Getting Started with WinLASI:

This document will help you get "up and running" with WinLASI. You will step through a basic Analogue Full-custom IC Design Flow taking about 10 hours at first pass.

Whilst not intended to teach how to do IC design, this note should complement the LASI Help files available from within each tool menu in WinLASI and give an introduction to the IC Design Flow itself and build a confidence with WinLASI.

I hope you enjoy using this program as much as I have.

Please feel free to send me an email telling of your comments and experiences; these will be valuable to me when writing the next version of these notes.

Simon Harpham © 15-02-2004
Silicon Devices (UK) Limited
SimonH@SiliconDevices.co.uk

Contents: ..... Error! Bookmark not defined.

---

1) Setting up WinLASI (and WinSPICE3)

Download WinLASI from:

http://cmosedu.com/cmos1/winlasi/winlasi.htm

and WinSPICE3 from:

http://www.willingham2.freeserve.co.uk/winspice.html

then follow the setup instructions given on each site. Read WinLASI update info at:

http://members.aol.com/lasicad/

this is particularly important if you are reading these notes in conjunction with a later version of WinLASI than 7.02.

IMPORTANT NOTE: These programs are provided free of charge to private individuals. If you intend to use these programs for commercial purposes please pay the licensing fees; they are extremely good value for money but will only remain so if we all support the people who have given so much of their time to developing these programs for us to use.

On the work area of your PC (under MS Windows) click on the "My Documents" icon, then in the window that comes up, choose:

[Tools] [Folder Options] [File Types] [New]

enter .cir in the dialogue box, then click [OK] [Change] [Other]. Use Windows Explorer to search for the winspice3.exe file. (In my setup it is to be found in C:\Program Files\WinSpice\) accept and Close the window.

Repeat the setup procedure for WinLASI, setting it to read .tlc files.

Setup WordPAD (in C:\Program Files\Windows NT\Accessories) to read

.rpt .hdr .ftr .ldm .nod and .usf files.

If you decide to run more than one version of WinLASI (eg Version 6.2, version 6.3.2.2 as well as version 7.0.2.2) you may choose to set up V7 to open .tlc and V6.3 to read .bp6 and version 6.2 to read .cl6 files.

NOTE: If this is the first time running any of the system utilities (such as LasiCkt, LasiDrc, Tlc2gds 7, etc) you may have to add the full path to the reader box to avoid the "Cant find Reader Program" error message, when you come to use those utilities.

In my setup the reader is at C:\WINNT\notepad.exe

---

2) Setting up a new design project

How to start a new design is not immediately obvious with WinLASI: if a new directory is created, then a .tlc file copied into it using Explorer, it will be found that the layouts will come up with lines with no “fill”. It will be useable, but confusing to the eye. You may also find that when downloading new examples, some layers don’t appear.

The solution is to copy over the layer data from an existing design, to save going through the process of installing the layer files.

Open a Windows Explorer window, then go to a library which contains the correct technology for your new cell (do not worry about the term "technology" at the moment; we will use an existing cell). Open the library to be found in your original LASI installation at:

C:\Lasi7\LibTutor

pick a cell (eg Dfbf311.tlc) double click on it to open with WinLASI.Click [Fit] in the menu at the top to see the whole layout.

Then use: [System] [Clone] entering the path to the new directory to be created in the dialogue box which appears, I used for example:

C:\Documents and Settings\All Users\Documents\WinLASI\SCRATCH\

TIP: Keep an Explorer window open, then click in the "Address" box at the top of the window, navigate to the path to where you want the new folder to be (don't create it yet) then select that path by clicking to the right of it to highlight, then click on [Edit] [Copy,]. Return to Contents to the LASI window click in the "path" box in the Clone window, then press <cntrl> c (saves re-typing the whole path} then type in the name of the new directory branch to be added (I have used “Example”) to the above path. Check [ ] Create System Icon, then fill-in the dialogue box [Example], click on [OK].

If the directory already exists you will get the message "directory exists", ignore it. Use the Explorer window to navigate to that new folder and you will see that it contains one file: Lasi7.usf

Leave the Lasi window on this, the cell “source” directory open, go to the target directory and select the .usf file, open with Wordpad and note the 27 "layer=..." lines; it will be become important that you get to know how to use these later on, particularly if you use LASI to do IP cell imports and conversions between process technology types. Close Wordpad.

In the source file window (the LASI window with DFBF311.tlc loaded) Click on [List] then pick an example cell to be copied over (eg DFBF311) to the target (Example) directory, click on [Export] [Browse] then locate the target library (as entered above) click on [Open].

The correct cell name should already be in the cellname box (DFBF311)at this point you could rename it by overtyping. You should now see the

Export Setup window with the correct destination path and cellname in it. Pick [OK], you should then see a message box saying "Export is complete ... 1 Cells + 27 layers" click on [OK].

You should now see two files in the target directory; the above Lasi7.usf file plus a new Dfbf311.tlc file should have appeared. Double click on the Dfbf311.tlc file, a second LASI window should open and you should see a third file, Cells7.dbd appear in the destination directory - open it, it will contain 0 (since it is empty).

In the target Lasi window, if you click on [list] you will not see any cells listed, so click on [Close] then [Import] [Browse]. Choose Cells7.dbd, click on [OK].

