



Low Power EDA on the Bleeding Edge

April 2015

This presentation may contain forward-looking statements regarding product development. Information or statements contained in this presentation are for informational purposes only and do not represent a commitment, promise, or legal obligation of any kind by Atrenta Inc. or its affiliates.





Managing Plan Misses

- Figuring out where you're at
- Techniques to reduce power

Longer-Term: System Power Modeling @RTL

In Theory

- Instrumented TLM, high-level power-aware synthesis, ...
- Mostly research labs and very high volume uPs

In Practice

- 80% of designs derivative, lots of IP content with no HL models and limited characterization above the gate-level
- Excel spreadsheets detail IP power params versus use-cases estimate time spent in each mode to get power contribution / IP / case
- Low tech but still the dominant planning technique in the industry

IP	Idle power	Low perf power	Hi perf power	Use-case 1	Use case 2	
A57						
PCIX						
DDR4						
SRAM						
MAC						

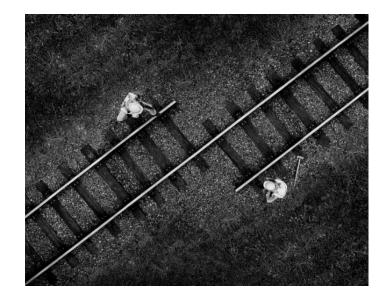
Atrenta Confidential © 2015 Atrenta Inc.

SPYG

The Plan and the Implementation..



- Plans are not always perfect
- IP power models are not always completely accurate
- Block designers trade off performance and power => some blocks overshoot budget
- You find that logic you thought could be power-switched in fact has to be always-on
 => higher energy drain
- So now you have to figure out what you've got
- And how you might make up the shortfall



Atrenta Confidential © 2015 Atrenta Inc.

SPYGLA





Managing Plan Misses

Figuring out where you're at
Techniques to reduce power

Longer-Term: System Power Modeling @RTL

What Have I Got and What Can I Tweak?

Power Explorer used as Central Cockpit for Efficient Analysis

- Complete picture of SoC power
- Re-analyze with same activity files, vary power parameters
- Efficient what-if analysis across multiple scenarios
- Experiment with reduction on other blocks: Vt mix, clock gating, lower voltage, power switch, ...
- What-if on reducing memory power, register power, ...

