Power Constraints:
From Sensors to Servers

Mike Muller
CTO

The Architecture for the Digital World®
1983 The Rise of the PC and ARM Technology

- Compass
- DynaTAC
- Paper
- Lotus 1-2-3
- Osborne
- Vinyl
ARM Design Philosophy

ARM Processors are:-
- Small in size
- Low cost
- Suitable as macrocells
- Efficient
  - low power consumption
  - low heat generation
1:45 – 3:15 **Session 3: Low Cost Processors**
Session Chair: John Mashey, Silicon Graphics
- Highly Integrated SPARC Processor Implementation
  S. Joshi, Sun Microsystems
- The LR33020 GraphX Processor: A Single Chip MIPS-RISC Based X Terminal Controller
  S. Dossi, LSI Logic Corporation
- The ARM600 Processor and FPA
  M. Muller, Advanced RISC Machines Ltd.

3:15 – 3:45 **Break**

3:45 – 5:15 **Session 4: Low Power Systems**
Session Chair: Dave Ditzel, Sun Microsystems
- A VLSI Chip Set for Personal Communications System
  R. Scavuzzo, AT&T Bell Labs
- SPARC90 - Chipset on a Chip
  J. Pendleton, Sun Microsystems
- Cold Chip Design Techniques
  R. Broderson, A. Chandrakasan and S. Sheng, UC Berkeley
1993 The Tipping Point

- Mosaic v1.0
- DVD
- Game Boy
- Zarlink Orion GPS
- 1st ARM GPS
- 1st ARM PDA
- 1st Consumer GSM
- 1st ThinkPad
2003 Unplugged

2004
2013 Mobile Computing
ARM7TDMI

GSM Base-Band
0.5u.

Wireless Enabled Product
0.05u

2000
### Dark Silicon Source – ITRS 2008

<table>
<thead>
<tr>
<th>Node</th>
<th>45nm</th>
<th>22nm</th>
<th>11nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year</td>
<td>2008</td>
<td>2014</td>
<td>2020</td>
</tr>
<tr>
<td>Area</td>
<td>1</td>
<td>x4</td>
<td>x16</td>
</tr>
<tr>
<td>Peak freq</td>
<td>1</td>
<td>1.6</td>
<td>2.4</td>
</tr>
<tr>
<td>Power @ 45nm freq</td>
<td>1</td>
<td>0.6</td>
<td>0.3</td>
</tr>
<tr>
<td>Power @ peak freq</td>
<td>1</td>
<td>1</td>
<td>0.6</td>
</tr>
</tbody>
</table>

**Exploitable Si (equivalent power):**

- 24% for 45nm
- 10% for 11nm

**Max Mobile SoC Power**

- **Sustained Tablet**
- **Sustained Phone**

Source: ITRS 2008 & ARM
Dark Silicon Source – ITRS 2008

<table>
<thead>
<tr>
<th>Node</th>
<th>45nm</th>
<th>22nm</th>
<th>11nm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Year</strong></td>
<td>2008</td>
<td>2014</td>
<td>2020</td>
</tr>
<tr>
<td><strong>Area</strong></td>
<td>1</td>
<td>x4</td>
<td>x16</td>
</tr>
<tr>
<td><strong>Peak freq</strong></td>
<td>1</td>
<td>1.6</td>
<td>2.4</td>
</tr>
<tr>
<td><strong>Power</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>@ 45nm freq</td>
<td>1</td>
<td>0.6</td>
<td>0.3</td>
</tr>
<tr>
<td>@ peak freq</td>
<td>1</td>
<td>1</td>
<td>0.6</td>
</tr>
<tr>
<td>Exploitable Si (equivalent power)</td>
<td></td>
<td>24%</td>
<td>10%</td>
</tr>
</tbody>
</table>

- **28nm**
  - 2014: 1
  - 2015: 1.7
  - 2018: 3.5
- **16nm**
  - 2015: 57%
  - 2018: 28%
- **10nm**
  - 2018: 40%

Source: ITRS 2008 & ARM
It’s Still All About the Power
This Year's Favorite Product