The example cell should now appear correctly (ie the polygons are "filled" rather than "outlines") in the target Lasi window.

You should now see 4 files in the destination (Example) directory; the new file is Loadbkup.tlc. This can be opened with LASI and contains the backup for the DFBF311 cell.

Whilst it may seem odd that you have to export to the new directory then open the file then import the cell, it makes sense if you consider that you have to place something into the directory for LASI to work with initially. It was convenient here to export from an existing source directory using LASI, rather than copying with Explorer, then explaining how to patch the .usf and .dbd files.

Repeat the import process to copy over the layout and schematic .tlc files (Dfbf311_lay.tlc and Dfbf311_sch.tlc) from the source directory

C:\Lasi7\Libtutor\

When you have finished, there should be 24 files in the target directory: 16 more appear after the layout import, being:

```
Dfbf311_lay.tlc  Nmos3_tok.tlc  Nmos4_tok.tlc  Nmos5_tok.tlc
Nmos6_tok.tlc    Nmos7_tok.tlc  Nmos9_tok.tlc  Pmos10_tok.tlc
Pmos13_tok.tlc   Pmos14_tok.tlc Pmos18_tok.tlc Pmos20_tok.tlc
Pmos23_tok.tlc   Pmos6_tok.tlc  Pmos9_tok.tlc  Swap.bpv
```

and 4 after the schematic import:

```
Dfbf311_sch.tlc   Gnd_sch.tlc    Nmos_sch.tlc   Pmos_sch.tlc
```

TIP: as LASI will only allow you to open one window in each folder, it may sometimes be convenient to make a copy of the schematic in a sub-folder (as well as in the target folder) to allow two windows to be opened during a layout phase of the design flow. (Remember to "Clone" the sub-directory folder).

After schematic activity, there will be 25 files, the new one being:

Copybkup.tlc

Close the source Lasi Window. We will now look at LasiCkt then LasiDRC.

---

3) Setting up Netlist Extraction from Schematic using LasiCKT

Choose the appropriate .tlc file (eg Dfbf311_sch.tlc in the above target directory) open with Lasi, then choose [System] [LasiCKT] (a new box will open at top left of screen) then choose [Setup] (and the "Compile Setup" dialogue box will open) then choose [List] and double click on INVF101_SCH, ignore Header File Name, Footer File Name and Alias File Name for the moment. Click on [OK].

Return to Contents to the schematic window select [Layr] - the seventh icon down in Menu 2 which is to the right of the main LASI window. (if the seveth icon is [Tlyr] click on the [Menu 2] icon to see [Layr])

You will see a list of all of the layers with one checked (probably layer 2 OTLN). Note the layer numbers for NTXT CTXT DTXT PTXT (probably 4, 5, 6 and 7) notice that these correlate to the "layer = ..." entries in the Lasi7.usf file.

Re-open the Compile Setup window and enter these numbers into the Node,Connector, Device and Parameter text boxes respectively, then make sure the following options are selected:

- that is [v] not [ ] and (o) not ( )

```
_____________________________________________________
Special Text Layers:

Real Node Text Layer 4        Device Name Text Layer 6
Connector Text Layer 5        Parameter Text Layer   7
_____________________________________________________
Options:

[v]Check for Floating Cells

[v]Check for Node Text on Multiple Conductors
                                [ ]Report Virtual Nodes
[v]Check for Connector Text on Multiple Conductors
                                [ ]Report Counts
_____________________________________________________

Virtual Node Prefix [vn      ]
               Connect Rules ( ) Circuit (o) Schematic]

Report File Name    [Lasickt.rpt      ]       [OK]
_____________________________________________________
```

Next we have to set up the interlayer connectivity. This is done with the Trace table: [System] [LasiCKT] [Trace]

Go back to the schematic window, click-on [Get] then draw a box around some of the blue interconnecting lines on the schematic, then click-on [Info]. There will be a layer reported in that box. For this example it is Layr=MET1(49)

Enter 49 in the Conductor 2 box under the "traced" column in the Layer Trace Table window. Click on [OK] then select [Go] in the LasiCkt7 menu at the top left otf the screen.

- the extraction routine should run, then report:

"Completed ...0 Errors 0 Warnings"

- if you get a lot of errors and warnings (108 Errors for this example) then you have not entered the correct layer in the step above. Notice that you can enter this number on any conductor layer and the netlist extraction from the schematic will still work. However when we get to the netlist extraction from layout, this will need to be entered into the correct conductor layer. Some schematics may use another layer (possibly layer 3 SCHM) to avoid this complication and minimise the possibility of making errors.

There are now 29 files in this directory, the new ones are:

LasiCkt.usf Dfbf311_sch.cir Dfbf311_sch.nod Lasickt.rpt

In the LasiCkt7 menu, choose [Run] then the Filename [Browse] then locate the target directory and click on Dfbf311_sch.cir, then [Open] [OK]. You should then see a SPICE netlist with the first and last few lines as follows:

```
*** SPICE Circuit File of DFBF311_SCH  02/10/04 10:27:44

* MAIN DFBF311_SCH IN_DATA IN_CLK IN_RST IN_SET O_Q O_Q_b
M1  vn3 IN_CLK VDD VDD CMOSPB L=2u W=23u
M10 vn1 vn5 VDD VDD CMOSPB L=2u W=12u
....
....
M7 vn5 vn4 vn18 0 CMOSNB L=2u W=5u
M8 vn18 IN_DATA 0 0 CMOSNB L=2u W=5u
M9 vn1 IN_SET VDD VDD CMOSPB L=2u W=10u
.END
```

This netlist has been extracted from the schematic. Details of how schematics should be prepared in order to create this netlist correctly is beyond the intent of this note, so you are directed to the LASI help files for further information on drawing schematics.

