CPU Hardening and Robust Power Network Design

ARM Physical IP Workshop
Chi-Mo Lee
June, 2014
CPU Hardening Questions from Designers

- **Exploration**
  - How to select the right libraries
  - Which Vts and channel lengths
  - How does my selection affect PPA

- **Hardening**
  - What IP can help my design the most
  - What IP most significantly affects PPA
  - What steps in the flow are most crucial
  - How should I create an optimized power network
  - What should I do to enable power gating
ARM IP Addresses these Questions

- **Processor Optimized Physical IP**
  - Fast Cache Instances + High Performance Kit

- Benefit from implementation knowledge and experience with CPU cores
  - CPU and GPU Hardening is more and more complex to implement

- Customized high-performance POP Reference Flow Methodology (scripts)
  - Two major EDA vendors supported
  - Easily reproduce POP PPA

- Support for multiple CPU configurations
  - Support for various L1 & L2 cache sizes

- **Automation Products to Ease Implementation**
  - Extended SB-OCV for improved accuracy
  - Power Network automation for fast exploration and implementation

- **Best PPA with low risk and short time-to-market**
FOM data: a comparison of normalized performance and leakage characteristics among different standard cell libraries available within ARM IP.

Select optimal VT combination according to FOM data.
Exploration – Benchmarking to Select VtGs and Channel Lengths

Achieved Frequency vs. Target Frequency
(SSg, 0.81V, 0C, cworst)

Leakage Power vs. Target Frequency
(TT, 0.9V, 85C, typical)

Area vs. Target Frequency

Target Frequency

Achieved Frequency

Leakage Power

Area

Target Frequency

LVT_C31
LVT_C35
SVT_c31
SVT_c35

LVT_C31
LVT_C35
SVT_c31
SVT_c35
ARM POP IP DEMO
USING TSMC28HPM CORTEX®-A53
POP IP Implementation Focus

- Synthesis and DFT Guideline
- Floorplan Guideline
- Placement Strategies
- Clock Tree Synthesis
- Routing Guideline
- IR Drop
ARM POP IP – Synthesis and DFT

- Physical aware synthesis uses an optimized floorplan
  - Macro and pin placement
  - Create bounds for logic clustering
  - Magnet placement for critical timing paths

- Defined path groups and path weight based on extensive design flow exploration
  - Many trials to find the best “recipe” to achieve optimal PPA

- Flow includes support for scan chain insertion
ARM POP IP – Floorplan and Placement

- Floorplan guideline
  - Analyze quality of results on many trials
  - The optimal floorplan is selected for POP IP delivery

- Power domain support
  - Provide UPF/CPF

- Includes recommended power grid
  - Specific targets for static and dynamic IR drop

- Power gating cell placement
  - IR drop target across power gating cells approximately 0.5% of Vdd
ARM POP IP – Clock Tree Synthesis

- Choosing the optimal clock-tree components – BUF vs INV, driving strength
- Clock nets NDR guideline – width and spacing
- Guidelines for routing layers, shielding, max. transition
- Recommendations for buffer drive strength, VT selection and channel length
- Suggestions for architectural clock gating placement
- CTS based on MCMM analysis
ARM POP IP – Routing

- Routing congestion is controlled in the floorplan and placement stages
  - No requirement for custom routing flows

- The POP IP flow addresses the following advanced items in routing scripts
  - Routing secondary power pins
  - DFM via insertion
  - Wire spreading
  - Layer aware routing for reducing cross-talk
  - Antenna violation fixing
ARM POP IP – Multi-bit Flip Flops (MBFF)

- MBFF’s are custom cells that cluster several single-bit FFs by sharing transistors
  - Usage is a trade-off between power and performance

- MBFFs may be introduced at the synthesis stage
  - Dependent on design targets – the POP IP user guide delivers complete usage analysis

