High Performance, Energy Efficient Implementation of ARM® Processors

Alan Gibbons, Principal Consultant, Synopsys Inc.
September 13th, 2017
The Power Gap for Application Processors

• User experience demands:
  • High levels of device functionality, HD graphics, fast response, long battery life
  • *Can* continue to extend CPU processing power to satisfy the performance requirements
  • *Cannot* continue to extend energy source to match – battery technology is not evolving fast enough

• Solution:
  ➢ Need to become much smarter in how we spend our power/energy budget
  ➢ Must maximize our entitled performance within an energy (or power) budget
  ➢ Energy efficiency is a system problem and needs a system solution - so industry collaboration also essential

Source: ITRS System Driver Chapter 2010 Updates
Collaboration Enables Early Adoption of ARM’s Latest IP

• ARM & Synopsys have been collaborating for over 20 years
  – Rapid, optimized implementation and verification of synthesizable ARM IP and sub-systems
  – Low power methodology and the development of energy efficient ARM based sub-systems
  – High performance verification, emulation and prototyping

• Why collaborate?
  – To deliver a better User experience and better QoR
  – Get early versions of IP, libraries and tools working together from the outset
  – De-risk the implementation for our mutual customers

© 2017 Synopsys, Inc.
Latest Collaboration – ARM DynamIQ™ CPU Sub-System

- Development of optimized implementations
  - ARM Cortex-A75 CPU
  - ARM Cortex-A55 CPU
  - DynamIQ™ Shared Unit (DSU)
- Tuned implementations
  - Optimizing performance, power and area
  - TSMC 16FF+ technology
  - ARM libraries and Synopsys EDA tools
- Available now
  - RIs are available for download from SolvNet
  - New QuickStart Implementation Kits (QIKs)
  - Additional support via Synopsys Design Services
Cortex-A75 Implementation – Major Challenges

**Power**
- Analyze Library for Balanced Power/Performance/TAT
  - OCV
  - Multi-Vt & gate-length
  - Multibit
  - Sequential Flops

**Area**
- Determine an Optimum Floorplan
  - Bounds
  - Placement Controls
  - Data Flow Analysis
  - Macro Placement

**Performance**
- Manage Crosstalk and Optimization for Best Frequency
  - Crosstalk Optimization
  - Concurrent Clock & Data (CCD)
  - Global Route-based Optimization
  - PrimeTime Delaycalc
## Performance and Power Managed Concurrently

### Meet Timing
- Enhanced physical guidance (eSPG)
- Enhanced layer-aware optimization
- Placement pre-clustering
- place_opt CCD
- New global route-based opt.
- CCS receiver cap modeling
- PrimeTime delay calc in route_opt
- Redundant VIA insertion
- Path-based analysis (PBA)
- Clock skew ECO
- Physical-aware ECO

### Reduce Power
- Timing-driven multibit register banking and de-banking
- Physical-aware clock gating
- Low power placement
- Incremental timing-driven multibit register banking and de-banking
- Clock gating optimization
- Low power placement
- High effort leakage flow
- Leakage-aware timing ECO
Cortex-A75 CPU Power Optimization Flow

- Deliver energy efficient performance
  - Not simply “high performance, low power” rather highest entitled performance within a power/energy budget
  - Optimal point(s) on a power v. performance curve

- Key considerations
  - Vt class availability
  - Multibit (MB) banking/de-banking
  - Leakage vs. timing vs. dynamic optimization
  - Leave headroom (both timing and power) for ECO

- Library impacts all these decisions

---

Meet Power Target

- MB banking/de-banking
  - 1bit, 2bit, 4bit
- VT selection
  - Across 12 vt/channel options
- QL (leakage) vs. Q (std) vs. QA (area)
- flop selection
- SI TNS Reduction, very congested, clock NDRs
- fix_eco_power to meet leakage target, expect 15-20% reduction
In DC, datapath delay is prioritized. Faster cells are used.

Pessimism reduces through the flow and CTS brings in useful skew, cells are swapped for power.

ECO brings in 4 new SVT classes and does positive slack recovery.