For the extracted netlist to be useful to us, we have to add a header file to set up the testbench around the circuit to be simulated and a footer file to set up the simulation environment.

SPICE Simulation from Schematic NLE

Copy the two netlists in the appendix to this note labelled Dfbf311_sch.hdr and Dfbf311_sch.ftr, into two text files saving each with those names into the target (Example) directory. You should now

have 31 files in the target directory.

NOTE: the device Models used here are simple Level 2 models which allow simulations to run. Ideally level 8 (level 49) models as provided from the foundary should be used, however to do so would require us to write the area, periphery and resistance rules for the layout based on the width and length of the devices to ensure junction capacitances and drain and source resistances are correctly extracted.

The original setup files for WinLASI were for SCNA (Scalable CMOS N-well Analog) technology with a lambda of 1.0. Unfortunately this technology (Orbit's 2.0 um CMOS) is no longer available, so we have to modify the default files to set them up for a new technology. Presently this is SCN3M_SUBM (SCalable N-well 3 Metal layer SUBMicron features) details can be found at:

http://www.mosis.org/Technical/Designsupport/mosis-design-example.html

Fortunately we can "re-use" the earlier layout libraries with a Lambda scaling of 0.30. This variant of SCMOS corresponds to the Agilent/HP 0.50um CMOS process. This allows us to reuse the typical N91B SPICE models from the web site in our simulations.

Note: the SPICE models used in the footer file of our example are generic models and do not represent any particular process.

To continue: go To the Compile Setup window in [System] [LasiCKT] [Setup] then enter the header file as Dfbf311_sch.hdr the footer file as Dfbf311_sch.ftr Click on [OK] then [GO] to create the netlist which should be ready for simulation.

If you have WinSPICE3 set up, you can just locate the Dfbf311_sch.cir file, double click on it and the simulation runs.

You should then see one plot with two traces in it: the first is the output transient response at -40degC, the second is the output response at +125degC. The Flip-flop has been configured as a divide two, producing a 5MHz square-wave from the (unplotted) 10MHz input.

TIP: always add a <return to Contents > at the end of the header and footer files, otherwise the compiler joins the next line and causes and error in SPICE.

You should still have 31 files in this directory.

---

4) Setting up Netlist Extraction from Layout using LasiCKT

Open DFBF311_LAY.tlc with Lasi, then choose [System] [LasiCKT] [Setup] [List] choose DFBF311_LAY, leave Header, Footer and Alias files blank for the moment. Choose (o) Circuit Rules [OK].

In the LasiCkt 7 menu, click on [Trace] then enter the following:

				Layer Name (not used in table)	
Trace Levels	Merged	Traced		Merged	Traced
Conductor 8					
Insulator 7					
Conductor 7					
Insulator 6					
Conductor 6					
Insulator 5					
Conductor 5					
Insulator 4					
Conductor 4		62			M3
Insulator 3		61			VIA2
Conductor 3		49			M2
Insulator 2		50			VIA1
Conductor 2		49			M1
Insulator 1		25			CON
Conductor 1	8 46	9 10 46		PCON POL1	NSEL PSEL POL1

These are the interconnect layers referred to in "CMOS Circuit Design, Layout and Simulation" page 157, as defined for this process. They can be verified by clicking on [Ldrw] in Menu 2 in the main LASI window.

Ignore the capacitances in the Layer Trace Table for the moment, we shall return to Contents to these later on. Ensure they are deselected.

Add the following cell names to the Smash List under [Smash]:

PAD
VIA
CONTACT
SFRAME

Click on [Go]. LasiCkt 7 should report back:

Completed 16 warnings 0 Errors

TIP: Layers used in a cell maybe checked with: [System] [Show] [List] then pick the cell of interest by double clicking on it, then [OK] by scrolling down the table presented you can see how many boxes, paths or text labels have been added on each layer.

You should now have 33 files in your target directory, the two new files are:

Dfbf311_lay.cir Dfbf311_lay.nod

Now you are set to extract a netlist from the target layout cell, ready for simulation.

SPICE Simulation from Layout NLE

Copy the two netlists in the appendix to this note labelled Dfbf311_lay.hdr and Dfbf311_lay.ftr, into two text files saving each with those names into the target (Example) directory. You should now have 35 files in the target directory.

