The IIT standard cell library Version 2.1

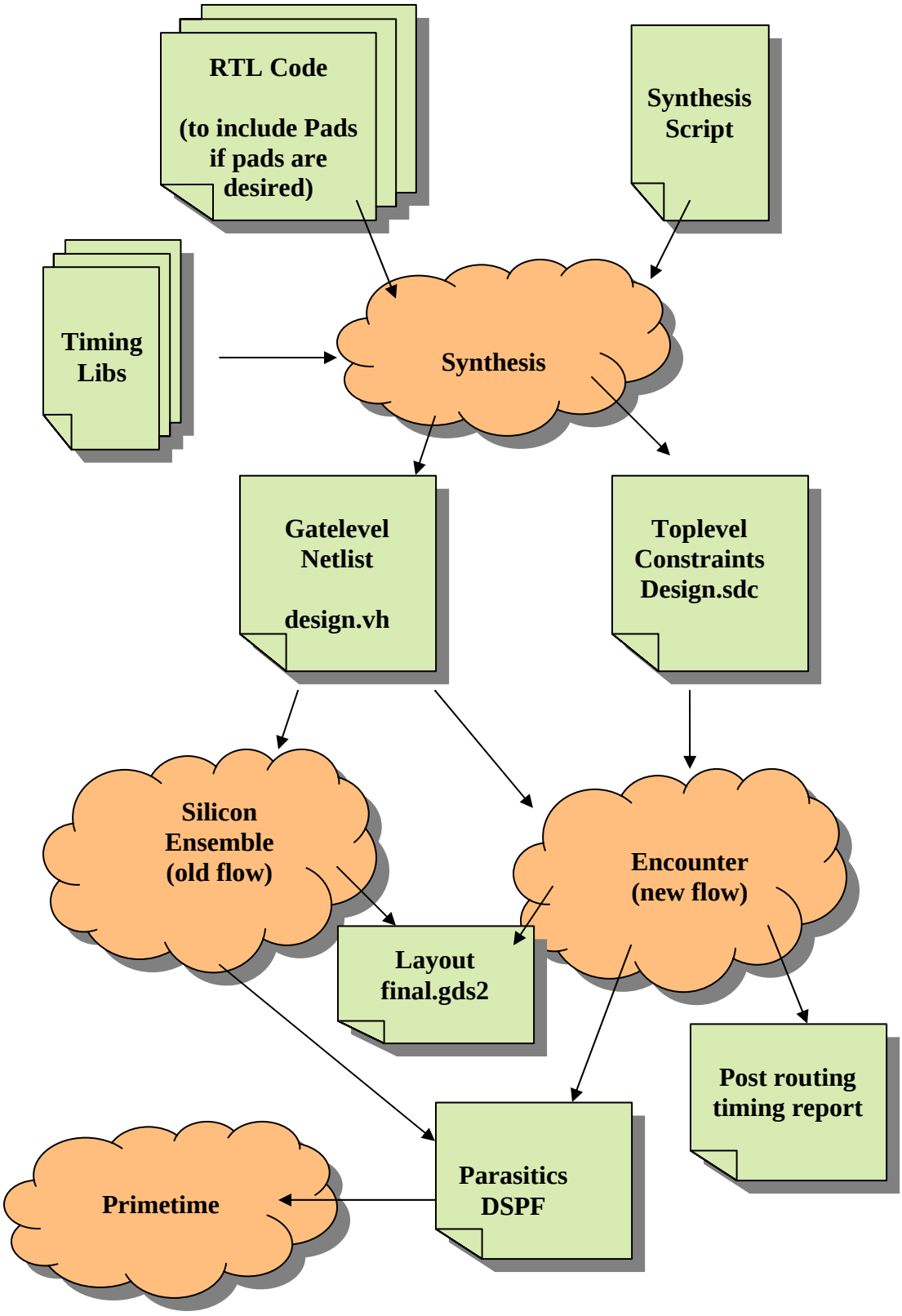
Highlights

- Support for AMI 0.35um library, including pads
- Added Primetime and Pathmill support to IIT ASIC Flow
- Support for stacked vias (for Virtuoso and Magic7)
- IIT ASIC Flow now fully supports Magic 7.3
- AMI 0.5 still supports Magic 6.5
- Magic 7.3 techfiles and setup instructions included
- All libraries now include Sue Schematics
- Many bug fixes in the schematics
- LVS of device widths now possible with Virtuoso and Gemini
- Added new script iitcells_ext2sim for easier netlisting with Magic
- Encounter and Silicon Ensemble now output DSPF for Primetime
