YPOLY. The arrows indicate the order the coordinates have to be entered.

(X1, Y1) is the bottom left vertex of the polygon. Y2 and Y3 are for the top left and top right vertices respectively. (X2, Y4) is the bottom right vertex of the trapezoid. The order of the coordinates is also indicated in figure 4.4.

The constraints for a `YPOLY` shape are $X2 > X1$, $Y2 \geq Y1$ and $Y4 \leq Y3$. You can use $Y1 = Y2$ or $Y3 = Y4$ to define a triangle with a vertical sides at the left or at the right, respectively.

4.6.5 DOT

Shape "`DOT`" is just a dot. The syntax is "`DOT` X, Y".

(X, Y) are the coordinates of the dot.

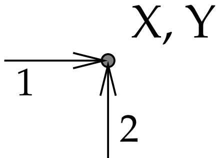


Figure 4.5: Dot shape

4.6.6 CIRCLE

Shape "`CIRCLE`" is a filled circle with center (X, Y) and radius R. The syntax is "`CIRCLE` X, Y, R".

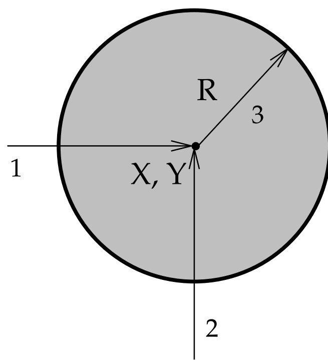


Figure 4.6: Filled circle

4.6.7 RING

The "`RING`" shape defines a ring or a ring segment. To define a closed ring, use "`RING` X, Y, R, W".

Here (X,Y) is the center of the Ring, R the outer radius and W the width of the ring, the outer radius minus the inner radius. The width has to be less than the radius to leave a small hole in the center (and > 0).

To create a ring segment add two more arguments, the start and end angle: `RING` X, Y, R, W, S, E.

These angles are specified in degrees, start with 0 to the right and increase counterclockwise, like a mathematical coordinate system. The ring shown in figure 4.7 has a start angle of S = 45 and an end angle of E = 315. These angles must be ≥ 0 and E must be greater than S and can be greater than 360. Setting W to R - 1 basically produces a circle segments.

4.6.8 RDOT

Shape "`RDOT`" is the same as "`DOT`", 4.6.5. There is only one more parameter, which is the number of repetitions of the dot. The syntax becomes "`RDOT` X, Y, REP".

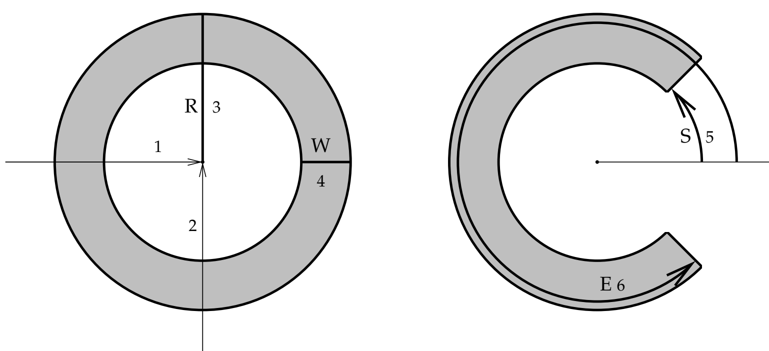


Figure 4.7: Left: A Ring is defined with four parameters. Right: Adding start and end angles gives a ring segment.

Basically a single dot is exposed for a time of REP $\times$ dwell time. The maximum value for the repetition is 65535, that is $2^{16} - 1$. If you need to expose one spot for even longer times write down the specific line repeatedly, either by hand or with the Psl console, described in section 4.9.

4.6.9 MSDOT

Shape "`MSDOT`" is similar to "`DOT`", 4.6.5. It is used to expose single dots for long periods of time. The third argument is the exposure time in milliseconds.

"`MSDOT` X, Y, T".

The time t is specified as a 32-bit value, that is, it can take values from 1 to 4294967295. As 4294967295 ms is equal to almost 50 days, this should be enough for any application beyond lithography.

4.6.10 SQUARENUT

The "`SQUARENUT`" shape defines a rectangle with a circular hole, like a square nut that can be screwed onto a bolt. The parameters are "`SQUARENUT` X, Y, R, W, H".

Here (X,Y) is the center of the shape, R the radius of the hole, W the width of the rectangle and H the height of the rectangle. The radius can be wider than W or H, but should be smaller than the distance from the center of the rectangle to a corner. This distance is $\sqrt{W/2 + H/2}$. If R is set to a higher value ECP reduces the radius to this value.

4.6.11 HEXNUT

The "`HEXNUT`" shape defines hexagon with a circular hole, like a more common six sided nut. The parameters are "`HEXNUT` X, Y, O, H".

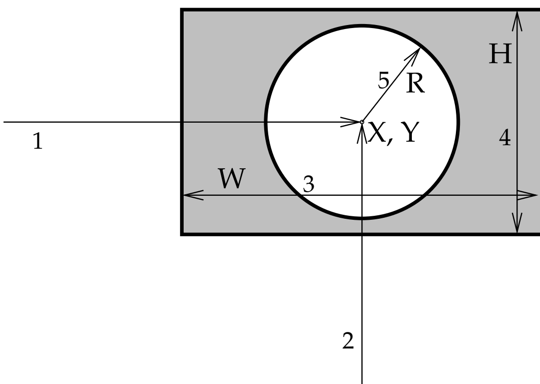


Figure 4.8: A square nut shape with W > H and H > 2 R

(X,Y) is the center of the shape, O is the outer radius, that is the distance from the center to a corner. H is the radius of the hole. The geometry of a regular hexagon is so, that the outer radius O is the same as the length of one side of the hexagon.

The hole radius H must be smaller then O, but can be larger then the inner radius I, the distance from the center to the middle point of an edge. ($I = \sqrt{3}/2 \cdot H \approx 0.8660 \cdot H$) When R is larger then I only six unconnected parts in the corners will be exposed.

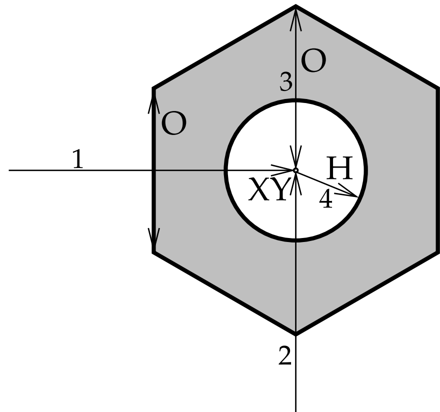


Figure 4.9: A six sided nut shape, created with HEXNUT. Note the vertical sides and the corners on top and at the bottom.

The hexagon of a HEXNUT has two vertical edges and two corners at top and bottom, on a vertical line through the center.

4.6.12 HEXZUT

The HEXZUT is a shape much like the HEXNUT (4.6.11), but rotated by 90°, the name is derived from HEXNUT, with the letter N rotated by 90° as well, giving a Z.

The parameters are the same as for HEXNUT: "HEXZUT X, Y, O, H", with center position, outer radius and hole radius.

A HEXZUT has two horizontal sides and corners at the left and right.

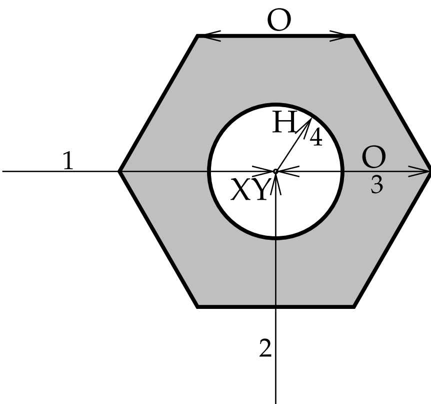


Figure 4.10: A six sided nut shape, rotated 90°, created with HEXZUT. Compare the position of the corners to figure 4.9.

The two hexagonal nut shapes work well with the hexagonal array codes, see sec:hexarray. With them a grid of un-exposed holes with sixfold rotational symmetry can be exposed easily.

N. B. For all shapes ECP also accepts numeric shape codes using a P and numbers that where used in earlier versions of the program. These P-numbers are explained in appendix B.1.

4.7 END

Finish the structure with the command END.

4.8 Repetition of Shapes

All shape can be repeated in x- and y-direction. There are two basic types how the repeated shapes are arranged, along a square grid or as a hexagonal tiling. For both arrays four more parameters are added after the shape.

For the "RING" shape (4.6.7) you have to use the version with the additional arguments, that is you must specify start and end angles of 0 and 360° for a whole ring. Otherwise the first two of the arguments are interpreted as angles, producing either an error or an undesired shape.

4.8.1 Rectangular Array

To arrange the shapes on a rectangular grid, four more numbers must be added after the shape data. They specify the period in x-direction, the period in y-direction, the number of repetitions in x-direction and the number of repetitions in y-direction.

To arrange shapes in a line specify an arbitrary period (e. g. 0) and a repeat count of 1 in the other direction.

It is also possible to use this syntax to repeatedly write a shape on the same place, for example to reduce the noise when writing lines. Set the period to 0 in both directions for this.

The following example pattern file specifies an array of circles:

```
D Circ
I 8
C 400
CIRCLE 12000, 6000, 3000, 12000, 9000, 3, 4
END
```

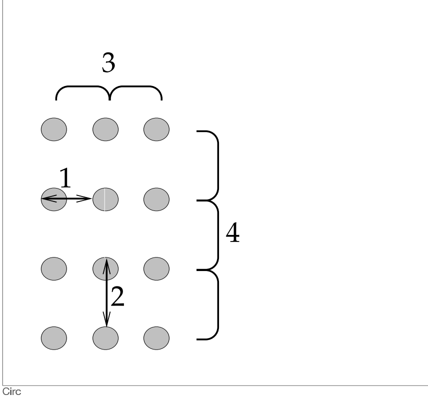


Figure 4.11: Array of circles created by adding four arguments to the definition of a single circle.

The numbers in figure 4.11 show the order in which these additional arguments must be entered.

4.8.2 Repetition of Structures

It is also possible to use the rectangular repetition syntax for all the shapes in one structure. For this you have to add the four repetition parameters after the structure name in the D-line.

Accordingly the following structure is the definition of a five by five grid of target shapes, not of a single one.

```
D targets, 10000,10000, 5,5
I 8
C 400
CIRCLE 5000, 5000, 500
RING 5000, 5000, 1500, 500
RING 5000, 5000, 2500, 500
RING 5000, 5000, 3500, 500
RING 5000, 5000, 4500, 500
END
```

The whole repeated structure must fit into the exposure field.

N. B. There has to be a comma separating the structure name from the array parameters.

4.8.3 Hexagonal Arrays

One shape can also be arranged in the pattern of a hexagonal tiling (tessellation). For this a keyword and three more numerical parameters are used. The keywords must be one of `hex1`, `hex2`, `hex4` and `hex5`. The number in the keyword is a "clock direction", it defines the direction from the lower left shape to the neighbor shape that is not on the rectangular grid. The two shapes defining this directions can be seen in figure 4.12.

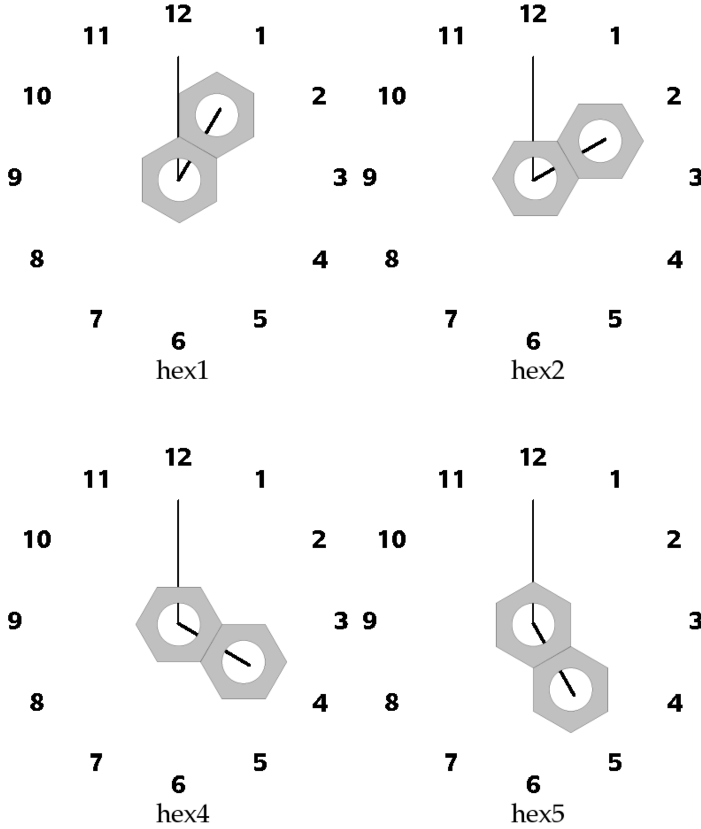


Figure 4.12: Clock faces defining the keywords for the arrangement of hexagonal shapes

The second parameter is the period P in one direction. The period in the other direction is calculated automatically. For hex1 and hex5 P is the distance to the next shape in a straight line above the first shape, for hex2 and hex4 the next shape straight to the right of the first shape. When using HEXNUT (for hex1 and hex5) or HEXZUT (for hex2 and hex4) shapes (see 4.6.11 and 4.6.12) using three times the outer radius (parameter three) exactly fills the plane, leaving just the circular holes. Mathematically the distance to the next shape in the other direction (to the right for hex1 and hex5, up for hex2 and hex4) is $^1/_{\sqrt{3}} \cdot P \approx 0.5773 \cdot P$. The position of the shapes are calculated as precisely as possible and the shapes exposed at the closest pixels positions.

The last two parameters are repeat counts in x- and y-direction.

The structure that is repeated $X \times Y$ times is *not* just the single shape defined before the

hex*N* keyword but two shapes, the one defined and its nearest neighbor in the direction indicated by the number in the array keyword. For a hexagonal array with repeat counts of X and Y the system will produce $2 \cdot X \cdot Y$ shapes.

Figure 4.13 shows the example arrays produced by the PAT file

```
D hex1
hexnut 5000, 5000, 3000, 1200, hex1, 9000, 8, 5
END

D hex2
hexzut 5000, 5000, 3000, 1200, hex2, 9000, 5, 8
END

D hex4
hexzut 5000, 7000, 3000, 1200, hex4, 9000, 5, 8
END

D hex5
hexnut 5000, 9000, 3000, 1200, hex5, 9000, 8, 5
END
```

The following features are marked in the figure:

- The position of the single shape defined in the PAT file is marked with an x.
- The length and direction of the period is marked with a black line.
- The two shapes that are repeated and that would be produced with a repeat count of 1 by 1 are marked in dark red.

As mentioned above the period is set to three times the outer radius of the hexagonal nut shape, $O = 3000$, $P =$9000 to completely tile the plane. The edges of all the hexagonal arrays are shaped so that an array can be stitched together with itself without leaving any open space.

While the hexagonal array mechanism works well with the HEXNUT or HEXZUT shapes, it can also be used with other shapes, like DOT or CIRCLE.

4.9 Psl Script

For complex and advanced structures, it is often tedious to manually type in the text commands. The Psl (Pattern scripting language) allows the use of for/next loops and definition of variables. Open the Psl Console by selecting the menu Tools/PSL Console. A session using this console is shown in figure 4.14

4.9.1 The Psl Console

The Psl Console dialog is divided in two areas. At the left is a tabbed text area. The first tab is the Script Editor. You can enter your script here or load a script from disk into this area using the Load button. The Save button saves this script to a file.

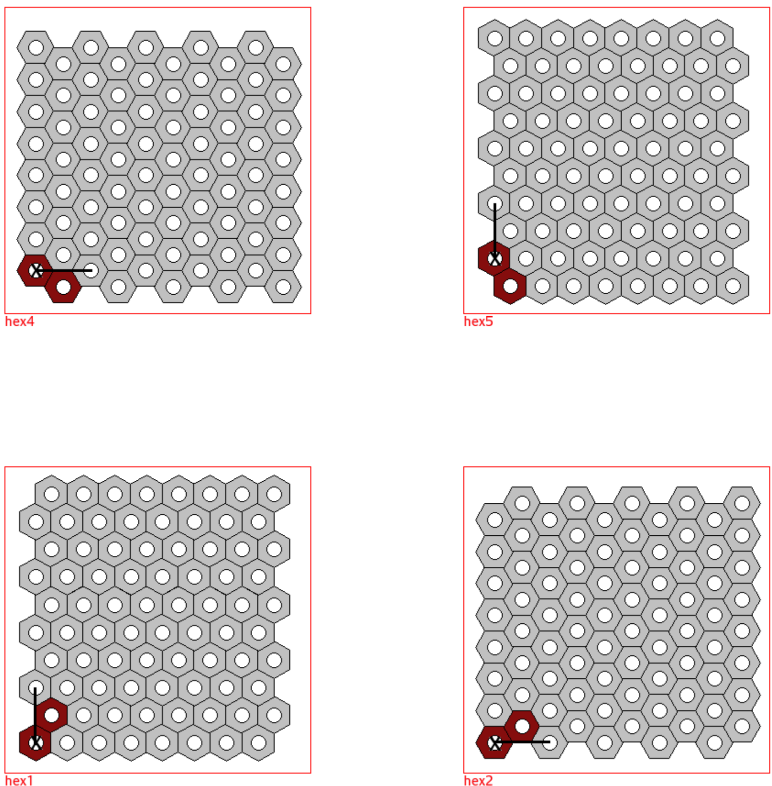


Figure 4.13: The four different hexagonal arrays, with certain features marked

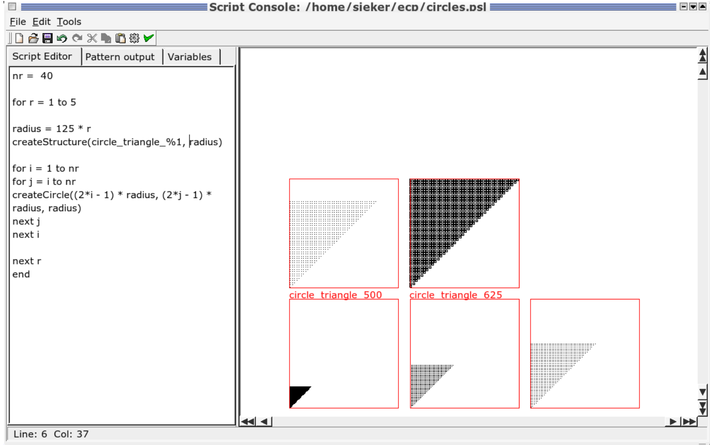


Figure 4.14: The Psl console after processing the script given as an example in this manual.

The second tab is the Pattern output. Clicking the Execute button runs the script in the Script Editor tab. If the script was valid the pattern text it creates is shown here. At the same time the pattern is drawn in the right part of the window. The last tab on the left, Variables, shows the variables used in the script and their values.

