DSP and Multimedia Processors

Jian-Jia Chen
(Slides are based on Peter Marwedel)
Informatik 12
TU Dortmund
Germany

These slides use Microsoft clip arts. Microsoft copyright restrictions apply.
Saturating arithmetic

- Returns largest/smallest number in case of over/underflows
- Example:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
<td></td>
<td>0111</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>+</td>
<td>1001</td>
<td></td>
</tr>
</tbody>
</table>

Standard wrap around arithmetic: (1)0000
Saturating arithmetic: 1111

(a+b)/2: correct 1000
Wrap around arithmetic: 0000
Saturating arithmetic + shifted: 0111 “almost correct"

- Appropriate for DSP/multimedia applications:
  - No timeliness of results if interrupts are generated for overflows
  - Precise values less important
  - Wrap around arithmetic would be worse.
Fixed-point arithmetic

Shifting required after multiplications and divisions in order to maintain binary point.
Multimedia-Instructions, Short vector extensions, Streaming extensions, SIMD instructions

- Multimedia instructions exploit that many registers, adders etc are quite wide (32/64 bit), whereas most multimedia data types are narrow
- 2-8 values can be stored per register and added. E.g.:

\[ \begin{align*}
&a1 \quad a2 \\
&b1 \quad b2 \\
&\text{plus} \\
&c1 \quad c2
\end{align*} \]

- 2 additions per instruction; no carry at bit 16

- Cheap way of using parallelism
- SSE instruction set extensions, SIMD instructions
Energy Efficiency of FPGAs

© Hugo De Man, IMEC, Philips, 2007
Reconfigurable Logic

Custom HW may be too expensive, SW too slow. Combine the speed of HW with the flexibility of SW

- HW with programmable functions and interconnect.
- Use of configurable hardware; common form: field programmable gate arrays (FPGAs)

Applications:
- algorithms like de/encryption,
- pattern matching in bioinformatics,
- high speed event filtering (high energy physics),
- high speed special purpose hardware.

Very popular devices from
- XILINX, Actel, Altera and others
Fine-Grained Reconfigurable Fabric

CLB: Configurable Logic Block
PSM: Programmable Switch Matrix
Additionally: I/O Blocks, RAM Blocks, Multiplier, CPUs, …

src: Xilinx Data Sheets
Fine-Grained Reconfigurable Fabric

**Logic Layer:** Perform Computations

**Configuration Layer:** Which Computations to Perform

**Partial Reconfiguration:** Only parts of the Fabric are reconfigured

---

src: Kalenteridis et al. "A complete platform and toolset for system implementation on fine-grained reconfigurable hardware", Microprocessors and Microsystems 2004
Coarse-Grained Reconfigurable Fabric

Pros:
- Reduced Area
- Higher Frequency
- Reduced reconfiguration latency

Cons
- Reduced Flexibility
- Area Wastage
  → In case computations do not suite the fabric (e.g. bit-level operations, bit shuffling)

src: PACT’s XPP 64-A1 architecture
Reconfigurable Computing: an example

Runtime reconfigurable processor:
+ Efficient use of hardware: through high degree of parallelism, multiplexed used of reconf. fabric => energy efficient
  – Reconfiguration Latency and Energy
Memory

Jian-Jia Chen
(Slides are based on Peter Marwedel)
Informatik 12
TU Dortmund
Germany

2015年 11 月 24 日

These slides use Microsoft clip arts. Microsoft copyright restrictions apply.
Memory

Memories?