- Encounter flow offers
    - timing-driven placement and routing
    - static timing analysis (renders Primetime optional)
    - clock tree synthesis
    - in-place-optimization and netlist modifications
    - Geometry and Connection Verification (DRC)
- Release is now structured into three main blocks
    - ./lib All library files
    - ./flow The IIT ASIC flow (script templates for most major tools)
    - ./ref_designs Reference Designs to verify and demonstrate usage
- The flow and reference design parts are optional for those only interested in the libraries

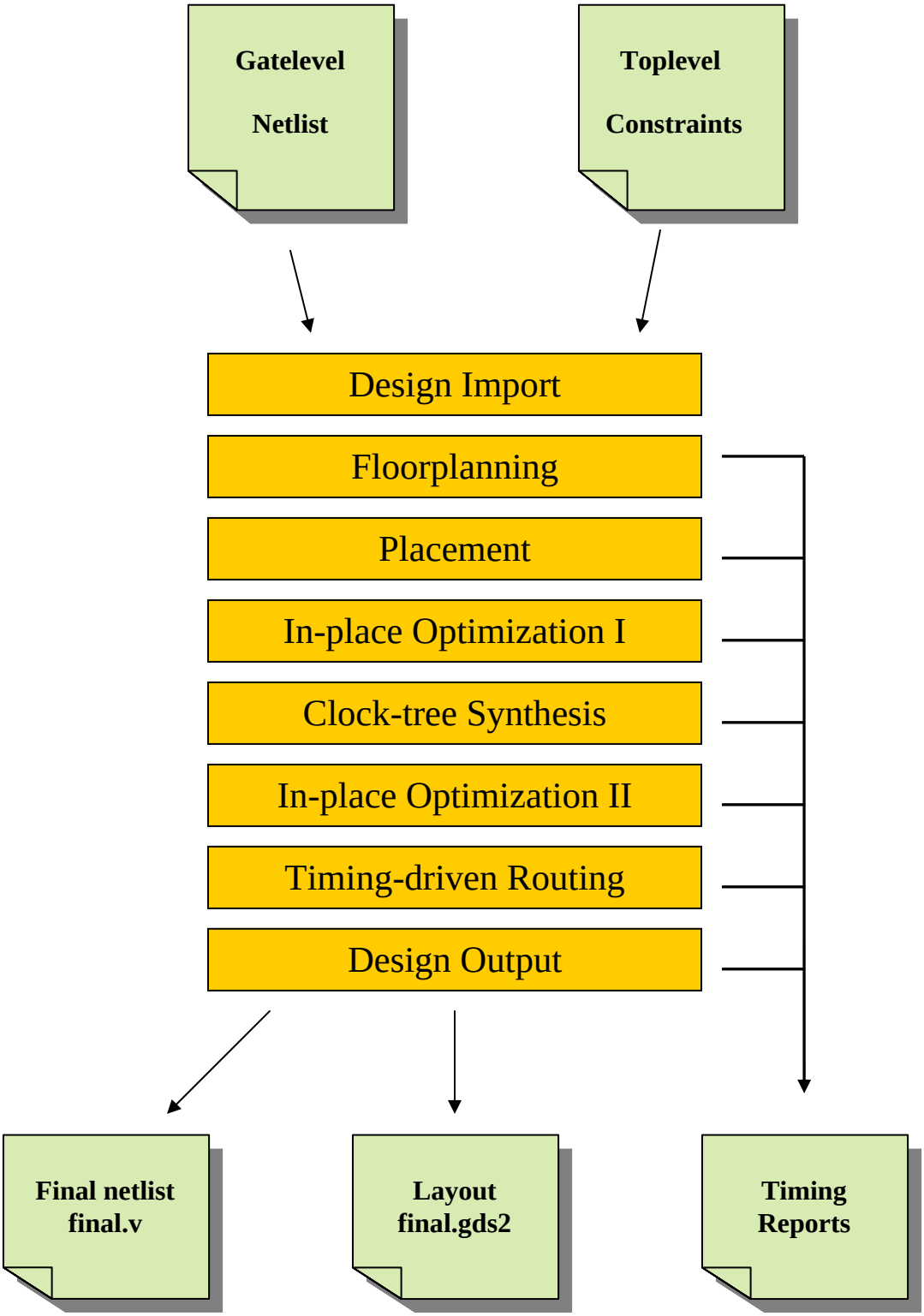
Where are the files located?