Clicking the check button inserts the pattern in the Pattern output tab into the current pat file. Closing the window with the close button or the window manager (X11) discards the pattern.

4.9.2 Structure of a Psl Script

The structure of a script looks as follows:

```
...
var = ...

createStructure(name)

... create shapes ...
...
...
end
```

4.9.3 Defining Variables

Any alphanumeric string not recognized by the script interpreter is automatically declared as a variable. For instance:

```
something
```

would create a variable named “something” with default value zero. Alternatively, use the assignment operator as follows:

```
length = 2000
```

4.9.4 createStructure

To create a Structure use `createStructure(`NAME`)` where NAME is the name of the structure. Spaces are not allowed. Use underscores instead. You can create more than one structure with one Psl script. Psl automatically adds an “`END`” after a structure when it finds another `createStructure` and at the end of a script.

Structure names can contain variables. This can be especially useful in for loops. List the variables after the name, separated by commas. Use a per cent sign (“%”) and the number of the variable in the list in the variable name. For example

```
var1 = 10
var2 = 20
createStructure(rectangle_%1_%2,var1,var2)
```

would create a structure named `rectangle_10_20`.

4.9.5 Creating Shapes

Similar to the `RECT`, `LINE` ... shape codes, you can use
`createRect`(X1, Y1, X2, Y2)
`createLine`(X1, Y1, X2, Y2)
`createXPoly`(X1, Y1, X2, X3 X4, Y2)
`createYPoly`(X1, Y1, Y2, Y3, X2, Y4)
`createDot`(X, Y)
`createCircle`(X, Y, R)
`createRing`(X, Y, R, W)
`createRDot`(X, Y, REP)
Use these in the same way as the corresponding shape codes. You can even do repetitions and use optional start and end angles for `createRing`.

4.9.6 Setting Exposure Parameters

`setInc(i)` sets the increment for the following shapes to the value of i, `setDwellTime(c)` sets the dwell time to c. They work like `I` i and `C` c in a pat file.

4.9.7 for/next Loop

The main feature of Psl is the for/next loop. It can be used to create many similar but not necessarily identical shapes. The syntax is

```
for lvar = min to max
...
various commands,
even more for loops,
some using lvar,
...
next lvar
```

where lvar is a loop variable that must not have been used before. This lvar is stepped through every integer value from min to max; min and max can be variables or integer numbers. For every value of lvar all commands in the loop, that is between for and next are executed. These commands can use lvar as a variable. You can use other for loops inside a for loop.

N. B. The syntax `for(NAME,start,final[,increment]) ... next NAME` that was described in earlier versions of this manual does no longer work.

4.9.8 Arithmetic Commands

You can also use the following commands anywhere for arithmetics. Please enter real numbers and valid arguments only. The script does not support the use of complex numbers.

+ $a + b$ adds b to a. $+a$ is a

– $a - b$ subtracts b from a, $-b$ is the negative of b

***** $a \star b$ multiplies a by b

/ a/b divides a by b.

Pi Pi is a variable predifined to 3.14159265 $\approx \pi$

sin() $\sin(a)$ returns the sine of a

cos() $\cos(a)$ returns the cosine of a

tan() $\tan(a)$ returns the tangent of a

exp() $\exp(a)$ returns e^a ($e \approx$ 2.7182818)

log() $\log(a)$ returns the natural logarithm of a ($\log(e)$ =1)

sqrt() sqrt(a) returns the square root of a ($\sqrt{a}$, a must be $\geq$ 0)

round() round(a) returns the integer closest to a

digitize() digitize(a,b) returns the integer that is divisible by b without a remainder and closest to a. This is the same as b * round(a/b).

All trigonometric functions work on arguments in radians. For example use $\sin(Pi \star d/180)$ for the sine of d when d is in degrees.

Many of these functions return real (not integer) values. When Psl finds an non-integer values in a shape creation command like `createRect`, it writes the smallest integer not larger than the number found ("floor()") in the output

Due to rounding errors and the limited precision of computer calculations it can happen that a variable value becomes negative, even if an exact calculation gives a value of 0. To avoid problems in these cases, PSL defines the special variable "EPSILON". For small negative numbers x, $-$EPSILON $\leq x \leq 0$, the square root is defined to be $\sqrt{x} := 0$. The value of EPSILON can be changed to set the amount of latitude.

4.9.9 Example PSL Script

This psl script creates five triangles of circles.

```
nr =  40

for r = 1 to 5

radius = 125 * r
createStructure(circle_triangle_%1,radius)

for i = 1 to nr
for j = i to nr
createCircle((2*i - 1) * radius, (2*j - 1) * radius, radius)
next j
next i

next r
end
```

5 Control Files

Control files ar the file ECP uses to communicate with the pattern generator for exposing the patterns. They are called ctl-files because of the "`.ctl`" file extension.

5.1 Introduction

A ctl-file consist of lines with text commands that control the e-beam system. You set up a number of variables like the beam current and define which pattern will be written at what position.

The different files referenced in a ctl file, for examle pat files, must be in the same directory as the ctl file. The software does not search for files in different directories.

The commands in a ctl file are *not* case sensitive. You can use for example "ORIGIN", "Origin" or "origin".

5.2 Commands

Commands in a ctl file can be divided into four groups, functions, definitions, simple commands and control structures, each with its own syntax.

Functions take one or more arguments in parentheses ("(" and ")"). Most of these functions do something, like writing a pattern onto the sample. An example is `draw (shape_1)`.

Definitions consist of a name, an equal sign ("=") and one or more values . They set up variables, that influence how functions and simple commands work. For example `sfile = pat_file` defines what pat file is used by `draw` functions.

Simple commands take no arguments. An example is `stage`.

Control structures have their own syntax. They are `for`, `next` and `end`.

The commands that define a position or a length, like `origin` or `+x` expect it in units of micrometers. It can be specified with a precision of 1 nm.[1]

The current stage position is at the center of the writing field used by the draw commands `adraw`, `idraw`, `image_draw` and `draw`, for `gdraw` it is at the center of the first writing field created by the command.

A brief summary of all the commands is given in Appendix C. Below is a detailed description.

5.2.1 adraw

`adraw` is similar to the `draw` (5.2.13) and `idraw` (5.2.36) commands. It provides a way to use the variables of two nested for loops in patterns names, see 5.2.20. The command takes three arguments, a base name and the variable names used in the for loops. The base name and the variable values, written with three digits and padded with zeros, are concatenated and form the pattern name.

[1]An exception are chip marks, they are defined inside a writing field and specified in pixels.

For example the commands

```
for a = 1 to 5
x = 0
+y = 200
for b = 1 to 5
+x = 200
stage
adraw(pat_,a,b)
next b
next a
```

produce a five by five square of patterns with names `pat_001001`, `pat_001002`, ... to `pat_005005`. If a pattern, e. g. `pat_002003`, is not defined in the pat file, `adraw`, unlike `draw`, simply skips it without producing an error.

> **N. B.** In older versions of ECP adraw required two arguments, the two variables, and used pattern names like `004005`.

5.2.2 align

When global marks have been defined with `gmcoord` (5.2.30), `align` performs another global mark detection with the marks used before. When a `rmark` (5.2.43) was specified, `align` uses that mark to calculate an offset and align the exposure without correcting the rotation. See section 11.3.3.

5.2.3 autorot

`autorot` needs two saved coordinates in the "Sample Rotation" buttons (see section 8.2.3), it calculates the rotation of the sample and corrects it in the same way as `rot=xxx` does.

5.2.4 bid_off_time

The `bid_off_time = t` assignment sets the time the beam is blanked after the exposure of a single pixel in beam induced deposition mode to t milliseconds.

5.2.5 cmark

`cmark` performs a chip mark scan. It scans the area at the position given with `cmcoord` in the current writing field. The scan is performed by deflecting the writing beam to the areas of the marks, not by moving the stage, otherwise the mark detection is very similar to the global mark detection (5.2.29). Usually you perform a global mark detection first and perform a chip mark detection for maximum precision. The chip mark detection can be repeated for every writing field.

When giving an `cmark` command, the marks are also drawn in the view area showing where the beam will scan over the sample.

See also section 11.5.

5.2.6 cmcoord

Specifies the coordinates for one chip mark used by `cmark`. Repeat this command for every chip mark in the writing field.

These coordinates are given in pixels and use the same coordinate system as pat files. So 0, 0 is at the lower left corner and for a writing field of, for example, 50 000 pixels use `cmcoord = 25000, 25000` to define a mark at the center.

> **N. B.** Be careful when specifying these coordinates, as the "target" area, where delicate structures are to be written can be contaminated by a scan at a wrongly specified chip mark position.

5.2.7 cmfile

`cmfile` is used to set the name of the file you want to use for making a chip mark. This file must be in the directory of your ctl-file. You can use the menu item "Mark/Make global..." for this or you can use a mark used as a global mark.

> **N. B.** While using a large image mark or a long scan length for a cross mark can be useful, a chip mark should be as small as possible to reduce the risk of contaminating the writing area.

5.2.8 cmfilepos

The functions of the two commands `cmfile` and `cmpos` can be combined together with the `cmfilepos` command. This command should be used when there are differently shaped marks chip marks in different parts of an exposure field. For each mark a different mark deffinition can be used, by giving the mark description file and then the mark coordinates with this command. An example command could look like "`cmfilepos ll_chip, 4000, 4000`".

5.2.9 cmark_fpr

One more parameter than with the `cmfilepos` command can be given with `cmack_fpr`. The letters `fpr` stand for *file name*, *position* and *rotation*. The extra parameter gives a rotation angle, in degrees. Some marks, especially correlation marks, can be rotated.

This command can be used when there are chip marks that are rotated and "point" to the middle of the sample. In a typical situation there would be four chip marks in the four corners of a writing field, rotated by 0, 90°, 180°and 270°. For example, with a mark rotated by 90°the command could be `cmark_fpr ll_chip, 4000, 46000, 90`

5.2.10 current

`current` Sets the current for the dose calculation. The current is given in picoampere (pA). If no current is specified ECP uses a default value hard coded into the program, usually 1000 pA.

5.2.11 dose

With `dose = x` the exposure dose is set to x times the dose specified in the pat files. When for example the parameters (current, field size, dwell time and increment) in a structure would expose it with 100 µC/cm^2, using a `dose = 2` would expose this pattern with 200 µC/cm^2.

Using this command overwrites changes to the dose made with `dscale` (5.2.14), the automatic correction made by `fcup` (5.2.16) and earlier calls of `dose`. To use the real, measured value for the current, use `fcup` again after using `dose`.

Setting the dose to less than one reduces the dwell time. Please make sure that the dwell time does not become shorter than the minimum dwell time of 1/(maximum clock). For example, when the hardware is limited to 10 MHz, and a ctl file uses "`dose = 0.7071`", the dwell times in the pat file must all be "`C 142`" or longer, making the effective dwell time 142 ns $\times$ 0.7071 = 100 ns, that is the limit in the example.

The easiest way to avoid this problem is to design the patterns for the smallest dose used and only use `dose` with values ≥ 1.

5.2.12 dose_table

`dose_table =` dose-table-file loads a file with a dose table. This maps the gray levels[2] of an image to exposure doses. The file must be in the directory of the ctl file. ECP adds a `.dtb` to the base name given in the command.

Each line in a dose table file (apart from empty lines, which are ignored) must contain two numeric values, firstly the gray level, that is a number between 0 and 255, where 0 stands for black and 255 for white, and secondly the dose that should be used to expose that gray level, in microcoulomb per square centimeter (μC cm^{-2}).

Dose values not specified in the dose table are interpolated linearly, if a lower and a higher value are specified. Thus a complete dose table file may look like this:

```
0 0
10 0
11 50
254 139
255 140
```

Values between 1 and 254 are interpolated, for example a pixel with a gray level of 100 would be exposed with 32.6 μC cm^{-2}. For resists with a known nonlinear relationship of exposure dose to resist thickness intermediate values need to be filled in to get a well defined resist relief.

The minimum dose to be used should be chosen with some care. ECP calculates the increment so that the smalles dose can be exposed with the minimum dwell time. While a dose table like

```
0 0
255 255
```

is valid, this would typically use a high increment so that pixels with a gray level of 1 could be exposed with 1 μC cm^{-2}. When a higher minimum dose is acceptable, this minimum dose should be specified in the dose table, with a dose of 0 explicitly specified for the graylevel directly below. Note the jump from dose 0 for graylevel 10 to 50 μC cm^{-2} for gray level 11 in the first example.

For black-and-white images without intermediat gray levels a dose table like this could be used:

[2] called "value" in "HSV" (*h*ue, *s*aturation, *v*alue) color definitions

```
0 0
127 0
128 100
255 100
```

5.2.13 draw

The syntax for the `draw` command is:
`draw (`pattern-name`)`
or
`draw (`pattern-name`, `increment`, `dwell time`)`
This command exposes the pattern on the sample. Before you can use `draw`, you have to specify from which pat file the pattern is taken with `sfile` (see 5.2.47). This second version of `draw` allows you to override the increment and dwell time specified in the pat-file.

5.2.14 dscale

To multiply the dose to use by a certain factor use `dscale = x`. This will expose the patterns below with the x-fold dose of those above the command. `dscale` commands work cumulatively, so for example using `dscale = 1.41421` and `dscale = 0.70711` later restores the dose to the original value.

With `dscale` two different kinds of test series are easy to do:

- To test the effects of dose changes, ctl file commands like this could be used:

  ```
  for foo = 1 to 10
  dscale = 1.05
  +x = 200
  stage
  draw (bar)
  next foo
  ```

- Writing a pattern with different sizes but a constant area dose can be done like this:

  ```
  for foo = 1 to 10
  fscale = 1.05
  dscale = 1.1025
  +x = 500
  stage
  draw (bar)
  next foo
  ```

  There are a few things to look out for. The different field sizes used should all be calibrated, see 5.2.21 and section 11.1. As the area dose scales with the inverse square of the field size, the correcting dose factor must be the square of the field size change. For a pattern of single pixel lines, the dose correction should be the same as that for the field size. For patterns of single dots, the dose stays the same. Patterns that combine several types of shapes, like area patterns and lines, can not be consistently scaled, as for example the lines get over-exposed when rectangles are exposed correctly.

The same problems that can arise with the `dose` command (5.2.11) apply to `dscale` as well. To avoid that the dwell time becomes to short take care that the product of the value set with `dose` and all following `dscale` values stays greater or equal to one. This can be done easily by setting `dose = 1` before an dose series and only increasing the dose with `dscale` or by taking care that for every decrease in the dose there has been a corresponding increase. An example for this is

```
for foo = 1 to 10
+x = 500
stage
draw (bar)
fscale = 1.05
dscale = 1.1025
next foo
+y = 500
+x = 500
for baz = 1 to 10
-x = 500
stage
fscale = 0.952381
dscale = 0.9070295
draw (quux)
next baz
```

An example of both kinds of uses can be seen in figure 5.1

5.2.15 end

`end` is the last line read by ECP. Without this command the program does not work. Lines after the `end` are ingnored. This can be used for testing.

5.2.16 fcup

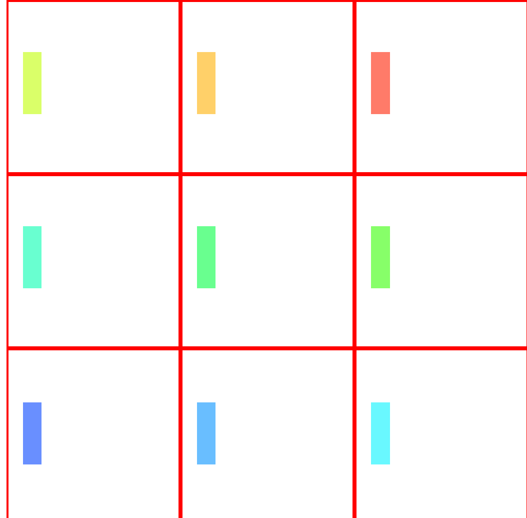
The `fcup` command tells ECP to move a Faraday cup into the beam, measures the current and move the cup out again. When the cup is located on the stage ECP moves to the stage position specified for the cup at "Stage/Coordinates" in the settings (see section 12.5.1) and back after the measurement.

The exposure current (see 5.2.10) is set to this value. If the current was already set this command also calculates a correction factor. The measured current and the correction factor are logged, when "Log current" is checked in the settings (see section 12.4).

Usually it is sufficient to check the current every few hours. For shorter exposures use `current` to set the current ECP uses for dosage calculations and use `fcup` to check what the current really is once at the beginning.

ECP corrects the exposure dose by changing the dwell time of the patterns written. When the current rises, ECP reduces the dwell time and vice versa. The patterns should take this into account, that is patterns that are used together with `fcup` commands should not use the shortest possible dwell time to allow ECP to reduce the dwell time should the current rise. The size of this time margin should be based on the temporal stability of the emission current.

If the calculated dwell time needed to correctly expose the pattern becomes too short, ECP uses the shortest allowed dwell time, that is the time "1/maximum clock" and over-exposes



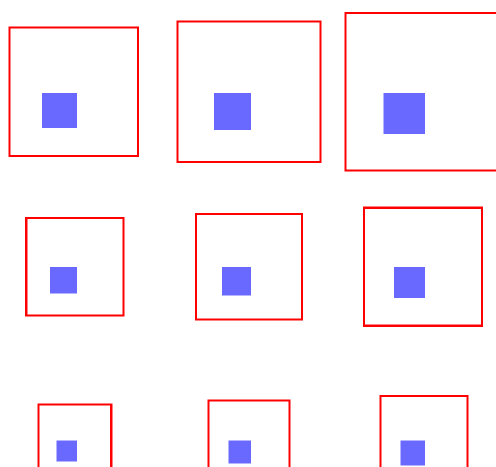


Figure 5.1: In the upper three by three grid the dose is increased with dscale. In the lower grid the field size is increased with fscale and the dose increased with dscale to get a constant dose for all fields.

the sample. To avoid this kind of overexposure, ECP can be configured to stop an exposure when a too short dwell time is needed. This is done in the settings ("System/Settings..." menu item) as an option under "File formats/CTL - Files". (See section 12.3) The "Allow Overexposure" box is checked by default, un-checking it makes ECP abort an exposure rather than overexpose as described.

5.2.17 fcup_in

Like the `fcup` command (5.2.16), an "`fcup_in`" moves an automatic Faraday cup, if one is installed, into the beam, blocking it. Unlike `fcup`, this command is a simple movement, the current is not measured and the cup is not moved out again. This command can be used at the end of an exposure to switch off the beam without controlling the high tension. With the Faraday cup in the beam path some parts like the edges of a beam blanker are no longer exposed to the beam current.

5.2.18 fcup_out

`fcup_out` moves a Faraday cup out of the beam path.

