Era of Customization and Implications to EDA

Jason Cong
Director, Center for Domain-Specific Computing
UCLA Computer Science Department
cong@cs.ucla.edu
www.cdsc.ucla.edu
Outline

- Why customization
- Opportunities for customization
- Opportunities for EDA
The Power Barrier and Current Solution

- 10’s to 100’s cores in a processor
- 1000’s to 10,000’s servers in a data center

Source: Shekhar Borkar, Intel
Cost and Energy are Still a Big Issue …

Hiding in Plain Sight, Google Seeks More Power

Google is building two computing centers, top and left, each the size of a football field, in The Dalles, Ore.

By JOHN MARKOFF and SAUL HANSELL
Published: June 14, 2006

THE DALLES, Ore., June 8 — On the banks of the windswept Columbia River, Google is working on a secret weapon in its quest to dominate the next generation of Internet computing. But it is hard to keep a secret when it is a computing center as big as two football fields, with twin cooling plants protruding four stories into the sky.
Next Significant Opportunity -- Customization

Adapt the architecture to Application domain

Parallelization

Customization

Rocket Nozzle
Sun's Surface
## Justification 1 – Potential of Customization

<table>
<thead>
<tr>
<th>AES 128bit key 128bit data</th>
<th>Throughput</th>
<th>Power</th>
<th>Figure of Merit (Gb/s/W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.18µm CMOS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.84 Gbits/sec</td>
<td>350 mW</td>
<td></td>
<td>11 (1/1)</td>
</tr>
<tr>
<td>FPGA [1]</td>
<td>1.32 Gbit/sec</td>
<td>490 mW</td>
<td>2.7 (1/4)</td>
</tr>
<tr>
<td>ASM StrongARM [2]</td>
<td>31 Mbit/sec</td>
<td>240 mW</td>
<td>0.13 (1/85)</td>
</tr>
<tr>
<td>Asm Pentium III [3]</td>
<td>648 Mbits/sec</td>
<td>41.4 W</td>
<td>0.015 (1/800)</td>
</tr>
<tr>
<td>C Emb. Sparc [4]</td>
<td>133 Kbits/sec</td>
<td>120 mW</td>
<td>0.0011 (1/10,000)</td>
</tr>
<tr>
<td>Java [5] Emb. Sparc</td>
<td>450 bits/sec</td>
<td>120 mW</td>
<td>0.0000037 (1/3,000,000)</td>
</tr>
</tbody>
</table>

- [1] Amphion CS5230 on Virtex2 + Xilinx Virtex2 Power Estimator
- [4] gcc, 1 mW/MHz @ 120 Mhz Sparc – assumes 0.25 u CMOS
- [5] Java on KVM (Sun J2ME, non-JIT) on 1 mW/MHz @ 120 MHz Sparc – assumes 0.25 u CMOS

Justification 2 -- Advance of Civilization

- For human brain, Moore’s Law scaling has long stopped
  - The number of neurons and their firing speed did not change significantly

- Remarkable advancement of civilization via specialization

- More advanced societies have higher degree of specialization
Key questions: Optimal trade-off between efficiency & customizability
Which options to fix at CHP creation? Which to be set by CHP mapper?
Goal

- A general, customizable platform for the given domain(s)
  - Can be customized to a wide-range of applications in the domain
  - Can be massively produced with cost efficiency
  - Can be programmed efficiently with novel compilation and runtime systems

- Metric of success
  - A “supercomputer-in-a-box” with 100X performance/power improvement via customization for the intended domain(s)
Outline

- Why customization
- Opportunities for customization
- Opportunities for EDA
Example: Core spilling – [Cong et al Trans. on Parallel and Distributed Systems 2007]

- CMP systems focus on improving overall throughput
  - Sequential or legacy applications might not see benefits
- Key idea – allow execution to be spilt from one core to next at run-time
  - Simulate increase in register file, instruction queue, ROB and LSQ size
  - Allocate cores intelligently to spilling core
Core spilling – [Cong et al Trans. on Parallel and Distributed Systems 2007]

- Results
  - Core spilling achieves more than 50% of the performance of ‘ideal’ 32-issue core by using 4-issue cores for single applications
  - 39% improvement for multiple application workload
  - Up to 40% reduction in latency for changing workloads
**Example: Customizable hybrid L1 cache**

- **Cache in conjunction with Scratchpad Memory (SPM) in L1**
  - **Cache: Hardware-controlled**
    - Transparent to software: a fast local copy of the global memory address space
  - **SPM: Software-controlled**
    - Not transparent to software: a separate address space from the global address space

- **Customizable**
  - Flexibly size the cache and SPM based on the application requirements
    - Cache: dynamic/random access
    - SPM: regular data access pattern