Library	Path	Comments
AMI 0.5um	./flow/ami05 Or ami05.stacks	- Default LEF file excludes stacked vias - Use ami05.stacks for stacked vias - Pad synthesis no longer supported (insert padframe in RTL code)
AMI 0.35um	./flow/ami035	- Includes TSMC 0.35um Pads - Uses stacked vias, use with Magic 7.3
TSMC 0.25um	./flow/tsmc025	- Uses stacked vias, use with Magic 7.3
TSMC 0.18um	./flow/tsmc018	- Now uses 6 metal layers - Uses stacked vias, use with Magic 7.3

The IIT ASIC Design Flow



What does the new IIT Encounter Flow Include?



How to setup a new design?

1. Simulate and test all Verilog RTL code
2. Make a run directory, e.g. “seultra” or “encounter”
3. Copy templates into that directory, e.g.
    - cp /import/cad2/iit_stdcells/flow/tsmc018/* ./encounter

How to run Synthesis with Synopsys Design Compiler

1. Customize compile_dc.scr
    a. Specify the names of all Verilog files
    b. Specify the toplevel names
    c. Specify the target clock frequency (in MHz)
    d. Specify the name of the clock pin
    e. Specify the input and output delays (or accept the default value of 1ns)
2. Run “dc_shell –f compile_dc.scr
3. The results
    a. The gate-level netlist (.vh)
    b. The toplevlel timing constraints (.sdc)
    c. Timing report: timing.rep
    d. Area report: cell.rep
    e. Power report: power.rep

How to run Synthesis with Cadence BuildGates

1. Customize compile_bgx.scr
    a. Specify the names of all Verilog files
    b. Specify the toplevel names
    c. Specify the target clock frequency (in MHz)
    d. Specify the name of the clock pin
    e. Specify the input and output delays (or accept the default value of 1ns)
2. Run “bgx_shell –f compile_bgx.scr
3. The results
    a. The gate-level netlist (.vh)
    b. The toplevlel timing constraints (.sdc)
    c. Timing report: timing.rep
    d. Area report: cell.rep
    e. Power report: power.rep

How to place&route with Silicon Ensemble (old flow)

1. Customize seultra.scr
2. Run “se_shell –f seultra.scr”
3. The results
    a. Layout file “final.gds2”

How to place&route with Encounter (new flow)

1. Customize encounter.conf
    a. Enter the toplevel name at the top
    b. Uncomment the “encounter.io” line if there are pads
2. Run “encounter –init encounter.tcl”
3. Either enter “win” to open the GUI with the chip or “exit” to quit
4. The results
    a. Layout file “final.gds2”
    b. Final netlist “.v”
    c. Final timing report: “timing.rep.5.final”
    d. Other timing reports from various place&route stages: “timing.rep*”

How to create a Magic 6.5 Layout (only AMI 0.5um)

1. If used “seultra”: run “iitcells_se2magic”
2. If used “encounter: run “iitcells_enc2magic”

Note: only works with ami05, not ami05.stacks, due to stacked via issues

How to create a Magic 7.3 Layout (all technologies)

