PowerArtist™: RTL Design-for-Power

Design Automation Conference 2014
Early Power Decisions → High Impact

### RTL Design-for-Power
- Power-Performance-Area Trade-offs
- Voltage / Power Domain Planning
- Block-level Clock and Data Gating
- Eliminate Redundant Activity

### Low Power Implementation
- Power Switch Sizing / Placement
- Clock Gater Cloning / Decloning
- Multi-Vt Optimization
- Power Integrity Verification
RTL Power ↔ Gate-level Power

Quicker Design Iterations

- Design Specification
- RTL Design
- Gate-Level Design
- Layout

~20 hours

~22 mins

Effective Design-for-Power

- RTL Power
- Power-per-Function
- Gate-level Power
- Power-per-Gate

Quicker Design Iterations

- Design Specification
- RTL Design
- Gate-Level Design
- Layout

~20 hours

~22 mins

Effective Design-for-Power

- RTL Power
- Power-per-Function
- Gate-level Power
- Power-per-Gate
PowerArtist: RTL Design-for-Power Platform

**RTL Power Analysis**
- Average, time-based
- Power-critical vector selection
- Regressions via TCL interface

**RTL Power Reduction**
- Clock, memory, logic
- Analysis-driven automation
- Interactive power debug

**RTL Links with Physical**
- PACE™: RTL power accuracy
- RPM™: RTL-driven physical power integrity
RTL Power: Ins and Outs

module PA ( ...
  always @ (posedge clk) begin
    dout <= din1;
  end
  assign out = sel ? dout : din2;
  ...
endmodule

RTL (VHDL, Verilog, System Verilog)

Clock tree, gating (SDC, PACE, user input)

Activity (FSDB / VCD / SAIF)

Capacitance model (WLM / PACE)

Power domains (UPF / CPF)

Power models (Liberty .lib)

RTL Power Analysis
Low Power RTL Design Methodology

- Perform design trade-offs
- Profile power vectors
- Check power vs. budget
- Debug power hotspots
- Reduce power automatically
- Monitor power vs. budget

Peak Power = 391mW
Average power = 239mW

RTL Power Regression Flow
### RTL vs. Gates: Accuracy and Performance

**Nvidia Case Study**

**RTL Power Accuracy:** ~15%

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Dynamic Power (mW)</th>
<th>Leakage Power (mW)</th>
<th>% of max dyn power</th>
<th>Dynamic Power (mW)</th>
<th>Leakage Power (mW)</th>
<th>% of max dyn power</th>
<th>Power Artist vs. PT-Flow</th>
<th>% of Power error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test 1</td>
<td>185.866</td>
<td>53.128</td>
<td>100.00%</td>
<td>188.789</td>
<td>55.789</td>
<td>100.00%</td>
<td>-3.1%</td>
<td>-13.74%</td>
</tr>
<tr>
<td>Test 2</td>
<td>183.906</td>
<td>54.900</td>
<td>100.00%</td>
<td>187.053</td>
<td>56.713</td>
<td>100.00%</td>
<td>-3.5%</td>
<td>-14.46%</td>
</tr>
<tr>
<td>Test 3</td>
<td>182.381</td>
<td>55.377</td>
<td>99.50%</td>
<td>188.012</td>
<td>58.988</td>
<td>99.90%</td>
<td>-3.7%</td>
<td>-13.52%</td>
</tr>
<tr>
<td>Test 4</td>
<td>181.259</td>
<td>55.127</td>
<td>100.00%</td>
<td>187.490</td>
<td>55.793</td>
<td>100.00%</td>
<td>-3.7%</td>
<td>-12.97%</td>
</tr>
<tr>
<td>Test 5</td>
<td>180.958</td>
<td>54.852</td>
<td>99.90%</td>
<td>187.122</td>
<td>55.670</td>
<td>99.90%</td>
<td>-3.6%</td>
<td>-7.64%</td>
</tr>
<tr>
<td>Test 6</td>
<td>180.706</td>
<td>54.377</td>
<td>101.00%</td>
<td>188.921</td>
<td>58.735</td>
<td>101.00%</td>
<td>-3.6%</td>
<td>-4.21%</td>
</tr>
<tr>
<td>Test 7</td>
<td>180.267</td>
<td>54.060</td>
<td>101.00%</td>
<td>188.224</td>
<td>58.681</td>
<td>101.00%</td>
<td>-3.4%</td>
<td>-4.81%</td>
</tr>
<tr>
<td>Average Power</td>
<td>180.295</td>
<td>54.060</td>
<td>101.00%</td>
<td>188.224</td>
<td>58.681</td>
<td>101.00%</td>
<td>-3.4%</td>
<td>-4.81%</td>
</tr>
</tbody>
</table>

