

## Lithography

*This support note shows how to use lithography section of WSxM program. It will describe each one of the menus that appear in the subprogram.*

Lithography is a part of the WSxM program. In this frame you can draw on a sample, define the tip motion and write on a sample.

### Introduction

WSxM Lithography section allows you to move the tip over the sample performing different actions. You can think in the tip as a pen, and the sample as a paper. If you can somehow define what does “draw” mean, you can “draw” with your tip over your sample, (like) as it was a pen. When you have a pen, you can move it far away from the paper and it will draw nothing. The same can be done with WSxM Lithography if you can define what “do not draw” means in your tip-sample system.

When you plot a simple figure, like a line or a set of concatenated lines, the tip will go to the desired starting point. Once reached that point, if you have defined what “draw” means in your system, these actions will be performed. This is what we have called **“Start drawing actions”** (we

will see this feature in more detail in a few lines). In our pen-paper example, this is like moving the pen to the paper until they contact. Then, the tip will move following a path, *drawing*. When the figure is complete, the system will perform “**Finish drawing actions**”. In the pen-paper example the pen goes out from the paper. Now the tip will move to the first point of the next figure, without drawing.

**Drawing a new lithography pattern**

There are two ways to input a lithography pattern:

1. Graphically. This is the easiest way and the one that we recommend.
  - 1.1. First of all, if the *design Window* is not open, you have to open it by pressing the **Design** command in the View menu or its associated button in the toolbar.
  - 1.2. In the settings dialog-bar you can change the size of the drawing area. This dialog contains some figure editing and simulating information, such as the size of the drawing area, the speed of the movements during the simulation process and a check-box that allows showing or hiding the position of the tip.
  - 1.3. Now you can plot figures by pressing the Plot menu or the appropriate toolbar command.

2. Editing the associated code.

- 2.1. You can see the code associated to the designed pattern by opening the code Window. You can add lines by typing new code, changing the existing, coping and pasting the existing and so on.
- 2.2. To make effective the changes made in the code window, you may press the Apply button (a check mark) or select this command in the Code menu.
- 2.3. If you do not see the effect of the changes you did, you can verify the compilation results window. This window will tell you if there is any syntactic error in your code. You can open it by pressing the toolbar button or the command in the View menu.

You can select a background image for your lithography pattern. When you open an image from the lithography main frame, this image will be placed in the background of the design window and the size in the settings will change. If you want to auto-fit the pattern to the image, you have to check this option in the browser settings. When you go to the new lithography main frame from the acquisition main frame, the image in the selected channel will be placed in the background.

You can save your session or open a saved one with the image browser, but always in the lithography main frame.

If you are editing some code and it has syntactic errors but you want to save it, you can do this by pressing Save Code or the button in the toolbar of the code window. If you save it by pressing **Save**, it will save the pattern in the design window.

**Simulating the lithography session**

After you have finished drawing your lithography pattern you can simulate it. The movement speed of the tip will be the one in the settings dialog. You can execute the whole session, only a single block (i.e. a multiline) or a single simple instruction (i.e. single line in a multiline). The real tip position will not vary while simulating, but there will be a simulated tip position.

**Doing a real lithography session**

There is a dialog-bar in the data acquisition main frame. If you open it, you will see a button to **activate/deactivate** the lithography session (in green if it is active and in gray if it is not) and some buttons to select the execution mode, described below.

If there is a lithography pattern in the main lithography frame, you can activate the lithography session in the acquisition main frame. To do this, press the **On/Off** button. You will see your pattern in the selected channel.

The movement speed is the same as the simulating speed. If you change this value in the Lithography main frame, you'll see the tip movement is faster or slower.

You can execute the whole session, only a line and so on by pressing the desired button.

To continue scanning after doing your lithography session, you may press the **Stop Scanning button** in the main channel window.

**General Frame (Design and Simulation)**

This is the general window that appears when you get into lithography. Menus that appear in (and belong to) this section are the following ones.

**FILE**



**OPEN**

Opens an image or lithography pattern to show it in the design view.

Moreover, you can load an image going to acquisition, and returning to lithography. The image that you have been acquiring will appear then in the view.



**SAVE**

Saves in a file the lithography pattern from the design view.



**REPRESENTATION**

Goes to the WSxM Representation section.



**ACQUISITION**

Goes to the WSxM Acquisition section.

## VIEW



### EDIT STARTING DRAWING CONFIGURATION

Shows start drawing configuration code. A draw configuration lets you decide what is the meaning of “start to draw” on the sample and “finish to draw” on it. In this view, you can introduce instructions that will be performed before drawing each instruction (on beginning).