> **N. B.** When using the `fcup_in` and `fcup_out` commands it is up to the user to ensure that the Faraday cup is moved out of the way for an exposure. For example using `fcup_in` at the end of one CTL file and then using another file without an `fcup_out` and without moving the cup by hand would leave the Faraday cup in the beam path and "expose" the inside of the cup instead of the sample.

5.2.19 field_rot, +field_rot, -field_rot

With `field_rot` the content of a writing field is rotated. This is distinct from the `rot` commands (see section 5.2.44), as *only* the written structure is rotated, the stage movement is *not* changed.

Similar to the `x`, `-x`, `+x` and `rot`, `-rot`, `+rot` commands, with `field_rot` an absolute command of field rotation is set. (`field_rot 0` resets any field rotation.) The `+field_rot` and `-field_rot` commands increase or decrease the amount of field rotation.

As the rotation is done digitally, for angles where the original writing field axes are not horizontal or vertical (angle close to 45°, 135°, ...), shapes in the corners of the writing field may cross the writing field borders. The point at which this starts depends on field calibration and chip mark correction values. This effect should not be relevant when all shapes fit into a circle around the center of the writing field with a radius of 25000 pixel. (Touching the midpoints of the four writing field borders.)

The structures are not shown rotated on screen, even when they will be exposed with a rotation.

frot

The commands `frot`, `-frot` and `+frot` can be used as shorter synonyms for the `field_rot` commands.

5.2.20 for - loops

The for loops are very helpful if you want to expose one or more patterns multiple times. The syntax is:

```
for x = 1 to E step S
... do something
next x
```

x is a variable and must be alphanumeric. The `1` is the starting point, E is the target and S is a step size. E and S must be numbers. The “step S” part is optional, if no step size is given, it defaults to 1. The commands between a `for` and the `next` with the variable of the `for` are run until the variable is greater or equal the target number and S is added to the variable at every iteration.

The following example shows how to fill a ten by ten grid with a pattern:

```
y = 0
for a = 1 to 10,
x = -200
+y = 200
for b = 1 to 10
+x = 200
stage
draw(RECT)
next b
next a
```

> **N. B.** The old syntax `for (1,E,1)`, where the second 1 is a step size can still be used in ctl files.

5.2.21 fscale

`fscale = x` scales the field size by x. Using for example first `fsize = 100` and later `fscale = 2` changes the the size of the writing field from 100 µm to 200 µm.

> **N. B.** Scaling the field size by x multiplies the area dose by $1/x^2$ and the line dose by $1/x$. To get a correct exposure, use `dscale` (5.2.14) as well

On some makes of SEM only discrete values from a list can be set for the magnification, ECP selects the best value automatically. When using a small factor for `fscale`, ECP may set the magnification to the same value for both field sizes. To write with the exact field sizes selected it is necessary to calibrate these field sizes before the exposure, see section 11.1, and to activate “Load calibration on field size change” in the calibration settings. The results of the calibrations must be saved with the names suggested by ECP or the program will not be able to find them during exposure.

5.2.22 fsize

`fsize` specifies the size of the writing field in micrometers.

For example `fsize=200` sets the field size to 200 µm

This (and fscale) of cause only works on systems that support dynamic field size change, mainly SEMs and e-beams with an attenuator.

When the field size is changed, the old field calibration is no longer useful. ECP either resets the calibration or loads a calibration from the file `calib/default_xxxx.cal` with xxxx replaced by the field size in micrometer. This file name and location is the same that the calib-files.

ECP decides to loads the data or just resets the calibration by using the "Load calibration on field size change" check box in the calibration settings (see section 12.1). If the calibration data file can not be loaded, ECP resets the calibration data.

> **N. B.** On a SEM, where the field size is set via the magnification, take care to correctly set the "Reference size SEM image" in the exposure field settings, as this value is used to calculate the magnification value to set.

5.2.23 gdraw

ECP provides commands to create diffraction gratings bigger than a writing field. These commands begin with a "g".

One of these commands is `gdraw(h,P)`. This draws a grating as set up with the other grating commands. `h` defines the orientation of the grating, "h" for horizontal drawing. P sets the period of the grating in Nanometers. So for example `gdraw(h,123.17)` draws a horizontal grating with a period of 123.17 nm.

5.2.24 ginc

With `ginc`=i it is possible to specify the increment in the same way it works in a pat-File (see section 4.3). So with, for example, `ginc = 3` only one point in three is exposed.

5.2.25 glength

`glength` specifies the length of the grating in micrometers. For example `glength = 2000` sets the length of grating to 2 mm.

5.2.26 gline

`gline` sets the width of a single line in the grating. The width is specified as the number of exposure pixels, for example `gline = 20`.

5.2.27 gtime

`gtime` sets the dwell time for the grating in nanoseconds. It is used like `C` in the pat-files. For example, with `gtime = 250` every pixel is exposed for 250 ns.

5.2.28 gwidth

`gwidth` is the width of the grating in exposure pixels, for example `gwidth = 2000` produces a grating 2 µm wide when the fieldsize is 50 µm and an exposure field is 50000 pixel wide.

To create a grating you have to define all of the `g`... variables mentioned above.

An example ctl file to draw a grating looks like this:

```
origin = 0,0
current = 200
fsize = 50
ginc = 2
gtime = 200
gline = 20
gwidth = 20000
glength = 2000
gdraw (H, 240.12)
```

N. B. You need no pat file to create a grating like this.

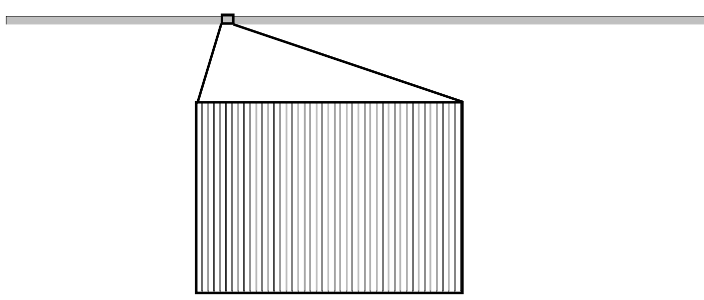


Figure 5.2: Overall and detail view of a grating defined with the `g...` commands

5.2.29 gmark (MANUAL / AUTO)

ECP provides more commands that begin with a “g”. They deal with global marks and begin with “gm”.

When the software reads the command `gmark(MANUAL)`, it starts a manual global mark scan. It moves to the global marks defined with `gmcoord` commands and starts a mark scan routine. It gets the exact position of a previously made structure and align the new file to the previously made layer. When you use the `AUTO`-option, the software tries to find the marks without user input. This is very helpful when the exposure runs over night and you want to align many different fields to different previously made layers.

5.2.30 gmcoord

`gmcoord = x,y` specifies the coordinates of a marks which will be used for a global mark scan. You can define up to 100 global marks with repeated `gmcoord` commands. When you specify two marks, the software calculates the rotation, the scale factor and the shift of the mainfield with the information of the two marks. When you use three marks (normally the best results), the software calculates different parameters for the x-axis and the y-axis. The result of your exposure should be better because the rotation and the scale factor of the two axis can differ. When you specify more than three marks, the program uses the first three marks for the first mark scan, the next three marks for the next mark scan and so on.

5.2.31 gmfile = filename

gmfile specifies the name of the mark file you want to use for the mark calibration. This file must be in the same directory as your structure. Use the menu item "Mark/Make Global" to copy it there.

See chapter 7 for details on defining marks.

5.2.32 gmfilepos

gmfilepos, like cmfilepos (see section 5.2.8, can be used when global marks in different parts of the sample are different from each other. Give the mark file name and the mark positions as parameters, analog to the use of cmfilepos. For example "gmfilepos ll_global 100, 0" would set up detection of the mark ll_global.mrk 100 µm to the right of the origin (5.2.41).

5.2.33 gmark_fpr

Like chip marks, global marks can be used rotated. Use gmark_fpr to set it up. This command, too, is analog to a chip mark command, cmark_fpr (see section 5.2.9). For example, to detect a global mark 2 mm to the right and 3 mm above the origin that has been rotated 180°, you could use "gmark_fpr tr_mark, 2000, 3000, 180"

These gm... commands are also described in section 11.3.

5.2.34 ht_off

When a long exposure is finished,[3] the SEM's high tension can be switched off with the ht_off command on some SEM types. This is useful, as it can improve the life time of a cathode by not having it run unnecessarily.

What this command actually does depends on how much control the SEM software offers to a remote controller like ECP. On some types there may be no way to change the state of the gun and this command does nothing. Or a ht_off call may simply switch off the high tension, as the name implies. On SEMs with a field emitter gun the command may also close the gun isolation valve between the sample chamber and the gun column, improving the gun life time by reducing contamination.

When this command can not turn off the gun, using the fcup_in command (see section 5.2.17) may be used to reduce stray exposure to the sample. There should be no harm in using both commands.

5.2.35 ht_on

The ht_on command can be used to switch on the SEMs gun after it has been switched off with ht_off (5.2.34), opening a gun isolation valve when necessary.

As a typical work flow includes setting up the SEM (focus, position on the sample etc.) by hand with the high tension on, this command is usually not needed.

[3] A typical scenario is an exposure that was started in the afternoon and finishes in the early morning, when no operator is around.

5.2.36 idraw

`idraw` is similar to the `draw` (5.2.13) command. It provides a basic way to use the loop variable of a for loop in patterns names.The command takes two arguments, the base name of the pattern to be drawn and the variable name used in a for loop. The base name and the value of the variable are concatenated.

For example the commands

```
for a = 1 to 5
+x = 200
stage
idraw(pat_, a)
next a
```

draw the patterns `pat_1` through `pat_5` one beside the other with a pitch of 200 µm. If a pattern, for exapmle `pat_3`, is not defined, `idraw`, unlike `draw` simply skips it without producing an error.

5.2.37 image_draw

`image_draw (`image-file`)` is used to expose arbitrary shapes by using image files. The image is drawn so that the longest side, the width or the height of the image, fits into the field size set with 5.2.22.

Each image pixel is drawn onto the sample with a doses defined by its gray level and by `dose_table` (5.2.12).

The image file must be in the directory of the ctl file. It can be in any format supported by the version of the Qt libraries used. PNG, BMP, XBM, XPM and PNM files always work, JPEG images can be used by almost all versions as well. GIF and some other files formats may also be supported, TIFF images usually can not be used. As different image types can be drawn, the file extension must be given with the `image_draw` command.

Grayscale PNG files are well suited for most applications. They use a lossless compression, that is they contain exactly the same information as plain bitmap files but are much smaller. When the image contains only large geometric shapes compression rates of better than 1:30 are possible, reducing disk usage.

> **Caveat:** Using images with transparency ("alpha channel") can lead to unexpected results. To ensure consistency, the background that can be "seen" through the transparency is treated as white,[a] and transparent pixels exposed with the dose for white from the dose table. Especially when the background color has been changed in the settings, it can be useful to check the dose with the "show dose" feature before an image exposure.
>
> [a] As white is the standard background color in ECP and most paint programs.

All image pixels are drawn with the same number of pixels, calculated so that the image is not larger than the field size, but the image can be drawn smaller than this. When the number of pixels XeDraw2uses is not divisible by the number of image pixels, the image will not exactly fill the writing field. This effect is usually not important when drawing images with only a few hundred pixels to a side, for example a company or university logo.

For example, with 50000 pixel in a writing field, a small image with 323 pixels will be drawn with 154 pattern generator-pixels for each image pixel, the result of the integer division of $50000/323$. So the whole image will take up $323 \times 154 = 49742$ pixels of the pattern generator,

filling 99.5 % of the 50000-pixel writing field. On the other hand, when using an image of 25001 pixel, each image pixel is represented by one pattern generator pixel. The result would be that the image appeared at only about half the set field size, 25001 pixels wide instead of 50000.

N. B. To draw structures with a precise size using images, the pixel number of the pattern generator should be divisible by the pixel number of the image. With 50000 pattern generator pixels, image with sizes of 50000, 25000, 12500, 10000, 6250, 5000 ... pixels fit *exactly* in the writing field.

Before using an `image_draw` command, `fsize` (5.2.22) and `dose_table` (5.2.12) should both be set.

The command "`image_pixsize`", that was required in older versions of ECP, is no longer necessary but it can be used to fill only a part of the exposure field with an image.

To change the size an image should take up on the sample use `fsize`.

5.2.38 image_pixsize

The command `image_pixsize = x` sets how large a single image pixel is drawn by `image_draw`. Every pixel takes up a square with the sides $x$nm long. The command can be used to draw an image so that it doesn't fill the whole writing field.

N. B. This command is no longer necessary to expose an image.

Usually the size of an image pixel is calculated automatically so that the image fills as much of the writing field as possible. This command can be used to overwrite that value.

5.2.39 mode

The `mode` command switches two different types of modes, controlling the stage coordinate mapping or the writing of single pixels.

Coordinate Mapping

ECP can be told to ignore the results of a global mark detection or a `rot` command, as far as the stage movement is concerned. This is called "direct draw" and activated with `mode = direct_draw`. The normal mode, mapping the stage coordinates according to the rotation and global mark detection results, which is active by default, can be reactivated with `mode = mask_draw`.

Single Pixels

Single pixels can be exposed in one of the following modes:

nanosecond mode The default mode is nanosecond mode. It is activated with `mode = ns`. Each pixel is exposed for the specified dwell time, given as nanoseconds with the `C` command in pattern files or in the `draw` command. This is the only mode that is needed for standard lithography.

millisecond mode In millisecond mode the dwell time is interpreted as millisecond, not nanoseconds. It is activated with `mode = ms`. Every pixel can be exposed for a time from 1 ms to 65535 ms, slightly longer than one minute.

Special care is needed when writing pattern files with millisecond counts shorter than the smallest nanosecond count allowed in nanosecond mode, for example when using `C 1` or `C 10` (one or ten millisecond per pixel) with the pattern generator clock set to 10 MHz, (smallest nanosecond dwell time allowed is `C 100`). In this case a control file with `mode =` ms should be written or loaded before writing the pattern file, to internally switch ECP to millisecond mode to accept these values.

Millisecond mode is used for applications beyond lithography, where the beam should stay on a material for a long time, for example to do material research.

single ion mode The single ion implantation mode is activated with `mode =` sii. In it the dwell time is ignored and each pixel is exposed until an external trigger signal is received on the “Control” connector of the pattern generator. The hardware must be specially set up for this mode. When the external signal is triggered by a single ion of an ion beam striking the sample, this mode will result in exactly one ion implanted at each position of the pattern file.

When after sii_timeout milliseconds (see 5.2.48) no trigger signal is received, the pattern generator moves on to the next pixel.

beam induced deposition mode `mode =` bid switches to beam induced deposition mode. In this mode each pixel is exposed for the specified number of nanoseconds, like in nanosecond mode. After the exposure of a single pixel the pattern generator blanks the beam for a period of time set by the `bid_off_time` command, see 5.2.4.

This mode is intended to be used for electron or ion beam induced deposition or etching. In a typical scenario the short exposure with the beam induces a chemical reaction with a gas present at low pressure above the sample. This reaction locally depletes the reagent at the point of the exposed pixel. During the waiting time gas can flow back to that point.

When none of these single pixel mode commands is given, the nanosecond mode is used.

Synonyms

The keywords used to set up the mode are not case sensitive, and there are a number of synonyms that can be used instead of the those described above.

direct_draw	d, direct
mask_draw	m, mask
ns	sii_off, siioff, ms_off, msoff, bid_off, bidoff, standard, fast
sii	sii_on, siion

There are no synonyms for “bid” or “ms”.

5.2.40 next

A `next` varname terminates the loop started with the `for` with that variable name. See 5.2.20 for details.

5.2.41 origin

With `origin = x,y` you specify the starting point of the exposure. The origin of the coordinate system used in the ctl file is set to these x and y values of the stage coordinate system.

5.2.42 recipe

The `recipe`-command sets the recipe or writing strategy file. See 5.3 for details on writing strategies.

The syntax is `recipe = filename`. the recipe file, with ane `.rcp` file extension, must be in the same directory as the control file.

5.2.43 rmark

`rmark = x, y` defines the position of a reference mark. Like `gmark` (5.2.29) you have to define a mark definition with `gmfile` (5.2.30) before you can use the mark. An `align` (5.2.29) command uses a `rmark` as a single reference mark to align the exposure. This can be used to correct a stage drift on a sample without a dedicated mark when a SEM image of, for exapmle, a scratch or a recognizable speck of dust is used as a mark. See also section 11.4

5.2.44 rot

It is basically impossible to mount the sample so that the edges are exactly horizontal and vertical. To align the structure to the edges of the sample, you must correct this rotation. Set the angle that the sample is rotated against the stage coordinate system with `rot = deg`, where deg is the angle specified in degrees. Use positive angles when the sample that is rotated counterclockwise, negative, when it is rotated clockwise (mathematical convention). See also section 5.2.3, or use the global mark mechanism for higher precision, see 5.2.29.

5.2.45 +rot,-rot

These command adds a rotation to the absolute rotation of the sample.

5.2.46 setfocus (INIT, AUTO, OFF, NOW)

When you want to correct the working distance difference of the sample, you can measure the distance when you calibrate the sample with "Working Distance Correction..." in the calibrate-menu, section 11.2. After setting up the correction, that is focusing at three points and storing those values, `setfocus(INIT)` reads the calibration data and starts the calibration. `setfocus(AUTO)` will correct the focus for every exposure field. `setfocus(OFF)` switches this feature off and with `setfocus(NOW)` the focus is corrected immediately at the current position.

Remember to keep the "Working Distance Correction" open or not to open a new one after closing it.

5.2.47 sfile

`sfile` sets the source file, that is the pat-file where all the structures are defined. You can use many `sfile` commands in a ctl file. `draw`, `idraw` and `adraw` always look for the patterns in the current source file, that is the file named in the last `sfile` command. The syntax is `sfile = pname` where "pname" is the name of the pattern file without the ".pat"; in the example ECP needs a file named "pname.pat" in the directory of the ctl file.

5.2.48 sii_timeout

This assignment sets the time in milliseconds that the pattern generator waits in single ion implantation mode when no trigger signal arrives until moving on to the next pixel.

5.2.49 stage

Moves the stage to the position defined by `origin` and `x,y,z`, ±`x,y,z` commands.

5.2.50 x,y,z

These commands set the target for the next `stage` command, for exapmle

```
x = 500
y = 1000
```

sets the stage move target to 0.5 mm to the right and 1 mm up from the origin defined with `origin`.

5.2.51 +x,+y,-x,-y

These command add a length to the stage move target or subtracts it. If you add, for example, the line `+x = 500` to the example above (5.2.50) the next `stage` would move the stage 1 mm up and 1 mm = 500 µm + 500 µm to the right from `origin`.

5.2.52 wait

`wait` is used for pausing the exposure for a certain time, specified in milliseconds, for example `wait (300000)` waits five minutes.

5.2.53 pause

`pause` pauses the exposure, opens a small dialog asking "Continue exposure" and waits for the user to click on "Continue" or to abort the exposure with "Abort".

