1TOPS/W
Software Programmable Media Processor

David Moloney, CTO, Movidius
19 August 2011
Movidius Background

• Started in 2005 looking at mobile gaming acceleration
  – Decided on multicore design to allow software derivatives and meet OPS/W/$ target
  – Existing processors poor cost/performance match for target workloads
  – Developed SHAVE vector processor with HW support for sparse data-structures (Matrix-Vector)
  – Expanded ISA to support C-complier
• Talked to mobile phone customers in 2007
  – Turned out their real problem was video
  – Back to the drawing-board!
• Initial 65nm Silicon & all IP on founder & angel funding
  – Allowed us to close A-round in October 2008!
• 65nm Myriad MM SoC in mass-production
  – Next generation 28nm SoC H1/2012 with 10x Perf/W
Mobile Video Processing Workload
Movidius SHAVE Processor

- **Streaming Hybrid Architecture Vector Engine**
  - Hybrid of RISC, DSP, VLIW & GPU architectural features
  - 128-bit vector arithmetic: 8/16/32-bit INT & fp16/fp32
- Unique proprietary architecture
  - Tailored to streaming workloads and architected for outstanding OPS/mW/$ performance
- Excellent Graphics and matrix mathematics support
  - HW texture unit for good graphics performance
  - Predicated execution to eliminate branches
  - Compiler-friendly architecture
  - HW support for compressed data-structures (ex. matrices)
SHAVE Instruction-Set

• RISC-style
  – Instruction predication
  – Extensive integer ISA
  – Excellent C-compiler support

• DSP-style
  – Zero overhead looping
  – Modulo addressing
  – Transparent DMA modes
  – FFT, Viterbi, and other DSP operation support
  – Parallel comparisons

• VLIW-style
  – Parallel functional units controlled by VLIW instr.
  – 8/16/32-bit x 1-4 SIMD INT

• GPU-style
  – Streaming operations
  – Floating-point operations (fp16/fp32 IEEE-compliant)
  – Texture-Management Unit and L1 Cache
SHAVE ISA Richness

# instructions

- IBM Cell
- Altivec
- TI C64xx
- Movidius SHAKE
- TI C55xx
- ADI BlackFin

![Movidius Logo]
Myriad Silicon Platform

- SW Controlled I/O Multiplexing
- MEBI
- SEBI
- SDIO x3
- SPI x3
- FLSH
- LCD x2
- Cam x2
- USB2 OTG
- I²C x2
- I²S x2
- JTAG
- UART x2
- TS
- GPS
- TIM
- RISC
- NAL
- SW Controlled I/O Multiplexing
- CMX 128kB
- L1
- TMU
- SVE0
- CMX 128kB
- L1
- TMU
- SVE0
- CMX 128kB
- L1
- TMU
- SVE0
- CMX 128kB
- L1
- TMU
- SVE0
- CMX 128kB
- L1
- TMU
- SVE0
- CMX 128kB
- L1
- TMU
- SVE0
- Stacked 16/64MB SDRAM die
- 32
- 64
- 128
- 50GFLOPS/W (IEEE 754 SP)

Movidius IP
Myriad 65nm CMOS LP Die

<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>FLOPS/core</th>
<th>Cores</th>
<th>GFLOPS</th>
<th>W</th>
<th>GFLOPS/W</th>
</tr>
</thead>
<tbody>
<tr>
<td>Myriad</td>
<td>2011</td>
<td>12</td>
<td>8</td>
<td>17.28</td>
<td>0.35</td>
<td>49.4</td>
</tr>
<tr>
<td>(1) KAIST</td>
<td>2011</td>
<td>12</td>
<td>8</td>
<td>5.8</td>
<td>0.28</td>
<td>21.1</td>
</tr>
<tr>
<td>(2) Intel</td>
<td>2007</td>
<td>80</td>
<td>80</td>
<td>1000</td>
<td>98.00</td>
<td>10.2</td>
</tr>
<tr>
<td>(4) Adapteva</td>
<td>2010</td>
<td>2</td>
<td>16</td>
<td>24.96</td>
<td>1.00</td>
<td>25.0</td>
</tr>
</tbody>
</table>
Technology - Platform Approach
# Myriad Example Applications