3. If used “seultra”: run “iitcells_se2magic7”
4. If used “encounter: run “iitcells_enc2magic7”

How to create a Cadence Layout

1. If used “seultra”: run “iitcells_se2icfb”
2. If used “encounter: run “iitcells_enc2icfb”

How to run a design with pads

1. Insert pads in the RTL code of your design (see the example designs)
2. Simulate the RTL code to make sure it still works
3. Run Synthesis as usual
4. For Silicon Ensemble: no change necessary
5. For Encounter
    a. Uncomment the “encounter.io” line in “encounter.conf”
    b. Make sure the name in “encounter.io” match the pads in the RTL code
    c. You can modify the “encounter.io” file to specify pads locations
    d. Now run Encounter as usual

What if I am using AMI 0.5um but I want stacked vias

1. Stacked vias make routing much easier because
    a. The router can change layers without “stair cases”
    b. The router can connect to the cell anywhere, not just on the pin
    c. Everything except the pin used to be an obstruction, even though it was still part of the net and valid for connection
2. Use “./flow/ami05.stacks” instead of “./flow/ami05”

3. Run as usual
4. Use Magic 7.3, not Magic 6.5 as your layout editor

How to run LVS with Sue against Magic?

1. A ./sue folder is automatically created by iitcells_xxx2magicX
2. Use sue to create a .sim netlist from the sue schematic
3. If pads are being used, manually add one instance each of PADVDD and PADGND (this is because the verilog import cannot import those)
4. To netlist an extracted Magic layout:
    a. Do :ext in Magic to extract the layout
    b. Do "iitcells_ext2sim layout_name lambda"
    e.g. "iitcells_ext2sim nrd_05 0.3
    Note that this will create nrd_05.sim and nrd_05_noRC.sim
    Use the _noRC.sim file for Gemini LVS
5. Now run LVS using Gemini
gemini ./magic/netlist_noRC.sim ./sue/netlist.sim

Installing Magic 7.3

1. Download the latest version from http://bach.ece.jhu.edu/~tim/programs/magic
2. Configure and compile the source code
```
./configure --prefix=/net/magic-7.3 --without-opengl
make
make install
```
(Put your desired target path for "prefix". At IIT we disabled Open-GL because of a local issue. Be sure to try it with Open-GL (default). If you use Solaris 8, IIT can provide pre-compiled binaries of Magic 7.3)
3. To obtain a command line version of "ext2sim" do this:
make veryclean
```
./configure --without-tcl
make
cp ext2sim/ext2sim /net/magic-7.3/bin
```
(This compiles the code again but does not install anything. Simply grab the ext2sim binary and put it wherever you want it).
4. Set $CAD_HOME to the magic install directory and add $CAD_HOME/bin to your path (typically in .cshrc):
```
setenv CAD_HOME /net/magic-7.3
set path = ($CAD_HOME/bin $path)
```
5. The techfiles are included in the IIT Asic flow release in /flow/techfiles. Copy them to $CAD_HOME/lib/magic/sys. To define a default techfile, create a symbolic link called "scmos.tech" to your desired target default file.
6. If you would like to keep your keyboard definitions from an older Magic installation, simply copy $CAD_HOME/lib/magic/sys/.magic from the old to the new magic installation.
7. If you are not compiling a new version of irsim, you need to copy "anXhelper" to $CAD_HOME/bin for irsim to be able to open an analyzer window.

8. If you don’t want the TCL command window, start magic with “magic –nocon”.

Known Problems and Solutions

1. Silicon Ensemble leaves gaps between pads
   This tools has a tendency to leave small gaps between the corner cells and pad rows. Use a layout editor to verify and move the pads if necessary.
2. Sue schematics with pads are missing PADVDD and PADGND
   Due to a limitation of “v2sue”, cells with no pins cannot be imported into Sue. That includes the corners, supply pads and spacer pads. This also means that LVS with device width comparison will fail, but normal LVS will still work. The user has to manually add PADVDD and PADGND to the schematic to get matching LVS.