Reconfigurable Computing for User Customizable SoC

Hiroto Yasuura
System LSI Research Center (SLRC)
Kyushu University
E-mail: yasuura@slrc.kyushu-u.ac.jp
Outline

- Background and Requirements
- Platforms for Reconfigurable Computing
  - DRP
  - DAP/DNA
- How to use Reconfigurable Computing in SoC
  - SysteMorph
- Conclusion
Why Reconfigurable?

- **Cost of Production**
  - Drastic increase of design and mask cost is requesting new system architectures, especially for small scale production less than 1M.

- **Customer Satisfaction**
  - Various kinds of customers, each of which has different requirement and knowledge. A customized system for each user is attractive.

- **Market Oriented SoC Design**
  - The direction of the market changes quickly and various new services are introduced.

- **Reliability and Security**
  - Repairs and debugging on customer site.
  - Changing system configuration for security. (cryptography etc.)

- **Global Environment Problem**
  - Grow out of the throwaway society.

- **Views of System Designers and Users**
Example: Mobile Phone

- **New Services**
  - I-mode (Internet Access: e-mail and WWW)
  - Built-in Digital Still Camera
  - Video Phone Service (MPEG-4 in NTT Foma)
  - Melody Calling
  - Music Down Load Service (MP3)
  - Electric Ticketing
  - Electric Money for Vending Machines
  - Simple Interface for Old People
  - Car Navigation Service

- **Needs for a new system architecture solution**
  - Reconfigurable computing is a possible solution.
System Level Optimization

- What
  - Parameters for optimization

- When
  - Design Stage, Compilation Stage, and Runtime

- Who
  - Designers, Service Providers, and also Users

- How
  - Reconfigurable Hardware Platforms
  - Software
  - Profiling and Design Optimization Techniques
Reconfigurable Computing

When?

Online Profiling & Reconfiguration

Runtime

DCO  EH  SysteMorph

Compile Stage

Offline Profiling & Reconfiguration

CO  SRC  Codesign

Design Stage

Offline Profiling & Optimization

SW  HW  HW&SW

What?

DCO: Dynamic Compilation/Optimization
CO: Compiler Optimization
EH: Evolvable Hardware
SRC: Static Reconfigurable Computing

By K. Murakami

H. Yasuura, Kyushu Univ.

SCESS 6
Runtime Reconfiguration

- **Dynamic**: Optimization is performed...
  - After SoC is shipped to the market
  - While SoC is used in the field
- **Online**: Profiling and optimization are performed...
  - In parallel with the execution of application programs
  - During idle or sleeping time
- **Adaptive**: Optimization is repeated...
  - In the form of a feedback loop
Analogy: Formula 1

The car is running in the course.

The pit crew is monitoring the behavior of the car.

After the reconfiguration, the car returns to the course.

Once the pit crew finds any hints for reconfiguration, the car pits in.

The car is now under reconfiguration.

By K. Murakami
When reconfiguration is done?

A system is not always active. Reconfiguration can be done in idle and sleep time.

H. Yasuura, Kyushu Univ.  
SCESS
A Possible Business Model: OSP (Optimization Service Provider)

Profile your habits
Profile your behaviors
Profile your favorites

Users in the field

Zzzzzz...

Improve your PDA while you sleep

Profiled Data
Configuration Data

OSP (Optimization Service Provider)

Pit Center

H. Yasuura, Kyushu Univ.
Customizable Mobile Phone
Your phone is evolving every battery charging!

On-line Profiling

Improving QoS in Sleep
- Sound quality
- Battery life
- Key operation
- New services
- Debugging

Out-of-Suit Optimization

Optimization Service Provider
Reconfigurable Computing for System on a Chip

- Background and Requirements
- Platforms for Reconfigurable Computing
  - DRP
  - DAP/DNA
- How to use Reconfigurable Computing in SoC
  - SysteMorph
- Conclusion
Platforms for Reconfigurable Computing

- Dynamically Reconfigurable Processor: DRP by NEC
- DAP/DNA by IP Flex
- Dynamically Reconfigurable Circuits by Sony
Platforms for Reconfigurable Computing

- Granularity of Reconfiguration
  - A Processor and Software
  - An Processor Array
  - Processing Elements
    - ALU, Multipliers, etc.
  - Logic Gates (FPGA)

- Timing of Reconfiguration
  - Every Clock Cycle
  - Every Task Execution
  - Every Power-on
Dynamically Reconfigurable Processor: DRP

8DRP cores on a Chip

A DRP core includes 64 PEs. STC controls PEs.