There is only one restriction: You cannot write here more drawing instructions<sup>1</sup>.

Example: If you want to perform an oxidation process, you may “draw” lines by applying a Bias Voltage. In this case, you can write here the Bias instruction and it will change automatically before plotting every line.

For more information about instructions, see Design.



### EDIT FINISHING DRAWING CONFIGURATION

Shows “finish drawing configuration” code. A draw configuration lets you decide what is the meaning of “start to draw” on the sample and “finish to draw” on it. In this view, you can introduce instructions that will be done after each drawing instruction (on finishing).

Only exists a restriction: You cannot write here more drawing instructions.

In the same example, you should set the bias to 0 or the previous value here.

For more information about instructions, see Design.




### SETTINGS


This dialog contains settings for lithography program. Here, you can choose:


- The **size** of the drawing area (or design window) (in nm)
- **Simulation speed** (in nm/s). Speed of the tip motion during simulation.
- **Drawing configuration**. (as explained before).
- You can load an existing configuration or make a new one by selecting “(new...)” configuration in list: A dialog lets you do it.
- **Show tip position**. If you don’t want show it in the view, you can unmark this option. Tip position is represented in the lithography design view as a cross.
- **Auto rescale** to fit to window when opening a new image. If this option is checked and you change the background image, instructions will resize automatically, to get the full area.


<sup>1</sup> A drawing instruction is an instruction that moves the tip, such as lines, and you can see it on lithography design view, on plotting it and simulating.

- **Grid Settings.** X and Y distance between grid points in the design view. If this value is too small, the grid will not appear.
- **Font for text instructions.** This lets you select a font file from Litho directory that is included with WSxM program.

 **COLORSETTINGS** Lets you select colors for different states of figures in the design view (normal, selected, executed), for tip, and the dot size to draw them.

 **DESIGN** This shows the frame where the image that you are acquiring or that was opened from a file (background image) appears, mixed with lithography patterns showing tip motion (that you can simulate here too), and actions to do while moving the tip. Using “design” you can directly draw patterns (figures), or do it by means of code (see next point). See Lithography Design Frame (Design and Simulation) for more information.

 **SOURCECODE** Opens the code view, which describes the lithography pattern. You can directly edit it, or add, delete or edit figures drawn in litho design view. See Source Code frame for more information.

 **COMPILATIONRESULTS** This is the view where the compilation results of the code appear. And not only this but also the result of compiling a “drawing configuration” (of starting or ending drawn).


## DISPLAY

You can see functionality of this menu in representation subprogram help.

## CODE

See Source Code frame.

## RUN

 **Go** Starts simulation. If you have selected show tip position, you can see it moving over the sample during simulation, conforming to the lithography pattern that you have designed in design frame and source code frame. Executed instructions will change the color (as you have decided in color settings).



Starts or continues simulation, but only of the next block. When the simulation of the block finish, it will stop there.



Starts or continues simulation, but only of the next instruction of the current block. If this instruction is a block, it will execute the first instruction of the subblock. When the simulation of the instruction finishes, it will stop there.



Pauses simulation. This option is only available when you are running a simulation. The simulation stops where it is when you have selected this option.



Restarts simulation. If you are running a simulation and select this option, simulation will go to the beginning of the code. If you have finished simulation by “step over” or “step into”, you need to restart simulation before starting again.

## SCREEN

You can see functionality of this menu in representation subprogram help.

## WINDOW



This option shows a dialog to rescale drawn instructions that appear in lithography design view. Rescale performs up or down instructions as you decide. An option is “keep proportion on scaling”, while is selected in the corresponding dialog.


If there are selected instructions, rescale will be on these. Else, rescale will be applied to on all drawn instructions of lithography design view.


## ***Lithography Design Frame (Design and Simulation)***


This is the window that appears when you select “view design”. Here you can see drawing instructions and simulations of the code.

New menus that appear here now are the following:


## EDIT


 **SELECT ALL** Selects all instructions that appear in the design view. You can also select an instruction individually by clicking on it, or even select all instructions that are in a rectangle. To do this, click once on starting selection point, and drop the mouse without unclick, to the finishing selection point, drawing the rectangle. Selection instructions will change color (as you have defined in the color settings). If you click the mouse right button over a selection, a contextual menu will appear, with actions to do with the selection.


