PUMA
by AMD

Gowthami J. Manikandan
Hashim Sharif
Markus H. Fischer

~ https://upload.wikimedia.org/wikipedia/commons/thumb/4/4a/Puma_face.jpg/1280px-Puma_face.jpg
History and Naming

- **AMD Bobcat** Family 14h (Q1 2011)
  - “difficult to compete in x86 market with a single core optimized for 10-100 W range”

- **AMD Jaguar** Family 16h (Q1 2011)
  - Next gen of Bobcat: More cores, FUs, memory, x86 instruction set extensions
  - Inside Xbox One and Playstation 4 customized with more powerful GPUs
  - Rushed to market

- **AMD Puma** Family 16h (Q2 2014)
  - Refresh of Jaguar
History and Naming

- **AMD Puma Family 16h (Q2 2014)**
  - 19% CPU core leakage reduction at 1.2V
  - 38% GPU leakage reduction
  - 500 mW reduction in memory controller power
  - 200 mW reduction in display interface power
  - Chassis temperature aware turbo boost
  - Selective boosting according to application needs (intelligent boost)
  - Support for ARM TrustZone via integrated Cortex-A5 processor
### 2014 LOW-POWER “MULLINS” APUs

<table>
<thead>
<tr>
<th>Model</th>
<th>Radeon™ Brand</th>
<th>SDP</th>
<th>TDP</th>
<th>CPU Cores</th>
<th>CPU Clock Speed (Max)</th>
<th>L2 Cache</th>
<th>Radeon™ Cores</th>
<th>GPU Clock Speed (Max)</th>
<th>DDR3 Speed (Max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A10 Micro-6700T</td>
<td>R6</td>
<td>2.8W</td>
<td>4.5W</td>
<td>4</td>
<td>2.2 GHz</td>
<td>2MB L2</td>
<td>128</td>
<td>500 MHz</td>
<td>DDR3L-1333</td>
</tr>
<tr>
<td>A4 Micro-6400T</td>
<td>R3</td>
<td>2.8W</td>
<td>4.5W</td>
<td>4</td>
<td>1.6 GHz</td>
<td>2MB L2</td>
<td>128</td>
<td>350 MHz</td>
<td>DDR3L-1333</td>
</tr>
</tbody>
</table>

### 2014 MAINSTREAM “BEEMA” APUs

<table>
<thead>
<tr>
<th>Model</th>
<th>Radeon™ Brand</th>
<th>TDP</th>
<th>CPU Cores</th>
<th>CPU Clock Speed (Max)</th>
<th>L2 Cache</th>
<th>Radeon™ Cores</th>
<th>GPU Clock Speed (Max)</th>
<th>DDR3 Speed (Max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A6-6310</td>
<td>R4</td>
<td>15W</td>
<td>4</td>
<td>2.4 GHz</td>
<td>2MB L2</td>
<td>128</td>
<td>800 MHz</td>
<td>DDR3L-1866</td>
</tr>
<tr>
<td>A4-6210</td>
<td>R3</td>
<td>15W</td>
<td>4</td>
<td>1.8 GHz</td>
<td>2MB L2</td>
<td>128</td>
<td>600 MHz</td>
<td>DDR3L-1600</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Radeon™ Brand</th>
<th>TDP</th>
<th>CPU Cores</th>
<th>CPU Clock Speed (Max)</th>
<th>L2 Cache</th>
<th>Radeon™ Cores</th>
<th>GPU Clock Speed (Max)</th>
<th>DDR3 Speed (Max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2-6110</td>
<td>R2</td>
<td>15W</td>
<td>4</td>
<td>1.5 GHz</td>
<td>2MB L2</td>
<td>128</td>
<td>500 MHz</td>
<td>DDR3L-1600</td>
</tr>
<tr>
<td>E1-6010</td>
<td>R2</td>
<td>10W</td>
<td>2</td>
<td>1.35 GHz</td>
<td>1MB L2</td>
<td>128</td>
<td>350 MHz</td>
<td>DDR3L-1333</td>
</tr>
</tbody>
</table>
Features we heard about...
## Functional overview