<table>
<thead>
<tr>
<th>SHAVE 1</th>
<th>SHAVE 2</th>
<th>SHAVE 3</th>
<th>SHAVE 4</th>
<th>SHAVE 5</th>
<th>SHAVE 6</th>
<th>SHAVE 7</th>
<th>SHAVE 8</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AAC</strong></td>
<td>SHAVE 2</td>
<td>SHAVE 3</td>
<td>SHAVE 4</td>
<td>SHAVE 5</td>
<td>SHAVE 6</td>
<td>SHAVE 7</td>
<td>SHAVE 8</td>
</tr>
<tr>
<td><strong>1080i H264 Decode</strong></td>
<td>SHAVE 4</td>
<td>SHAVE 5</td>
<td>SHAVE 6</td>
<td>SHAVE 7</td>
<td>SHAVE 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Cam Corder</strong></td>
<td><strong>H264 Decode</strong></td>
<td><strong>Video Edit Effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>3D H264 Decode</strong></td>
<td><strong>3D Stereo Pipeline</strong></td>
<td><strong>Resize</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>3D Rect. &amp; Auto-Convergence</strong></td>
<td><strong>3D Stereo Video Pipeline</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>2D-3D</strong></td>
<td><strong>Anaglyph Conversion</strong></td>
<td><strong>Resize</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

20/Apr/2011
MoviSim ISS Architecture

Runtime
- Task Allocator
- Memory Allocator

OpenCL

SABRE Debugger

MoviSim
- Messaging
- Instrumentation
- Thread 0
- Thread 1
- Thread n-1
- Model API
- Thread m-1
- Thread m
- XML Parser

SHAVE ISS
- SHAVE ISS
- SHAVE ISS
- LEON/ARM ISS
- Heterogeneous Core ISS
- DRAM
- Simulation Engine
- XML Architecture Description
Fragerak 28nm Platform

450GFLOPS/W (IEEE 754 SP)
Any questions?

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n°248481 (PEPPHER Project, www.peppher.eu)
Abstract

• The rationale and architecture behind a new software programmable multimedia coprocessor for mobile devices is outlined.
• The focus of the architecture is on power-efficient operation, allowing functions which are traditionally implemented in fixed-function hardware to be implemented competitively in software.
• For instance the sustained single-precision IEEE 754 rate is 50GFLOPS/W allowing existing applications to be ported with great ease. The device supports 8, 16, 32 and some 64-bit integer operations as well as fp16 (OpenEXR) and fp32 arithmetic and is capable of an aggregate 1 TOPS/W maximum 8-bit equivalent operations in a low-cost plastic BGA package with integrated 16 or 64MB SDRAM.
• New architectural features such as support for random-accessible sparse data-structures are implemented for the first time improving memory utilization and bandwidth efficiency. Power efficiency is paramount and the device contains a total of 11 power-islands with 8 dedicated to each of the integrated SHAVE processors, allowing very fine-grained power control.
• Comparisons to previous work based on 65nm silicon and applications are shown to illustrate the power of the device.
Myriad GOPS/Watt
(Total/Arithmetic)

Myriad
GOPS/W

GOPS/W (total) 1004
GOPS/W (arith) 181

16 60 40 20 40
11 18 21 22 22
7 8 8 8 8
80 40 40 20 20
32 48 16 22 16
24 48 24 21 21
8 48 16 45 45
1 2 8 1 1
16 16 16 16 16

int8 int16 int32 fp16 fp32

OP/W tot
RISC
DMA

PEU BRU LSU0 LSU1 VAU IAU SAU CMU TMU DMA

Movidius
BW Hierarchy

Myriad 65nm

- 547GB/Sec
- 190:1
- 2.88GB/Sec
- 2:1
- 1.44GB/Sec

Frigrak 28nm

- 4864GB/Sec
- 42:1
- 115GB/Sec
- 18:1
- 6.4GB/Sec

Bottom-Line - Very High Sustainable Performance
## BW Hierarchy (Detail)