**RTL Power: ~30X faster**

- **Power Artist vs. PT-Flow Flow Runtime**
  - **Total Runtime for 7 tests**
    - Power Artist: 1 hour
    - PT-Flow: 1 hour
  - **Significant power analysis speedup can be achieved**
  - **Synthesis runtime is the main bottleneck**

- **1 license used for PA runs**
RTL Capacity: Large Designs / FSDBs
Samsung Case Study

Critical Signal Flow for RTL Power Estimation (2)

- Experimental result with Design-B in RTL
  - The second experiment was done with quad-core CPU block
  - Design size is Tens of Million Gates, with 32nm library

FSDB captures only power-critical signals identified by PowerArtist

- FSDB size: 1/4
- TAT: 4X faster
- Loss of accuracy: 2%
RTL Power Analysis
PowerArtist RTL Power Analysis

Activity Analysis
- Total Logic / Clock Activity per Hierarchical Instance
- Qualify Coverage per Power Mode
- Identify Power Bugs

Average Power Analysis
- Understand Power: Where? Why?
- Per Hierarchy, Category, Mode, Clock / Voltage Domains
- Qualify Power Efficiency with Multiple Metrics

Time-based Power Analysis
- Power Waveforms per Hierarchical Instance
- Waveforms per Category: Clock, Memory, Logic
- Identify Peak Power and Time
Clock Gating Efficiency
*Temporal and Structural Metrics*

**Example**
- 16 of 20 bits are gated
- 5 of 10 cycles are gated
- 2 of 5 enabled cycles had data toggles

<table>
<thead>
<tr>
<th>SCGE</th>
<th>DCGE</th>
<th>CGEE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Definition</strong></td>
<td>% Gated Bits</td>
<td>% Gated Clock Cycles</td>
</tr>
<tr>
<td><strong>Type of Metric</strong></td>
<td>Structural</td>
<td>Temporal (en, clk)</td>
</tr>
<tr>
<td><strong>Value</strong></td>
<td>80%</td>
<td>50%</td>
</tr>
</tbody>
</table>
Clock Gating Efficiency
Temporal and Structural Metrics

<table>
<thead>
<tr>
<th>Total Flops</th>
<th>Gated Flops</th>
<th>CGE</th>
<th>Wasted Clock (%)</th>
<th>Pin Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>429180</td>
<td>283248</td>
<td>63.07</td>
<td>11.9mW</td>
<td>orpsoc top</td>
</tr>
<tr>
<td>6923</td>
<td>802</td>
<td>0.00</td>
<td>0.00</td>
<td>orpsoc_top.i_orlk.i_tap_top</td>
</tr>
<tr>
<td>36812</td>
<td>2709</td>
<td>10.21</td>
<td>102mW</td>
<td>orpsoc_top.i_orlk.i_dbg_top.i_dbg_wb</td>
</tr>
<tr>
<td>3245</td>
<td>3245</td>
<td>0.00</td>
<td>0.00</td>
<td>orpsoc_top.i_orlk.i_dbg_top.i_dbg_crc32_d1_out</td>
</tr>
<tr>
<td>61554</td>
<td>7516</td>
<td>11.15</td>
<td>1.2mW</td>
<td>orpsoc_top.i_orlk.i_or1200_top.or1200_cpu.or1200_fpu.fpu0</td>
</tr>
<tr>
<td>14820</td>
<td>0</td>
<td>0.00</td>
<td>1.1mW</td>
<td>orpsoc_top.i_orlk.i_or1200_top.or1200_cpu.or1200_fpu.fpu8.u6</td>
</tr>
</tbody>
</table>