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int Rename</td>
<td>Integer Register Rename Unit</td>
</tr>
<tr>
<td>FP PRF</td>
<td>Floating-point Physical Register File</td>
</tr>
<tr>
<td>Int PRF</td>
<td>Integer Physical Register File</td>
</tr>
<tr>
<td>LAGU</td>
<td>Load Address Generation Unit</td>
</tr>
<tr>
<td>SAGU</td>
<td>Store Address Generation Unit</td>
</tr>
<tr>
<td>BU</td>
<td>Bus Unit</td>
</tr>
<tr>
<td>VALU</td>
<td>Vector ALU</td>
</tr>
<tr>
<td>VIMUL</td>
<td>Vector Integer Multiply Unit</td>
</tr>
<tr>
<td>St Conv.</td>
<td>Store / Convert Unit</td>
</tr>
</tbody>
</table>
Instruction Fetch & Decode

- Implements AMD64 (x64 / x86-64)
- Fetches instructions as 32B blocks
- Decode unit scans 2x 16B windows per cycle
- Dispatcher can send up to 2 macro-ops to retire unit and corresponding micro-ops to schedulers
## Macro- and Micro-Ops

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Macro-ops</th>
<th>Micro-ops</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>REP MOVS [mem],[mem]</td>
<td>Many</td>
<td>Many</td>
<td>Microcode</td>
</tr>
<tr>
<td>ADD reg,reg</td>
<td>1</td>
<td>1: add</td>
<td>Fastpath single</td>
</tr>
<tr>
<td>ADD reg,[mem]</td>
<td>1</td>
<td>2: load, add</td>
<td>Fastpath single</td>
</tr>
<tr>
<td>MOVAPD [mem],xmm</td>
<td>1</td>
<td>2: store, FP-store-data</td>
<td>Fastpath single</td>
</tr>
<tr>
<td>VMOVAPD [mem],ymm</td>
<td>2</td>
<td>4: 2 × {store, FP-store-data}</td>
<td>256b AVX Fastpath double</td>
</tr>
<tr>
<td>ADDPD xmm,xmm</td>
<td>1</td>
<td>1: addpd</td>
<td>Fastpath single</td>
</tr>
<tr>
<td>ADDPD xmm,[mem]</td>
<td>1</td>
<td>2: load, addpd</td>
<td>Fastpath single</td>
</tr>
<tr>
<td>VADDPD ymm,ymm</td>
<td>2</td>
<td>2 × {addpd}</td>
<td>256b AVX Fastpath double</td>
</tr>
</tbody>
</table>
Branching

- **Next-Address Logic**
  - Find next block to load

- **Branch Target Buffer**
  - Where did branch previously go?
  - Mispredict penalty $\geq 14$ cycles
  - 2 levels with each 1024 entries

- **Branch Target Address Calculator**
  - Check and correct BTB predicted address

- **Return Address Stack (RAS)**
  - As call is fetched, address of following instruction is pushed

- **Indirect Target Predictor**
  - 512 entries
  - If different targets, use 26 bit global history

- **Advanced Conditional Branch Direction Predictor**
  - Non-take $\rightarrow$ Always taken $\rightarrow$ apply prediction
  - Uses same 26 bit global history
Integer Execution

- Int Scheduler receives macro-op (20 entries)
- Issue 1 micro-op per pipe (2 pipes)
- Executes 00
- Tracks dependency information
- Mul
  - up to 32x32 bit
  - Fully pipelined, 3 cycles latency
  - 64x64 bit → Data pumping
    - 6 cycles latency
    - throughput = 1 op every 4 cycles
- Div
  - 2 bits of result per cycle
**Integer Execution**

- AGU Scheduler receives macro-op (12 entries)
- Issue 1 micro-op per pipe (2 pipes)
- Executes 00
- Tracks dependency information

