Reconfigurable Architectures for Wireless Systems: Design Exploration and Integration Challenges

Joseph R. Cavallaro, Michael C. Brogioli, Alexandre de Baynast, and Predrag Radosavljevic

Rice University, Houston, TX USA
{cavallar, brogioli, debaynas, rpredrag}@rice.edu

www.ece.rice.edu/~cavallar
Outline

- Background – ASIPs
- Context on WG6 Issues
- Hardware Partitioning and Design Exploration
- Imagine and TTA
- Interconnect Challenges
- System Simulation
Background

- WWRF11 Oslo June 2004

- Application Specific Instruction Processor Design Flow
  - Channel Equalization for HSDPA in MIMO Environment
  - Multiple Equalizer Algorithms to same Architecture
  - Design Exploration for Area, Time, Power Constraints

- Issues Remain in System Integration
Context of WG6

White Paper on:

- Element Management, Flexible Air Interfaces, SDR
  - 4.3 SDR Baseband Reference Models and Architectures
  - 4.6 4G/Beyond 3G Verification Tools
    - PRAGA Platform and Design Flow
  - 6 Enabling Technologies
  - 6.1 Cognitive Radio
- XG and JTRS Initiatives
ASIPs (Application Specific Instruction Processors):
⇒ Excellent tradeoff between efficiency of ASICs and flexibility of DSPs
Flexible processors for mobile handsets:
⇒ Different modifications of wireless base-band algorithms (processing in slow/fast fading, low/high scattering environments)
⇒ Support for evolution of standards (3GPP, 4G, 802.11x, WiFi, etc)

Efficient processors to achieve high-demanding real time requirements:
⇒ Customized architecture is needed
⇒ Extension of ASIP instruction set with application-specific operations

Examples: Imagine Media Processor and Transport Triggered Architecture (TTA)
Design Exploration Strategies

Algorithm mapping:
Design of algorithms for efficient mapping and performance

Architecture scaling:
Having designed the algorithms, find a low power processor

Workload adaptation:
Having designed the processor, improve power at run-time

Data-Parallel Systems
Example MIMO Downlink Equalization for 3G HSDPA

- Physical layer of mobile handset in MIMO downlink
  ⇒ ASIP architecture based on TTA
- Flexible architecture solution for different modifications of channel equalization algorithm
- Highly optimized for the most computationally complex version of channel equalization
Example TTA Equalizer ASIP

Coprocessor for CG filter update
Example Imagine 3G BS ASIP
Interconnect Challenges

- System partitioning and interconnect modeling

- At higher level:
  - Fabric between processors and co-processors,
  - memory, and peripherals

- At lower level:
  - aggressive process technology scaling
  - increasing operating frequencies
  - delay, noise, and power problems

- Massive network servers down to mobile wireless handheld devices.
Intra- and Inter-chip Communication

- Interfaces between DSP, ASIP, ASIC, FPGA
- Example 3G Multi-user Detector on Multiple DSP-FPGA
- Bus-based Vbus in TI C6X DSP

SoC Core Socket-based OCP-IP Initiative for Host and Multiple Co-Processor Cores
System Simulation: Spinach Composable Software Modules

- Make software modules act and connect like real hardware
  ⇒ 1:1 Mapping between software modules and real hardware components. . .facilitates intuitive hardware modeling
  ⇒ Well abstracted port API.
    • Example: Processors only knowledge of “memory” state is through requests to imem/dmem via port interfaces.
  ⇒ Accurately supports asynchronous events
- Eliminating global state = modularity and composability.
  ⇒ Rapidly prototype systems in minutes (not hours/days)
  ⇒ No global machine state means less software engineering overhead
  ⇒ Enables complete flexibility, configurability
- Higher-level than Mentor Seamless SoC Simulator
Spinach Modules

- Processing Elements
  - Bit true cycle accurate TI C6x DSPs
  - FPGA based Coprocessors (user defined, fully flexible)
  - MIPS R4000 microcontrollers

- Memory System
  - Bus arbiters, multiported memories
  - Caches and cache controllers, SRAM and DRAM controllers.

- Interconnect
  - Mux, demux, pipe delays, user defined functions.

- Input/Output
  - DMA assists, medium access assists, I/O harness

- Support for multiple clock domains
Typical Simulator Configuration

- On chip instruction memory
- Memory Bus Arbiter
- TI C6x DSP
- FPGA Coproc
- Coproc Mem
- MIPS R4K
- Memory Bus Arbiter
- DMA 0
- DMA N
- On chip data memory
Case Study: Coprocessor based Matrix Multiplication

- Idea: Use custom FPGA coprocessor with DSP.
  - Run dot product of matrix multiply vectors on coprocessor.
  - Use host DSP for synchronization, DMA control, all other code.

- Simulated System.
  - 167MHz TI C62x DSP
  - 64k single cycle on-chip instruction and data memories.
  - Coprocessor software controlled via memory mapped registers. Asynchronous. More on this later…
  - Data transfers to coprocessor via on-chip DMA engines.
Experimental Setup

- Workloads:
  - 16 bit fixed point matrix multiply kernels compiled at –O3 in CCS
  - Array sizes and offsets known statically at compile time
    - Compiler can aggressively schedule and unroll loops.
  - DSP controls DMA of data to/from coprocessor
- Coprocessor Specifications.

<table>
<thead>
<tr>
<th></th>
<th>DSP only</th>
<th>8 wide coproc</th>
<th>16 wide coproc</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 element dot product</td>
<td>N/a</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>16 element dot product</td>
<td>N/a</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>32 element dot product</td>
<td>N/a</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>64 element dot product</td>
<td>N/a</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>128 element dot product</td>
<td>N/a</td>
<td>21</td>
<td>11</td>
</tr>
</tbody>
</table>
Future Directions for 4G

- System on Chip (SOC)
  - Integration Challenges with DSP – FPGA/ASIP Co-Design, Simulation, Verification – Very Error Prone – Needs Standardized Interfaces

- Gigabit/sec Systems
  - Modular Terminals will Support from Voice to over 1 Gbps Wireless through Several Interfaces
  - OFDM, MC-CDMA in High Mobility
Research Topics and Challenges

- Architectures and Design Environments (From algorithm mapping and analysis up to SoC and Reconfigurable):
  - Hardware Abstraction Layer (HAL) development for reconfigurable SoC systems
  - DSP host to ASIP and ASIC co-processor system integration for System on Chip (SoC) design
  - SoC simulation environment based on DSP with programmable ASIP co-processors and fixed ASIC blocks