Note: SCGE : Static Clock Gating Efficiency
DCGE : Dynamic Clock Gating Efficiency

CGE: Static, Dynamic Flop: Power, Activity
100% Static CGE 0% Dynamic CGE

CGEE, Power Impact
RTL Power Reduction
PowerArtist RTL Power Reduction

1. Interactive Power Debug
   - Block-level Power "Bugs"
   - Large Power Savings

2. Automated Power Reduction
   - Instance-level Power Reduction
   - 15 Analysis-driven Techniques

3. Customizable Power Reports
   - TCL Queries to OADB
   - Automation Beyond PowerArtist Reports
Debug Power: Visualize-Analyze-Reduce

<table>
<thead>
<tr>
<th>Instance</th>
<th>Module/Cell</th>
<th>Total Par</th>
<th>Total Dyn</th>
<th>Clk Gated</th>
<th>CGE</th>
<th>Infer'd Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>rem</td>
<td>#register#</td>
<td>27.75uW</td>
<td>27.36uW</td>
<td>None</td>
<td>--</td>
<td>#1</td>
</tr>
<tr>
<td>remain</td>
<td>#register#</td>
<td>27.75uW</td>
<td>27.36uW</td>
<td>None</td>
<td>--</td>
<td>#1</td>
</tr>
<tr>
<td>mul</td>
<td>#register#</td>
<td>27.75uW</td>
<td>27.36uW</td>
<td>None</td>
<td>--</td>
<td>#1</td>
</tr>
</tbody>
</table>

Identify Block-level Clock Gating Enable
Block-Level Power Reduction

Block-level Activity Analysis: Clock and Data Ports

1.1 Clock Pins

<table>
<thead>
<tr>
<th>Redundant Cycles</th>
<th>Total Cycles</th>
<th>Pin Name</th>
<th>Mode</th>
<th>Instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>201</td>
<td>CLKA</td>
<td>read</td>
<td>top.core1.t1.dpmem.m1</td>
</tr>
</tbody>
</table>

1.2 Input and Redundant Pins

<table>
<thead>
<tr>
<th>Redundant Toggles</th>
<th>Total Toggles</th>
<th>Pin Name</th>
<th>Mode</th>
<th>Instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>AB[8]</td>
<td>read</td>
<td>top.core1.t1.dpmem.m1</td>
</tr>
</tbody>
</table>

Clock Activity per Hierarchy

Constant high activity
Missed clock gating?

Wasted Activity per Mode

Redundant activity in read mode

Block-level Clock Gating

Block-level Data Gating

Clock Active, Data Inactive

Clock Inactive, Data Active
Instance-Level Power Reduction

Clock / Clock Gating
- Clock gating coverage
- Clock gating efficiency
- Sequential and combinational

Control Logic and Datapath
- Redundant activity
- Don’t care conditions
- Datapath operand isolation

Memory Subsystem
- Redundant read/write
- Splitting memories
- Exercising sleep modes
Analysis-Driven RTL Power Reduction

Wasted activity/power when sel is 0
Analysis-Driven RTL Power Reduction

Multi-cycle ODC sequential analysis

Pre-compute based new clock gate enables
Analysis-Driven RTL Power Reduction

15 Power Reduction Techniques

- Clock, Memory, Logic
- Sequential, Combinational
- Vector-based, Vectorless
- Hierarchical, SoC capacity

Maximize Power Savings
Minimize Design Impact

Top 5 RTL changes → 50% identified power savings
Power Reduction Case Studies

**MUX Reduction Technique:**
- Scan clocks toggling in functional mode
- Redundant data activity in registers wasting power

**GMC Technique:**
- Redundant data toggles in read mode
- Cycle-based analysis reports % Redundant Cycles
Power Database Access with TCL API