 **SELECT NONE** Deselects actual selected instructions.


 **CUT SELECTION.** Cuts selected instructions to the clipboard (as if you pressed Ctrl+X).

 **COPY SELECTION.** Copies selected instructions to the clipboard (as if you pressed Ctrl+C).


 **PASTE.** Pastes in the design view instructions from the clipboard (as if you pressed Ctrl+V).


 **DELETE SELECTION.** Clears selected instructions (as if you pressed Supr).


 **UNDO.** Undoes last action(s) done in design window (as if you pressed Ctrl+Z).


 **REDO.** Undoes last undo(s) done in design window (as if you pressed Ctrl+Y).

## PLOT

 **DOT** Draws a dot in the design view. When you select it, you can click on a point of the view to select the position of the dot.

 **LINE** Draws a line in the design view. When you select it, you can click on a point of the view to select the start position of the line, and click again to select the end position.

 **MULTILINE** Draws a multiline in the design view. When you select it, you can click on a point of the view to select the start position of the multiline, and click again to select the end position of each partial line.

 **TEXT** This option shows a dialog to write a text in the design window. You can decide the height of text. After you select

OK on this dialog, the position of the text has to be defined by clicking at the corresponding location in the design view. Font can be selected in lithography settings frame.

## Source Code Frame (Design and Simulation)

Here, you can see the instructions represented in the design view, and add new instructions or modify the existing ones. The language is not case sensitive, that is, you can type *LINE* or *line* and it will treat both words as if they were the same one.

Instructions available are the following ones:

### Assignment Instruction.

- Description.  
Assigns a value to a variable.
- Format.  
`variable-ident = math-expression ;`

### Block.

- Description.  
Let you organize code, and simulates it as you want, using step over and step into options.
- Format.  
`begin  
(....)  
end`
  - Where (...) are more blocks or other instructions (without any limit)

### Comments.

```
// This is a comment
/* This is other way to write a comment */
```

### DoRamp Instruction.

- Description.  
Acquires from the specified input channel while applying a ramp in the specified output channel. Optionally, it can apply ramps in two different output channels simultaneously.

➤ Format.

```
DoRamp(InputChannels = ("IChnList"),
       OutputChannel = "OChn",
       NumPoints = numPoints,
       InitialValue = Unit,
       RampAmplitude = Unit,
       Speed = UnitPerSecond,
       Point = (NanometersX, NanometersY),
       SaveData = bool,
       Smooth = bool,
       LimitChannel = "LChn",
       LimitValue = Unit,
       RelativeLimit = bool,
       OutputChannel2 = "OChn2",
       InitialValue2 = Unit,
       RampAmplitude2 = Unit,
       ForwardAndBackward = bool)
```

- IChnList: list with input channel names.
- OChn: output channel name.
- NumPoints: number of points to acquire. It must be between 2 and 4096.
- InitialValue: initial value for acquiring the curve (optional).
- RampAmplitude: ramp amplitude.
- Speed: acquisition speed.
- NanometersX, NanometersY: coordinates where ramp will be applied (optional)
- SaveData: whether or not the curve must be saved (optional). The value in "Saving Options" will be used if omitted.
- Smooth: whether or not the acquired curve must be smoothed (optional). This argument is false by default.
- LChannel: limit channel name.
- LimitValue: limit value (in LimitChannel unit)
- RelativeLimit: whether or not the LimitValue is a relative limit (optional). This argument is false by default.
- OChn2: second output channel (optional).
- InitialValue2: initial value for the second output channel (optional).
- RampAmplitude2: ramp amplitude for the second output channel (optional).
- ForwardAndBackward: whether or not to make both, forward and backward ramps (optional). This argument is false by default. If present, backward ramp will have the same amplitude as forward ramp (with the opposite direction).

➤ Remarks.

- The Unit for InitialValue, RampAmplitude and Speed is the one selected as default unit for the specified output channel in the Output Channels Manager. The Unit for InitialValue, RampAmplitude also applies for the second output channel, if specified. Speed is only referred to the main output channel.
- IChnList must go between parentheses.
- Channel names must be written between quotes.
- If either OChn2 or InitialValue2 or RampAmplitude2 is specified, OChn2 and RampAmplitude2 are then required. If limit channel and limit value are specified, ramp acquisition will stop when the value measured in limit channel is greater than the limit value. This limit can be either relative or absolute. These values are mutually inclusive: you must either omit both or set the two values.
- If you want to use this instruction to acquire an F-Z curve, remember to remove the feedback before (with a SetFeedback instruction). Otherwise, your curve will not be properly acquired.