### Vt Class/Channel Mix Changes As Implementation Progresses

<table>
<thead>
<tr>
<th>ULVT</th>
<th>LVT</th>
<th>SVT</th>
</tr>
</thead>
<tbody>
<tr>
<td>c16</td>
<td>c16</td>
<td>c16</td>
</tr>
<tr>
<td>c18</td>
<td>c18</td>
<td>c18</td>
</tr>
<tr>
<td>c20</td>
<td>c20</td>
<td>c20</td>
</tr>
<tr>
<td>c24</td>
<td>c24</td>
<td>c24</td>
</tr>
</tbody>
</table>
Floorplanning

- Floorplanning is and always has been, a key element in ARM CPU implementation
- Module and macro placement critical for hitting aggressive QoR targets
- Design topology, power switch mesh and power supply network also key considerations
- Macro placement, bounds and blockages impact both timing and power
QoR Challenges in Placement

• Design topology
  – Analysis of Critical Module Timing on Cortex-A75 CPU

• Critical paths to & from CORE sub-module
  – CORE connects heavily to DSIDE and DENGINE
  – Critical paths seen throughout the flow (challenging to fix downstream)

• CORE being “pushed” out of center of core area and near IOs

• Created FMAX-limited paths due to long-path buffering across block
Cortex-A75 CPU Floorplan Changes

Move RAMs To Guide Module Placement

Floorplan changes that allowed CORE to float to the center and close to DSIDE
Crosstalk: An Ongoing Challenge on ARM Cores

- Lower geometry processes always have a crosstalk component
- ARM CPUs have traditional SI prevention
  - Clock NDRs
  - Congestion-aware placement
  - Logic and density controls
- The Cortex-A73 flow used NDRs to dramatically reduce crosstalk
- We have used all these techniques plus more on the Cortex-A75 CPU
Early Results - Cortex-A75 CPU

• When starting the Cortex-A75 CPU flow, used best practices from Cortex-A73 RI
  – Clock NDRs
  – Crosstalk threshold noise ratio of 20%
  – Congestion optimization for placer settings
  – 80ps max_transition limit

• Early results: crosstalk still an issue
  – Large TNS increase at route_auto stage
  – Leakage increase at route_opt stage

More attention needed to address crosstalk
Crosstalk Mitigation – Cortex-A75 CPU

New optimization solutions with better control of placement in both core area and macro channels resulted in dramatic FMAX & power improvements.
Concurrent Clock and Data Optimization - CCD

**place_opt with data only: higher area & power**

-100ps 200ps

**place_opt w/ useful skew: lower area & power**

50ps 50ps Delay 150ps

---

**Cortex-A75 CPU**

<table>
<thead>
<tr>
<th></th>
<th>WNS</th>
<th>TNS</th>
<th>Leakage Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>place_opt + route_opt</td>
<td>-23</td>
<td>-95</td>
<td>100%</td>
</tr>
<tr>
<td>place_opt CCD + route_opt</td>
<td>-20</td>
<td>-68</td>
<td>98%</td>
</tr>
</tbody>
</table>

---

**Apply useful skew**

Size up clock buf

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CLK</td>
<td>90ps</td>
<td>10ps</td>
</tr>
</tbody>
</table>

**Datapath area/power recovery**

Size down/swap LVT

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CLK</td>
<td>90ps</td>
<td>10ps</td>
</tr>
</tbody>
</table>

---

**Cortex-A75 CPU**

<table>
<thead>
<tr>
<th></th>
<th>WNS</th>
<th>TNS</th>
<th>Leakage Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>route_opt (Baseline)</td>
<td>-112</td>
<td>-134</td>
<td>100%</td>
</tr>
<tr>
<td>Power CCD + route_opt</td>
<td>-99</td>
<td>-126</td>
<td>99%</td>
</tr>
</tbody>
</table>
Results on Cortex-A75 CPU

Cortex-A75 CPU PPA

- TNS (ns)
- FMAX (%)
- Leakage (%)

Implementation Stages:
- DCG
- place_opt
- clock_opt
- route_opt
- Signoff
- ECO

DC Graphical + IC Compiler II + PrimeTime ECO
Summary

- Manage power and performance concurrently – power is not an after thought
- Analyze library for balanced power, performance and turnaround time
- Determine an optimum floorplan – module and memory placement critical
- Manage crosstalk as well as concurrent optimization of clock and data
Latest RIs Available Now

Synopsys Reference Implementations (RIs) for Cortex-A75/-A55 are ready

- CPU and DSU flows
- TSMC 16nm FFC process
- ARM POP™ IP – core optimized standard cells & fast cache RAMs
- Complete implementation and static verification flows

Contact your Synopsys AC for additional information
RIs available on SolvNet (solvnet.synopsys.com/ARM-RI)