To be able simulate the netlist extracted from the layout, we have to convert the MOS labels to a format that WinSPICE3 understands; the easiest way is to write a macro in Word to convert the following text strings from the left column to the right column:

```
-----------------------------------------------------------------
 NMOS3_TOK        NMOS L=2u W=3u
 NMOS4_TOK        NMOS L=2u W=4u
 NMOS5_TOK        NMOS L=2u W=5u
 NMOS6_TOK        NMOS L=2u W=6u
 NMOS7_TOK        PMOS L=2u W=7u
 NMOS9_TOK        NMOS L=2u W=9u

 PMOS6_TOK        PMOS L=2u W=6u
 PMOS9_TOK        PMOS L=2u W=9u
 PMOS10_TOK       PMOS L=2u W=10u
 PMOS13_TOK       PMOS L=2u W=13u
 PMOS14_TOK       PMOS L=2u W=14u
 PMOS18_TOK       PMOS L=2u W=18u
 PMOS20_TOK       PMOS L=2u W=20u
 PMOS23_TOK       PMOS L=2u W=23u


-----------------------------------------------------------------
```

Copy this text and save as a text file called Token.paf; ensure you have at least two blank lines a the end of the file, otherwise LASI fails to complete the substitution.

You should now have 34 files in the directory.

Go to [System] [LasiCKT] [Setup] [List] select Dfbf311_lay. Add the header and footer file names back in , add Token.paf as the Alias File Name. Click [OK] then [GO] (... then "organise" if LASI asks to do so).

Click on [Run] in the LasiCkt 7 menu. LASI will take a few moments to complete; you should then see a "clean" error report (remember to

switch cell names to Dfbf311_lay.cir and switch to using layout rules)

There will now be 36 files in the Example directory, the two new files are:

Dfbf311_lay.cir Dfbf311_lay.nod

Open Dfbf311_lay.cir and Dfbf311_sch.cir then compare the netlists for the Flip-flop: you will see that M2, M10, M13, M25 and M29 are different widths.

Locate Dfbf311_lay.cir then double click on it, when the simulation has run, compare this result with that for Dfbf311_sch.cir (you can have more than 1 WinSPICE3 session active at once).

Next we shall compare the nodal connectivity of the Layout against the nodal connectivity of the Schematic (Layout Versus Schematic).

---

5) Setting up Layout Versus Schematic checking in LasiCKT

Start WinLASI in the Example directory. Click on [System] [LasiCKT] [LVS]

then enter:

```
Layout Node List File:        Dfbf311_lay.nod
Schematic Node List File:     Dfbf311_sch.nod
Base Name of Circuit:         Dfbf311
Name of Error File:           Lvserror.rpt
```

Then click on [Compare] then click on [Run] then enter Lvserror.rpt as the filename followed by [OK] you will see a report like this:

```
*** Node Compare Error List  02/11/04 12:38:06

Comparing Dfbf311_sch.nod to Dfbf311_lay.nod
Comparing Dfbf311_lay.nod to Dfbf311_sch.nod

Finished ... 0 Errors
```

You will now have 37 files in the directory. A file called Lvserror.rpt has appeared and is the example above. It is worth opening the two node files and doing a visual compare to satisfy yourself that they are in fact the same.

---

6) Setting up Capacitance Extraction from Layout using LasiCKT

Once you have a schematic and layout extraction setup and got them to simulate in SPICE, it is useful to set up the parasitic capacitance extraction for the layout netlist extraction.

Load Dfbf311_lay.tlc into WinLASI, then click on [System] [LasiCKT] [Setup] enter Dfbf311_lay then select circuit rules. Click on [Trace] then enter the following values:

Pair		Area	Periphery	Layers
9	0	0.50	0.00	NSEL GND (ie Substrate)
10	0	0.70	0.00	PSEL GND
46	0	0.09	0.00	POL1 GND
49	0	0.03	0.07	MET1 GND
46	9	2.40	0.00	POL1 NSEL
46	10	2.40	0.00	POL1 PSEL

switch the (o)fF selector on.

TIP: WinSPICE3 does not recognise aF.

Check the "[v] Node to Gnd Caps" box
But not the "[ ] Node to Node Caps" box. Then [OK]

re-netlist both Dfbf311_lay and Dfbf311_sch with the correct headers for each file, then simulate both .cir circuits (remember to edit the MOS names into parameters). Compare the results again; you will see a difference: the Flip-flop falling edge is delayed by around 15ns at +125degC with respect to the -40degC simulation and that they are both about 30ns slower than the falling edge in the simulation for the schematic’s netlist.

Redo the capacitance extraction with:

Check the "[v] Node to Gnd Caps" box
check the "[v] Node to Node Caps" box. Then click [OK]

Re-run the simulation and you will see that the flip-flop delay at +125deg has increases slightly.

Author’s note: doing these simulations at Vdd=2.2V is rather unfair on the original design as it was intended to be run with a 5V. The point is to be aware of the effects of temperature and parasitics on what may appear to be an acceptable design at nominal voltage, temp and process.

---

7) Setting up the Lasi's 3D Layout Viewer

Once a layout is available click on [System] [Lasi3D] [Setup] then enter layers into the Layer List:

---

1 2 3 4 5 6 7 8
8 9 10 46 25 49 50 51

9 10 11 12 13 14 15 16
61 62

Then Browse for Dfbf311_lay.tlc click on [OK] then [Go] then after Lasi3D 7 has run, click [OK] on pop-up, then go to [Import] then browse for Dfbf311_lay_3D.tlc, click [Open] [OK] then [OK}

You should see an angled isometric 3D plot of the layers in the Dfbf311 cell. Note: if there were lower ranking cells in the cell being viewed, they will not be displayed.