Customize and Automate Power Reduction, Reports, Regressions
- Quick access to power and design properties
- Accomplish custom tasks with few lines of TCL

Power Queries
- `getPropVal instance/net`
- `getClockPower`
- `getNetPower`
- `getClockEnableExpr`

Design Queries
- `getMemories/Flops/Combs`
- `getFanout`
- `getModulePorts`
- `reportDesignStats`

Report Creation
- `reportCGEfficiency`
- `diffPdbPower`
- `reportPower`
- `reportReductions`

Design Navigation
- `dls`
- `dpwd, dcd`
- `dpushd, dpopd`
- `show`
Custom Power Reports
50% Idle Power Reduction in Mobile SoC

Inefficient enables waste power

Block-level clock gates control significant power

PowerArtist clock gating report → identifies inefficient clock gates

<table>
<thead>
<tr>
<th>Instance Name</th>
<th>Enable Efficiency</th>
<th>Clock Power</th>
<th>Clock</th>
<th>En Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>or1200_cpu.ckg12</td>
<td>0</td>
<td>5.17E-03</td>
<td>clk</td>
<td>or1200_cpu.en_blk</td>
</tr>
<tr>
<td>or1200_cpu.or1200_ctrl.ckg5</td>
<td>0.1</td>
<td>1.36E-03</td>
<td>gclk_blk</td>
<td>or1200_cpu.or1200_ctrl.n1</td>
</tr>
</tbody>
</table>

Power Efficiency = 0

Single clock gate controls >5mW
RTL Power Regressions

- 30+ blocks per typical SoC
- 2+ vectors per block
- Vectors written for power: idle, active
- Daily block-level, weekly chip-level regressions monitor power changes
- Power metrics track power efficiency
- PowerArtist identifies where power changed

RTL (Verilog, SV, VHDL)

Testbench

Simulator

FSDB

RTL Power Analysis, Reduction, Regression
RTL Links with Physical Design
PACE™: Physical-Aware RTL Power Budgeting

module PA (...
    input posedge clk;
    output dout;
    input din1;
    input din2;
    input sel;
    assign out = sel ? dout : din2;
    ...
endmodule

- Clock Distribution
- Parasitics
- Multiple Vt
- Low-power Structures
- Optimization

PACE Bridges the RTL vs. Layout Gap → Predictable RTL Power Accuracy
RTL PACE vs. Gate-Power: Mobile SoC @14nm

Total Power Correlation
Gate-SPEF vs. RTL-PACE vs. RTL-WLM

Clock Power Correlation
Gate-SPEF vs. RTL-PACE

RTL-PACE Power within 20%
RTL-PACE Clock Power within 20%
RTL Power-Driven Power Integrity

module PA (
    ...;
    always @ (posedge clk)
    begin
        dout <= din1;
        assign out = sel ? dout : din2;
        ...
endmodule

• Shrinking geometries → Increasing di/dt
• Gate vectors too late
• Layout late for changes
• Error-prone guesstimates

RPM Enables PDN Planning → Early, Optimal, Robust
RPM Case Studies

Peak and di/dt Cycle Selection on a GPU Core

Frame: DIDT
Start time: 0.0817704
Finish time: 0.0817706
Average leakage for supply VDD: 0.00257393
Average power for supply VDD: 0.185336
Peak power for supply VDD: 0.219776

Frame: CYCLE_POWER
Start time: 0.0806005
Finish time: 0.0806007
Average leakage for supply VDD: 0.002569
Average power for supply VDD: 0.250168
Peak power for supply VDD: 0.266678

Early Voltage Drop Analysis

Early Package Resonance Analysis

CPM(Layout)+Pkg
CPM(RPM)+Pkg
Pkg only
Related Presentations @ DAC2014

- Power Analysis Using PowerArtist for WaveLogic3 ASIC – 100Gbs Coherent Metro Optical Modem
- Achieving RTL Power Efficiency and Automated Power Reduction
- Methods for Achieving RTL to Gate Power Consistency