🗟 🕅 V 🖾 📊 Searc	:h ᢏ]	•	e e						C	_		
Register Name	Register Width	Clock E Driving Clock Name		Internal Power	Register Powe			ock Gating		Ve	Vie	Mc
ethmac.temp_wb_ack_o_reg_reg		ethmac.wb_clk_i	33.333 MHz	755.366 nW	11.749 nW		N	N	N	2.000	-	~
ethmac.\temp_wb_dat_o_reg_reg[0:31]	32	ethmac.wb_clk_i	33.333 MHz	19.206 uW	375.980 nW	610.153 nW	N	N	N	2.000	2.000	0.194
ethmac.temp_wb_err_o_reg_reg	1	ethmac.wb_clk_i	33.333 MHz	595.673 nW	11.749 nW	0.000 W	N	N	N	2.000	2.000	0.000
ethmac.CarrierSense_Tx1_reg	1	ethmac.mtx_clk_p	12.562 MHz	216.047 nW	11.749 nW	0.000 W	N	N	N	0.754	0.754	0.000
ethmac.CarrierSense_Tx2_reg	1	ethmac.mtx_clk_p	12.562 MHz	216.047 nW	11.749 nW	0.000 W	N	N	N	0.754	0.754	0.000
ethmac.Collision_Tx1_reg	1	attended with all a						N	N	0.754		
etriffac.collision_txt_reg	1	ethmac.mtx_clk_p	12.562 MHz	216.047 nW	11.749 nW	0.000 W	N	IN .	IN	0.754	0.754	0.000
mory View			€	216.047 nW	11.749 nW	0.000 W	N 	N	N		0.754	0.000
mory View		Clock Details	••••••	Power	<u></u>			 Activity D	Details			,
mory View		Clock Details Clock D	Record H	Power sakage S	<u></u>	ddr1 Ac 1 Addr2	Ac Addr1 Ac	Activity E Addr2 At at				,
mory View	ry Cell Name 1g	Clock Details Clock D	Record H	Power sakage S	witching I.A	ddr1 Ac 1 Addr2	Ac Addr1 Ac	Activity E Addr2 At at	Petails ta1 Activ ata2 Ac	tivi]⊇1 Activ	rity 22 Activ	vity Read1 F
mory View	ry Cell Name 1g	Clock Details Clock D	Record H	Power sakage S	witching I.A	ddr1 Ac 1 Addr2	Ac Addr1 Ac	Activity E Addr2 At at	Details ta1 Activ ata2 Ac	tivi]⊇1 Activ	rity 22 Activ	vit) Read1 F
mory View	ry Cell Name 1g	Clock Petails Clock h Stock Frr A NA 22	Record H	Power sakage S	witching I.A	ddr1 Ac 1 Addr2	Ac Addr1 Ac	Activity E Addr2 At at	Details ta1 Activ ata2 Ac	tivi]⊇1 Activ	rity 22 Activ	vit) Read1 F
mory View	ry Cell Name 1g	Clock Petails Clock h Stock Frr A NA 22	Internal L 584800 uW 4.145	Power sakage S	witching I.A	ddr1 Ac 1 Addr2	Ac Addr1 Ac	Activity D Addr2 At at 0.000 0	Details ta1 Activ ata2 Ac	n Tri ⊇1 Activ 0.000	vity 22 Activ 0.000	vity Read1 F
mory View	ry Cell Namx 1g	Clock Petails Clock h Stock Frr A NA 22	@ I Internal Le 564800 uW 4.145	Power eakage 5 28830 uW 525. Details	witching I F	adırı Arij Addır2 0000 0.000	Ac Addr1 Ac 0.000	Activity D Addr2 At at 0.000 0	 Details tat Activ atn2 Ac	n Tri ⊇1 Activ 0.000	vity 22 Activ 0.000	vit) Read1 F

Atrenta Confidential © 2015 Atrenta Inc.

SPYG

Power Exploration Accuracy

Factors in Accuracy

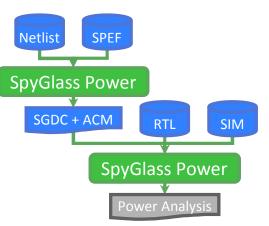
- Want to work with pre-implementation RTL, where can still optimize
- But need to model with post-implementation accuracy
- Requires calibration of multiple estimates against similar production RTL: Vt mixes, clock tree, capacitances, drives, etc

Correlation

- Intuitive set of structural models (not scaling factors!)
 - Advanced Capacitance Model (ACM), clock tree, drive distribution, Vth mix
- Models automatically set from netlist of same design class (same technology node, similar timing characteristics)
- Multi-variate regression analysis

Typical Deviation

vs. reference power @ gate <15%</p>







Managing Plan Misses

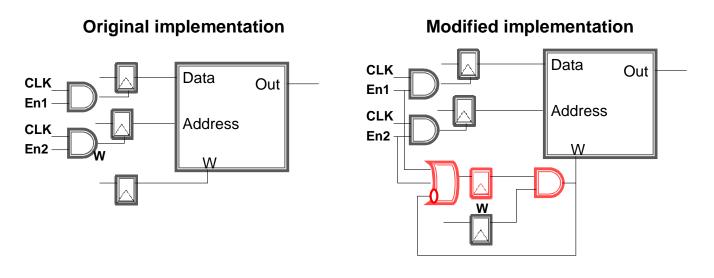
Figuring out where you are at
Techniques to reduce power

Longer-Term: System Power Modeling @RTL

Memory Power Reduction



Redundant write (one example): If the Data and Write address are stable, then every write after the first one is redundant and can be removed



Memory power reduction typically has a more significant impact on power than register optimizations

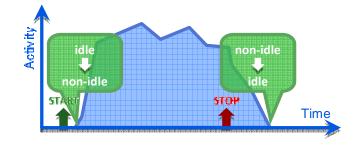
Additional Clock Gating

May be Additional Gating Opportunities Above Register Level

- Especially in legacy IP
- But not always clear when you can and cannot gate
- Empirical analysis a practical starting point, not requiring detailed understanding of IP architecture