### Myriad 65nm

<table>
<thead>
<tr>
<th>VRF</th>
<th>SRF</th>
<th>IRF</th>
<th>LSU</th>
<th>IDC</th>
<th>L1</th>
<th>ISB</th>
<th>L2</th>
<th>SDRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>Bytes</td>
<td>16</td>
<td>4</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>8</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>Ports</td>
<td>12</td>
<td>12</td>
<td>17</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>BW</td>
<td>34.56</td>
<td>8.64</td>
<td>12.24</td>
<td>2.88</td>
<td>2.88</td>
<td>1.44</td>
<td>5.76</td>
<td>2.88</td>
</tr>
<tr>
<td>#SHAVES</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Total BW</td>
<td>276.48</td>
<td>69.12</td>
<td>97.92</td>
<td>23.04</td>
<td>23.04</td>
<td>11.52</td>
<td>46.08</td>
<td></td>
</tr>
<tr>
<td></td>
<td>547.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Fragnak 28nm

<table>
<thead>
<tr>
<th>VRF</th>
<th>SRF</th>
<th>IRF</th>
<th>LSU</th>
<th>IDC</th>
<th>L1</th>
<th>ICB</th>
<th>L2</th>
<th>XCB</th>
<th>SDRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>Bytes</td>
<td>16</td>
<td>4</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>8</td>
<td>16</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Ports</td>
<td>12</td>
<td>12</td>
<td>17</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>BW</td>
<td>153.6</td>
<td>38.4</td>
<td>54.4</td>
<td>12.8</td>
<td>12.8</td>
<td>6.4</td>
<td>25.6</td>
<td>12.8</td>
<td>6.4</td>
</tr>
<tr>
<td>#SHAVES</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>6.4</td>
<td></td>
</tr>
<tr>
<td>Total BW</td>
<td>2457.6</td>
<td>614.4</td>
<td>870.4</td>
<td>204.8</td>
<td>204.8</td>
<td>102.4</td>
<td>409.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4864</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
LSU HW Sparse-Data Support

IRF

bitmap

field

0 | 0 | 0 | 0 | 0
---|---|---|---|---
32 | 1 | 0 | 0 | 4

1 | 0 | 0 | 0 | 4
---|---|---|---|---
16 | 1 | 0 | 0 | 8

2 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 1 | 0 | 12

3 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 0 | 0 | 16

4 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 0 | 1 | 20

5 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 1 | 0 | 24

6 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 1 | 0 | 28

7 | 1 | 0 | 0 | 4
---|---|---|---|---
32 | 1 | 1 | 0 | 32

addr_gen

fen[2:0]

word_cnt

addr[1:0]

IRF[base_addr]

RAM_addr[31:0]

IRF

bm7 bm6 bm5 bm4 bm3 bm2 bm1 bm0

addr_gen

instr_[7:0]

pw_config

RAM_wr RAM_rd bru_hold

LSU

HW Sparse-Data Support
**Sparse Data-Structure Example**

<table>
<thead>
<tr>
<th>bitmap</th>
<th>description</th>
<th>data</th>
<th>64-bit RAM word</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>sx</td>
<td>sx</td>
<td>base+0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0.0</td>
<td>sy</td>
<td>base+2</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0.0</td>
<td>x₀</td>
<td>base+4</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.0</td>
<td>x₁</td>
<td>base+6</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0.0</td>
<td>x₂</td>
<td>base+8</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>sy</td>
<td>y₀</td>
<td>base+10</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0.0</td>
<td>y₁</td>
<td>base+12</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0.0</td>
<td>y₂</td>
<td>base+14</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0.0</td>
<td>y₃</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0.0</td>
<td>z₀</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1.0</td>
<td>z₁</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0.0</td>
<td>z₂</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0.0</td>
<td>z₃</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0.0</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0.0</td>
<td>bmp</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td></td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>x₀</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>x₁</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>x₂</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>x₃</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>y₀</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>y₁</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>y₂</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>y₃</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>z₀</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>z₁</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>1</td>
<td>z₂</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td>z₃</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td>addr</td>
<td>addr</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td>next str. addr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
<td>next str. Bitmap</td>
<td>bmp</td>
</tr>
<tr>
<td>31</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
References