For the memory, efficiency is again a concern:

- capacity
- energy efficiency
- speed (latency and throughput); predictable timing
- size
- cost
- other attributes (volatile vs. persistent, etc)
Memory capacities expected to keep increasing

Figure ORTC2  ITRS Product Function Size Trends:
MPU Logic Gate Size (4-transistor); Memory Cell Size [SRAM (6-transistor); Flash (SLC and MLC), and DRAM (transistor + capacitor)] -- Updated

[ITRS Update 2008]
Where is the power consumed?
- Stationary systems -


- Switching (dynamic) power, logic dominating
- Overall power consumption a nightmare for environmentalists
Where is the power consumed?
- Consumer portable systems -


- Based on current trends

- Memory and logic, static and dynamic relevant
Energy consumption and access times of memories

Example CACTI: Scratchpad (SRAM) vs. DRAM (DDR2):

16 bit read; size in bytes;
65 nm for SRAM, 80 nm for DRAM

Source: Olivera Jovanovic,
TU Dortmund, 2011
Access times and energy consumption for multi-ported register files

Cycle Time (ns)  Area ($\lambda^2 \times 10^6$)  Power (W)

Source and © H. Valero, 2001
Trends for the Speeds

Speed gap between processor and main DRAM increases

- Speed gap between processor and main DRAM increases over time.
- CPU performance increases at a rate of 1.52 p.a.
- DRAM performance increases at a rate of 1.07 p.a.
- Memory access times >> processor cycle times
- "Memory wall" problem

Similar problems also for embedded systems & MPSoCs

[P. Machanik: Approaches to Addressing the Memory Wall, TR Nov. 2002, U. Brisbane]
However, clock speed increases have come to a halt

The speed gap does not disappear – may be, it is not getting larger any more.
Memory Hierarchy

- Register (SRAM)
- SRAM (Cache or Scratchpad)
- Main Memory (DRAM)
- File Cache (DRAM, Flash)
- Secondary Storage (Disks, Flash)
- Teritary Storage (Optische Speicher, Bänder)

persistent
Recover Your Knowledge on Cache

- Fully-associative mapping, direct mapping, set-associative mapping
- Cache replacement policy
- Temporal locality and spatial locality
Set-associative cache $n$-way cache

$|\text{Set}| = 2$

Address

Tag | Index
---|---

way 0 | $\$(\text{€})$ | way 1

Tags | data block | Tags | data block

$\geq 1$
Hierarchical memories using scratch pad memories (SPM)

SPM is a small, physically separate memory mapped into the address space.

Address space

<table>
<thead>
<tr>
<th>0</th>
<th>scratch pad memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>FFF..</td>
<td>no tag memory</td>
</tr>
</tbody>
</table>

Examples:
- Most ARM cores allow tightly coupled memories
- IBM Cell
- Infineon TriCore
- Many multi-cores, due to high costs of coherent caches

Selection is by an appropriate address decoder (simple!)
Influence of the associativity

[P. Marwedel et al., ASPDAC, 2004]
Summary

- Processing
  - Multimedia/DSP processors
- FPGAs (skipped)
- Memories
  - “Small is beautiful”
    (in terms of energy consumption, access times, size)
Spare Slides
Cycles/access as a function of the size of the list

Figure 3.10: Sequential Read Access, NPAD=0

* prefetching succeeds

§ prefetching fails
Impact of TLB misses and larger caches

Elements on different pages; run time increase when exceeding the size of the TLB

Larger caches are shifting the steps to the right

Figure 3.14: Advantage of Larger L2/L3 Caches
Comparison of currents using measurements

E.g.: ATMEL board with ARM7TDMI and ext. SRAM

![Image of ATMEL board]

<table>
<thead>
<tr>
<th>Current Source</th>
<th>Current (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prog Main/ Data Main</td>
<td>116</td>
</tr>
<tr>
<td>Prog Main/ Data SPM</td>
<td>77.2</td>
</tr>
<tr>
<td>Prog SPM/ Data Main</td>
<td>44.4</td>
</tr>
<tr>
<td>Prog SPM/ Data SPM</td>
<td>1.16</td>
</tr>
</tbody>
</table>

Current
32 Bit-Load Instruction (Thumb)

- Core+SPM (mA)
- Main Memory Current (mA)
Why not just use a cache?

2. Energy for parallel access of sets, in comparators, muxes.

[Graph showing energy per access in nJ for different memory sizes and cache configurations.]

[R. Banakar, S. Steinke, B.-S. Lee, 2001]