

AccuCellの コンフィギュレーション・ファイルの自動生成

AccuCellには、ユーザーがライブラリおよび各セルのコンフィギュレーション・ファイルを自動で作成することができる自動コンフィギュレーション・コマンドlib2cfgがあります。このコマンドの実行には、既存のLibertyフォーマットのライブラリが必要であり、図1にその機能が示されています。

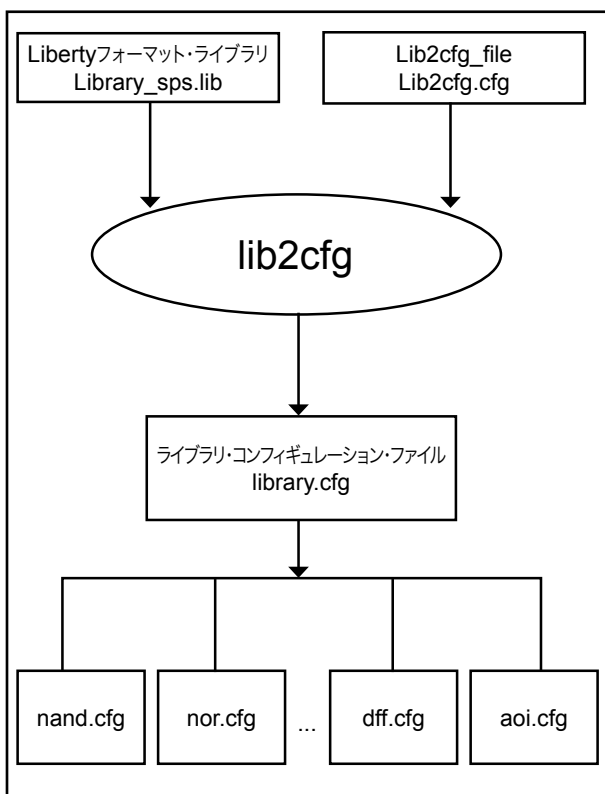


図1: 入力と出力

このコマンドは、次に示される2つの引数を必要とします。

lib2cfg *library_sps.lib* *lib2cfg_file*

このコマンドの実行には、*lib2cfg_file*が必要となります。このファイル(ブランクでも可)の例を次に示します。このファイルは通常、生成されたコンフィギュレーション・ファイルの内容の管理に使用されます。

```

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
  
```

図2: lib2cfg_fileの例

- POWERSおよびGROUNDはそれぞれ、セルのPGLルールにグローバル・ノードを定義します。
- SUBCKT_EXTNは、各セルのネットリストの拡張子<ext>を定義し<cell_name>.<ext>という形式にするために使用されます。
- SLOPE 0は、lib2cfgにより読み取られた.libファイルに予め定義された傾きを無視し、セルの.cfgファイルのSLOPE_TABLEの傾き値の定義をオフにします。SLOPE 1(またはlib2cfg_fileから削除)では、lib2cfgにより読み取られた.libファイルに定義されている傾きの定義を各セルに反映させます。
- lib2cfg_fileのIN_FILE_NAMEは、セルのネットリストをすべて含んだ個別のファイルを、個別のセルの.subckt定義として指定します。または、各セルのネットリストすべてが入ったディレクトリへのパスの指定でも可です。後者の場合、SUBCKT_EXTNも使用してください。

lib2cfg.cfgがブランクのとき、Lib2cfgは一般的なコンフィギュレーション・ファイル、cell_listファイルならびに各セルのディレクトリを図3のように作成します。各セルに対するセル・コンフィギュレーション・ファイルも作成します。

コンフィギュレーション・ファイルを自動生成するには、library.tclファイルを次のように作成します。

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

最後に、システム・プロンプトで次のように入力します。

```
accucell library.tcl
```

次の例は、INV、NOR、NANDおよびMUXセルを含むライブラリに対するlib2cfgの実行結果を示しています。

```
[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
```

図3: 自動生成コマンドから生成されたファイル

```

...
...
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");
    }
  }
}

```

図4: Libertyライブラリのサンプルsample_library_sps.lib

まとめ

使用されるLiberty.libファイルの性質により、生成されたコンフィギュレーション・ファイルはキャラクタライズ実行前に修正が必要となる場合があります。しかし、多くの場合は必要な全てのコンフィギュレーション設定が完了し、そのまま実行可能なファイルになります。したがって、ほとんどのキャラクタライズ作業において生産性を大きく向上させることができます。