Automatic Generation of Configuration Files for AccuCell

AccuCell provides an auto configuration command, lib2cfg, that allows users to automatically create configuration files for a library and for each cell. A pre-existing Liberty format library is required to run this command and Figure 1 shows its function.
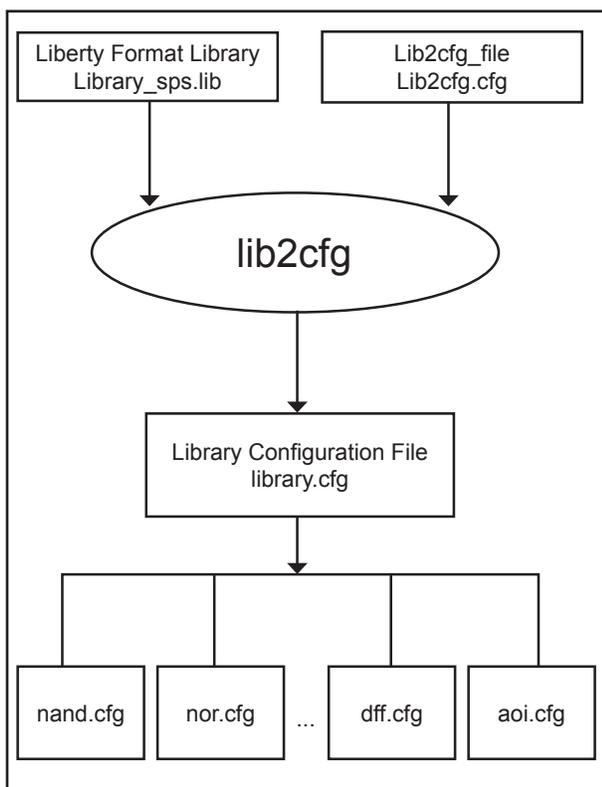

Figure 1: Input and output.

The command needs two arguments as shown below:

Lib2cfg *library_sps.lib lib2cfg_file*

Lib2cfg_file is required to run this command. Examples of this file (which can be left blank) are provided below. This file is generally used to control the contents of generated configuration files.

```
POWERS VDD

GROUNDS VSS

SUBCKT_EXTN net

SLOPE 0

IN_FILE_NAME /home/user/spice_netlists

or

POWERS VDD

GROUNDS VSS

IN_FILE_NAME. my_netlists.net
```

Figure 2. Example of "Lib2cfg_file".

- The POWERS, GROUNDS each define the global node for the PG rails of the cells.

- SUBCKT_EXTN can be used to define the <ext> extension of the netlist for each cell in the form of <cell_name>.<ext>.

- SLOPE 0 turns off the defining of SLOPE_TABLE slope values in the cell .cfg file so as to ignore the predefined slopes in the .lib file read by lib2cfg. Setting SLOPE 1 (or leaving it out of the lib2cfg_file) includes slope definitions in each cell as defined in the .lib file read by lib2cfg.

- IN_FILE_NAME in the lib2cfg_file may point to an individual file containing ALL cell netlists as individual cell .subckt definitions, OR it may be a path to a directory containing a set of netlists for each cell. In the latter case, SUBCKT_EXTN should also be used.

With a blank lib2cfg.cfg, Lib2cfg will create a generic configuration file, a cell_list file and directories for each cell as shown in Figure 3. A cell configuration file is also created for each cell.

To auto generate configuration files, create a library.tcl file as follows:

```
lib2cfg sample_library_sps.lib lib2cfg.cfg
```

Lastly, at the system prompt type:

accucell *library.tcl*

The following example illustrates the result of running lib2cfg for a library containing cells INV, NOR, NAND and MUX.

```
[rhel4-64txc test]$ ls
lib2cfg.cfg  library.tcl  sample_library_sps.lib
[rhel4-64txc test]$ accucell library.tcl &

Loading ByteCode Loader.....
AccuCell(R) 2.4.9.C, Fri Aug 13 14:11:28 PDT 2010, Proprietary and
Copyright (c) 1984 - 2010, Silvaco Inc.

starting reading sample_library_sps.lib
sample_library_sps.lib read successfully
Processing library sample_library with 6 cells.

[rhel4-64txc test]$ ls
delay.1  lib2cfg.cfg  mux2   nor2                        sample_libr
inv      library.tcl  nand2  sample_library.cell_list  sample_libr
```

Figure 3. Generated files from auto-generation command.

```
…
…
cell (inv) {
  area : 0;
  pin (A) {
    direction : input ;
    capacitance : 0.01167;
    rise_capacitance : 0.01166;
    fall_capacitance : 0.01167;
    rise_capacitance_range (0.01166 , 0.01166) ;
    fall_capacitance_range (0.01167 , 0.01167) ;
    clock : false;
    max_transition : 1.0;
  }
  pin (Y) {
    direction : output;
    max_capacitance : 0.29498;
    function : "(!A)";
    internal_power () {
      related_pin : "A";
      rise_power (pwr_template4x4) {
        index_1 ("0.12500, 0.25000, 0.50000, 1.00000");
        index_2 ("0.01000, 0.02000, 0.04000, 0.08000");
        values  ("0.04925, 0.06377, 0.09399, 0.15582", \
                 "0.06145, 0.07462, 0.10282, 0.16216", \
                 "0.08851, 0.10006, 0.12573, 0.18125", \
                 "0.14490, 0.15451, 0.17700, 0.22763");
      }
      fall_power (pwr_template4x4) {
        index_1 ("0.12500, 0.25000, 0.50000, 1.00000");
        index_2 ("0.01000, 0.02000, 0.04000, 0.08000");
        values  ("0.02038, 0.03705, 0.06922, 0.13245", \
                 "0.00949, 0.02813, 0.06254, 0.12779", \
                 "0.01637, 0.00482, 0.04329, 0.11335", \
                 "0.07218, 0.04830, 0.00485, 0.07340");
      }
    }
    timing () {
      related_pin : "A";
      timing_sense : negative_unate;
      cell_rise (delay_template4x4) {
        index_1 ("0.12500, 0.25000, 0.50000, 1.00000");
        index_2 ("0.01000, 0.02000, 0.04000, 0.08000");
        values  ("0.07676, 0.10078, 0.15069, 0.25288", \
                 "0.11439, 0.14147, 0.18932, 0.28745", \
                 "0.18412, 0.21727, 0.27138, 0.36649", \
                 "0.31137, 0.35516, 0.42304, 0.53144");
      }
```

Figure 4. Liberty Library Sample "sample_library_sps.lib".

Conclusion

Depending on the nature of the Liberty.lib file used, the generated configuration files may require modification prior to running characterization. However, in most cases all required configuration setup will be complete with ready-to-run files. Therefore a significant productivity boost for most characterization efforts will result.