5.3 Recipes

ECP uses specially designed algorithms to steer the beam over the areas of the shapes. These algorithms are optimized for lithography use and high writing speeds. For other applications, especially ion beam milling or deposition, it can be necessary to change these writing strategies. This can be done with so called recipes.

Recipes are created with the "Edit Writing Strategy" dialog from the "Extras" menu.

On the left-hand side of the dialog the writing of the dialog the behavior of a single writing pass. (See 4.5.1 to process single shapes for longer times.) The way the shapes are written is simply selected by clicking the radio button beside the icon showing the writing mode.

For the basic strategy "spiral/intelligent fill" there are two more controls. Small shapes with a high aspect ratio (much wider than high or much higher than wide) are usually written as

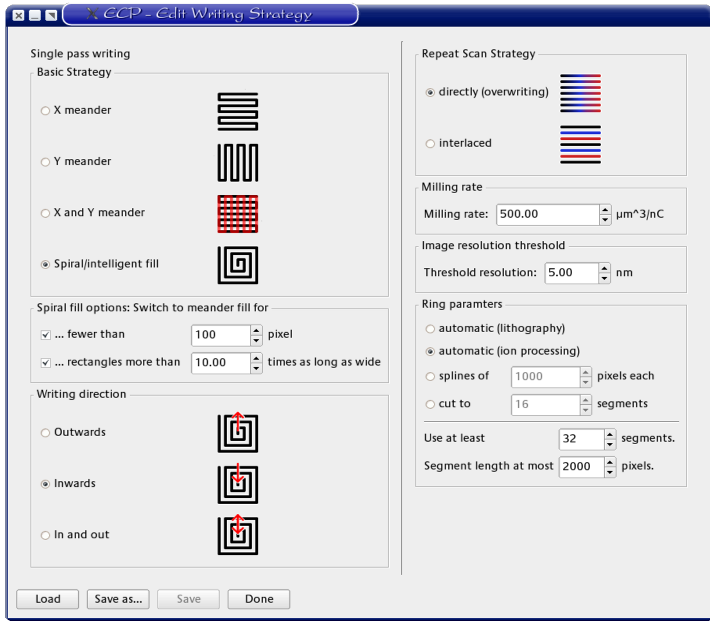


Figure 5.3: Screenshot of the edit writing strategy dialog

meanders. The two lines in the middle of the left side control when this change happens. Unchecking these check boxes means that the shapes will always written as spirals, when that basic strategy is active.

The "Repeat Scan Strategy" on the right-hand side sets how consecutive scans are executed: "directly" means that all exposure are at exactly the same positions. With "interlaced" and an increment ("`I`", see 4.3) in the pat file set to values above one, the exposure steps up one pixel from one scan to the next, filling the spaces left by the increment on a single scan.

Below that is a control for the milling rate. This rate is used to control the processing when a specific processing depth is desired.

XeDraw2 uses two different algorithms to expose images. A precise one and a slightly faster one when the difference can't be detected because of the limitations of real hardware. The minimum size of image pixels where XeDraw2 uses the precise algorithm can be set here.

The last box controls, how single-pixel rings are cut up into splines. The more shorter splines are used, the more precise the exposure will be, but the calculation takes time and rings with fewer segments will be exposed more quickly. The two first radio buttons select automatic modes that should work well for the two different uses, lithography and ion processing, respectively.

After setting all recipe parameters the information can be saved in a recipe file with the "Save as . . . " and "Save" buttons. Recipe files prepared earlier can be loaded with the "Load" button. "Done" closes the editor dialog.

5.4 Proximity Correction

The proximity effect can be corrected by use of different algorithms. We have implemented one in the software that changes the doses used by the e-beam so that the resulting patterns matches the one drawn as close as possible.

5.5 Overlay files

Overlay (ovl) files are used to combine several ctl files.

For an ovl file, the main window is divided in three parts. As with most file formats, to the right the shapes in the file.

The list on the top left shows the ctl files in this ovl file. The first thing to do after creating a new ovl file is to add several ctl files here, using the "Edit/Add File..." menu item. Each file is shown as a different layer, starting with layer one for the first file in the list, and drawn in the way set up in the layer definition file used.

When several ctl files have been loaded, clicking on a file name shows the text of this file in the lower left side edit area.

The ovl file can now be used to change the ctl file so that they match, for example by changing `x` and `y` values of start coordinates. Special care should be taken that the positions of global marks or chip marks in the different files match.

To align shapes or marks the zoom and measure tools can be used to determine how much exactly an element should be moved and then change the values by this amount.

After checking and correcting the overlay of the ctl files, they can be exposed one by one, processing the sample in between if necessary.

6 GDS-II Files

GDS-II[1] files are currently the standard for data exchange of IC mask designs. While ECP cannot directly expose data contained in a GDS-II (or "gds"[2]) file, it can read gds files. These files can be viewed and manipulated.

When viewing a gds file ECP's window is divided into two areas, the tree view and the graphical display area.

The gds file format is organized in layers and hierarchically. The layer structure means that the different parts of for example an IC design, like different implants or contacts, can be stored in one file. To represent complex and repetitive structures, gds allows to define substructures, for example single logic gates or large logical units of a chip. These substructures can contain other substructures and one structure can be a substructure (be referenced) in many other structures.

This relationship of different structures is displayed in what is called the "Structure Tree" in the lefthand view. After loading a gds file this structure tree is collapsed, that is only the "root" node is visible. With a double click on a node, including the root node, any structures in the hierarchical level below that are shown in the tree view. Clicking on a structure once shows only this structure on the right.

How much detail is shown in the graphical view is controlled with the "Increase Recursion" and "Decrease Recursion" items of the "View" menu and the spin box in the tool bar. With a recursion level of 0 only shapes on the selected level and frames for the substructures included at that level are shown. Increasing the level of recursion fills these frames with the shapes and frames in the next level of substructures.

Which layers are drawn can also be controlled. Open the "View/Layer Properties..." dialog. Here two check-boxes control if the layer is drawn with the fill pattern indicated or just as a frame and if it is drawn at all. After making changes click the apply button to see the effect on the display. The colors and fill patterns are taken from a layer definition file. This can be loaded manually with "View/Load layer properties..." or automatically. See section 12.3.

The "Properties" column contains a "SREF" or "AREF" for each substructure. This and the icons in the first column indicate if this substructure appears at this point as a *s*ingle reference—just once—or as an *a*rray reference, repeated vertically or horizontally.

6.1 Editing GDSII Files

As ECP is not designed to create or edit gds files, it provides only simple tools to edit them. They are more useful to fix small problems that have been discovered at a late time than to create patterns.

The edit tools for gds files are also described in section 3.2.4.

Selecting a part of a structure, and selecting the "Edit..." item from the "Tools" menu or the context menu displays a list of the structures, sorted alphabetically, on the left.When you select a structure from the Structure Tree view of the main window, the "Reference" column

[1] graphic design system II

[2] with a "`.gds`" file extension

of the Edit Reference dialog shows which simpler structures are used in this structure. You can add another structure to the active one by dragging it from the left column to the middle. You can also modify a structure. The controls are the same as the "Transform" controls with the possibility to use the sub-structure repeatedly as an array. Just set "Reference Type" to "Array" and input the appropriate values in the "Array" box.

The Values in these edit fields specify how the original structure is modified when it is included in the higher level-structure. So pressing "Apply" more then once when, for example, "Scale" is > 1 scales the structure only once, unlike the "Transform" dialog.

The "Transform" item can be used to modify a whole structure. There are currently no tools to modify the shapes like boundaries or paths in the structure.

6.2 Exposing Data from GDSII Files

To use the data from gds files for an exposure it has to be converted to ctl and pat files. To do this, first call "Flatten" from the "Tools" or context menu for the structure you wish to write. Next select the "_FLAT" structure and call "Extract Layer..." on this flattend structure. Select the layer you want extracted in the dialog. On this flattened structure with the single layer, for example "..._FLAT_5", use "File/Export/CTL" to convert the gds to a ctl and a pat file. This opens an options dialog (figure 6.1). Set the layer to export, the one selected before, the file name (the files will be created in the directory of the gds file), the field size and the number of pixels. The field size and pixel number values default to those defined in the settings. There is usually no need to change the pixel number.

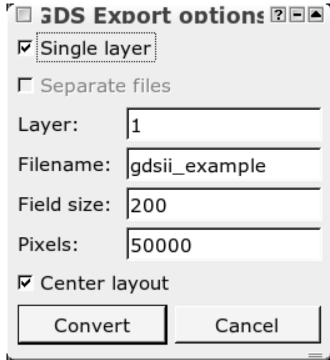


Figure 6.1: GDS export options dialog

After an export the ctl file can be loaded into ECP and the pattern exposed.

7 Alignment Marks

Alignment marks are used to calibrate the exposure field of the lithography system and to align an exposure to an existing structure on the sample. Marks are commonly fabricated by a lift-off process (e.g. Au marks) or by etching ridges or trenches into the sample.

To align an exposure at an existing structure ECP must recognize these marks. This chapter describes how to define them in ECP. Alignment and calibration with these marks is described in chapter 11.

7.1 Editing Marks

ECP can use cross shaped marks, marks defined with commands used in pat files and SEM images with a wide range of parameters. The position of a mark is detected by scanning the electron beam over the mark and recording the secondary electron (SE) signal. The edges of the mark usually show up as distinct peaks in the scan signal.

If you want to create a new parameter set for mark detection, select “New Mark” in the menu “Mark” or click the button . You can select if you want to define a cross mark or an image mark. An image mark can be defined either with geometry commands used in pat files or by taking a SEM scan of an actual mark on your sample.

The properties of an existing mark can be modified by selecting “Edit Mark” in the menu “Mark” or by clicking on +. ECP will show you a file dialog with a list of existing mark files.

7.1.1 Cross Marks

For cross marks a threshold method is employed to determine the position of the edges. The result of several scans can be averaged in order to obtain a higher accuracy.

The mark editor for crosses shows the geometry of the mark and the position of the scan lines (shown in green) used for mark detection. The “Cancel” button discards all changes to the mark parameters and closes the editor. “Save & Exit” saves the parameters to disc. In the tab labeled “Geometry”, the name of the mark, the length and width of the arms can be changed.

> **N. B.** The values in a mark definition are specified in pixels, so it may be neccessary to use a specific a mark definition for every field size used.

If a parameter is changed, pressing ’Enter’ or the ’Apply’ button at the bottom of the editor will update the display.

Use the tab “Scan” to adjust the parameters of the line scan. “Increment” specifies the pixel increment of the scan. “Dwell time” is the dwell time per pixel. These parameters are the same as those used in pat files, see sections 4.3 and 4.4. Reasonable dwell times for mark detection are in the range of 2000 to 20000 ns. “Offset X” specifies the horizontal offset of the vertical scans from the origin of the mark. “Offset Y” specifies the vertical offset of the horizontal scans from the origin of the mark. “Scanning length” is the length of the line scan. The number of points in a scan is given by the scan length divided by the increment. XeDraw2can store up to 4000 points in the buffer. This translates into a maximum scan length of 4000 for an

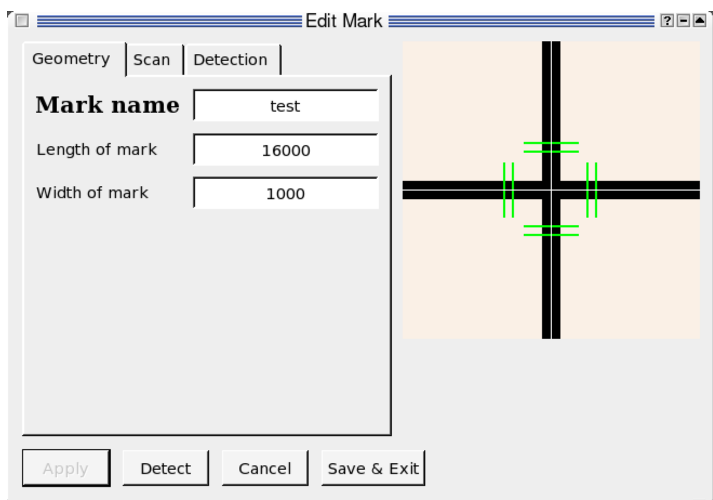


Figure 7.1: Editing the geometry of the mark

increment of 1. If you have to use a longer scan, the increment has to be increased. “Scanning pitch” is the distance between multiple scans. “Number of scans” specifies how many scans are performed on each arm of the mark. Each scan is repeated “Scan repetition” times. If SE signal is clear and has good contrast, “Scan repetition” can be set to 1. For a noisy signal, repeating the scan several times can result in a better signal to noise ratio.

The third tab “Detection” is used to define the parameters for detecting the mark. ECP uses a threshold method to detect the mark. The scan signal is smoothed to reduces the effect of noise. The width of the smoothing can be set with “Smoothing interval”. When some points are not recorded correctly at the beginning or at the end of a scan, they can be dropped. The number of points dropped at the beginning and at the end is set with “Drop points”. The threshold level is set with “Edge detection level”. When the signal crosses this level for the first time this is used as the beginning of the mark, when it crosses this level again, this is used as the end of the mark. The signal levels go from 0 for black to 255 for white. The parameter “Margin of det. width” sets how much the width in the detected signal can differ from the mark width set on the first tab to still be accepted as a valid mark scan.

Table 7.1 summarizes all parameters and the corresponding minimum, default and maximum values.

Often it is sufficient to just set the width of the mark. If the mark can not be detected with “Test Mark” you can try to increase the values of “Scanning length” (increasing “Increment” if necessary) or “Margin of det. width”. The other parameters can usually be left at the preset values.

7.1.2 Image Mark

To use a SEM image as a mark first go to the position of the mark using the stage control (chapter 8).

The mark editor for image marks, like the cross mark editor, has three tabs. Set the name of the mark and its size on this tab. The increment and dwell time are also set here. As the area

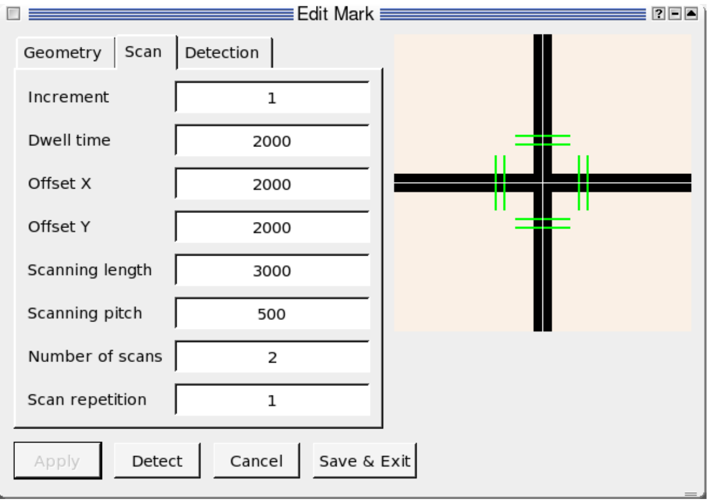


Figure 7.2: Editing the geometry of the mark

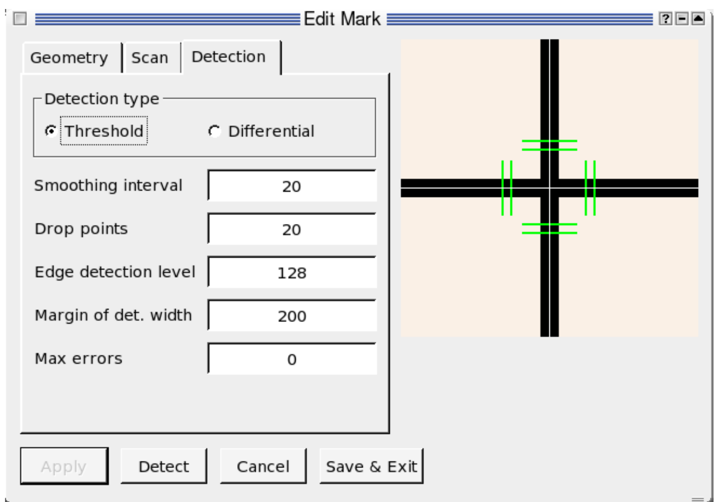


Figure 7.3: Editing the geometry of the mark

Table 7.1: Cross mark parameters

Parameter	Minimum	Default	Maximum
Length	1000	16000	50000
Width	10	1000	10000
Increment	1	1	64
Dwell time	250	2000	20000
X-Offset	100	2000	10000
Y-Offset	100	2000	10000
Scan length	200	3000	20000
Scan pitch	200	200	10000
Scan repetition	1	1	10
Number of scans	1	2	10
Smoothing interval	0	20	100
Drop points	0	20	N.A.
Detection level	0	128	255

of the mark is scanned two-dimensional using small values for the increment slow down the scan. Usually scans with a width and height of 100 $\times$ 100 are sufficient. Set the increment to “Size of mark / 100” for this.

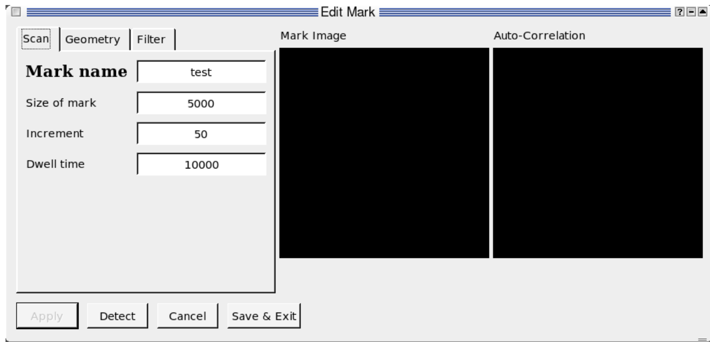


Figure 7.4: Editing an image mark

Select “Scan image” on the second tab to use a scanned image. Click on the “Apply” button to record an image. This is shown in the square in the middle. The autocorrelation of this image is shown to the right. The autocorrelation of a useful mark is light in the middle and darker at the outside. If the autocorrelation is a uniform gray, than ECP will have problems detecting this mark.

Click on “Save & Exit” to save the mark definition to disk. The recorded image is not stored on disk but in memory, so you have to record an image again with “Detect Mark” (7.3.2) after restarting ECP.

The first three parameters on the third tab are used only for marks defined by geometry. The fourth sets the correlation amplitude detection threshold. Only when the maximum of the amplitude of the correlation function between the image template and the recorded image during a mark detection is above this value, the mark is considered valid.

Figure 7.5: Using a scanned image as a mark

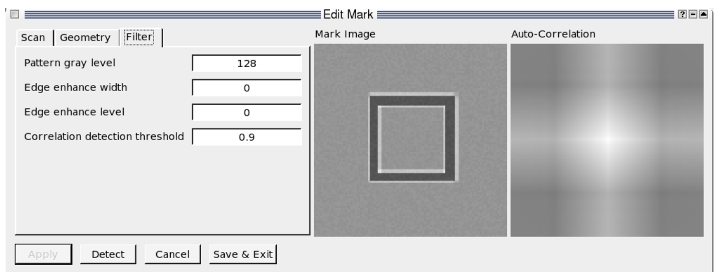


Figure 7.6: Setting up the detection threshold for scan and geometry image marks and the edge simulation filter for geometry marks.