Charger

Cartridge

Fans

Valve

Single Plate

PEM

H^+

cathode

2x H_2O

anode

O_2

Fuel Cell Stack

2 x H_2

Load

Intelligent Energy™
Heterogeneous Computing

Bottom Left

1998
Manual Partitioning
C & Assembler

ARM + DSP

Top Right

ARM + GPU

2013
Manual Partitioning
C++ & OpenCL/RenderScript
Heterogeneous Computing Performance Drivers

**Mobile Image Processing**
- Expression detection
- Scene component aggregation
- Blemish reduction
- Should not interfere with GUI
- 2020: 100GOp/s @ 1W

**Advanced Driver Assistance**
- Collision avoidance
- Driver Assistance
- Enhanced Perception
- Driver monitoring
- 2020: 40GOp/s @ 1W, ASIL-D

**High Performance Computing**
- Simulating atoms
- Predicting atmosphere
- Scalability
- Programmability
- 2020: One Exaflop @ 20MW
What’s Good About GPU style uArchitectures for Efficiency

- Relaxed single-threaded performance
  - No dynamic scheduling
  - No branch prediction
  - No register renaming, no result forwarding
  - Longer pipelines
  - Lower clock frequencies

- Multi-threading to obtain performance
  - Tolerate long latencies to memory

- Increasing the ALU/control ratio
  - Short-vectors exposed to programmers
  - SIMT based execution
But hard to optimize using GPU programming languages
Heterogeneous Compute

Homogeneous Architecture

- C++ and OpenMP programming in the same language as the CPU
- Same binary, process, ABI, instruction set, mature familiar tools and ecosystem
- Not necessary to restructure the program for asynchronous execution
  - Very low latency offload
- Not necessary to restructure program to mark up memory
Performance vs Effort

- We’ve implemented SGEMM, a matrix-matrix multiplication benchmark, in various ways, to investigate the tradeoff between programmer effort and performance payoff
  - 1 core 500 MHz SIMT CPU vs 1 GHz Cortex-A15-like CPU

<table>
<thead>
<tr>
<th>SGEMM version</th>
<th>Speedup</th>
<th>Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARM in C</td>
<td>1x</td>
<td>Low</td>
</tr>
<tr>
<td>ARM in C with NEON intrinsics, prefetching</td>
<td>15x</td>
<td>Medium-High</td>
</tr>
<tr>
<td>ARM in assembly with NEON, prefetching</td>
<td>26x</td>
<td>High</td>
</tr>
<tr>
<td>SIMT ARM in C</td>
<td>35x</td>
<td>Low</td>
</tr>
<tr>
<td>SIMT ARM in C, unrolled</td>
<td>44x</td>
<td>Low-Medium</td>
</tr>
<tr>
<td>SIMT ARM in C, unrolled, blocked, 16 entries per work item</td>
<td>125x</td>
<td>Medium</td>
</tr>
<tr>
<td>Mali GPU x 4 way</td>
<td>136x</td>
<td>High</td>
</tr>
</tbody>
</table>
Extending big.LITTLE MP for Thermal Management

Linux Kernel Scheduler

Performance Monitor

Power Model

Intelligent Power Allocation (IPA)

Voltage

Frequency

Performance Allocation

GPU

big

LITTLE

big

LITTLE

MHz

MHz

MHz

Control Temperature

IPA

Traditional Thermal Control

Time (s)

Temperature
It’s Still All About Heterogeneous Systems
Aging/Wearout in Mobile Systems?

- Aging normal associated with high performance systems

- Degradation due to Bias Temperature Instability (BTI) aging is relevant for DVFS systems
  - High voltage and performance mode causes $V_t$ increase over lifetime
  - Timing is critically affected in low voltage mode

- Workload based aging analysis flow
  - Mid-size ARM CPU: ~100K instances, ~10K flip-flops
  - 28nm library, $V_{dd} = 0.9\, V$, Temp = 105 $^\circ$C, Lifetime = 3 years
  - Workloads: mp3, web-browse, 3D render, Dhrystone, Video H264

- Timing degradation from ~2.3% (mp3) to ~9% (Video H264)
Is it Safe?

- Memory no longer dominant FIT contributor
  - ECC with physical interleaving reduces the FIT rate to ~0
  - Physical interleaving: multi cell error → single bit error
  - Temporal scrubbing to correct single bit errors

- Unprotected logic now dominates SoC FIT rate
  - No “cheap” way to protect - spatially distributed bits
Is it Safe?