**For Instruction.**

➤ Description.

Repeats an instruction a specified number of times.

➤ Format.

`for (var = init to end step s) do instruction`

- Var must be a valid variable identifier.
- Both, 'init' and 'end' must be either numeric constants or numeric constant variables, and 'end' must be greater or equal than 'init'. (if not, step must be negative)
- Instruction can be a block.
- Var cannot be assigned in instruction (it can be read, of course).
- Step is optional. If not present, its value is either 1 or -1, depending on the loop limits).

**GetValue Instruction.**

➤ Description.

Reads a value from a channel. This instruction can be used as a numeric value (as a part of any other instruction).

➤ Format.

`GetValue (InputChannel = "IChn", SaveData = bool)`

- IChn: input channel name.
- SaveData: whether or not the value must be saved (optional). The value in “Saving Options” will be used if omitted

#### If Instruction.

- Description.  
Executes an instruction only if a certain condition evaluates to true.
- Format.  
`if ( condition ) then instr-1 [else instr-2]`
  - The ‘else’ part is optional.
  - Both, instr-1 and instr-2 can be instruction blocks.

#### Line Instruction.

- Description.  
Draws a single line. This is a drawing instruction.
- Format.  
`Line (begin = (NanometersX1, NanometersY1),  
end = (NanometersX2, NanometersY2) ,  
InputChannels = (“IChnList”),  
NumPoints = NumPoints,  
Smooth = bool).`
  - NanometersXi and NanometersYi are absolute coordinates of a point.
  - IChn: list with input channel names.
  - NumPoints: number of points to acquire. It must be between 2 and 4096.
  - Smooth: whether or not the acquired curve must be smoothed (optional). This argument is false by default.
- Remarks.
  - Begin argument is optional. If it is not specified, the beginning of the line will be the actual tip position (it can be the finishing point of the previous line).
  - IChnList, NumPoints and Smooth are optional.

#### MoveTo Instruction.

- Description.  
Moves the tip to the specified position.
- Format.  
`MoveTo (point = (NanometersX, NanometersY)).`

- NanometersX and NanometersY are absolute coordinates of a point.

#### Point Instruction.

➤ Description.

Plots a point. This is a drawing instruction.

➤ Format.

```
Point (point = (NanometersX, NanometersY),  
Milliseconds = msec).
```

- NanometersX and NanometersY are absolute coordinates of a point.
- msec is the time to wait at the specified position.

#### ReadDigitalBit Instruction.

➤ Description.

Reads a single bit from the user input digital signals port. This instruction can be used as a numeric value (as a part of any other instruction).

➤ Format.

```
ReadDigitalBit (BitIndex = idx).
```

- idx is the bit index, from 0 to 7 (7 is the most significant bit)
- This instruction is only available to **Dulcinea Electronics** users.

#### ReadDigitalByte Instruction.

➤ Description.

Reads the byte in the user input digital signals port. This instruction can be used as a numeric value (as a part of any other instruction).

➤ Format.

```
ReadDigitalByte ().
```

- This instruction is only available to **Dulcinea Electronics** users.

#### SetDrawingConfig Instruction.

➤ Description.

Changes the current drawing configuration

➤ Format.

```
SetDrawingConfig (Name = "ConfigName").
```

- ConfigName is the configuration name. It must be written between quotes.

### SetDynamic Instruction.

➤ Description.

Activates or deactivates the oscillation on the dynamic card and changes the PLL state.

➤ Format.

```
SetDynamic (Oscillation = Bool, SwitchRelay = Bool, PLLActive = Bool, Index = BoardIndex)
```

- Oscillation is stopped or restarted depending on the value of parameter Oscillation (false / true), if present.
- When PLLActive parameter is present, it activates / deactivates the PLL on the specified dynamic board.
- When SwitchRelay is true, the oscillation is started / stopped by changing the corresponding bit in the digital channel. When it is false, it just resets / restores the amplitude channel value. Default value for this parameter is false.
- BoardIndex determines over which dynamic board the changes are made. It can be either 1 (for the first dynamic board) or 2 (for the second dynamic board). Its default value is 1.

### SetFeedback Instruction.

➤ Description.

Activates or deactivates feedback loop and set feedback parameters.