- Three-operand LEA mapped to SAGU
  - 2 cycles latency
  - Result inserted back into ALU1
Integer Execution

- Retire Control Unit
  - Tracks outstanding operations
  - Can receive 2 macro-ops per cycle
  - Tracks up to 64 macro-ops in-flight
- Macro-op finish == all its micro-ops finish
- Commits in order
- RCU handles register renaming
- Physical Register file
  - 64 registers
  - 33 to 44 registers available for renaming
Floating Point Execution

- Native support for operands
  - 32/64 bit Single/Double Precision
  - 80 bit Extended Precision
  - 256 bit packed Single Double
- 128 bit load/store paths

- Receive up to 2 macro-ops/cycle
- Issue 1 micro-op per pipe (2 pipes)
- Schedule Queue: 18 micro-op entries
- RCU can track 44 micro-ops in-flight

- Int ops with FP component are held in FP RCU until Int macro-op retires in Int RCU
# Floating Point Execution Times

<table>
<thead>
<tr>
<th>Instruction Class</th>
<th>Latency</th>
<th>Throughput</th>
<th>Execution Pipe</th>
<th>Unit(s)</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIMD ALU (most)</td>
<td>1</td>
<td>2 / cycle</td>
<td>Either</td>
<td>VALU0, VALU1</td>
<td>Integer</td>
</tr>
<tr>
<td>Floating-point logical</td>
<td>1</td>
<td>2 / cycle</td>
<td>Either</td>
<td>FPA, FPM</td>
<td>Floating-pt</td>
</tr>
<tr>
<td>SIMD IMUL</td>
<td>2</td>
<td>1 / cycle</td>
<td>Pipe 0</td>
<td>VIM</td>
<td>Integer</td>
</tr>
<tr>
<td>Floating-point multiply single-precision</td>
<td>2</td>
<td>1 / cycle</td>
<td>Pipe 1</td>
<td>FPM</td>
<td>Floating-pt</td>
</tr>
<tr>
<td>Floating-point add</td>
<td>3</td>
<td>1 / cycle</td>
<td>Pipe 0</td>
<td>FPA</td>
<td>Floating-pt</td>
</tr>
<tr>
<td>Store/Convert (many)</td>
<td>3</td>
<td>1 / cycle</td>
<td>Pipe 1</td>
<td>Store/Convert</td>
<td>STC</td>
</tr>
<tr>
<td>Floating-point multiply double-precision</td>
<td>4</td>
<td>1 / 2 cycles</td>
<td>Pipe 1</td>
<td>FPM</td>
<td>Floating-pt</td>
</tr>
<tr>
<td>Floating-point multiply extended-precision (x87)</td>
<td>5</td>
<td>1 / 3 cycles</td>
<td>Pipe 1</td>
<td>FPM</td>
<td>Floating-pt</td>
</tr>
<tr>
<td>Floating-point DIV/SQRT</td>
<td>Iterative</td>
<td>Iterative</td>
<td>Pipe 1</td>
<td>FPM</td>
<td>Floating-pt</td>
</tr>
</tbody>
</table>
Features we DID NOT hear about ...
SKIN TEMPERATURE AWARE POWER MANAGEMENT

WITHOUT STAPM

- Tablet power is limited by steady-state maximum skin temperature i.e. TSP – defined as the power that if consumed indefinitely, will cause skin temperature to hit the user sensitivity limit.

- Die temperature at TSP will be below physical limits (typically around 100°C), leaving several minutes where we could run much faster than TSP allows untapped performance potential for a period of time before the skin temperature gets too hot.

