FH-OÖ Hagenberg/ESD
Metrikorientierter Hardwareentwurf, WS 2015
Rainer Findenig, Markus Schutti © 2008 (R 1664)

5. Übung: PROL16: Power to the People!

Name(n): Punkte:

1 Power Analysis

In dieser Übung sollen Sie die Stromaufnahmne Ihres PROL16 analysieren. Der entsprechende Ablauf ist in Abbildung 1 dargestellt.

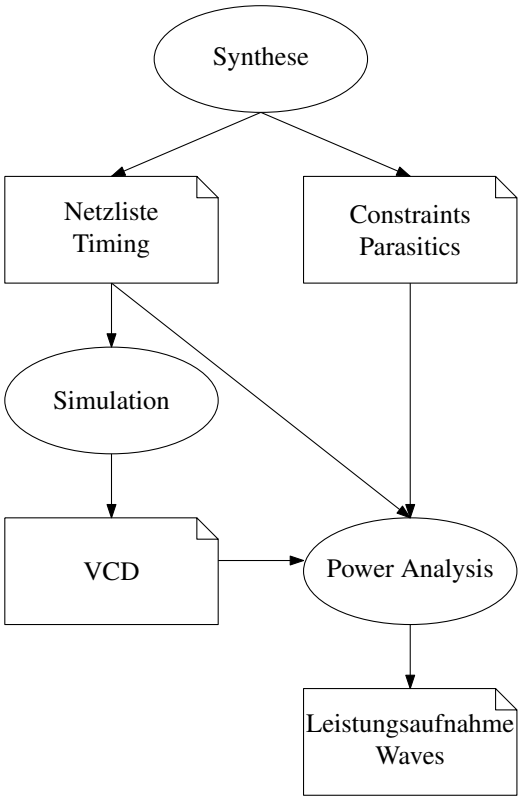

Abbildung 1: Ablauf der Power-Simulation.

Für die Power-Simulation sind also die folgenden Schritte auszuführen:

- Das Syntheseskript ist so zu erweitern, dass neben der Netzliste und der Timing-Information (SDF) auch eine SDC-Datei (*Synopsys Design Constraints*, Befehl `write_sdc`) und eine *Parasitics*-Datei (`write_parasitics`) erzeugt werden. Beachten Sie dabei, dass eine Verilog-Netzliste erstellt werden muss.

- Die Postlayout-Simulation ist wie gewohnt durchzuführen, und dabei eine VCD-Datei (*Value Change Dump*, Befehle `vcd file` und `vcd add`) zu erzeugen. Diese Datei beinhaltet die Switching-Informationen des Entwurfs. Das Ergebnis der Simulation hängt in diesem Fall natürlich stark vom exekutierten Programm ab; dieses sollte daher möglichst repräsentativ sein. Für diese Übung sollen Sie Ihr Testprogramm aus dem fünften Semester verwenden.
- Abschließend können Sie mit der Netzliste, den Constraints, den Parasitics und der VCD-Datei die Power-Simulation durchführen. Verwenden Sie dazu das Programm Synopsys PrimeTime PX (`primetime`) mit dem folgenden Skript:

```
set AMS_DIR     [getenv AMS_DIR]
set SYNOPSYS    [getenv SYNOPSYS]
set TECH        c35_3.3V
set WORK_DIR    work
set script_path [getenv SYNOPSYS_SCRIPTS];

set search_path  ". \
                  $AMS_DIR/synopsys/$TECH \
                  $AMS_DIR/synopsys/generics \
                  $SYNOPSYS/libraries/syn    \
                  $SYNOPSYS/dw/sim_ver"

set synthetic_library dw_foundation.sldb
set target_library    c35_CORELIB.db
set link_library      "* $target_library  $synthetic_library"
set link_create_black_boxes false

set power_enable_analysis TRUE
set power_analysis_mode time_based

########################################################
# link design
########################################################
set power_enable_analysis true
read_verilog netlist/prol16.v
current_design cpu
link

read_sdc ./prol16.sdc
read_parasitics ./prol16.spf

########################################################
# read switching activity file
########################################################
read_vcd -strip_path cpu_tb/dut ../sim/prol16.vcd

########################################################
# timing analysis
########################################################
check_timing
update_timing
report_timing

########################################################
# power analysis
########################################################
```

```
check_power
set_power_analysis_options -waveform_output prol16
update_power
report_power -hierarchy
```

-

□ Interpretieren Sie die durch `report_power` ausgegebenen Werte!

2 Clock Gating

Um die Leistungsaufnahme zu verringern sollen Sie Ihren PROL16 nun mit *Clock Gating* synthetisieren. Verwenden Sie dazu die Option `-gate_clock` zum Befehl `compile`. Das Ergebnis dieses Prozesses können Sie mit `report_clock_gating -verbose -gated -ungated` prüfen. Sie werden bemerken, dass viele Register „gated“ sind, jedoch nach wie vor einige als „ungated“ definiert sind. Erklären Sie das Verhalten speziell für die Register `RegTmpA`/`RegTmpB` und `Carry`/`Zero` (Tipp:
□ `set_clock_gating_style`)!

2.1 Postlayout-Simulation

Prüfen Sie das Ergebnis in der Postlayout-Simulation. Erstellen Sie danach ein Programm, das in einer Endlosschleife zuerst einige Zeit wartet und dann auf ein bestimmtes Register einen Wert schreibt. Vergleichen Sie im Wave-Fenster den Takt am Eingang dieses Registers (also am Eingang
□ eines Flipflops dieses Registers) mit dem Takteingang Ihrer CPU!

2.2 Power-Simulation

Führen Sie die Power-Simulation, wie im ersten Abschnitt beschrieben, nun mit der Netzliste mit Clock Gating aus. Wie stark ist die Verbesserung? Wo ergibt sich die größte, wo die geringste
□ Einsparung, und warum?

2.3 Glock Gating und DfT

Nach dem Einfügen der Clock-Gating-Zellen können Probleme beim Einfügen einer Scan-Chain
□ entstehen! Erklären Sie das auftretende Problem!