Activity Trigger Detection

- Automated analysis of activity, RTL analysis and formal proving of derived triggers
- Use to determine when can gate the clock (in idle)
- Demonstrated to save ~30% on a video processor
- Can also highlight potential power bugs (spikes)



SPYG

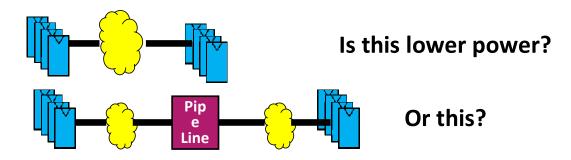


Physically-Dependent Optimization



Really Early-Stage but Maybe Late if you are Desperate..

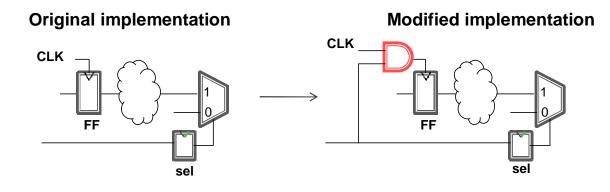
- Normally optimize RTL for power, lose some of gain in timing closure
- To not lose gains, need to manage power/timing tradeoffs
- Below, with one stage for combo logic need big drivers to meet timing
- Split the logic with a pipeline, drivers smaller but pipeline adds power
- Which is better requires physically-aware power analysis



Register-Level Optimization



- Observability Don't-Care: State change blocked downstream, therefore can gate upstream
- In the same vein: Stability condition check, enable strengthening



- Low-level savings unlikely to select more than a few big hitters
- Logic proven by sequential equiv check, but also changes timing
- Can also mess up CDC need to couple closely with CDC analysis

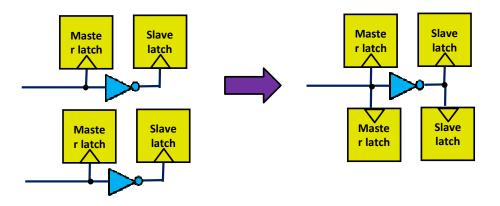
Atrenta Confidential © 2015 Atrenta Inc.

SPY

Fine-Tuning Flops



- Combining single flops into dual, quad, ... flop macros which use a common inverter between master and slave stages
- Saves one inverter for a dual flop, 3 for a quad flop, ...
- Only makes sense for flops which will be physically close
 - On busses possible to implement in RTL dominant power saving
 - Otherwise close by chance opportunistic saving based on placement



Atrenta Confidential © 2015 Atrenta Inc.

SPYGL





Managing Plan Misses

- Figuring out where you are at
- Techniques to reduce power

Longer-Term: System Power Modeling @RTL

System Power Modeling @RTL



The Best Way to Model SW Loads is in Emulation

- Typically very fast, but becomes impossibly slow if dumping activity for power estimation
- Today average power estimation based on software simulation
- But software simulators can't model realistic loads to capture potential peak power problems
- Solution has to be emulation-based, but requiring less dumped nodes for estimation

Power Model Abstraction

- Active standard development in (IEEE) P1801 SLP
- An extension of UPF for abstracted power modeling
- Defines how the models can be represented, but not of course how the models should be created

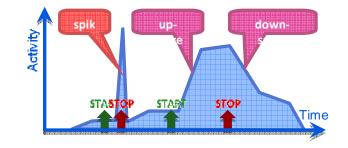
System Power Modeling @RTL

Default Would be to Create the Models Manually

- Reasonable approach for IP vendors
- May be more challenging for internal IP legacy, original developer long gone, many tweaks, ...

An Alternative – Empirical Model Development per IP/Block

- Based again on activity trigger analysis
- Here use to find triggers for and averaged power in major modes of operation
- Still an R&D activity, but looks promising









Thank You!

This presentation may contain forward-looking statements regarding product development. Information or statements contained in this presentation are for informational purposes only and do not represent a commitment, promise, or legal obligation of any kind by Atrenta Inc. or its affiliates.