  - This is STAPM

---

1. Chart is a model representative of a nominal “TSP App” (3DMark05) running on an 11.6” Discovery tablet with a 4 core Mullins. TDP 4.5W, TSP 3.5W

2. Actual duration of boost opportunity is a function of platform design and component choices and will vary.
SKIN TEMPERATURE AWARE POWER MANAGEMENT WITH STAPM

- Boost aggressively until Tskin reaches user defined maximum
- Reduce power only when necessary to adhere to Tskin, max limit
- Most use-cases for mobile devices are short in duration → Result is higher performance most of the time
- All without using more power

DYNAMIC SKIN-TEMPERATURE AWARE POWER MANAGEMENT CAN ENABLE UP TO 63% PERFORMANCE INCREASES¹ ON KEY WORKLOADS

¹ Based on 3DMark11-P and PCMark8 V2 Home on 3.5W TSP Mullins with and without STAPM enabled. Pre-production engineering sample of “Mullins” quad-core APU with next generation AMD Radeon graphics (model number TBD), 2x2GB DDR3-1333MHz RAM, Windows 8.1, and unreleased reference driver.
AVOIDING POWER WASTE WITH INTELLIGENT BOOST CONTROL

- Intelligent Boost is designed to avoid power waste that results from boosting applications that benefit very little from higher frequency.
- Enables long battery life and cool operation while maintaining great performance.
- Power management micro-controller tracks application behavior real-time to determine frequency sensitivity.
- Boost behavior is adjusted accordingly.
AMD SECURITY TECHNOLOGY
ENGINEERED TO WORK TOGETHER

COMPREHENSIVE, HARDWARE-BASED SECURITY

- Isolation for security and privacy sensitive applications and data that are important to everyone
- Trusted Execution Environment (TEE) provides standard interfaces and APIs for application portability
- AMD PSP with ARM TrustZone complements existing system security capabilities, and is designed to provide greater assurance and acceleration where and when it’s needed

- Secure technology components
- Standard processing path
- Security aware applications, secure services, or trusted applications
- Secure processing path
Memory Hierarchy
Memory Ordering

- Features two pipelines - one for loads and one for stores
- Memory pipeline out of order
- Loads can pass earlier loads and earlier unaliased stores
- Load and store µops allocate entries
  - AGU scheduler
  - Memory ordering Queue
  - Store also allocate an entry in the Store Queue
Memory Ordering

- The AGU scheduler holds 12 entries in Jaguar and 8 for Bobcat
- Virtual address calculation done in Load and Store AGU
- The AGUs in PUMA are optimized for simpler addressing
- Once virtual address has been calculated, entry written to MOQ
- Data access
  - Access the L1 data cache
  - Check store queue for ordering constraints
  - Check store queue buffer for store forwarding
L1 Data Cache

- **L1 D Cache characteristics**
  - 32 KB, 8 way associative, 64 B cache lines
  - Load to use latency 3 cycles
  - Writeback policy
  - Parity protected, LRU replacement algorithm
  - 128 bit read/write each cycle

- **Physical and virtual address only differ in the tags**

- **Data cache access takes two clock cycles**
  - Virtual address translation in parallel with tag read
  - Second cycle reads out data from the cache line
Data TLB

- **L1 DTLB characteristics**
  - Fully associative
  - 40 entries for 4KB pages, 8 entries for 2MB pages

- **L2 DTLB characteristics**
  - 512 entry, 4-way associative for 4KB pages
  - 256 entry, 2-way associative for 2MB pages

- **L2 TLB misses are speculatively resolved**
  - Hardware page table walker
  - 16-entry directory cache for intermediate translations
L2 Cache

- **L2 DCache characteristics**
  - Unified, shared, write back
  - 1-2MB and 16-way associative
  - ECC protection for tags and data
  - Inclusive of all higher 8 level caches
  - Tracks 4 outstanding requests to bus interface

- **PUMA vs Bobcat**
  - 25 cycle hit latency as opposed to 17 cycles
  - Shared cache for all four cores
  - Higher hit rate than Bobcat
L2 Cache

- **Prefetching**
  - Prefetcher for each core
  - Prefetcher can track up to 8 streams

- **Coherency**
  - MOESI protocol
  - 16 entry snoop queue for coherency requests