7.1.3 Geometry Mark

A new mark defined by geometry is created as an image mark. Set the parameters on the first ("Scan") tab as you would for an image scan. Select "Define Geometry" on the second tab.

The geometry mark is defined by commands as they are used in a pat file (see chapter 4).

> **N. B.** The field used for mark detection is in the middle of the writing field, that is the middle of the mark detection area has the x and y coordinates ("exposure field pixels/2"),("exposure field pixels/2"). For an exposure field size of 50000 and a mask size of 4000 the area used for mark definitions is between 23000,23000 and 27000,27000.

Currently only the shapes `RECT`, `CIRCLE`, `XPOLY` and `YPOLY` can be used in a mark definition.

Pressing "Apply" displays the mark in the middle and its autocorrelation to the right. The same rules as for image marks (7.1.2) apply.

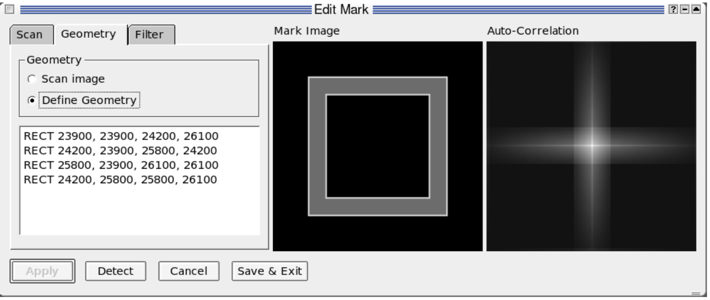


Figure 7.7: Defining a mark by geometry

The third tab, "Filter", defines how the pattern appearance is modified to resemble a SEM image."Pattern gray level" sets the value or gray level in the mark image. The "Edge enhance width" and "Edge enhance level" parameters define how the edge effect is simulated. The width defines how many pixels on the edge of the pattern form the edge and the level defines how much the edge is brighter than the pattern itself.

When the geometry defined on the second tab is not an exact match, the mark can still be detected by setting "Correlation threshold" to a low value, for example 0.2. This can be useful for a global mark scan, but for field calibration with this kind of mark usually yields insufficient results.

7.2 Test Mark

After specifying the parameters, select "Test Signal" from the "Mark" menu or press the button to display the scan signal. The program will show a list of available mark files.

7.2.1 Cross Mark

When you select a cross mark, this shows a graph of the scan. You can use this to set up image parameters like brightness or contrast. The scan should show an arm of the cross as a plateau, see figure 7.8.

The “Change” button switches between scans over the four cross arms.

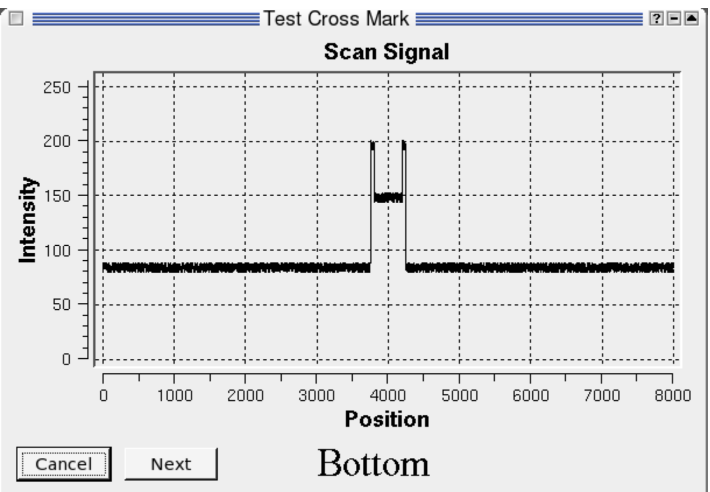


Figure 7.8: Testing the signal of a cross mark. (simulation)

7.2.2 Image Mark

For an image mark, defined by a image scan or by geometry, this shows the two-dimensional scan at the current position, see figure 7.9.

7.3 Mark Detection

A mark detection is performed by selecting the “Detect Mark” from the “Mark” menu or by clicking on the ⌖ button.

7.3.1 Cross Mark

When you select a cross mark, a graph similar to that of that shown for “Test Signal” (figure 7.8) is shown.

The main parameters are displayed at the top of the window. “Target Width” is the specified width of the mark, “Detected Width” is the actual width of the mark. If the difference of the two values is larger than the tolerance set with “Margin of det. width”, the scan is not valid and not used for the calculation of the position. The first edge, shown with a blue vertical line, is the point where the red curve of the smoothed signal crosses the green line of the threshold value for the first time. The point where the red crosses this line again is the second edge.

7.3.2 Image Mark

The dialog for detecting an image mark, defined by an image scan or by geometry, is divided into three square areas. They are filled by pressing “Detect”, see figure 7.12.

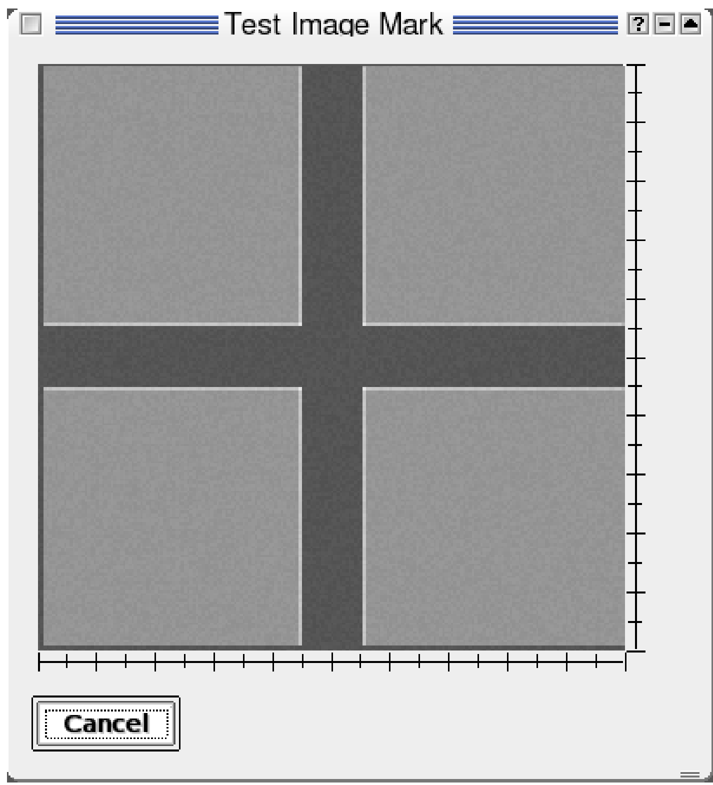


Figure 7.9: Testing the signal of an image mark. (simulation)

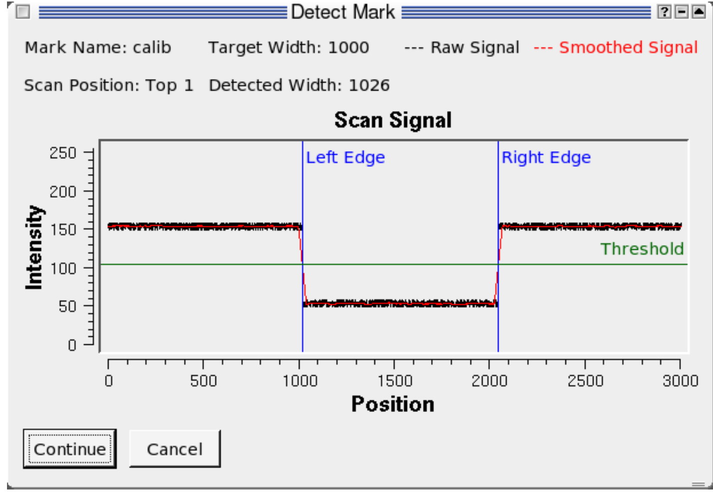


Figure 7.10: “Detect Mark” dialog for a cross mark (simulation)

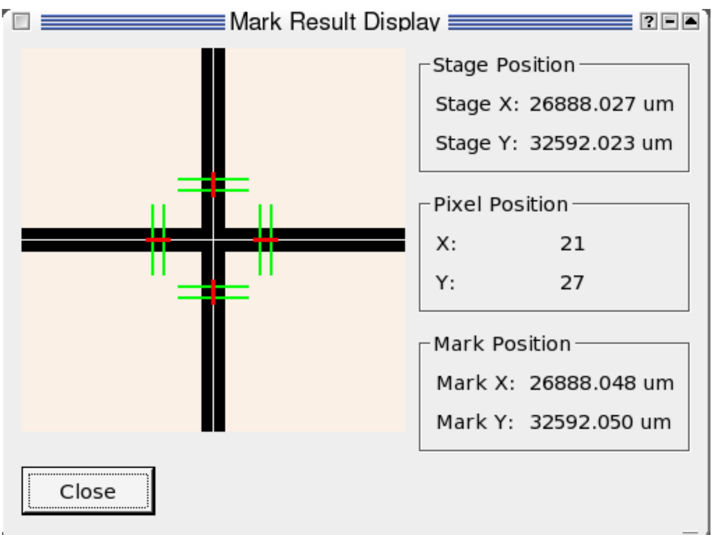


Figure 7.11: Result of a cross mark detection (simulation)

The top left square shows the image template, that is the image that the image at the current position is compared to. If the mark is defined by an image scan, pressing "Detect" records this image.

The square to the right shows the image just recorded. The correlation function of these two images is calculated and displayed in the bottom left square. From its maximum ECP calculates the actual position of the mark and displays in the bottom right area. The "Correlation amplitude" shows how well the template and the scanned image match; 0 means no similarity and 1 is a perfect match.

Below that the Correlation amplitude detection threshold is shown. If the actual correlation amplitude is larger than this, the mark is valid.

The last line in this field shows if the mark was detected or not. For a mark defined by a scan image the correlation is usually quite high. For a mark defined by geometry this correlation can be much lower, especially when the geometry definition is not precise.

For using either a single mark for field calibration or more than one mark for alignment see chapter 11.

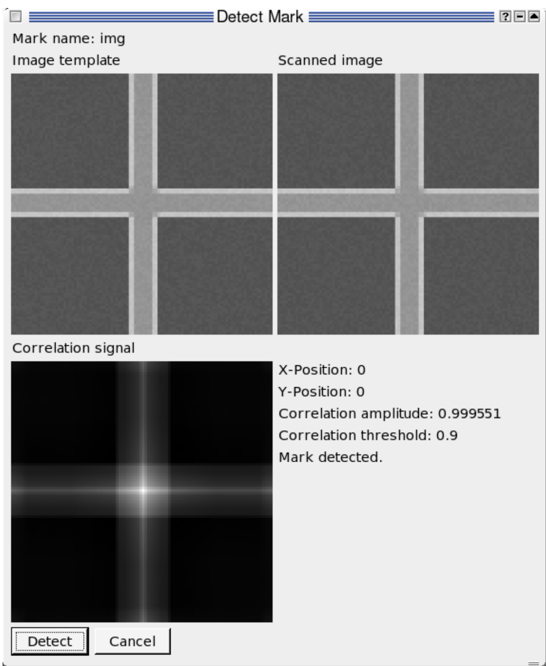


Figure 7.12: Detecting a mark with an image scan (simulation)

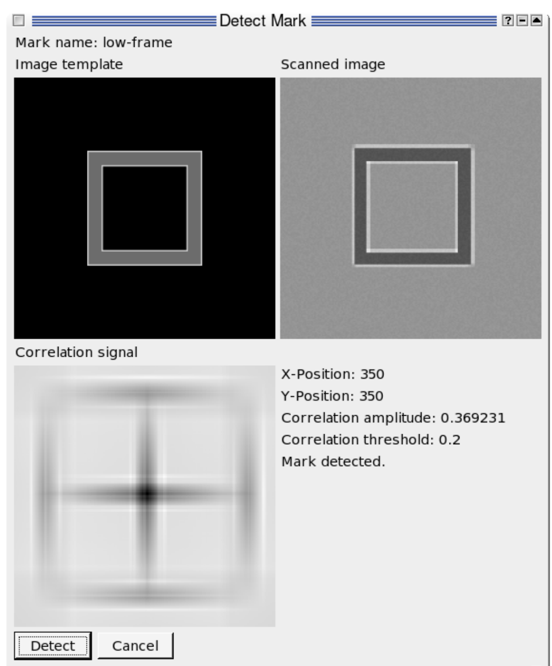


Figure 7.13: Detecting a mark with an imprecisely defined by geometry (simulation)

8 Stage Control

If your system has a motorized stage connected to the control PC, ECP offers a convenient way to position the stage by using the “Stage Control” from the “EBL” menu.

Figure 8.1 shows the Stage Control dialog. The current x and y position of the stage is always shown in the right fields at the top of the dialog.

8.1 Moving Around

This dialog provides several ways to move the stage to a certain position

8.1.1 Entering Coordinates

To move the stage to a position with know coordinates enter these values in µm into the “X:” and “Y:” input fields at the top left of the dialog and press “Move”. The stage will be positioned at that point.

> **N. B.** Turn off the beam using the “Beam Control” from the “EBL” menu when moving great distances to avoid scanning over parts of the sample you want to write on later.

8.1.2 Using the Stage Drawing

The fastest way to position the stage roughly in the desired area is to click into the drawing of the stage and to press “Move”.

The light gray area shows the area reachable by the stage, the red cross marks the current position of the stage and the blue cross marks the position selected by a click. When “Manual mode” is turned off the coordinates of the blue cross are shown in the coordinate entry fields and can be modified before pressing “Move”

This drawing is defined in a file and can be adapted to the stage size and shape and to the number and position of sample holders actually used. This is described in section 8.3.

8.1.3 Manual Mode

The toggle button “Manual Mode” switches the manual move mode on or off.

Select the distance you want to move the stage by in a single step by clicking on the appropriate radio button, from 0.1 µm to 5000 µm.

Switch to the “Num Lock” mode on your keyboard to comfortably use this mode. Use the number keys “1” through “9” (without “5”) to move the stage in the indicated direction. If you press, for example, “7” the stage moves the selected distance to the left and the same distance up, showing an area below and to the right of that before the move.

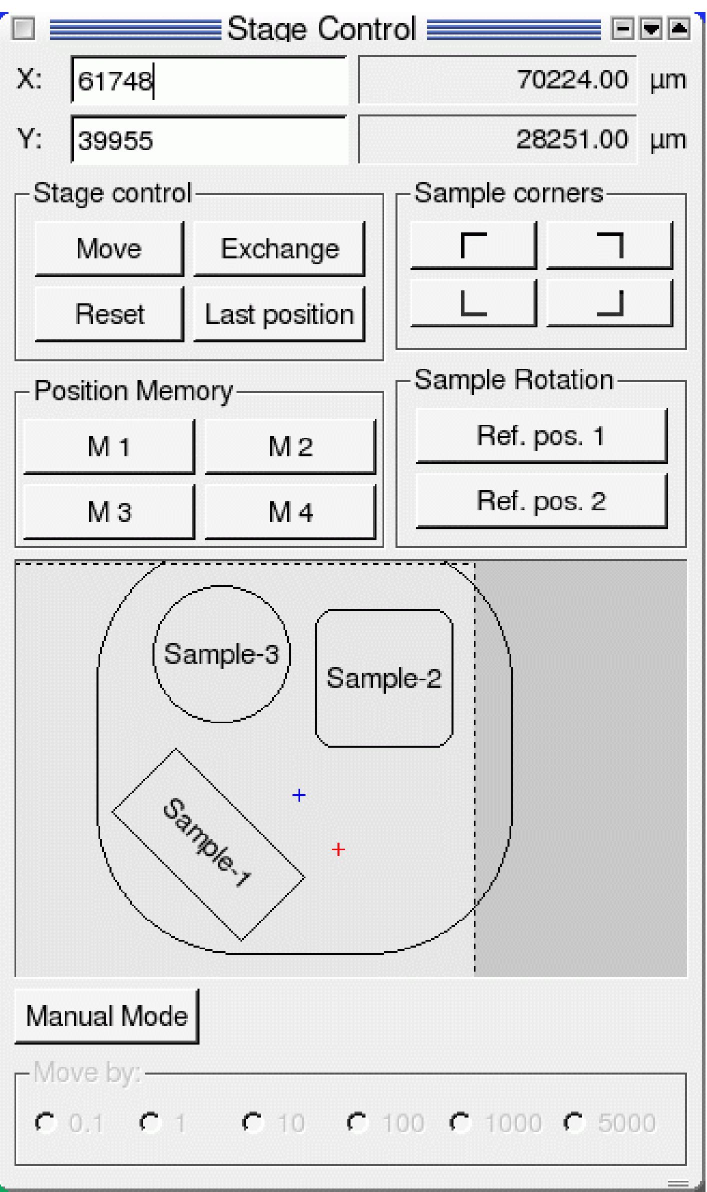


Figure 8.1: Stage control

8.2 Memory Buttons

The stage control offers three groups of buttons to store stage positions, the "Sample Corners", "Position Memory" and "Sample Rotation" buttons. If you start the program all these memories are empty and the buttons have a normal appearance. If you press one of these buttons the current position is stored in the button and it takes on a pushed-in appearance to indicate it is in use. If you press a used memory button you can move the stage to the stored position by using the "Go" button in the dialog that appears, clear the memory with "Delete" or do nothing but check the stored position with "Cancel".

8.2.1 Sample corners

These four buttons represent the four corners of a rectangular sample. If you store the position of at least two opposing corners in these buttons ECP can calculate the size of the actual sample and automatically position the pattern in the middle of the sample using the "Center exposure" function from the Tools menu. See section 3.2.4.

8.2.2 Position Memory

The buttons "M1" to "M4" are general purpose memories that can be used to store the position of interesting parts of the sample.

8.2.3 Sample Rotation

The "Ref. pos. 1" and "Ref. pos. 2" position memories are used to automatically calculate the rotation of the sample. When you store two points in these buttons that are on a horizontal or vertical line in the system of the sample, for example the straight edges of the sample and use the `autorot` function (see section 5.2.3) in a ctl file, the pattern will be exposed aligned to these lines. This becomes inaccurate if the points are to close together.

8.3 Configuration files

The default file is "default.cfg". Other configuration files can be loaded using the "Load Stage Configuration" dialog.

The layout of the sample holder is based on a list of rectangles and circles.

Here is an example file that defines a square sample holder with rounded corners and two rectangular and one circular sample.

```
0
0
100000
90000
rect windows=3
x=18000
y=5000
width=90000
height=90000
label=
```

```
round=70

x=35000
y=20000
width=40000
height=20000
label=Sample-1
round=0
angle=45

x=65000
y=50000
width=30000
height=30000
label=Sample-2
round=30

circ windows=1
x=45000
y=70000
radius=15000
label=Sample-3
```

The first four lines specify the minimum and maximum coordinates of the drawing of the stage that is defined in the rest of the file. Please note that these don't have to be identical to the minimum and maximum stage coordinates specified in the "Settings" menu. The coordinates given here are for display purposes only. The accessible area, as defined by the coordinates of the "Settings" menu, is shown in light gray inside the display area. These lines contain just the numbers of the x and y positions of the lower left corner and the upper right corner respectively.