By NEC

H. Yasuura, Kyushu Univ.
• 16 instructions can be stored in the instruction memory of each PE.
• An instruction specifies connections and operation of each processor.
• The STC specifies the address of instruction.

By NEC
Dynamic Reconfiguration of DRP

The connection among PEs and operations of PEs can be changed in every clock cycle.

By NEC
DAP/DNA - IP Flex (http://www.ipflex.com)

External Memory

BUS Controller

Instruction Cache

Data Cache

DNA Buffer

Configuration Memory

DNA Matrix

RISC Core

Traditional 32b embedded processor: DAP

32b ALU or MUL
Features of DAP/DNA

DAP/DNA reconfigurable processor has the advanced features, including:

- The DNA-Matrix architecture with dynamically reconfigurable hardware.
- Reconfiguration of the DNA-Matrix in one clock.
- Parallel data processing, not sequential data processing (Neumann Cycle), and extremely high performance with low power consumption due to the low clock frequency.
- 1-2 digits higher performance compared to existing solutions such as the CPUs and DSPs.
- Dramatic reduction of the development cost and period compared to ASIC and fully custom devices.
- Hardware design with software method (C language) enables flexible the design changes.

600MTr. 225m²

DNA (Distributed Network Architecture) Matrix Architecture

The DNA-Matrix is a dataflow type accelerator arrayed 148 dynamic reconfigurable operation units. The wiring among elements can be changed dynamically and can quickly constitute parallel/pipeline processing system according to each application operation unit processing. The DNA-Matrix internal constituent information is stored in configuration memory, and its constitution changes in one clock depending on applications.

- 148 of 32bit Operation Units
- Data transfer between elements at single cycle
- Operating Frequency 100MHz

By IP Flex
Reconfigurable Computing for System on a Chip

- Background and Requirements
- Platforms for Reconfigurable Computing
  - DRP
  - DAP/DNA
- How to use Reconfigurable Computing in SoC
  - SysteMorph
- Conclusion
Silicon Sea-Belt Project
Just-in-Time (Dynamic, Online & Adaptive) HW/ISA/SW Co-optimization Technology

Applications:
- High Performance Computing
  - Molecular Orbit Computation (Chemistry)
  - Reducing Cost and Energy
- Mobile Devices
  - Mobile phones
  - Sensor networking
  - Reducing Energy and Increase Service Quality
Design Issues in SysteMorph

- What to profile
- How to profile them
- How to discover hints for optimization
- What to optimize
- How to optimize them
- How to reconfigure HW/ISA/SW
Functionality Morphing: An Example of SysteMorph

- Design issues in SysteMorph
  - What to profile
  - How to profile them
  - How to discover hints for optimization
  - What to optimize
  - How to optimize them
  - How to reconfigure HW/ISA/SW

- Solutions in functionality morphing
  - Online hot-path profiling
  - Offload the functionality of hot-paths from SW to HW
  - Online HW resynthesis
  - Reconfigurable co-processor
  - Dynamic binary rewriting

By K. Murakami
Functionality Morphing - Offload Hot Program Path to HW -

Application programs are running...

(1) Monitor program path
(2) Detect and predict hot program path
(3) Transform the function of the hot path into a logic function
(4) Reconfigure the hardware of a reconfigurable co-processor (RCP)
(5) Replace the hot path with a co-processor call

Application programs are under optimization...