There are now 39 files in the target directory, the two new files are

Lasi3D.usf Dfbf311_lay_3D.tlc

---

8) Setting up Design Rule Checking using LasiDRC

From the layout window [System] [LasiDRC] [Setup] the list dialogue box should already contain [DFBF311_LAY] in it, if not use [List] to locate it.

Go to the DRC Filename dialogue box and choose [Browse] then search for Scmos.drc file, which you will find in

C:\Lasi7\Resource\Scmos.drc in the original installation.

[Open] [Ok] [Fit] [Ok] then [Go] in the LasiDrc 7 menu.

When the DRC has run, click on [Run] [Browse] then pick LasiDRC.rpt.

You should then see a short report like this:

*** LASI Design Rule Check Flag Report

Cell: DFBF311_LAY
Check 1: 1.1 Well width >=10 lam
Date: 02/15/04 Time: 19:11:59
Resolution= 0.1um* Distance= 10um Rescale= 1 MapSize= 1
Area(s) Flagged:

Which shows the DRC ran but only one check was carried out.

Go back to the LasiDrc 7 menu then click on [Setup] then click on [Checks, scroll to the bottom and note the number of the last check, click cancel, then enter that number (106) in the "Finish Check [ ]" box, click [OK].

Then in the LasiDrc 7 menu, click on [Go] ... wait till it finishes then .... go back to [Run] pick LasiDRC.rpt again, then click [OK].

Note that the DRC run pauses at four places and displays a dialogue box which tells you which check is being run when it pauses (Check 29, Rule 8.5). When it stops, it will tell there are 4 Total Flags.

Click on [Run] in the LasiDrc 7 menu, to see a report which starts like this:

*** LASI Design Rule Check Flag Report

Cell: DFBF311_LAY
Check 1: 1.1 Well width >=10 lam
Date: 02/15/04 Time: 19:18:50
Resolution= 0.1um* Distance= 10um Rescale= 1 MapSize= 1
Area(s) Flagged:

```
*** LASI Design Rule Check Flag Report

Cell: DFBF311_LAY
Check 2:  1.2 Well spacing >=9 lam
Date: 02/15/04    Time: 19:18:51
Resolution= 0.1um*  Distance= 9um  Rescale= 1  MapSize= 1
Area(s) Flagged:

....etc
```

You will now have 45 files in the directory, the six new ones are:

```
LasiDrc.usf    Lasidrc.rpt   Dfbf311_lay_29_1.pcx  Dfbf311_lay_29_2.pcx
Dfbf311_lay_29_3.pcx  Dfbf311_lay_29_4.pcx
```

The last four contain the “DRC error” information; to view them go to the LasiDRC 7 menu then click on [Map] you will then see a list of .pcx files (the four listed above). Pick one then click [Open]. You will see a window with simplified representations of the cell(s) with the violations to the DRC rule being checked when that file was written, being highlighted in white. If you have the main LASI 7 layout window visible, you can click on [locate] in the DRC error (.pcx file) window and see the layout move to where the DRC errors are.

Go back to the LasiDrc 7 menu and open Lasidrc.rpt again. Used Notepad’s find utility to look for “.pcx”. Notice that the line referring to each .pcx file, includes a name indicating the rule number which is being violated.

You will now see four extra lines, containing the coordinates of the area in which the error lies:

```
*** LASI Design Rule Check Flag Report

Cell: DFBF311_LAY
Check 29:  8.5 Via1 to poly1 or active edge spacing>=2 lam
Date: 02/15/04    Time: 17:31:35
Resolution= 0.1um*  Distance= 2um  Rescale= 1  MapSize= 1
Area(s) Flagged:
Area: L= -15.05 B= -44.02 R= 87.35 T= 19.98   Map: DFBF311_LAY_29_1.pcx
Area: L= 82.55 B= -44.02 R= 184.95 T= 19.98   Map: DFBF311_LAY_29_2.pcx
Area: L= -15.05 B= 15.18 R= 87.35 T= 79.18   Map: DFBF311_LAY_29_3.pcx
Area: L= 82.55 B= 15.18 R= 184.95 T= 79.18   Map: DFBF311_LAY_29_4.pcx
```

We see that the error for Check 29, Rule 8.5, is given as:

Via1 to poly1 or active edge spacing>=2 lam

In earlier installations of Lasi, there were a number of Rulexx.tlc files, under the MOSIS directory, which illustrated each Rule being checked. Unfortunately these are not in the more recent standard installations, as they were written for an older 2um technology.

You may request AllMOSIS.tlc from me at: SimonH@SiliconDevices.co.uk. Or look at page 878 of "CMOS Circuit Design, Layout, and Simulation" by Jacob, Li and Boyce.

Once you have a copy of the file, load AllMOSIS.tlc into LASI. This is a Rank 2 cell which reads all the Rulexx.tlc files as Rank 1 cells. Scroll around until you locate Rule 8. Load Rule8.tlc into LASI; go to [List] double click Rule8(1) then click [Fit]

Compare this layout with the one for the Dfbf311 cell. You will see that it is the upper example of check 8.5 that is violating the "<=2u poly to via spacing" in most cases.

You will see that most of the violations are <1um spacing with a bridged on layer 1 metal – in other words, the gaps are too small and will merge if the IC is fabricated in a 2um process. However since the gaps are bridged by metal, then the errors are benign and maybe ignored. The fact that the gaps exist is an artifact of the fact that the layout has been assembled from "standard cells".