All other lines contain a keyword, a "=", and a value.

The next line gives the number of rectangular elements in the layout. Each of these rectangles is defined in a block of four to seven lines followed by a blank line. These lines contain the x and y position of the lower left corner, the width and height of the rectangle. These four lines are mandatory. The fifth lines contains a label that is drawn in the resulting image. The sixth line defines how much the corners of the "rectangle" should be rounded and the seventh lines defines how many degrees the rectangle is rotated clockwise. The order of these lines is important. No line may be skipped but the last three lines may be omitted. So if you want to draw a tilted rectangle with no label and sharp corners you have to define an empty label ("label=")and a "round"-value of 0.

The definition of circular shapes is very similar to that of a rectangles. One line sets the number of circles drawn. Each circle is defined in a block of three or four lines similar to the definition of a rectangle. Three lines define the x and y position of the center and the radius of the circle. The last line defines a label for the circle. The label line may be omitted.

9 XeBaSe

XeBaSe is the common controller for the Xenos stage and the Xenos blanker.

9.1 Stage

The Xenos stage is a device for positioning a sample in two dimensions, x and y, inside the sample chamber of a SEM.

The positioning is done with piezoelectric motors that directly move the sample platform. The arrangement of motors and bearings makes it possible to precisely position the stage without backlash.[1]

The position is measured with a build in laser based system.

The platform can be moved by hand, but after moving it a stage reset (see section 9.7.3) should be performed to get accurate absolute measurements again.

While the stage works both in a high vacuum and at atmospheric pressure, it must not be used if the pressure is in the range from 10 Pa to 10 kPa (0.01% to 10% of atmospheric pressure, 0.0015 psi to 1.5 psi). This is the pressure range where the conductivity of air is highest, causing the danger of electrical sparking.

N. B. Do not move the stage while evacuating or venting the sample chamber.

9.1.1 Handling the Stage

While not in the SEM sample chamber, contamination of the stage should be avoided.

- Handle it only with gloves (lint free or latex/nitrile).
- Store it in a climatized, dry place.
- Never touch the white friction bars.
- Never touch the optical grid on the lower side of the top platform.

9.1.2 Using a JEOL Airlock

When using the Stage in a JEOL SEM certain points shourd be observed.

The most important point is to correctly position the stage in the sample camber before removing it through the air lock. The stage must be centered in the x direction and moved to low y values, towards the exchange rod. Otherwise it will not fit into the air lock.

When the stage is not centered in x, it will be caught on the edge of the airlock and may fall out of the clamp on the transfer rod; when the stage is positioned "up", at high y values, the stage platform will block the plate of the inner airlock valve. In both cases a normal operation of the airlock will not be possible, making it neccessary to vent the whole sample camber.

The correct stage position before an exchange can be assured in differet ways:

[1] On stages with geared drives a position should always be approached from the same side to compensate for play in the gears.

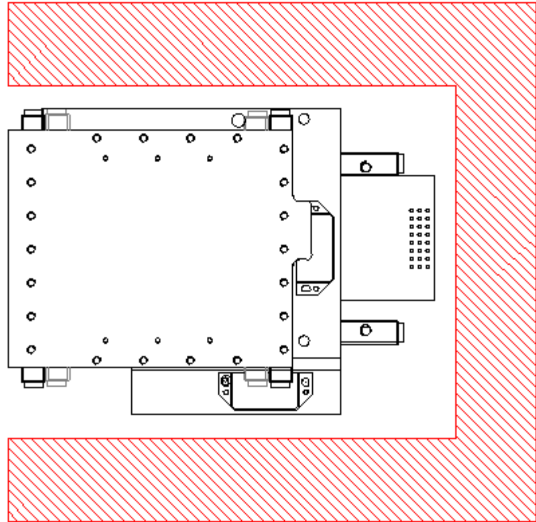


Figure 9.1: Correct stage position during airlock use

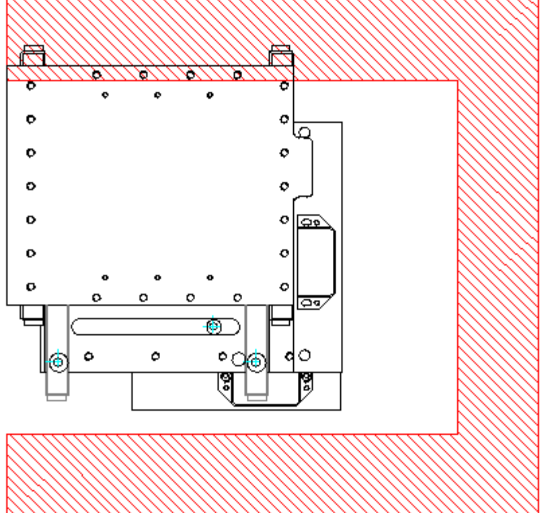


Figure 9.2: The stage may catch on the side of the airlock

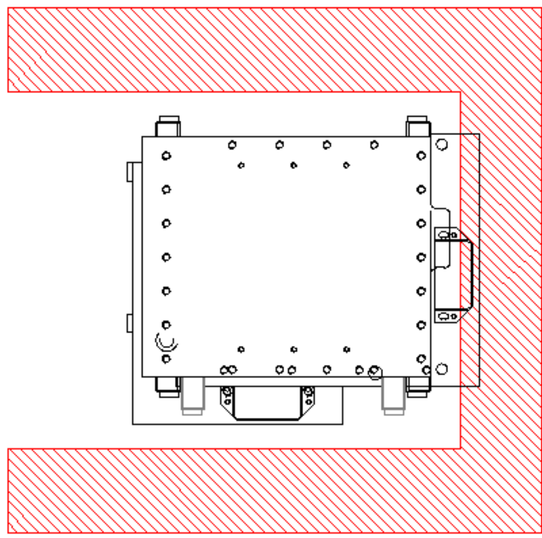


Figure 9.3: The stage will block the vane of the inner airlock valve

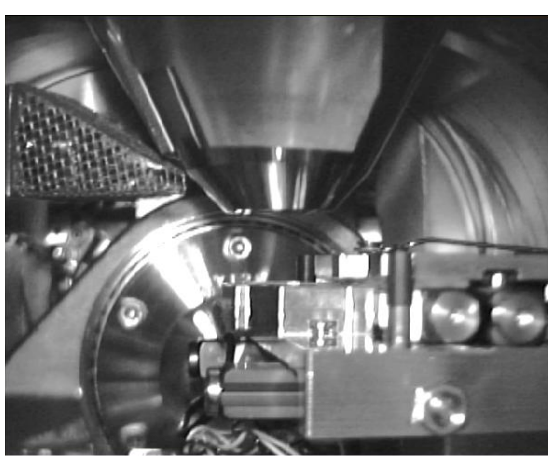


Figure 9.4: Correct stage position seen on a chamberscope. Note the two screw heads on the ends of the guides in the lower right corner are at the same distance from the camera. This indicates that the x position is correct. The stage platform is at the right, the correct y positon.

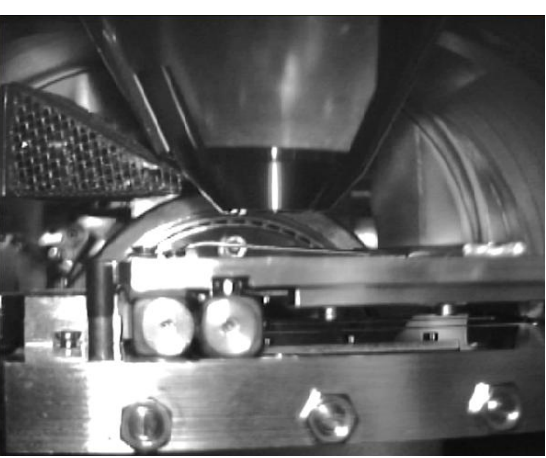


Figure 9.5: Wrong position for stage transfer. The stage is too far to the left in the picture and the airlock vane would be blocked.

- On a chamberscope (IR camera inside the sample chamber). The correct position can be seen in figure 9.4.
- By performing a stage reset. See section 9.7.3. After a reset the stage is in a "safe" position.
- By pressing the "Exchange" button in the stage menu in ECP.

When using the "Exchange" button, the exchange position must have been set up in ECP's settings.

It should also be ensured, that no cables will obstruct the stage before removing it through the airlock.

When inserting the stage through the airlock, it should be brought into a position where it fits into the airlock and through the inner valve, too. This position is the same as for removing the stage and the moving platform can be pushed into position by hand.

SEM Stage

When the Xenos stage is used the JEOL SEM stage *must not* be moved. Instead the SEM stage must be brought to the exchange position by pressing the "Exchange" button at the airlock or in the SEM software.

9.2 Blanker

The Xenos blanker is a motorized combination of an electrostatic beam blanker and a Faraday cup to measure the beam current. It is installed so that it can be moved in and out of the electron beam of a SEM.

There are three positions the blanker can be moved to:

in
: In the "in"-position the blanker is positioned in the electron beam. This is the operating position for the blanker, in this position the beam can be switched on and off with the optical blank signal.

cup
: The "cup"-position is used to measure the electrical current of the electron beam. The beam falls into a Faraday cup and can be measured through the BNC-connector on the blanker flange. This Faraday cup is also called the probe current detector (PCD).

out
: In the "out"-position the arm with the blanker plates and the Faraday cup is moved out of the electron beam.

The beam on-beam off signal is put in through a fiber-optical line directly to the box installed at the beam column.

The same motor as those driving the stage are used, so the same warning applies: the blanker arm must not be moved when the pressure at the motor is in the range from 10 Pa to 10 kPa during evacuation or venting. This is especially relevant on SEMs using a tungsten cathode.[2]

[2] On some SEMs, typically those with field emitter cathodes or with a specimen air lock, the blanker is permanently in a vacuum and this restriction is not relevant.

9.3 XeBaSe Central Controller

All components are connected to the XeBaSe Central Controller (XCC). On the front of the box are the connector for the DCU (see section 9.6) and a USB connection to the PC running ECP or BSC. On the back is a power switch, the connector for the power cord and a number of cables to the other components.

9.4 Blanker Tension Supply

The high tension (up to 150 V) that is applied to the blanker plates is produced in another box, the Blanker Tension Supply (BTS). The BTS is connected to the XCC and has its own power cord. It is switched on and off through the XCC and needs no extra power switch.

9.5 Saving Parameters

A number of parameters, including the current position of the stage and blanker and the blanker tension are saved into an eeprom memory when the device is switched off. This is indicated by a brief flash of the LED in the button.

9.6 Display and Control Unit

The Display and Control Unit (DCU) is the "joystick box" that is used to control the stage or blanker when this is not done through the USB connection. It also displays the blanker tension and position or the stage position.

It has two buttons, a joystick and a two-line LCD. One of the buttons also has an LED.

9.6.1 Mode Switching

The left *Mode* button switches between a stage control mode and a blanker status display mode.

9.7 Controlling the Stage

If the display does not display the x and y positions, press the *Mode* button until it does. The stage can then be moved with the joystick.

Three different speeds—fast, medium and slow—can be selected for use at different magnification. The currently selected speed is indicated in the display between the `X` and `Y` and the number with a letter `F`, `M` or `S`. The speed is switched with presses of the right *Change* button.

9.7.1 Parameter Discovery

When the stage is new, it may be necessary to "train" the stage, so that in runs smoothly. This is done by slightly deflecting the joystick in the four directions, up, down, left and right, and waiting for the stage to start moving. This process discovers some regulating parameters that are used by the speed control feedback loop. This parameter discovery should be done at all three speed settings.

9.7.2 Hard Limits

When the stage unexpectedly stops during a move, it probably has reached its hard limit, the border of the area of movement. This is indicated by a `B`[3] replacing the speed indication in the display. In this case the stage can not be moved with either the joystick or through USB commands. To unblock the stage press the right "Change" button for at least two seconds untill the `B` is replaced with the speed indication again.

9.7.3 Stage Reset

If a stage reset is necessary, press the *Mode* button for about two seconds. This will trigger the reset procedure in which the stage calibrates itself and returns to its home position. During the calibration process the stage will move to different positions, as can be seen by the changing x and y coordinates on the LCD.

9.7.4 Velocity Warnings

When the stage table is moved too fast, the relative positioning system may lose track of the true position, losing count of the distance from the origin (position 0,0). This is indicated by a "`V`" in the display.

This will happen when the stage is moved by had with the electronics running. Perform a stage reset in this case or press the "Change" button for two seconds to just clear the warning.

During normal operation, when moving the stage with the joystick or via USB, there should not be any velocity warnings. When the "`V`" appears repeatedly, please contact customer service.

9.8 Controlling the Blanker

To go to the blanker control mode, press the *Mode* switch until "`Blanker`" appears in the display. In this mode, the blanker can be moved through the three positions, in (blanking), cup (probe current detection), and out (retracted out of the beam path) by presses on the *Change* button.

9.9 Setting Up the Blanker

To set the blanker and PCD positions, the blanker controller must be changed into *Adjust* mode. When in *Blanker* mode, press the *Mode* button for about two seconds, until "`Adjust`" appears in the display.

In *Adjust* mode the blanker arm can be moved with the *Change* button the same way as in *Blanker* mode, cycling through the out, blank and cup positions. It can also be moved with the joystick. Moving the joystick to the left moves the blanker arm out of the beam into the blanker housing, moving it to the right moves the arm in the other direction.

Setting up a position is done by first selecting that position with the *Change* button and then moving the blanker arm to the correct position with the joystick.

[3] for "blocked"

9.9.1 Out Position

Setting the out position is both important and simple. Move the joystick all the way to the left and wait until the blanker arm has reached the limit. A few seconds after reaching the limit the XCC recognizes that the arm has stopped and sets this position to 0.

If the position value does not go to zero automatically, press the *Mode* button until it does. (See section 9.9.4)

Setting the out position is important, because it is used as the origin (position 0). The other two positions use this position as reference.

9.9.2 Blanking Position

After setting the out position, the blanking position should be set. For this a live SEM image is needed. In the out position the image can be seen normally.

Press the *Change* button to move the blanker arm to the old blanker position. At some time the image disappears when the tip of the arm blocks the beam. When the arm is moved further, either during the automatic move or when the arm is moved with the joystick, the beam will briefly go through the blanker slit. During this time the SEM image will briefly appear and disappear again. The arm should then be moved around slowly until the position of the blanker slit is found, that is, until the SEM image appears again. The arm should be position so that the electron beam falls through the middle of the blanker slit. Note the values shown on the display for the two positions where the image appears and disappears at the two edges of the slit and move the arm to the position directly in the middle between these values.

While doing this set-up, make sure that no signal to blank the beam goes into the blanker. Select "Beam on" in the "Beam control" dialog when ECP is running. When in doubt, disconnect the fiber optic blanker line at either end during the set-up.

After successfully setting up the position press the *Change* button until the LED in the button flashes to write the new position to eeprom. Reconnect the blanker signal line if you disconnected it earlier.

9.9.3 BCD Position

Setting the BCD (Faraday cup) position is done by watching the beam current, measured with a picoamperemeter, while moving the blanker arm. The blanker is in the BCD position when the current measured is highest. The Faraday cup position value is about 8 mm less than the blanking position (arm withdrawn 8 mm into the blanker housing).

The Faraday cup itself is about 0.2 mm wide and it is located inside a 2 mm wide gap in the shielding of the blanker arm assembly. This means that there may be two jumps in the current measured when searching for the cup. At first the current will go up from a low to a medium value when the beam goes from the shielding to the material of the cup. When moving about one millimeter further the current will go up to a higher value and go down to the medium value when moving a short distance more. This second jump is an indication that the beam fell into the Faraday cup. This area of the highest measured current is the right position for the PCD.

Press the *Change* button until the LED in the button flashes here, too, to write the new position to eeprom after successfully setting up the position.

9.9.4 Resetting Position Memory

If the position is not sets itself to zero when trying to move past the "out" mechanical limit (see section 9.9.1), the counter can be reset by hand by a long press of the *Mode* button.

9.9.5 Setting Blanker Tension

In the blanker *Adjust* mode, the blanker tension (voltage) is set by moving the joystick up or down. What blanker tension to use depends on the acceleration tension used on the SEM.

To select the right blanker tension, the blanker must be working. That is, the arm must be moved to the blanker position and there must be a way to switch on and off the blanker signal. This can be done with ECP and the XeDraw2. In ECP go to the "EBL" menu and select "Beam control...", the blanker can be switched off and on with the "Beam on" and "Beam off" radio buttons. (Beam on means the blanker signal is off/low, beam off means the blanker signal is on/high.)

Switch blanking on ("Beam off") and, starting with low values, slowly increase the blanker tension by moving the joystick up until the SEM images completely disappears. On most SEM types the image will simply go black, but on some SEMs the image may go white instead. This is caused by the arrangement of the secondary electron detector close to the blanker. With an "in lens detector" scattered low-energy electrons can reach the detector and register as a signal that is displayed as white. This has no influence on the effectiveness of the blanker, none of the high-energy electrons will reach the sample when it is set up correctly.

To check that the blanker is working, toggle the blanker signal; The SEM image should appear and disappear.

9.9.6 Leaving Adjust Mode

To leave the *Adjust* mode press the *Mode* button.

10 General Controls

Some general functions of Ecp that can be activated from the menu bar even when no file is loaded are described below.

10.1 SEM Image Scan

The Sem Image Scan can be used to get an image of the sample. You can open this dialog with the "EBL/SEM Image..." menu item or by clicking the button, see figure 10.1

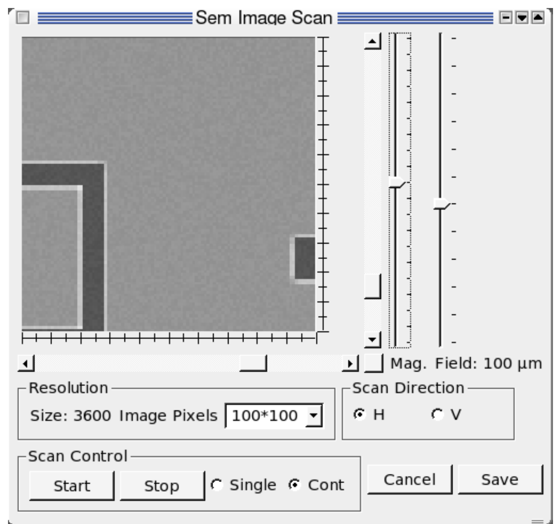


Figure 10.1: The Sem Image Scan window (simulated image)

You can choose between three scan sizes, 100×100, 200×200 or 400×400 pixel and between horizontal ("H") and vertical ("V") scan. You can also start and stop the scans or scan the area just once by selecting "Single" and pressing "Start".