- **Data Banks**
  - Tags are split into 4 banks with hashed addresses
  - Each tag array is associated with a data array
  - Each 512 KB data array runs at half core frequency to save power
  - 16B per core clock cycle on each data array - aggregate bandwidth of 64B/cycle
• Radeon 3200 Integrated Graphics Processor (IGP)

• Discrete GPU: Radeon 3450 (for mobile platforms) and Radeon 3800 series (for notebooks and tablets)
  • Radeon 3800: Industry’s first 55 nm GPU!
  • ATI Powerplay technology: Enhanced energy efficiency (Performance-per-watt ratio)

• AMD’s **Hybrid CrossFireX** technology allows the IGP to team up with the on-board GPU and achieve upto 70 % improvement in performance (over a single GPU)
GPU: Radeon 3800 series

Major stages in Graphics processing:
- Shader Processing: Vertex and pixel shading
- Texture Mapping
- Rendering

GPU features:
- Frequency: 700 MHz
- No. of streaming processors: 320 (Shaders)
- No. of TMUs and Raster units: 16 + 16
- L2 Texture cache, color cache, shader cache
- Memory bus width: 256-bits

The IGP: Radeon 3200

• Core Architecture: RS780

• Second Unified Shader architecture: RS600
  • Performs both pixel shading (color computations) and vertex shading (controlling movement, lighting etc. in 3D models)
  • Unified architecture allows switching between pixel and vertex shading depending on the amount of work at each stage
  • Clock frequency ~500MHz

• No. of pipelined shader processing units: 40
The IGP: Radeon 3200

- Texture Mapping Units (TMU): 4
  - They rotate, resize and perform other texture operations on the shaded vectors.

- Raster Operating Units (ROP): 4
  - Responsible for converting the graphics vectors into a ‘rasterized’ image (pixels) that can be displayed on the screen.

- Pixel Rate: Maximum amount of pixels that can be rendered/written into local memory in one second: 2GPixel/sec
Unified Video Decoder (UVD)

AMD’s dedicated video decoding ASIC for hardware video decoding (off-loading decoding work from the CPU at the expense of some area)
ATI Hybrid CrossFireX Technology

• CrossFireX: allows combination of multiple (upto 4) discrete GPUs to achieve enhanced graphics performance.

• Hybrid CrossFire X: allows combination of on-board IGP with the discrete GPU to enhance power.

There are several load balancing techniques that distribute the workload between the IGP and GPU(s).
Hybrid CrossFireX (Contd.)

Super-tiling
- Optimum load-balancing
- Does not work with OpenGL
- Each card needs to compute the entire geometry of the scene

Scissor Mode
- Support dynamic load-balancing
- Works with both OpenGL and Direct3D
- Each card needs to compute the entire geometry of the scene

Alternate frame rendering
- Computing load is divided-up optimally
- Enables full geometry acceleration
ATI PowerXpress

<table>
<thead>
<tr>
<th>Mode</th>
<th>Plugged to AC power?</th>
<th>On-board IGP</th>
<th>Discrete GPU module(s)</th>
<th>Optimization</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC Mode</td>
<td>Yes</td>
<td>On</td>
<td>On</td>
<td>&quot;Maximum performance&quot;</td>
</tr>
<tr>
<td>DC Mode</td>
<td>No</td>
<td>On</td>
<td>Off</td>
<td>&quot;Maximum battery life&quot;</td>
</tr>
</tbody>
</table>

• PowerXpress technology enables dynamic switching between standalone IGP mode and the Hybrid mode depending on whether the device is plugged-in.
• When the device is connected to power supply, both, IGP and discrete GPU modules are enabled for enhanced load-balancing and better 3D rendering capabilities
• When disconnected from the power supply, the system operates in the standalone IGP mode to increase battery life.
Competition