➤ Format.

```
SetFeedback (Index = feedback_index, Active = Bool, P = PValue, I = IValue, SetPoint = SetPointValue, InputChannel = "iChn", SmoothInput = SmoothValue)
```

- All arguments are optional.
- Active (true or false) determines feedback to be turned on/off.
- PValue is the feedback proportional parameter (P).
- IValue is the feedback integral parameter (I).
- SetPointValue is value of the set point in the feedback loop. Units are the same as you have decided in the corresponding panel of the WSxM acquisition frame.
- InputChannel is the feedback channel.

- Index is a number specifying the feedback type. You can choose among the following constant values:

|                                  |      |
|----------------------------------|------|
| <code>feedback_index_main</code> | (1)  |
| <code>feedback_index_2nd</code>  | (2)  |
| <code>feedback_index_3rd</code>  | (3)  |
| <code>feedback_index_4th</code>  | (4)  |
| <code>feedback_index_xy</code>   | (10) |
| <code>feedback_index_z</code>    | (11) |

You can use either the number or the constant.

- SmoothValue is the smooth factor of the feedback signal. It must be a number between 0 and 100. The use of this parameter is the same as the “Smooth Input” parameter in the Main Feedback dialog.
- The meaning and need of some arguments may vary depending on the feedback chosen:

`PValue`, `IValue`: when using either `feedback_index_xy` or `feedback_index_z`, `PValue` represents the Power feedback parameter. In such a case, the `IValue` is ignored. When using other feedback type, these values are mutually inclusive: you must either omit both or set the two values.

`SetPoint`: this value is ignored when using either `feedback_index_xy` or `feedback_index_z`.

`InputChannel`: this argument is ignored when using `feedback_index_xy`.

`SmoothInput`: this argument is only valid with main feedback.

#### SetLaser Instruction.

- Description.  
Activates or deactivates the laser.
- Format.  
`SetLaser (Active = Bool)`

- This instruction is only available to **Dulcinea Electronics** users.

#### SetMovementSpeed Instruction.

- Description.  
Sets the tip motion speed.
- Format.  
`SetMovementSpeed (horizontal = nm/sec, vertical = nm/sec)`

- The unit is nm/sec.
- Both arguments are optional.

#### **SetPlaneScan Instruction.**

- Description.  
Activates or deactivates the planescan.
- Format.  
`SetPlaneScan(active = Bool)`

#### **SetValue Instruction.**

- Description.  
Sets a value in a channel, as discussed below.
- Format.  
`SetValue (Bias = BValue) or`  
`SetValue (OutputChannel = "OChn", Value =`  
`OValue)`
  - OChn: output channel name, between quotes. Don't use the following channels: "X scan", "Y scan", "Z".
  - OValue: value to set in the specified output channel, in channel default unit.
  - BValue. voltage to set in the bias output. Value must be specified in millivolts.

#### **Sleep Instruction.**

- Description.  
Waits for a time doing nothing.
- Format.  
`Sleep (milliseconds = msec)`
  - msec is time to wait.

#### **Text Instruction.**

- Description.  
Draws a single line of text. This is a drawing instruction.
- Format.  
`Text (text = "Text", position = (NanometersX,`  
`NanometersY), size = Size, font = Font, stretch`  
`= Stretch))`
  - Text is the text to put in the view

- NanometersX and NanometersY are coordinates of the position.
- Size, is the height of the text in nanometers.
- Font is the name of the font used when drawing the text. If omitted, default font will be used.
- Stretch, is the stretching factor (optional). Its default value is 1.

#### Variable declaration.

➤ Description.

Declares a variable with the name 'ident'.

➤ Format.

Float identifier.

Integer identifier.

Boolean identifier.

- This will declare a new variable, of the specified type.
- Variables must be declared before using them.
- You can declare variables only in the main block and before any other instruction. However, you can use them in you drawing configurations too.
- Float variables can hold rational numbers. You can use "Numeric" instead of "Float" to declare this kind of variables.
- Integer variables can hold integers in the range -2147483648 to 2147483647
- Boolean variables can hold the logical values true and false.
- Optionally, you can declare a const variable, by writing the const keyword before its name. In such case, you must provide its constant value as in an assignment expression.

#### While instruction.

➤ Description.

Repeats an instruction until the specified condition is false.

➤ Format.

```
While (condition) do instr
```

- Instr can be a block.

### WriteDigitalBit Instruction.

- Description.  
Writes a single bit to the user output digital signals port.
- Format.  
`WriteDigitalBit (BitIndex = idx, Value = val)`.
  - `idx` is the bit index, from 0 to 7 (7 is the most significant bit)
  - `val` can be either 0 or 1.