<table>
<thead>
<tr>
<th></th>
<th>Cache</th>
<th>SPM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>access time</strong></td>
<td>hit: 1 cycle</td>
<td>miss: L cycles</td>
</tr>
<tr>
<td><strong>energy</strong></td>
<td>4.57 nJ</td>
<td>1.53 nJ</td>
</tr>
</tbody>
</table>
Adaptive Hybrid Cache [ISLPED’2011]

- **Way-wise reconfigurable cache**
  - Configure several ways of cache as SPM
  - Column cache [Chiou et.al. DAC’00]

- **Hybrid cache (Block-wise reconfigurable cache)**
  - Virtual local store [Cook et.al. UCB TR’09]
  - Unified mapping of SPM blocks onto cache blocks

- **Adaptive hybrid cache (AH-Cache)**
  - Dynamically remap SPM blocks from high-demand cache sets to low-demand cache sets.

Rician noise

Cache sets

Run time

Cache sets

Run time

Cache sets

Run time
Example: Customizable Multiband RF-Interconnect

- In TX, each mixer up-converts individual baseband streams into specific frequency band (or channel)
- N different data streams (N=6 in exemplary figure above) may transmit simultaneously on the shared transmission medium to achieve higher aggregate data rates
- In RX, individual signals are down-converted by mixer, and recovered after low-pass filter
Terahertz VCO in 65nm CMOS

Demonstrated an ultra high frequency and low power oscillator structure in CMOS by adding a negative resistance parallel tank, with the fundamental frequency at 217GHz and 16.8 mW DC power consumption.

The measured 4th and 6th harmonics are about 870GHz and 1.3THz, respectively.

higher harmonics (4th and 6th harmonics) may be substantially underestimated due to excessive water and oxygen absorption and setup losses at these frequencies.

“Generating Terahertz Signals in 65nm CMOS with Negative-Resistance Resonator Boosting and Selective Harmonic Suppression” Symposium on VLSI Technology and Circuits, June 2010
Mesh Overlaid with RF-I [HPCA ’08]

- 10x10 mesh of pipelined routers
  - NoC runs at 2GHz
  - XY routing
- 64 4GHz 3-wide processor cores
  - Labeled aqua
  - 8KB L1 Data Cache
  - 8KB L1 Instruction Cache
- 32 L2 Cache Banks
  - Labeled pink
  - 256KB each
  - Organized as shared NUCA cache
- 4 Main Memory Interfaces
  - Labeled green
- RF-I transmission line bundle
  - Black thick line spanning mesh
RF-I Logical Organization

• Logically:
  - RF-I behaves as set of N express channels
  - Each channel assigned to src, dest router pair \((s,d)\)

• Reconfigured by:
  - remapping shortcuts to match needs of different applications
Example: Accelerator-Rich Architecture (AXR-CMP) [SAW’2011]

- **Architecture of AXR-CMP:**
  - Multiple cores and accelerators
  - Global Accelerator Manager (GAM)
  - Shared L2 cache banks and NoC routers between multiple accelerators
**Accelerator Chaining and Composition**

- **Chaining**
  - To have an efficient accelerator to accelerator communication

- **Composition**
  - To create the virtual feeling of having larger accelerators for the applications
Experimental Results – Performance

- **Performance improvement over OS based approaches:** on average 4.1 X, up to 10X

- **Performance improvement over SW only approaches:** on average 97X, up to 208X
Outline

◆ Why customization

◆ Opportunities for customization

◆ Opportunities for EDA
  
  ▪ EDA can make use of customized computing for acceleration
  
  ▪ Customized computing creates many interesting problems/opportunities for EDA
Emulation at NVIDIA

In 1995, CEO Jensen Huang “spent $1 million, a third of the company’s cash, on a technology known as emulation, which allows engineers to play with virtual copies of their graphics chips before they put them into silicon. That allowed Nvidia to speed a new graphics chip to market every six to nine months, a pace the company has sustained ever since.”
- from Forbes, 1/7/08

One of the largest emulation labs in the world

Mike Butts - RAMP - August, 2010
Another Successful Example - Brion Technologies

- Computational lithography with FPGA acceleration
- Each server node is equipped by 2 CPUs and 4 FPGAs [Cao’04]
  - Looking at performance/accuracy trade-offs of FFTs

Source: System and method for lithography simulation, US patent 7117477 (by Brion Technologies)
ASML to buy Brion for $270 million

Mark LaPedus
12/18/2006 11:36 AM EST

ASML to buy Brion for $270 million
SAN JOSE, Calif. — In a major expansion into the design-for-manufacturing (DFM) arena, ASML Holding NV on Tuesday (Dec. 19) announced plans to acquire Brion Technologies Inc., a provider of semiconductor design and wafer production technology, for $270 million in cash.