It is left as "an exercise for the reader" to satisfy themselves that ALL of the errors are benign.

Once you have located and edited the errors, check the

[v]Remove old PCX file

switch, then re-run the DRC on all 106 checks.

If you chose to clean-up the errors, you will now have 41 files with no PCX files once the cell is "DRC clean", otherwise there will still be 45 files in the directory.

TIP: to save time when checking for one rule, enter its number in both the start and finish boxes.

NOTE: it is "good practice" to re-extract the net-list from the layout and do a node-list compare with the schematic, then re-run the simulation on the extracted circuit just to ensure there have been no compromising changes made during the DRC corrections.

NOTE: the DRC decks can be applied to any GDS11 file imported into LASI via the GDS2TLC routine we will set up in a later section. Something that is very useful for importing IP cells from other sources and from other technologies.

---

9) Setting up a "TapeOut" from WinLASI using TLC2GDS

It is strongly recommended that you read the help files for TLC2GDS first:

[System] [TLC2GDS] [Help] Read it.

[Setup] enter the following ....

```
Main Cell File Name [Dfbf311.tlc      ]
GDS File Name       [Dfbf311.gds      ]
--Layer Options-------------------------------------------------------
TLC Layer Filter (1-256) [9 10 25 42-51 61 62                        ]
Layer Datatype Map File  [Dfbf311.ldm                                ]
-----------------------------------------------------------------------
GDS Library name        [DEFAULT.DB] Lambda Length in um [1.0        ]
GDS Units per Phys Unit [1000      ] Default Path width  [0          ]
--Conversion Options--------------------------------------------------
[v]Check for Proper GDS IC Protocol [v]Convert Text to GDS text Records
[v]Sort Cells in Ascending Rank Order
---Legacy Options-----------------------------------------------------
[ ] Lower Case Cell Names           [ ]Scale Unit LSB Correction
-----------------------------------------------------------------------

             Report File Name [Tlc2gds.rpt   ]        [OK]
```

You will now have 46 files in the directory; the new file is

Tlc2Gds.usf

Click [Go] in the TLC2GDS window. Reply yes to "LDM file Dfbf311.ldm will be created?” and yes to all of the queries during runtime.

You will now have 49 files in the directory; the new files are:

Dfbf311.gds Dfbf311.ldm Tlc2gds.rpt

Click on [Run] in the Tlc2Gds 7 menu, then [Browse] to find Tlc2gds.rpt. You should get the following report with no errors:

```
*** TLC to GDS Conversion Report 02/13/04  18:32:31

GDS File Name: Dfbf311.tlc
GDS Library Drawing Name: DEFAULT.DB
GDS Scale: 1000 per um
LASI Scale: 100 per um
Rescale Ratio: 10
Lambda: 1um
```

```
Filtered Layers: 9 10 25 42-51 61 62

Structure                      Boundaries  Paths       Text      refs
-------------------------------------------------------------------------
DFBF311                           237         0          0          0

Completed ... 19562 Bytes in GDS File
```

Note the presence of Dfbf311.ldm in the (Example) directory. If you need to alter the layer map for a new process (that is, you wish to export IP cells between technologies), erase this file, re-run the converter, then put new numbers in the "Map to GDS Layer [ ]" box for each layer you wish to move.

Locate the layer map file (Dfbf311.ldm) using Windows Explorer, then open in a text editor to see a list like this:

```
*LDM built from Dfbf311.tlc
25,0,25
50,0,50
51,0,51
49,0,49
44,0,44
45,0,45
43,0,43
42,0,42
46,0,46
```

It is very important that you get to understand this file and how it works and the meanings of the layers if you intend to any really serious layout work using WinLASI. It really is the heart of how this tool works and explains WinLASI’s utility as an IP cell layer converter between the larger proprietary tools common in the IC design industry.

The left column applies to the gds11 file (input or output) and the right column is the equivalent “LASI layer”. If you intend to translate cells between technologies, then be very disciplined about which layers you translate the left hand side to, on the right hand side.

The resulting output file (Dfbf311.gds) should now be in "camera ready" GDSII format. If the file was for a whole chip, then we would have just created a "database" ready to "TapeOut". However this was only one cell, being our Flip-flop example.

To verify that the conversion was accurate we have to use the GDS2TLC converter ....

You should still have 49 files in the (Example) directory.

---

10) Reading a GDSII file into WinLASI using GDS2TLC

From the Layout window [Sys} [GDS2TLC] [Help] read the files.

To view the new tlc files and then compare them to the originals, we need to create a subdirectory. Remember to "clone" the new directory to get the layer information; from the target directory:

[System] [Clone] write .\GDS2TLS in the "Full Path of New Folder[ ]" box [OK]. Use Widows explorer to verify the folder is created and it contains a Lasi7.usf file.

