

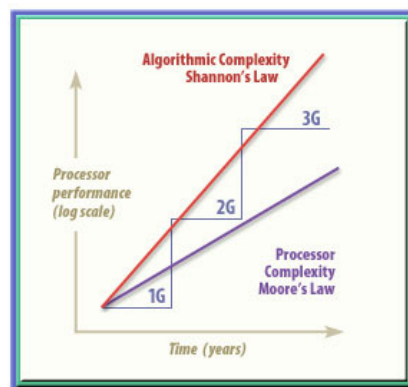
Review of:

Re-Configurable Computing in Wireless

- Paper by:
 - Bill Salefski, Levent Caglar (Chameleon Systems, Inc.)
- Published in:
 - 38th Design Automation Conference, Las Vegas
 - June 18-22, 2001
- Survey by:
 - Haoyu Song

The Motivation

- Design Dilemmas
 - Algorithmic complexity is increasing faster than processing power;
 - The dynamics of wireless communication standards;
 - The realities of building large system on chip (SOC) solutions.
- Solution
 - Re-configurable Computing
 - Chameleon's Re-configurable Communications Processor (RCP)



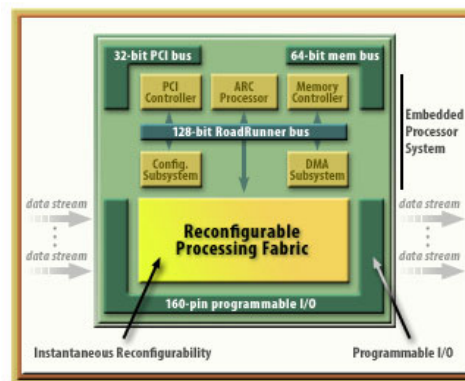
Jan Raboey, Berkeley Wireless Research Center

Style of the Paper

- Overview of the Chameleon RCP
- Major architecture features of the Chameleon RCP
- Chameleon RCP design methodology and tools
- Performance compare with DSP, FPGA and ASIC approaches.

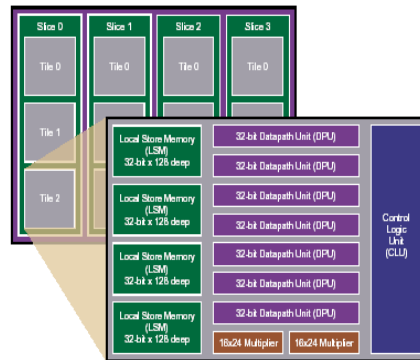
Chameleon RCP Architecture

- 3 major subsystem:
 - Reconfigurable Processing Fabric (RPF)
 - Programmable input and output (PIO) banks
 - Embedded processor subsystem
- Linked with a high-performance 128-bit bus.



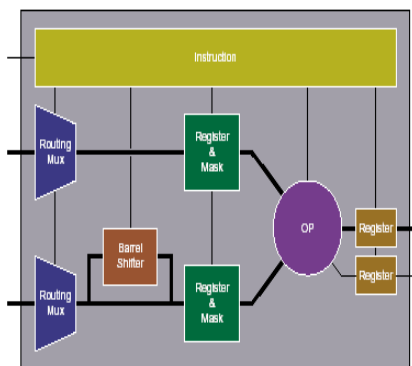
Re-configurable Processing Fabric (CS2112)

- Structure of RPF
 - Data processing units
 - Local storage
 - Interconnect structure
- RPF for CS2112
 - 4 slices
 - 3 tiles / slice
 - 7 32-bit DPU / tile
 - 2 Multipliers(MPU) / tile
 - 4 Local-Store Memories(LSM) / tile
 - 1 Control Logic Unit (CLU) / tile



RPF: Data-path Unit & MPU

- DPU
 - 8 user-definable instructions stored in Instruction memory
 - Word length 32 bits, can also work on 4 8-bit and 2 16-bit data stream
 - 1 core: Operator
 - 2 branch data paths
- MPU
 - 16x24-bit or 16x16-bit single-cycle multiplications

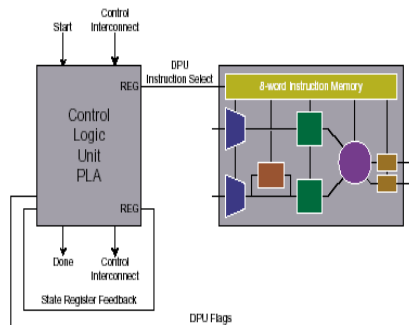


RPF: Local-Store Memory

- Multi-ported, 32-bit by 128-word RAM
- Re-configurability: LSMs can be assembled into different memory configurations through programming.
- Common use: Quickly load data.