**PCMark 8 V2 – Home**
- A6-6310
- Pentium 3556U (Haswell)
- A4-6210
- Pentium N3510 (Bay Trail M)

**3DMark 11 Performance**
- A6-6310
- Pentium 3556U (Haswell)
- A4-6210
- Pentium N3510 (Bay Trail M)

**BaseMark CL**
- A6-6310
- Pentium 3556U (Haswell)
- A4-6210
- Pentium N3510 (Bay Trail M)
Sources

● http://www.realworldtech.com/jaguar/6/
● https://www.bottomupcs.com/hardware_support_for_virtual_memory.xhtml
History and Naming

- **AMD Bobcat Family 14h (Q1 2011)**
  - “difficult to compete in x86 market with a single core optimized for 10-100 W range”
  - Fusion: Bobcat core(s) used together with GPU cores in accelerated processing units (APUs)
  - Targeted at low-power products: Netbooks, Tablets, embedded market
  - TDP of 18 W or less
  - Tera Scale 2 GPU
  - 64 bit core, up to 2 cores per chip
  - Out of order + speculative execution
  - 2x ID, 2x pipelined INT ALU, 2x pipelined FP ALU, SIMD (MMX and SSE up to SSE4A)
History and Naming

- **AMD Jaguar** Family 16h (Q1 2011)
  - Inside Xbox One and Playstation 4 customized with more powerful GPUs
  - Up to 4 cores
  - Shared L2 cache, more addressable RAM
  - Extra integer division unit
  - Double (128 bit) FPU path
  - Improved power saving
  - Additional instruction sets: SSE4.1 and 4.2, AES
  - TDP between 25W and 3.9W
History and Naming

- **AMD Puma Family 16h (Q2 2014)**
  - Revision of AMD Jaguar
  - 19% CPU core leakage reduction at 1.2V
  - 38% GPU leakage reduction
  - 500 mW reduction in memory controller power
  - 200 mW reduction in display interface power
  - Chassis temperature aware turbo boost
  - Selective boosting according to application needs (intelligent boost)
  - Support for ARM TrustZone via integrated Cortex-A5 processor
Execution Unit

- 28 nm
- 64 bit registers
- Up to 4 cores per chip
- Out of order and speculative execution
- Register file size: 64 entries (presumably 64 int and 64 float)
- L1 Branch Target Buffer: 1024 entries
- Branch misprediction penalty = 15-16 cycles
Execution Unit

- Return Address Stack: 16 entries
- 2x ALU, 2x FPU, LDU, STU with 128 bit (256 bit vectors are split)
- Double (128 bit) FPU path
- Extra integer division unit
- ALU scheduler: 20-entry
- Address Generation Unit (AGU) scheduler: 12 entries
Execution Unit

- Instruction sets:
  - **MMX**, SSE, SSE2, SSE3, SSSE3, SSE4a, SSE4.1, **SSE4.2**, AVX, F16C, CLMUL, AES, BMI1, MOVBE, XSAVE/XSAVEOPT, ABM (POPCNT/LZCNT), and AMD-V

- Physical addressing width: 40 bit

- Power and Clock Gating
### AMD Virtualization

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Virtualization Extensions to the x86 Instruction Set</strong></td>
<td>Enables software to more efficiently create VMs so that multiple operating systems and their applications can run simultaneously on the same computer</td>
</tr>
<tr>
<td><strong>Tagged TLB</strong></td>
<td>Hardware features that facilitate efficient switching between VMs for better application responsiveness</td>
</tr>
<tr>
<td><strong>Rapid Virtualization Indexing (RVI)</strong></td>
<td>Helps accelerate the performance of many virtualized applications by enabling hardware-based VM memory management</td>
</tr>
<tr>
<td><strong>AMD-V Extended Migration</strong></td>
<td>Helps virtualization software with live migrations of VMs between all available AMD Opteron processor generations</td>
</tr>
<tr>
<td><strong>I/O Virtualization</strong></td>
<td>Enables direct device access by a VM, bypassing the hypervisor for improved application performance and improved isolation of VMs for increased integrity and security</td>
</tr>
</tbody>
</table>