The slider to the right of the image area labeled "Mag." controls the magnification, that is the distance between the dots on the sample where the value is recorded. At lowest magnification the pixel are taken from a whole exposure field, at highest magnification and highest resolution no pixel are skipped. The magnification can also be changed by a clicking into the image area (a left click and a mouse movement zoom in, a right click zooms out) or with he mouse wheel. Changing the magnification or clicking on the "Mag." slider restarts a stopped scan.

When the magnification is not at the lowest, the horizontal and vertical scrollbars can be used to pan around in the image. Pressing the button to the right and below the image re-centers these scrollbars.

The slider labeled "Field" controls the field size if an attenuator is connected to the Xe-Draw2.

The "Cancel" button closes the SEM scan window, the "Save" button can be used to save the current image. An image of a real sample saved with this function can be seen in figure 10.2.

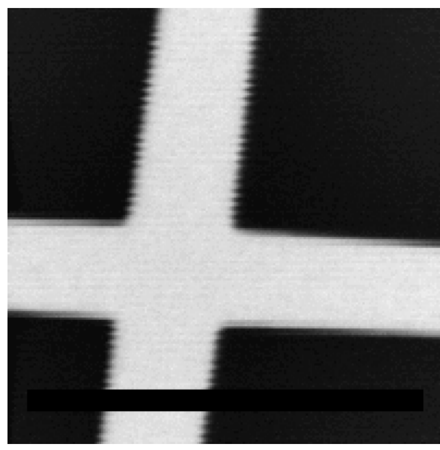


Figure 10.2: Image of a cross mark, resolution 200×200 pixel

N. B. Starting an exposure closes the SEM Image Scan window. This reduces the risk of destroying an exposed structure by accidentally scanning over it.
Move the sample after an exposure with the Stage control or by adding a stage move command after exposing the last field in a ctl file before reopening the SEM Image Scan window.

N. B. When a scanned image is completely black, it may be that the beam is blanked. Ensure that the blanker control (see 10.2) is set to "Beam On".

10.2 Beam Control

The "Beam Control" dialog can be activated from the "EBL" menu, as well.

The appearance of this dialog depends on the hardware installed. The "Blanker Control" is always available. By clicking on "Beam Off" you can blank the beam, for example to move the stage without exposing sensitive parts of the sample. Click on "Beam On" to turn the beam on.

When a probe current detector (PCD) or a moveable blanker is installed you can move it with the position control buttons. There are three possible positions, but depending on the hardware some of them may not be available. The positions are

Out The PCD or blanker is moved out of the beam.

Blanker The PCD is moved to blanking position. If there is no blanker, this is the same as the *Out* position.

PCD The PCD is in the probe current detection position and the current is measured. The value is shown below the buttons.

10.3 Dose Calculator

The increments and dwell times used in the pat and ctl files are somewhat unwieldy. To get a pair of values that matches a desired exposure dose Ecp provides the "Dose Calculator" from the "Extras" menu. It can calculate parameters for an area, a line or a dot. The "clock", "pixel", "field size" and "current" parameters are initially set to the default values. To use the

calculator enter the desired values. Clicking on "Calculate dwell time" gives values for the time and increment that can be used in a pat file. The calculator can also be used to calculate the dose that will be written when exposing with a set of given parameters. Enter a dwell time and increment and click on "Calculate dose" for this.

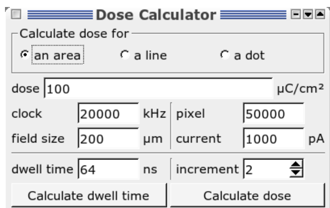


Figure 10.3: Calculate the exposure dose

11 Calibration

When stitching together different fields it is important that the shapes drawn by the electron beam have exactly the right size and are not tilted. To assure this it is necessary to calibrate the scaling (“Scale X” and “Scale Y”) and rotation (“Rot X” and “Rot Y”) using a calibration mark. On defining a calibration mark see Chapter 7.

11.1 Field Calibration

To perform a field calibration a mark must be centered in the SEM image. Use the stage control (see chapter 8) to achieve this. Remember to turn off the beam if you move across parts of the sample you wish to write on. A field calibration can be performed on a cross mark or on an image mark. Since the image is recorded when the calibration is started, any recognizable shape on the sample can be used, for example a grain of dust.

When using very clean samples without predefined marks you can mount a second sample with a mark at exactly the same height, scratch the resist and use parts of the scratch as image mark or rely on calibration data recorded earlier.

When you select the menu item “Calibrate/Field Calibration” you must select a file describing the mark you wish to use. On systems that can change the writing field size you are first asked for the field size to calibrate. The dialog is preset to the current field size. The field size is than set to this value either by setting the magnification on a SEM or by setting an attenuator. Clicking “OK” immediately sets the selected field size, possibly loading an old field calibration from `default_xxxx.cal`, “Cancel” does not change the field size or load any calibration data.

When the mark description is loaded the “Field calibration” dialog appears. Click “Calibrate” to start the calibration. the “Test Mark” dialog is shown.

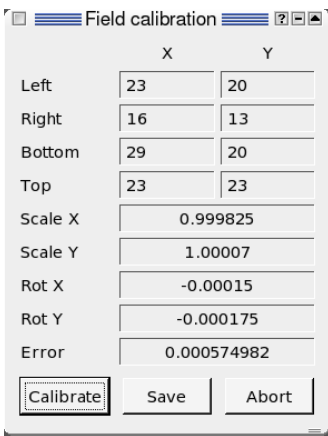


Figure 11.1: Field Calibration dialog after a calibration

Select “OK” when done. This starts the first run of the calibration. The result is shown in the field calibration dialog. This calibration is an iterative process, so clicking “Calibrate” a

few more times may give greater precision. When the error is small enough, click "Save" and select a file name to save the calibration. The suggested file name `default_xxxx.cal` is the one ECP uses when changing to field size xxxx when "Load calibration on field size change" is active in the settings. Clicking "Abort" closes the calibration dialog.

11.2 Working Distance Correction

The "Working Distance Correction" works by storing the focus setting, usually called "working distance" at three stage positions. For best results these points should be outside the sample at three of its corners or some distance away from the sample. Move the sample to these positions using ECP's (see section 8.1) or the SEM's stage control. At each point take great care to get the best focus value,[1] as this is used as the base for setting the focus at other points. After setting the focus press the "Submit" button. The current stage (x and y) and focus (or working distance or z) values are stored and displayed.

When three points have been defiend ECP uses the values to define the plane of the sample surface and calculate the working distance at other places of the sample. To check this function, move to another point outside the actual writing area and click "Calculate". Setting the working distance to the value shown as "Z - Position" in the fourth line should bring the image into focus.

To use the correction in a ctl file use `setfocus(INIT)` and `setfocus(AUTO)`, see section 5.2.46.

Leave the window open (or minimized/iconified) when you want to check the focus values ECP sets or close it to minimize the communication between ECP and the SEM.

> **N. B.** After closing the window, selecting the "Working Distance Correction..." menu item deletes the stored values and ECP is ready to newly define the correction.

11.3 Global Marks

When you have more than one mark on your sample, you can use the distance and angle between them for a calibration with the global mark commands in a .ctl file.

These global marks can be used to find structures again when exposing a sample more than once, for example when exposing different layers on a chip, one after the other.

11.3.1 Setting Up Global Marks

To use global marks define the positions of the marks in the ctl file using `gmcoord` commands. These coordinates are in the local coordinate system, that is the values specify the distance from the point defined with `origin`. You need to define at least two mark positions for a calibration, but with three marks the calibration for x and y rotation become independent. Using more than three marks increases the precision.

The mark description file used is defined with `gmfile`. The file used must be in the same directory as the ctl file. Use the menu item "Mark/Make global" to copy it there. The global mark functions work with both cross and image marks. The image mark for a global mark scan, especially an automatic global mark scan, should be defined by geometry; during a

[1] When there is an range of focus values that all give the sharpest image, select a value in the middle of this range.

global mark detection the image for a scan image mark is recorded at the preset stage position and always accepted as a valid mark there.

11.3.2 Aligning the Sample

After defining a mark definition with `gmfile` and some `gmcoord` commands, use `gmark` to align the exposure fields to the sample. ECP determines the real positions of the marks, the distances between the real and the expected positions and calculates and applies correction factors for offset, rotation and scaling, each in x and y direction separately. With these corrections successive layers are exposed precisely aligned.

There are two ways to let ECP perform these corrections: automatically and with user interaction. These are activated with the commands `gmark(AUTO)` and `gmark(MANUAL)`. In automatic mode ECP moves to all mark positions and tries to find the marks automatically. This can be used when the sample position is already known with some precision. When ECP can not find a cross marks increasing the scan length in the mark definition gives more leeway to find marks a little off the expected position.

In manual mode ECP moves to the specified position and opens a SEM scan window and the stage control.

> **N. B.** In older versions of ECP you have to open the stage control window before starting an exposure with a manual global mark scan.

You can than precisely center the mark in the SEM image. After clicking "Okay" to the "Stage position okay?" question ECP performs a mark test (see section 7.2) and tries to detect the mark when the test window has been closed. When the mark could not be found ECP offers the possibilities to "Retry" positioning and detecting the mark, to "Skip" this mark or to "Abort" the exposure.

When less than two marks are found during the automatic or manual mark detection ECP stops the exposure with an error.

With two marks found ECP calculates four correction parameters, one for scaling in both direction, one for x and y rotation and two for the offset or shift, in x- and y-direction.

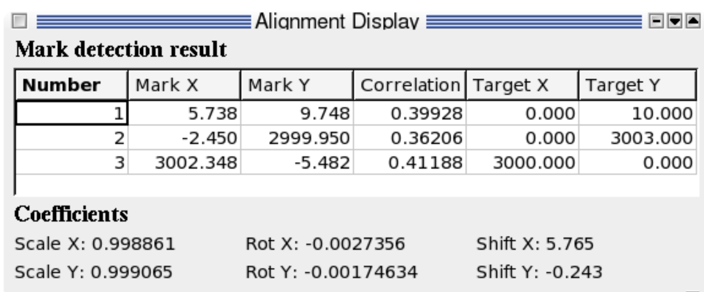
When three or more marks are found, six parameters are calculated, scaling, rotation and shift, each independently for x- and y-direction.

After the alignment the expected ("Target") and real ("Mark") positions of the marks and the parameters calculated from these are shown in a "Alignment Display" window. See figure 11.2. When a mark has not been detected, the coordinates at "Mark" are replaced with "`----`". For an image mark the correlation amplitude detected during the scan is also displayed. This can be used to gauge the quality of the different marks. For a cross mark this column shows only "`--`" as no correlation is calculated.

11.3.3 Repeating Alignment

After performing a global mark detection with `gmark` there are two ways to perform another detection.

Use `align` to reuse the mark definitions and to perform another automatic global mark detection with the marks used by the last `gmark`. For example this can be useful to correct a drift caused by a stage that expands as it gets warmer. You can set up and perform a global mark detection with `gmark` at the beginning of an exposure and use `align` several times during a long exposure to correct the drift.



Number	Mark X	Mark Y	Correlation	Target X	Target Y
1	5.738	9.748	0.39928	0.000	10.000
2	-2.450	2999.950	0.36206	0.000	3003.000
3	3002.348	-5.482	0.41188	3000.000	0.000

Figure 11.2: Display of alignment results

The other way to perform more than one alignment is to define new marks with new `gmcoord` commands and calling `gmark` again. This can be useful when there is a grid of marks on the sample. You can than perform an alignment with marks right next to the current writing position.

To define the positions of the marks it is convenient to places one mark at the origin (`gmcoord = 0, 0`) and define all positions, those of the other marks and of the actual pattern, relative to this mark. After mounting the sample with roughly the right orientation just move to this mark, load the control file, select the menu item "Tools/Map exposure origin", perform or load a field calibration and start the exposure.

11.4 Using a Reference Mark

Drift on a sample without marks can be corrected with `rmark`. This command defines the position of a reference mark. A scratch in the sample or a grain of dust may be used as a reference mark. First define the mark definition with `gmfile`. Use an image mark and record an image at the position of the reference mark with "Mark/Detect Mark" and "Detect". Set the position of the mark in the ctl file with `rmark = x, y`.

> **N. B.** The position of the reference mark is set in the coordinate system of the sample, not of the stage. You may want to use the position of the mark as the exposure origin: set `rmark = 0, 0` and use "Tools/Map exposure origin".

To correct a drift use `align` later during the exposure. This moves the stage to the position of the mark, checks the exact position and calculates the difference between the two. The target of the next `stage` move command is corrected for this difference.

11.5 Chip Marks

To get a precise alignment in one writing field, ECP can perform a chip mark detection. This is very similar to a global mark detection, but the marks are located in the writing field and the beam is moved to scan the marks, not the stage.

Just like global marks the positions of chip marks must be defined with `cmcoord`. The positions are the target positions of the center of the mark in pixels, as in pat file descriptions. The mark description file must be specified with `cmfile`; the same rule as for global mark files applies, that is the chip mark file must be in the directory of the ctl file. A chip

mark scan is started with `cmark`. As chip mark scans are always performed in an automatic manner, no mode ("AUTO" or "MANUAL") is given. The results of the global mark scan are shown in the same "Alignment Display" window as the results of global mark scans. A global mark scan can be repeated (e. g. after moving the stage to a new writing field) simply by repeating `cmark`. As with global marks, repeating `cmark` without giving new coordinates with `cmcoord` reuses the coordinates. This is useful when many writing fields have their chip marks at the same positions. New coordinates can be given by giving new `cmcoord`s. The next `cmark` will use these coordinates.

A chip mark scan calculates and then uses parameters for rotation, scaling and shift.

12 Settings

The settings menu can be accessed from the "System" menu.

12.1 Calibration

This menu configures the automatic calibration procedure (Auto Calibration), which is invoked from the "Calibration" menu.

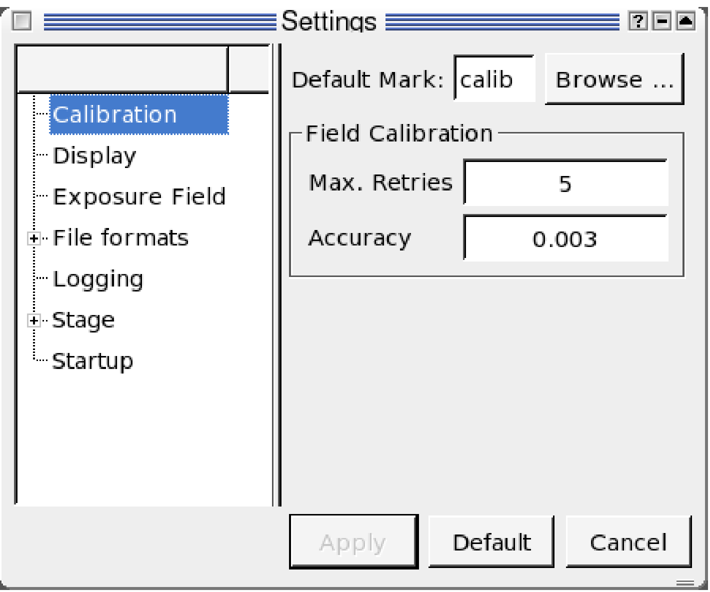


Figure 12.1: Screenshot of the calibration configuration.

The "Default Mark" is the mark used for calibration process.

"Max. Retries" and "Accuracy" specify the number of calibration attempts and the maximum calibration error for automatic field caribration. If you have a system with a double deflector, the error and attempts can be specified separately for the main- and subfield. The calibration procedure is repeated until the residual error is below the specified error or the maximum number of attempts is reached.

On systems where the field size can be changed the "Load calibration on field size change" box determines if calibration data should be loaded on a field size change (checked) or if the calibration should be reset (not checked). See section 5.2.22.

12.2 Exposure Field Settings

This menu configures the field size and the number of pixels in the field. Normally, you do not need to change the settings for every exposure. You will see settings for the exposure fields like shown in figure 12.2.

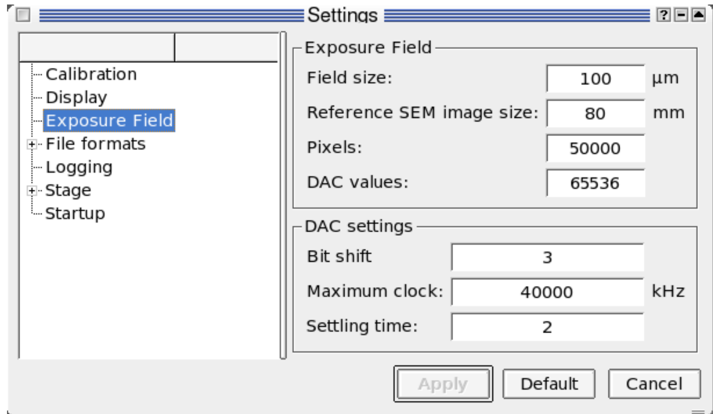


Figure 12.2: Screenshot of the field configuration

In the first box you can set the default field size, that is the size of the writing field used, when it is not set in a ctl file (“`fsize`”, section 5.2.22).

“Reference SEM image height:” sets the reference size used for calculating the magnification. When this is set to the default value of 80 mm,and the magnification of the SEM is set to, for example, 800, the SEM image would show an area 100 µm (= 80 000 µm/800) high.

The “Pixels:” field defines how many pixel a shape defined in a pat file can have in either direction. For single deflection systems ”Exposure field” has a fourth setting, the “DAC values:”. This sets the real values that can be addressed by the main dac. It is usually $2^{16} = 65\,536$ and should not be changed. The “Pixels” value must be smaller than this value.

Apart from the reference image size and the field size, these values should usually not be changed.

The last few things are the DAC settings. Normally you should not change anything here. We normally use a bitshift of three at sending the data from the PC to the XeDraw2. The maximum clock can be set to 10 MHz. The settling time is set to two.

12.3 File Formats

This menu configures the default display style for the different file formats supported by the ECP. You can also change the display mode manually once you open a file.

If the option “Fill” is checked, the patterns are filled. Otherwise, only the outline is drawn. If the “Transparent” option is checked, the patterns are drawn in a transparent mode. In other words, you can see through different layers. This is mainly useful for file formats with several layers (OVL and GDS files).

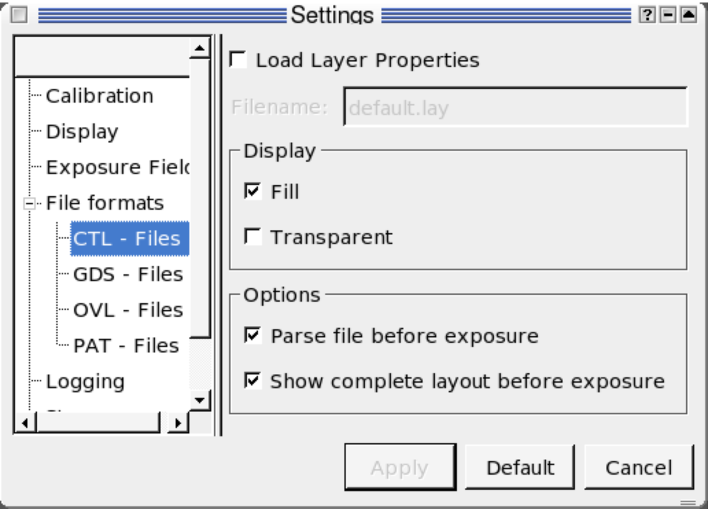


Figure 12.3: Screenshot of the File formats configuration.