Instruction Execution
Profiler
Reconfigurable Fabric
Processor Core
Target Programs
SysteMorph Software

ISA
Instruction Execution
Profiler
Reconfigurable Fabric
Processor Core
Target Programs
SysteMorph Software

By K. Murakami

H. Yasuura, Kyushu Univ.
Offline vs. Online Profiling

Offline Profiling
Summary of program behavior based on whole program trace
Good for:
- CO (Compiler Optimization)
- SRC (Static Reconfigurable Computing)

Online Profiling
Prediction based on current execution window of program
Good for:
- DCO (Dynamic CO)
- SysteMorph

By K. Murakami
Online Hot-Path Profiling
- Algorithm -

**Program Path**

**Our Algorithm**
- Profile the history of branch instruction’s behaviors (taken or not-taken, branch target)
- If the execution frequency at a path head exceeds the threshold, select the path head (‘A’ in the figure) as a candidate of the hot path head
- Traverse the object code, starting with the candidate (‘A’), based on the branch history, and predict the hot path

**Existing Algorithms**
- Efficient path profiling
- NET prediction

By K. Murakami

H. Yasuura, Kyushu Univ.  SCRESS  27
Example: Dynamic Software Pipelining

Hot Path Detection

Software Pipelined Loop

Modified Code of basic blocks not included in the hot path (cool parts)

Basic Block

Hot Path

Cool parts
DSWP適用後の速度向上比

speed up

benchmark

164gzip 176gcc 181mcf 197parser 256bzip2 300twolf
## Analysis

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>#Path</th>
<th>#HotPath</th>
<th>#HotPath of DSWP Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>164.gzip</td>
<td>401</td>
<td>38(97.8%)</td>
<td>12(30.5%)</td>
</tr>
<tr>
<td>181.mcf</td>
<td>165</td>
<td>38(98.7%)</td>
<td>6(5.1%)</td>
</tr>
<tr>
<td>176.gcc</td>
<td>15874</td>
<td>168(45.9%)</td>
<td>50(17.14%)</td>
</tr>
<tr>
<td>197.parser</td>
<td>2720</td>
<td>126(81.5%)</td>
<td>15(23.7%)</td>
</tr>
<tr>
<td>256.bzip2</td>
<td>115</td>
<td>11(99.8%)</td>
<td>1(32.1%)</td>
</tr>
<tr>
<td>300.twolf</td>
<td>106</td>
<td>106(87.8%)</td>
<td>19(19.17%)</td>
</tr>
</tbody>
</table>
Functionality Morphing Prototyping:
IP Flex DAP/DNA Powered by SysteMorph

SysteMorph Software

Target Programs

SysteMorph Software

Target Programs

Online Profiling

Binary Rewriting

Hints for Optimization

Adaptive HW/ISA/SW Co-optimization

Instruction Execution

Instruction Execution

DAP

DNA

DAP

DNA

ISA

DAP

DNA

By K. Murakami
Offline vs. Online HW Synthesis

**Offline HW Synthesis**
HW configuration is synthesized prior to the program execution
Good for:
- SRC (Static Reconfigurable Computing)

**Online HW Synthesis**
HW configuration is synthesized in parallel with the program execution
Good for:
- SysteMorph

---

By K. Murakami

H. Yasuura, Kyushu Univ.
Reconfigurable Computing for System on a Chip

- Background and Requirements
- Platforms for Reconfigurable Computing
  - DRP
  - DAP/DNA
- How to use Reconfigurable Computing in SoC
  - SysteMorph
- Conclusion
Reconfigurable Computing

- Several platforms are ready to use. System architectures and application techniques for SoC are key issues.
  - Cost Reduction
    - Reduce design, mask, and test costs.
    - Reuse an SoC device into different consumer products.
  - Customer Satisfaction
    - Optimize devices for multiple applications and varieties of usage.
  - Speed-up of Business
    - Introduce new services without change of devices.
  - Reliability and Security
    - Repairs and debugging on customer site.
    - Frequent update of security procedures.
  - Environment and Ecology
    - Reuse devices in revised services.
    - Minimization of energy consumption for each user.
Thank you for your attention.

Silicon Sea-Belt
- Low Energy SoC with RF
- SysteMorph
- SiP Design Tools
- Embedded Software Design Methods
- EDA for SoC