RPF: Control Logic Unit Connection

- DPU Instructions
 - I/O routing
 - Shifting
 - Masking
 - Register enables
 - LSM read
 - LSM write
 - Flag generation
 - DPU Operator
- CLU
 - Implement a FSM to select DPU/MPU instructions



Other Features

- **Dynamic Interconnect**
 - No signaling protocol
 - Nearby DPUs in full crossbar connection
 - Distant DPUs routing with 1-clock pipeline delay
 - Routing multiplexers control the routing selection
- **Programmable I/O**
 - 4 banks of 40 programmable I/O pins
 - Programmed to interface to a variety of external data sources: SRAM, ADC and FPGA
 - Raw data bandwidth of about 2.0 Gbits/sec
- **Embedded Processor Subsystem**
 - ARC processor
 - 8K instruction cache and 8K data memory
 - 32-bit PCI I/F, DMA, 64-bit memory controller

Reconfiguration

- **System Reconfiguration**
 - A configuration controller
 - 2 configuration planes: Active and background plane.
 - Less than 50,000 bits and 3 ms required to program all slices
 - One clock cycle switching
 - Data in LSM retained when reconfiguration
- **Instruction-Based Reconfiguration**
 - DPU and Dynamic Interconnect controlled by CLU and specified in Instruction Memory.
 - Enable clock-by-clock basis reconfiguration through programming.

Design Methodology

- Design Process
 - Define the data flow through the system
 - Map the operations to the architecture elements
 - **Parallelism**
 - Instruction level - operation
 - Task level - process
 - Enter the design into RCP programming tools
- Architecture Mapping
 - Pipelining
 - **Interleave other data streams for the same or similar algorithm through the data path**
 - Reconfiguration as Design Component

Reconfiguration As Component Example

- Frame-based application, frame requires several distinct processing steps. Using reconfiguration instead of having separate processing units.
- Using reconfiguration to vary the algorithm depending on environmental considerations.
- Completely different algorithms based on the traffic mix.



In a cdma2000 chip-rate application, for example, tasks such as pseudorandom noise generation and rake finger searches are not parceled out to various pipelined subprocessors. Instead, the entire processing fabric is dedicated first to pseudorandoms, then to demodulation, then to finger searches, with the task *reassigned in a single cycle*.

— EE Times

Compare to Other Approaches

- **FPGA and ASIC Design**
 - Similar
 - RCP is completed when DPU and control functions are specified and mapped onto RCP.
 - After get net list, FPGA/ASIC still need floor planning, timing analysis, processor integration, place and route.
 - RCP tools runtime required only a few minutes
 - FPGA/ASIC tools runtime tend to be in hours
- **DSP Design**
 - Similar in use assembly language
 - RCP offer more parallelism and more flexibility

Design Tools

- **Fabric Design**
 - Designer specify the programming for individual DPUs
 - Tools map, par actual fabric
 - Based on Verilog
- **Embedded Processor Design**
 - Industry-standard GNU-based tool chain
 - Chameleon system libraries - eBIOS
 - Fabric control functions
 - Initiating processing on the fabric
 - Loading a configuration in the background planes
 - Scheduling the next fabric loads
 - Communicating with system host processor
- **System Verification**
 - Software simulator
 - Standard PCI-based evaluation board

Performance Comparison - DSP

	RCP	TI TMS320C62-300
24-tap FIR filter with 16-bit data, 200 samples	1.9 us	11.0 us
1024-point complex FFT or Inverse FFT with 16-bit real and 16-bit imaginary data, 1024 samples	42.6 us If 4 instances running in parallel, Can output 1024 samples / 10 us.	68.7 us

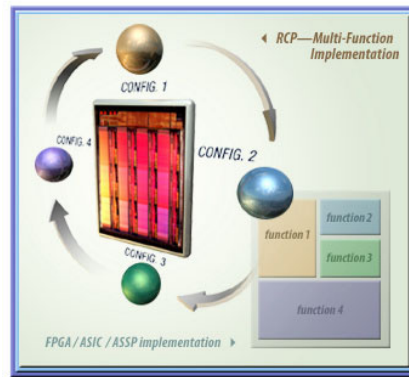
- RCP is about 6 times faster

Performance Comparison - FPGA

- Performance similar
- Significant cost difference
 - RCP's fixed micro architecture result in fewer unused wires and routing channels comparing with FPGA's more fine-grained architectures
- No experiment data present

Performance Comparison - ASIC

- ASIC provide ultimate performance but fail in time-to-market and unable to satisfy the need of flexibility.
- Reconfiguration versus Dedicated Hardware - **higher performance, lower cost and lower power consumption.**



Conclusions

- Providing a reconfigurable solutions to today's processing requirements.
 - meets the increasing computational demands of the wireless industry while providing the programmability to enable equipment providers to update deployed equipment with software programming.
- Validating the concept of modular computing,
 - the RCP architecture implements an interconnected array of computational elements and memories connected with a programmable interconnect.
- The paper has nearly nothing to do with the wireless computing
- Lacking detailed case study and performance analysis