12.4 Logging

When you change to the *Logging*-field, you can see figure 12.4. The software can keep track of the important parameters. With "Log mark detection", you log the coordinates of the detected marks, the calculated rotation, scale and shift factors. When "Log calibration" is set, you get all the calibration values like used mark, detection level and so on saved. When you enable "Log exposure", the start and finish time are logged. When you use the automatic correction of the current by checking with the Faraday cup, you get the correction factor and the time when the dwell time was changed if "Log current" is enabled. You can check whether the quality of your sample improved or degraded by tracking the current.

12.5 Stage

This menu configures the stage.

12.5.1 Stage Coordinates

The four values in the box "Limits" are minimum and maximum values of the stage coordinates.

The coordinate of the exchange position, that is the position used by an automatic sample handling system, can be edited in the box "Exchange position".

"Home position" is useful when the system needs to go back to a specific point.

When a Faraday cup is installed on the stage instead of in the beam column, the cup position can be specified with "Faraday Cup Position".

The orientation of the coordinate axes can be specified in the axis orientation boxes. "Positive" means the usual orientation (x axis points right, y axis points up).

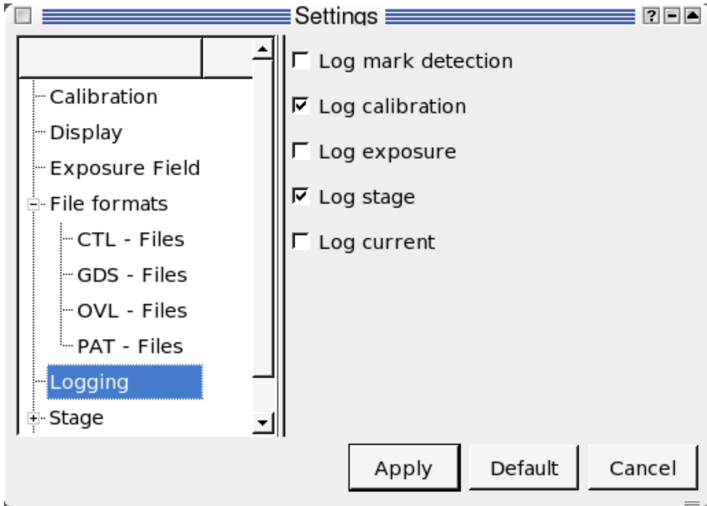


Figure 12.4: Screenshot of the logging configuration

12.5.2 Laser Stage

Coarse positioning refers to adjusting the laser within tolerance. Maximum retries limits how many iterations the system tries. The system will stop when it is within tolerance or it has tried too many times. The axis angles are not very critical. They are just extra offsets for faster calibration.

12.6 SEM Interface

When the SEM is connected in a way that can be configured, that is done here. This is the case with SEMs connected through a serial line or through an tcp/ip connection *not* using a special "dll"-library.

For serial connections the device can be set in the "Serial port name" box. For Windows this is usually `COM1`, `COM2` or something like that. It can depend on the Unix or Linux flavors used. For the Version initially installed, SuSE GNU/Linux, it should be something like"`/dev/ttyS0`" or "`/dev/ttyS1`".

For SEMs that use a IP connection there are two fields: "Host name" and "IP port number". The host name can be entered as a numeric ip address or as a symbolic host name that can be resolved by the computer running ECP. Since the SEM is usually on a small private network, any name resolution must be done by the computer running ECP. The "IP port number" is the number the SEM is listening on for connections.

Usually none of these controls need to be changed once the system has been set up by Xenos staff.

12.7 Startup

This menu configures the actions ECP performs at startup.

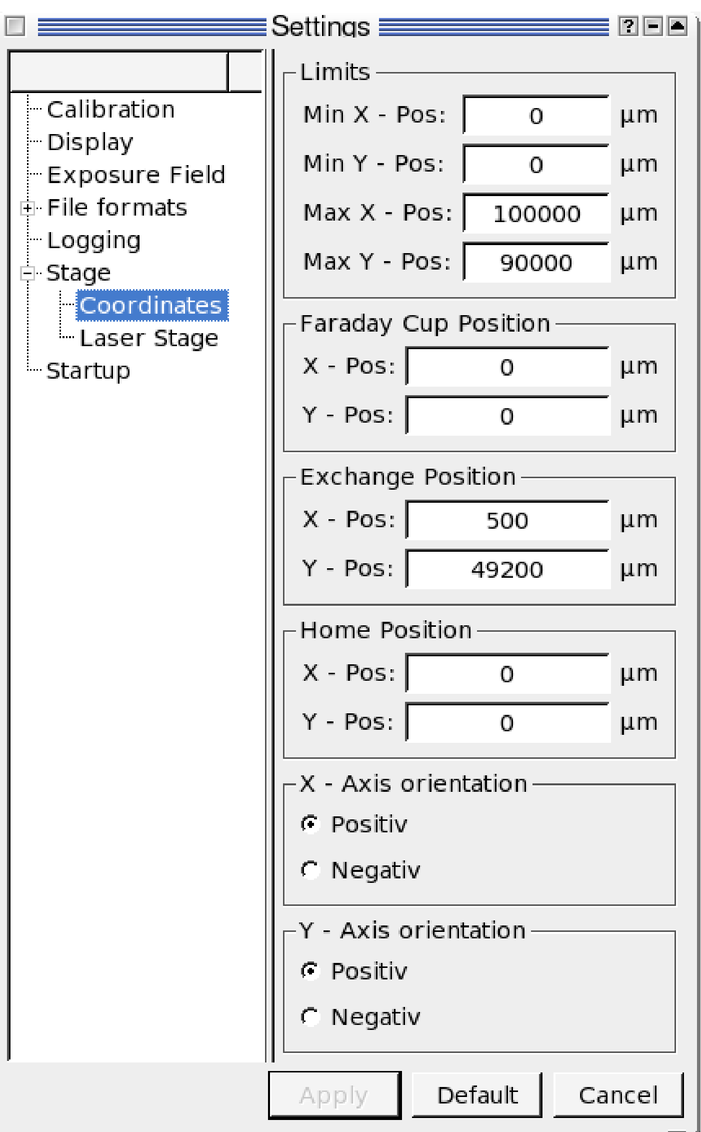


Figure 12.5: Screenshot of the stage coordinates settings

Show splash screen	—	Show the logo during startup.
Load PGN kernel	—	Load the dsp kernel of the XeDraw2 during startup. Loading the kernel is only necessary if the XeDraw2 has been switched off. The kernel can also be loaded at a later stage using the "Load XeDraw2 kernel" function in the "Systems" menu.
Reset PGN	—	Reset the XeDraw2 at startup time.
Reset stage	—	Perform a stage reset at startup time.

Table 12.1: Startup Settings

When the "Init calibration" box is checked ECP either resets the field calibration, that is it sets x and y scale to 1 and x and y rotation to 0 or loads the last saved calibration data, depending on which radio button is selected

"Init focus correction" works like "Init calibration", resetting or loading the focus correction parameters.

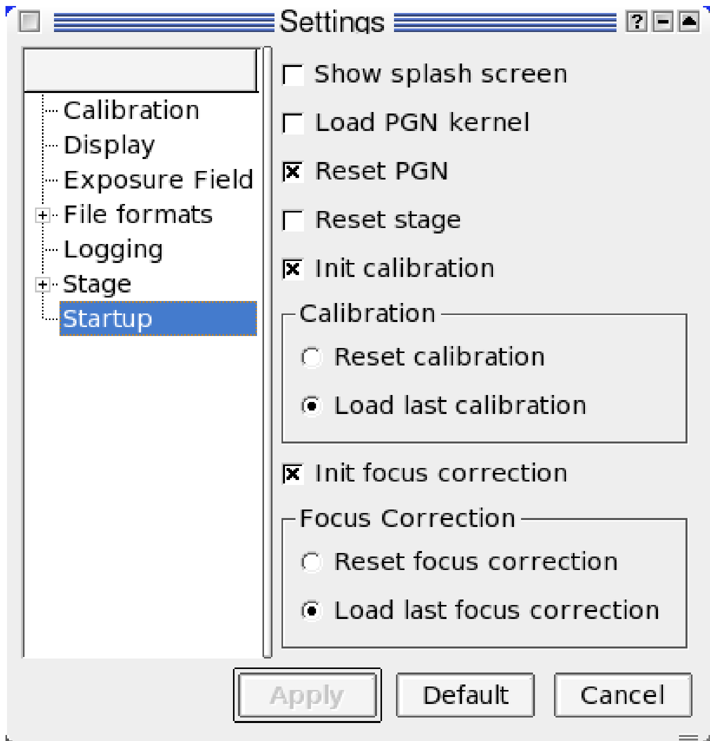


Figure 12.6: Screenshot of the startup configuration

Appendix

A Key Shortcuts

A.1 Global shortcuts

Key	—	Action
Alt+F	—	Opens the File menu
Alt+E	—	Opens the Edit menu
Alt+V	—	Opens the View menu
Alt+T	—	Opens the Tools menu
Alt+C	—	Opens the Calibration menu
Alt+M	—	Opens the Mark menu
Alt+S	—	Opens the System menu
Alt+H	—	Opens the System menu
Ctrl+N	—	New File
Ctrl+O	—	Open File
Ctrl+S	—	Save File
Ctrl+P	—	Print File
Ctrl+W	—	Close File
Ctrl+Q	—	Quit
Ctrl+C	—	Copy
Ctrl+X	—	Cut
Ctrl+V	—	Paste
F5	—	Redraw
Ctrl+-	—	Zoom out
Ctrl++	—	Zoom in

A.2 Text Editor

The text editor has the following shortcuts:

Key	—	Action
Backspace	—	Delete the character to the left of the cursor
Delete	—	Delete the character to the right of the cursor
Ctrl+A	—	Select the whole text (all).
Ctrl+B	—	Move the cursor one character left
Ctrl+C	—	Copy the marked text to the clipboard
Ctrl+D	—	Delete the character to the right of the cursor
Ctrl+E	—	Move the cursor to the end of the line
Ctrl+F	—	Move the cursor one character right
Ctrl+H	—	Delete the character to the left of the cursor
Ctrl+K	—	Delete to end of line
Ctrl+N	—	Move the cursor one line down
Ctrl+P	—	Move the cursor one line up
Ctrl+V	—	Paste the clipboard text to the cursor position
Ctrl+X	—	Cut the marked text to the clipboard
Ctrl+Z	—	Undo the last operation
Ctrl+Y	—	Redo the last operation
Ctrl+Insert	—	Copy the marked text to the clipboard (Windows only)
Shift+Insert	—	Paste the clipboard text to the cursor position (Windows only)
Ctrl+Delete	—	Cut the marked text to the clipboard (Windows only)
Left Arrow	—	Move the cursor one character left
Ctrl+Left Arrow	—	Move the cursor one word left
Right Arrow	—	Move the cursor one character right
Ctrl+Right Arrow	—	Move the cursor one word right
Up Arrow	—	Move the cursor one line up
Ctrl+Up Arrow	—	Move the cursor one word up
Down Arrow	—	Move the cursor one line down
Ctrl+Down Arrow	—	Move the cursor one word down
Page Up	—	Move the cursor one page up
Page Down	—	Move the cursor one page down
Home	—	Move the cursor to the beginning of the line
Ctrl+Home	—	Move the cursor to the beginning of the text
End	—	Move the cursor to the end of the line
Ctrl+End	—	Move the cursor to the end of the text
Shift+Wheel	—	Scroll the page horizontally
Ctrl+Wheel	—	Zoom the text

A.3 Stage Control

Use these keys for manual move in the stage control, the keys are shown as they are arranged on the numpad.

	/ or ÷ Decrease step size	* or × Increase step size
7 Move up and left	8 Move up	9 Move up and right
4 Move left		6 Move right
1 Move down and left	2 Move down	3 Move down and right

B Pattern reference

All patterns can be repeated in x and y direction. In this case, the pattern data is followed by four more number, which specify the period in x-direction, the period in y-direction, the number of repetitions in x-direction and the number of repetitions in y-direction. To use repetition, all optional arguments must be given between the regular arguments and the repetition arguments.

`D` name	—	Pattern name
`I` X	—	Increment
`C` X	—	Dwell time in nanoseconds
`RECT` x1, y1, x2, y2	—	Rectangle
`LINE` x1, y1, x2, y2	—	Line
`XPOLY` x1, y1, x2, x3, x4, y2	—	Trapezoid, horizontal edges
`YPOLY` x1, y1, y2, y3, x2, y4	—	Trapezoid, vertical edges
`DOT` x, y	—	Dot
`CIRCLE` x, y, r	—	Circle
`RING` x, y, r, w(, s, e)	—	Ring or ring segment
`RDOT` x, y, rep	—	Repeated dot
`MSDOT` x, y, t	—	Dot exposed for t milliseconds
`SQUARENUT` x, y, w, h, r,	—	Rectangle with circular hole
`HEXNUT` x, y, o, h, r,	—	Hexagon with circular hole and vertical sides
`HEXZUT` x, y, o, h, r,	—	Hexagon with circular hole and horizontal sides

B.1 Numerical shape codes

Older versions of ECP used numbers to identify shape codes. Old pat files using these numbers can still be used, but using shape names makes pat files more readable. The number codes have these meanings:

```
P 0,   — RECT
P 1,   — LINE
P 2,   — XPOLY
P 3,   — YPOLY
P 4,   — DOT
P 5,   — CIRCLE
P 7,   — RING
P 9,   — RDOT
P 21,  — HEXNUT
P 22,  — HEXZUT
P 23,  — SQUARENUT
```

N. B. The shape code number is parsed as another numeric argument, so lines using the `P` format have one comma more than lines using a mnemonic to identify the shape. For example `RECT 1000, 2000, 3000, 4000` is equivalent to `P 0, 1000, 2000, 3000, 4000`

C CTL Reference

adraw(b, i, j)	—	draw pattern with a name of basename b and the values of i and j
align	—	repeat aligment using global marks or reference mark
autorot	—	calculates the rotation with 2 points
bid_off_time = t	—	sets the BID mode beam off time to t ms
binfile	—	specifies the binary pattern where the patterns are found
cmark	—	perform a chip mark detection
cmcoord = x,y	—	specify the coordinates of one chip mark
cmfile = fn	—	specify the filename for global mark scan
current = X	—	use a current of X pA to calculate the dose
dose_table = fn	—	use the file fn.dtb as a dose table for image_draw
draw (X, I, C)	—	exposes pattern X, optional parameter: I = increment, C = dwell time
end	—	specifies the end of the file
fcup	—	move the cup in, calculate correction factor for dwell time (in the case that the current is not stable)
fcup_in	—	move the cup into the beam path
fcup_in	—	move the cup out of the beam path
for a = 1 to X step Y	—	loop with variable a, 1 start, X target, Y step size
fscale = X	—	scale the exposure field size by X
fsize = X	—	fieldsize for dynamic change (for SEM)
gdraw (H, P)	—	draw a horizontal grating with a period of P nm
ginc = I	—	Set the increment for a grating to I
glength = X	—	Length for grating in µm
gline = X	—	Width of one line in an array in pixels
gline = X	—	Width of a grating in pixels
gmark({MANUAL, AUTO})	—	global mark scan, either manually or automatically
gmcoord = x,y	—	specifies one global mark
gmfile = fn	—	specify the filename for global mark scan
gtime = C	—	Set the dwell time for gratings to C ns
gwidth = X	—	Width for grating
idraw	—	indicated drawing (rect1, rect2, rect3, . . .) with variable as pattern name
image_draw (fn)	—	Draw the image file fn (specify the whole file name, usually with an ending like .png or .jpeg)
image_pixsize = x	—	draw one image pixel as a square of $x \times x$nm^2

mode = x	—	sets the exposure mode: direct_draw = do not correct stage coordinates, mask_draw = normal coordinate mapping, ns = nanosecond mode, ms = millisecond mode, sii = single ion nmode, bid = beam induced deposition mode
next a	—	step to the next repetition of a
origin	—	specify origin coordinates for exposure
pause	—	pauses the exposure and waits for user input
rmark = X,Y	—	Define a reference mark
rot = X	—	specifies the rotation (only for small angles)
setfocus ({INIT, AUTO, OFF, NOW})	—	only for LEO stages
sfile = X	—	specifies the pat (or source) file
sii_timeout = t	—	sets the timeout in SII mode to t ms
stage	—	moves the stage
x, y, z = X	—	Position to move the stage to
±x, ±y, ±z = X	—	Value to move the stage by
wait	—	pauses the exposure for the specified time in milliseconds

D Step-by-Step Startup Lists

When starting up ECP it is important to switch on or power cycle the XeDraw2 before starting the software. The following check lists can be used when starting or restarting ECP

D.1 Startup List

1. Boot the PC and log in
2. Switch on the XeDraw2, wait for the power LEDs to come on
3. Start ECP

D.2 Restart List

When restarting ECP the XeDraw2 should be powered down and up again:

1. Switch off the XeDraw2
2. Wait for a few seconds, at least until all power LEDs have gone out.
3. Switch the XeDraw2 on again, and wait for the power LEDs to come on.
4. Restart ECP
5. If a dialog appears with a warning "Lock file NN exists! This usually means another instance of ECP is running. Disable I/O interfaces?" check if another ECP is running.
6. If another ECP is found the old one has now lost the connection to the XeDraw2 and should be shut down, after saving files if necessary.
7. In the new ECP's dialog box click on "Use I/O anyway"

When the "active" LED does not come on after starting or restarting ECP,[1] the kernel must be loaded manually with these steps:

- Activate the "System/Boot XeDraw2 kernel ..." menu
- Click on the open button in the dialog to boot the standard kernel.

When the XeDraw2 is switched on and no kernel is loaded, the pattern generator may take over the scan controll, causing a black screen in the SEM software untill the kernel is loaded.

N. B. Start ECP and load the XeDraw2 kernel immediately after starting the pattern generator when "Load XeDraw2 kernel" is not checked in the settings.

[1] The LED should come on automatically when "Load XeDraw2 kernel" is checked on the "Startup" page in the settings.

E Notes on Third Party Software

Some versions of ECP contain software with additional, free licenses.

E.1 Kiss FFT

ECP uses "Kiss FFT". With respect to Kiss FFT, please note the following:

Copyright © 2003–2004, Mark Borgerding

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the author nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Kiss FFT can be found at `http://freshmeat.net/projects/kissfft/`.