Click on [System} [GDS2TLC] [Setup] then enter the following ....

```
GDS File Name              [Dfbf311.gds         ]
Path of all TLC Cell Files [Path* (- see below) ]
--Layer Options-----------------------------------------------------
Layer Datatype Map File  [Dfbf311.ldm                               ]
TLC Layer Filter (1-256) [9 10 25 42-51 61 62                       ]
-------------------------------------------------------------------

    Physical Units [um     ]   LASI units per Phys Unit [Scale**    ]

--Conversion Options------------------------------------------------
[v]Convert GDS Text    [v]Use Default Opt. Recds [ ]Write Layers TLD
[v]Check for Open Poly [ ]Omit Layers Record     [ ]Write Names File
[v] [v]Convert Boundaries to  Boxes
-------------------------------------------------------------------
             Report File Name [Gds2tlc.rpt   ]        [OK]
```

*Path = C:\Documents and settings\All Users\Documents\Winlasi\Scratch\Example\GDS2TLC\ however just .\GDS2TLC\ can be typed in.

**Scale = 100 Lasi Units/Phys Unit; look in [System] [Scale]. Note that this is not equivalent to the “GDS Units per Phys Unit” of the TLC2GDS routine.

Notice that we have used the sub-directory we created in the target directory, to prevent the original .tlc files from being overwritten.

[Go] When routine stops, open the Examples directory to see that two files have been added to make a total of 51 files in the Example directory, with two files in the GDS2TLC sub-directory. The new files are:

```
Examples:         Gds2Tlc.usf  Gds2tlc.rpt
      |
      GDS2TLC:    Dfbf311.tlc (Lasi7.usf was “cloned” earlier)
```

Go to the Temporary directory and double click on Dbfb311.tlc to open a new LAZI 7 layout window, then “import” the Dfbf311.tlc cell. When

done, you should see the flip-flop layout. It may initially appear different to Dfbf311_lay.tlc, but this is because the text is missing. Resize the two layout windows you now have open, such that one is on top of the other. You can then toggle between them by clicking the mouse buttons on the icons in the menu at the bottom of the work area. Compare the two layouts - text apart, they should be identical apart from where the conversion to GDSII format may have removed vertices etc.

Check that the sizes of the two layouts are the same (the top right vertx should be at (168.75, 75.91); and the vias should be 4lambda across. If not go back and re-check the value of scale of the “LASI units per Phys Unit” entered in GDS2TLC window above.

You should now have 4 files in the temporary directory: the two new files are:

Cells7.dbd Loadbkup.tlc

The target directory should still have 51 files in it.

Importing IP cells

Clone a new folder .\IPcell_Import\ under the Example directory.

Go to: http://cmosedu.com/cmos1/lasiproj/scells.htm

Then download:

40p2200.tld - tiny chip (final size of 2.2 mm x 2.2 mm) padframe using the AB process (lambda of 0.8u and SCMOS design rules)

near the bottom of the page. Save it into the IPcell_Import directory you have just created. Load it into LASI. If you have not associated .tld files with LASI, do so now ... once Lasi starts, [close] the load cell dialogue box, then go to the top menu and click on [Import] switch from tlc to tld, then browse for 40p2200.tld select it then click OK.

You should then see the tiny cell pad ring. [List] double click on IO, you should now see the contents of the Ii-output pad cell.

There will now be 11 cells in the IPcell_Import directory.

---

11) Appendices

SPICE Netlist Header Files

Dfbf311_sch.hdr

Save the following text as a text file named: Dfbf311_sch.hdr

```
************************************************************************
* Test DFF1_PFD      SimonH@SiliconDevices.co.uk
************************* Created 11-06-03 *************************
************************ Dfbf311 Test-Bench ************************

Vdd V 0   DC 5V
Rdd V VDD 10

*                    V1  V2  td   tr   tf    Pw   Per
*                     |   |   |    |    |     |     |
  Vin C 0 DC 0 PULSE(.5  1.5  0ns 49ns 49ns   1ns 100ns)
* Vin C 0 DC 0 PULSE( 0   2   0nS  1ns  1ns 100ns 200ns)

Rfb  O_Q_b IN_DATA 1000
Rin  C IN_CLK 1000
Cin  IN_CLK 0 30f
Cldq O_Q 0 0.2p
Cldin IN_DATA 0 0.2p

* Subcircuit call: (not used in Example)
* X1 (VDD IN_DATA IN_CLK IN_RST IN_SET O_Q O_Q_b) DFBF311_SCH

Rrst VDD IN_RST 1000
Rset VDD IN_SET 1000

************************ End of Test-Bench ************************
```

Dfbf311_lay.hdr

Save the following text as a text file named: Dfbf311_lay.hdr

```
*********************************************************************
* Test DFF1_PFD      Simon.Harpham@SiliconDevices.co.uk
************************ Created 11-06-03 ************************
************************ Dfbf311 Test-Bench ************************

Vdd V 0   DC 5V
Rdd V Vdd 10

*                     V1  V2  td   tr   tf    Pw   Per
*                      |   |   |    |    |     |     |
* Vin C 0 DC 0 PULSE(.5  1.5  0ns 49ns 49ns   1ns 100ns)
  Vin C 0 DC 0 PULSE( 0   2   0nS  1ns  1ns 100ns 200ns)

Rfb  q_ data 1000
Rin  C clk 1000
Cin  clk 0 30f
Cldq q 0 0.2p
Cldin data 0 0.2p

* Subcircuit call: (not used in Example)
* X1 (Vdd data clk rst set q q_ ) DFBF311_LAY

Rrst Vdd rst 1000
Rset Vdd set 1000

************************ End of Test-Bench ************************
```