### WriteDigitalByte Instruction.


- Description.  
Writes a byte to the user output digital signals port.
- Format.  
`WriteDigitalByte (Value = val)`.
  - `val` must be in the range 0 .. 255

### ZMove Instruction.

- Description.  
Moves tip to a given height.
- Format.  
`Zmove (height = Nanometers, speed = nm/sec)`
  - `Nanometers` is height to move the tip. Negative value goes away from the sample, that is tip-sample distance is increased. Positive values are allowed but should be used with care.
  - `nm/sec` is average movement speed (optional) .

Commands available for this frame are:

 **OPEN CODE** Opens a code from a file, and puts it in the lithography code frame, but without loading. You can also load a Bitmap File (\*.BMP). In this case, you will be asked to select the distance between the points of the bitmap (in nanometers). A new Lithography pattern made of `Point` instructions will be created. There will be points only in the dark zones of the bitmap.

 **SAVE CODE** Saves code from the frame in the actual file as is, without checking it. If code is totally new, and you

have not loaded it from a file, this option will ask you for a file name.



**SAVE CODE AS**

Saves code from the frame in a new file, without checking it. The system will prompt for the desired name.



**APPLY**

Applies code changes to the design view. Instructions are compiled, and results appear in the lithography compilation results view. If it was success, new code will be represented in the lithography design view, and you can simulate it. During simulation the next line to execute will change its color. If there are errors, instruction won't appear in the view. Warnings only indicate that something can be wrong, in spite of code compile and will be shown in the view.

**Starting Drawing (or Finishing Drawing) Configuration frame (D&S)**

Here, you can write instructions that will be executed before (or after) each drawing instruction.

Only exists a restriction: You cannot write here more drawing instructions.

Example: If you want to perform an oxidation process, you may “draw” lines by applying a Bias Voltage. In this case, you can write the SetBias instructions in the start drawing frame and it will change automatically before plotting every line. In the finishing drawing frame, you should set the bias to its previous value

Available commands are:



**OPEN CONFIGURATION**

Opens an existing configuration, and put it in the start draw configuration frame and in the finish draw configuration one.




**SAVE CONFIGURATION**

Saves in actual configuration, the instructions that appear in both configurations, the start drawing and the finish drawing one.





**SAVE CONFIGURATION AS**

Saves both configuration files, the start drawing and the finish drawing one. The system will prompt for the desired name.

 **COMPILE** Compiles both configurations, the start drawing configuration and the finish drawing one, showing results in the Compilation results window. Errors must be corrected, but warnings only indicate you that something can be wrong, in spite of configuration compile.


## Acquisition controls (Real execution)

### LITHOGRAPHY DIALOG-BAR


 **OFF/ON** Activates/deactivates the lithography session. Activation will stop the scan (as if pressed  in the selected channel acquisition window) and show the lithography pattern over the acquired image.

### RUNNING COMMANDS


For more details about these commands, please refer to their corresponding command in design and simulation section.

 **Go** Starts or continues running.

 **STEP OVER** Runs a single block.

 **STEP INTO** Runs a single instruction.

 **PAUSE** Pauses the execution.

 **RESTART** Reinitializes the execution.

## OTHER RELATED CONTROLS



Activates / deactivates the scan. The scan must be paused while performing the lithography session in order to allow the tip to move freely over the sample. Activating lithography will automatically pause the scan.

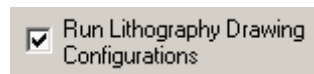
If you want to continue scanning after your lithography session, you must press the On/Off button in the Lithography dialog-bar and the scan will be activated again.



### ACTIVATION BIT

Enables / Disables the activation bit for Lithography. If this bit is set, current Lithography program

will be executed when the specified bit becomes active (value 1). Bit value is checked at the end of each scanned line.



### LITHOGRAPHY IN RETRACE

Enables / Disables the execution of Lithography Drawing configurations in

the Retrace mode. If this option is enabled, Start Drawing Configuration will be executed when the retrace begins, and Finish Drawing Configuration will be executed when it ends. This option is available in the Retrace Dialog Bar.



**Nanotec Electrónica S.L.**  
 Centro Empresarial Euronova 3  
 Ronda de Poniente 12, 2º C  
 E28760 Tres Cantos (Madrid)  
 SPAIN

Phone: + 34 91 804 33 47  
 Fax : + 34 91 804 33 48  
 Web: [www.nanotec.es](http://www.nanotec.es)  
 CIF: ES000B82006057

09/09/2010