- Use MBFFs to save power and area
  - ~10% saving at dynamic and leakage power without timing driven mode
  - ~5% saving at dynamic and leakage power with timing driven mode
  - Average ~2% impact on WNS with timing driven mode but can be fixed by doing timing ECO
POP IP Reference Flow

- POP IP Leakage Reduction Technology
- Artisan Physical IP
- Benchmarking Datasheet
- POP IP User Guide

POP IP Reference Flow

- Synopsys/Cadence based reference flow (scripts) to help customers reproduce the POP IP results

File Structure
- Makefile
- Flow/Scripts

1. Synthesis & DFT
2. Placement & preCTS opt.
3. CTS & postCTS opt.
4. postCTS & hold fixing
5. Routing
6. postRoute opt. & Hold fixing
7. LRT/Leakage opt.
8. Sign off
Three Parts of the Power Grid

1. Structures to support **static** IR drop based on the peak power density
   - Limited frequency response due to impedance between the power regulator and the standard cells

2. Structures to support fast transient response to **dynamic** IR
   - Requires a low impedance path to nearby capacitance, either inherent to the design or from decap

3. Power gating switches and enabling the power gating switches
   - Optional

   - Must consider IR drop
IR Drop Sources

- Total budget for static IR drop plus a weighted average dynamic IR drop is either 5% or 10%
  - From power regulator to cells
  - Aligns with the voltages in the worst case PVT corners
- Adjust as needed for each design
Power Grid Basic Physical Structure – Power Gated

Upper 2D
- Handles global supply distribution
- Reduces IR drop, ball to last thin layer
- Improves routing

Last thin layer
- Interface, thick metal via and thin metal totems

Thin totems
- Attach lowest upper grid strap to the top of the lower 2D static IR grid

Lower 2D supply grid (Static IR)
- Distributes current to cell rails
- Improves di/dt
- Reduces IR drop

Lower 2D supply grid (Dynamic IR)
- Reduces resistance and hot spots
- Reduces on-chip inductance
- Reduces VIA resistance
- Avoids large delay degradations
Example Lower 2D Power Grid
POWER GATE PLACEMENT AND SIZING
Power Gating Options

- Headers and footers offered
  - Buffered versions reduce the overhead for sleep signal distribution
  - All taps in these cells reduce the overhead for tying the substrate
- Power gates are two-rows high below 28nm
  - Better device density
  - Less sleep signal overhead
  - Single height provided for hot spots
- Three sizing choices for device width
  - Supports tuning to the power density of the given block
- Available in various VT/CH combinations
  - Faster implant channel versions for hot spots
- ALL implementations assume wells remain ON
Power Gate Placement

- Spacing along a pair of rows is determined by:
  - Maximum distance from any transistor to a substrate tie
  - Size (transistors inside) the power gates
  - VT/CH of the power gating switches
  - Peak power density of the block in question
  - Area allocated to the power gates
- Placement strategies recommended by ARM
  - Two row checkerboard placement (less area)
  - Column placement (generally easier)
- Care must be taken at block edges and near internal blockage that disrupt the power gate placement

![Diagram of power gate placement strategies](Site Rows Checkerboard placement Column placement)
Example of Checker Board PG with Grid
Power Gate Rough Initial Sizing

1. Get the equivalent linear resistance of each power gating switch
   - Included in power model files (Apache, Voltage-Storm, or Liberty)
   - Use slow or global slow process corner
2. Determine the current from each power gate at the budgeted power gate collapse
3. Use peak power density determine the amount of area each power gate can support
4. Adjust the power gate size/placement pitch to match this area per power gate target
5. Consider the impact on power grid collapse for switched network due to fewer static IR straps
   - Static IR straps are adjusted to be located only over the power gate cells for the switched supply
   - Switched rail will demonstrate greater IR drop than the unswitched rail
   - ARM recommends a split strap approach if the power density is high enough
6. In some cases a higher current VT/CH may be required to meet the area impact goals
   - Comes at a strong leakage increase during power down
SLEEP SIGNAL DISTRIBUTION
Three primary objectives:

1. Minimized overhead by using buffers internal to power gates as much as possible
2. Avoid exceeding the maximum inrush current
   - Current flowing through power gates to charge the switched supply
   - Current required to toggle the gate capacitance of the power gates
3. Bring power grid up evenly to avoid crowbar

For power gating without retention
- Longer power up/down time is usually acceptable
- Use tree to trickle chain followed by parallel hammer groups

For power gating with retention
- Power up/down time is critical
- This indicates a desire to keep the inrush current at a maximum without going over for as long as possible
- Use geometrically increasing trickle chains
EXAMPLE OF WHAT NOT TO DO AS A LEARNING AID
Serial Trickle and Serial Hammer Chain

Example

No different than having power gate with part of the device width on one enable chain and the rest on a second enable chain (classic mother/daughter)
Single Serial Trickle and Hammer Chain

- **Advantages**
  - Easy to build with low overhead
  - Single enable signal

- **Disadvantages**
  - EM failure likely on initial switches
  - This is not speculation, ARM is aware of real cases in silicon
  - Steep IR drop gradient results in crowbar current during power-up
  - Very slow
  - Inefficient use of internal buffers

- This is one of the least efficient ways to implement power-on
IR Drop Gradient & Crowbar Current

- Initially powering up the grid from one point, can lead to a voltage gradient across the power domain.
- This can lead to crowbar current during power-up.
- This wastes power.

![Diagram showing IR drop gradient and crowbar current](image-url)
EXAMPLES OF WHAT TO DO
Tree to Trickle Chain with Hammer Groups

Legend

- GPGBUF
- Trickle HEADTIE or FOOTTIE
- Trickle HEADBUFTIE or FOOTBUFTIE
- Hammer HEADTIE or FOOTTIE
- Hammer HEADBUFTIE or FOOTBUFTIE

Continued fan-out
Tree to Trickle Chain with Hammer Groups

- Trickle Cell location is such that the number of rows skipped times the number of cells skipped in a row is equal to the hammer to trickle ratio.
- Alternating rows use *BUFTIE and *TIE
- Trickle cells are always placed in *TIE rows
Tree to Trickle Chain with Hammer Groups

- **Advantages**
  - Inrush current controlled by the trickle to hammer ratio
  - Reduced IR drop gradient because of rapid spatial spread of trickle cells
  - Minimal always-on buffer usage
  - Efficient use of internal buffers
  - Minimal wiring overhead
  - Turn on time is much faster than tradition mother/daughter
    - Trickle chain distribution is fast
    - Hammers turn on fast because of parallel groups and FO=2 within groups

- **Disadvantages**
  - Power-up latency is not minimal
  - Must limit the number of parallel hammer groups to avoid too much gate capacitance switching
Geometrically Increasing Trickle Chains

```
trickle_ena[0]  \arrow{trickle chain group 0}
                   \arrow{trickle_ena[N-1]  \arrow{trickle chain group N-1}}
```

**Legend**
- **GPGBUF**
- **T** Trickle HEADTIE or FOOTTIE
- **BT** Trickle HEADBUFTIE or FOOTBUFTIE
- **Hammer HEADTIE or FOOTTIE**

---

Continued

fan-out
Geometrically Increasing Trickle Chains

- **Advantages**
  - Minimal power-up latency
  - Reduced IR drop gradient because of rapid spatial spread of trickle cells
  - Can minimize the disadvantages at the cost of more latency by having a small number of trickle chains followed by a hammer group

- **Disadvantages**
  - More dependence on external buffers
  - Still small, but one HFN per chain instead of just two for trickle tree to hammer group method
  - Complex analysis to determine progression rate and delay tuning between progressions
Thank You

The trademarks featured in this presentation are registered and/or unregistered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. Any other marks featured may be trademarks of their respective owners.