SPICE Netlist Footer Files

Dfbf311_sch.ftr

Save the following text as a text file named: Dfbf311_sch.ftr

```
************************ Library Definition ************************
* Definitions for Schematic:
* Level 2 model nchan model for Orbit CN20
.MODEL CMOSNB NMOS LEVEL=2 PHI=0.60 TOX=4.350E-08 XJ=0.2U TPG=1
+ VTO=0.8756 DELTA=8.5650E+00 LD=2.3950E-07 KP=4.5494E-05
+ UO=573.1 UEXP=1.5920E-01 UCRIT=5.9160E+04 RSH=1.0310E+01
+ GAMMA=0.4179 NSUB=3.3160E+15 NFS=8.1800E+12 VMAX=6.0280E+04
+ LAMBDA=2.933E-02 CGDO=2.8518E-10 CGSO=2.8518E-10 CGBO=4.0921E-10
+ CJ=1.0375E-04 MJ=0.6604 CJSW=2.1694E-10 MJSW=0.178543 PB=0.800

* Level 2 model pchan model for Orbit CN20
.MODEL CMOSPB PMOS LEVEL=2 PHI=0.60 TOX=4.350E-08 XJ=0.2U TPG=-1
+ VTO=-0.8889 DELTA=4.8720E+00 LD=2.9230E-07 KP=1.5035E-05
+ UO=189.4 UEXP=2.7910E-01 UCRIT=9.5670E+04 RSH=1.8180E+01
+ GAMMA=0.7327 NSUB=1.0190E+16 NFS=6.1500E+12 VMAX=9.9990E+05
+ LAMBDA=4.229E-02 CGDO=3.4805E-10 CGSO=3.4805E-10 CGBO=4.0305E-10
+ CJ=3.2456E-04 MJ=0.6044 CJSW=2.5430E-10 MJSW=0.244194 PB=0.800
************************ Commands for Spice3 ************************
.OPTIONS METHOD=GEAR ITL5=0  RELTOL=0.001  ABSTOL=1n VNTOL=1m
*#destroy all
*#run
*#set editor="C:\Program Files\Windows NT\Accessories\WordPad.exe"
*********************************************************************
************************ Simulations & Outputs ************************
.control
 where
 echo

 set temp = 125
 alter @Vdd[DC] = 2.2
 tran 0.2n 200n UIC
 rusage temp

 set temp = -40
 alter @Vdd[DC] = 2.2
 tran 0.2n 200n UIC
 rusage temp

 plot tran1.v(O_Q) tran2.v(O_Q)

 status
 rusage time solvetime totiter trantime tranpoints space
.endc

*********************************************************************
******************** www.silicondevices.com ************************
*********************************************************************
```

Dfbf311_lay.ftr

Save the following text as a text file named: Dfbf311_lay.ftr

```
********************** Library Definition ***********************
* Definitions for Layout:
* (copied from Dfbf311_lay.ftr)
.MODEL NMOS NMOS LEVEL=2 PHI=0.60 TOX=4.350E-08 XJ=0.2U TPG=1
+ VTO=0.8756 DELTA=8.5650E+00 LD=2.3950E-07 KP=4.5494E-05
+ UO=573.1 UEXP=1.5920E-01 UCRIT=5.9160E+04 RSH=1.0310E+01
+ GAMMA=0.4179 NSUB=3.3160E+15 NFS=8.1800E+12 VMAX=6.0280E+04
+ LAMBDA=2.933E-02 CGDO=2.8518E-10 CGSO=2.8518E-10 CGBO=4.0921E-10
+ CJ=1.0375E-04 MJ=0.6604 CJSW=2.1694E-10 MJSW=0.178543 PB=0.800

.MODEL PMOS PMOS LEVEL=2 PHI=0.60 TOX=4.350E-08 XJ=0.2U TPG=-1
+ VTO=-0.8889 DELTA=4.8720E+00 LD=2.9230E-07 KP=1.5035E-05
+ UO=189.4 UEXP=2.7910E-01 UCRIT=9.5670E+04 RSH=1.8180E+01
+ GAMMA=0.7327 NSUB=1.0190E+16 NFS=6.1500E+12 VMAX=9.9990E+05
+ LAMBDA=4.229E-02 CGDO=3.4805E-10 CGSO=3.4805E-10 CGBO=4.0305E-10
+ CJ=3.2456E-04 MJ=0.6044 CJSW=2.5430E-10 MJSW=0.244194 PB=0.800

********************** Commands for Spice3 ************************
.OPTIONS METHOD=GEAR ITL5=0  RELTOL=0.001  ABSTOL=1n VNTOL=1m
*#destroy all
*#run
*#set editor="C:\Program Files\Windows NT\Accessories\WordPad.exe"
*******************************************************************
********************** Simulations & Outputs ************************

.control
 where
 echo

 set temp = 125
 alter @Vdd[DC] = 2.2
 tran 0.2n 200n UIC
 rusage temp


 set temp = -40
 alter @Vdd[DC] = 2.2
 tran 0.2n 200n UIC
 rusage temp

 plot tran1.v(q) tran2.v(q)

 status
 rusage time solvetime totiter trantime tranpoints space
.endc

*******************************************************************
******************** www.silicondevices.com ***********************
*******************************************************************
```