SPECInt 2000

94%

6%

5%

0.7%

0.3%

SoC SER FIT rate per node

Unprotected Memory SER

Protected Memory SER

Logic SER

200nm 150nm 100nm 50nm 0
Is it Safe?

- Protecting every flip-flop incurs significant area, performance and power penalty
- Identify flip-flops which dominate design FIT rate
  - Fault injection provides each flip-flop’s vulnerability
- Critical flip-flops are replaced by SER optimized flip-flops
- Design is iterated in ECO mode to achieve targets
It’s All About the Memory Efficiency

- Engaging the Ecosystem
  - Wendy Elsasser @ ARM chairing Future Memory JEDEC Task Group
  - Continued participation in HMCC

- 2018 post DDR4/LPDDR4
  - Focus on useable bandwidth
  - Abstracted packet interface

- SoC targets
  - Mobile: >60GB/s ~1.5W
  - Server: >150GB/s ~10W
  - Networking: >300GB/s ~15W
It's Still All About the Memory
and Bottom Left
Scalable Area, Performance and Battery Life

<table>
<thead>
<tr>
<th>Mobile Implementation</th>
<th>High-end Wearable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cortex-A7 MP2 1.2 GHz, 2.25 mm²</td>
<td>Cortex-A7 MP2 500 MHz, 1.1 mm²</td>
</tr>
</tbody>
</table>

Area calculations based on 28HPM node, 9T
Scalable Area, Performance and Battery Life

Mobile Implementation
Cortex-A7 MP2
1.2GHz, 2.25 mm²

High-end Wearable
Cortex-A7 MP2
500MHz, 1.1 mm²

Optimized Wearable
Cortex-A7 UP
500MHz, 0.36 mm²

Optimized 40LP Wearable
Cortex-A5 UP
500MHz, 0.49 mm²

6x reduction in size
9x reduction in leakage
>1800x leakage reduction
7x reduction in power
Scalable Area, Performance and Battery Life

Optimized Wearable
Cortex-A7 UP
500MHz, 0.36 mm²

Optimized IOT
Cortex-M0
40MHz, 0.05mm²

Area based on 28HPM node, 9T, unless otherwise indicated
Applications
Cortex-A5 System
250MHz

Comms
Cortex-M0 System
40MHz

Sensor Hub
Cortex-M0+ System
1MHz – 20kHz
Bottom Left Efficiency 1.2V to 0.2V AM

**12 Power Domains**

<table>
<thead>
<tr>
<th>Test Logic</th>
<th>2kB Boot ROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug Interface</td>
<td>16kB 6T SRAM 1.2V</td>
</tr>
<tr>
<td>Serial Interface</td>
<td>4kB 10T SRAM LV</td>
</tr>
<tr>
<td>Host Interface</td>
<td>4kB 10T SRAM LV</td>
</tr>
<tr>
<td>RTC</td>
<td>Bus Fabric</td>
</tr>
<tr>
<td>128bit AES H/W</td>
<td>Cortex-M0+</td>
</tr>
</tbody>
</table>

**Sensor Hub**

Cortex-M0+ System

1MHz – 20kHz

65nm

1.2V to 0.2V
On Chip Regulators
Direct Battery operation
SRAM contribution to system standby/retention power is >80%
- Level Shifters must be optimized for range & reduced variability
- Power gate voltage drop significantly impacts frequency and leakage
- EDA tools assume RC dominates but at low voltage gate-delay dominates
Minimum Power Point
- 200mV @ 20 kHz
- 1/6000 \times \text{power}
- 1/3000 \times \text{frequency}
- Energy Scavenging

Minimum Energy Point
- 400mV @ 1MHz
- 1/1000 \times \text{energy}
- 1/100 \times \text{frequency}

Circuits can be made to work but the system level leakage needs great care
- 200mV Min Voltage – useful for power limited energy harvesters
- 400mV Min Energy – best for energy limited batteries
- 600mV Near Threshold – half the energy savings but much lower complexity & risk
Great, but what can you do with it?

- Thermo-electric 5K gradient @ 300uW
- 5 year CR2032 @ 8uW
- Indoor 4mm² PV cell @ 2uW

Power (uW) vs. kHz

Sensor @100kHz
MCU @1MHz
BLE @8MHz
It’s still all about the power

It’s still all about heterogeneous solutions

It’s still all about the memory

In the future the cloud drives device requirements