The move propels lithography-tool giant ASML (Veldhoven, Netherlands) into a new DFM market. Founded in 2002, Brion (Santa Clara, Calif.) is a player in the growing field of computational lithography, which encompasses design verification, reticle enhancement technologies and optical proximity correction.

Brion's computational lithography technology enables semiconductor manufacturers to simulate the realized pattern of integrated circuits and to correct the mask pattern to optimize the manufacturing process and yield, the company said.

Brion has various arrangements with third parties, including a deal with ASML's competitor, Nikon Corp. It is unclear if Brion will maintain its DFM arrangement with Nikon.
Acceleration of Lithographic Simulation [FPGA’08]

- Lithography simulation
  - Simulate the optical imaging process
  - Computational intensive; very slow for full-chip simulation

\[
I(x, y) = \sum \lambda_\kappa \times \\
\left| \sum \tau \left[ \psi_\kappa(x-x_1, y-y_1) - \\
\psi_\kappa(x-x_2, y-y_1) + \psi_\kappa(x-x_2, y-y_2) - \psi_\kappa(x-x_1, y-y_2) \right] \right|^2
\]

- 15X+ Performance Improvement vs. AMD Opteron 2.2GHz Processor
- Close to 100X improvement on energy efficiency
  - 15W in FPGA comparing with 86W in Opteron

- XtremeData X1000 development system (AMD Opteron + Altera StratixII EP2S180)

AutoPilot™ Synthesis Tool

Algorithm in C
Outline

- Why customization
- Opportunities for customization
- Opportunities for EDA
  - EDA can make use of customized computing for acceleration
  - Customized computing creates many interesting problems/opportunities for EDA
Example: High-Level Synthesis from C/C++ to Customized Circuits [SOCC’2006]

- Advanced transformation/optimizations
  - Loop unrolling/shifting/pipelining
  - Strength reduction / Tree height reduction
  - Bitwidth analysis
  - Memory analysis ...

- Core behavior synthesis optimizations
  - Scheduling
  - Resource binding, e.g., functional unit binding register/port binding

- \( \mu \)-Arch-generation & RTL/constraints generation
  - Verilog/VHDL/SystemC
  - FPGAs: Altera, Xilinx
  - ASICs: Magma, Synopsys, …
AutoESL/AutoPilot Compilation Tool (based on xPilot)

- Platform-based C to FPGA synthesis
- Synthesize pure ANSI-C and C++, GCC-compatible compilation flow
- Full support of IEEE-754 floating point data types & operations
- Efficiently handle bit-accurate fixed-point arithmetic
- More than 10X design productivity gain
- High quality-of-results

Developed by AutoESL, acquired by Xilinx in Jan. 2011
Example: Simulation for Customizable Heterogeneous Platforms (CHPs)
Example: Compiler Support for Customizable Hybrid Cache [DAC’2011]

```c
int amplitude[N]; // Global variable
int state[N];     // Global variable
...
for (i = 0; i < N; ++i)
    if (state[i] & pos )
        d += amplitude[i];

int amplitude[N]; // Global variable
int state[N];     // Global variable
int* SPM = &amp1itude[0];
    **spm_pos**(SPM);
    **spm_size**((2*N*sizof(int)));
...
for (i = 0; i < N; ++i)
    if (SPM[N+i] & pos)
        d += SPM[i];
```

SPM base address

SPM size

RASP!
RASP (Reuse-Aware SPM Prefetching) flow

- Prefetch-enabled: hide memory access latency
- Reuse-enabled: reduce amount of data transfers

**Compiler Support for Customizable Hybrid Cache (Cont’d)**

- **Reuse Analysis**
- **Reuse & Prefetching Co-Optimization**
- **Optimized Code for Hybrid Cache**

**Hybrid Cache Configuration**
- maximal SPM size

**C/C++ Program**
- LLVM-2.7 implementation
- prefetch latency

**Architecture Parameters**
- RASP prefetch latency
- maximal SPM size

---

**Initiate SPM buffer size**

**Calculate SPM utilization ratio r for unactive reuse edges**

**Activate reuse edge u→v with largest r**

**Update local/reuse regions of downstream vertices of v**

**Exceed SPM size?**
- **Yes**
- **No**
Concluding Remarks

- We believe that customization is the next transformative approach to energy efficient computing.
- Many EDA problems can benefit from customized computing for acceleration.
- Design automation and optimization is crucial for the success of customized computing.
  - New business opportunities for EDA.
Acknowledgements

- This research is partially supported by NSF Expeditions in Computing Award and GSRC

- Collaborators
  - Jason Cong
  - Glenn Reinman
  - Frank Chang
  - Mohammad Ali Ghodrat
  - Michael Gill
  - Karthik Gururaj
  - Hui Huang
  - Chunyue Liu
  - Yi Zou