FPL 2007
The 17th International Conference on Field Programmable Logic and Applications
Amsterdam, Netherlands 27-29 August 2007

Design Methodology and Trade-offs Analysis for Parameterized Dynamically Reconfigurable Processor Arrays

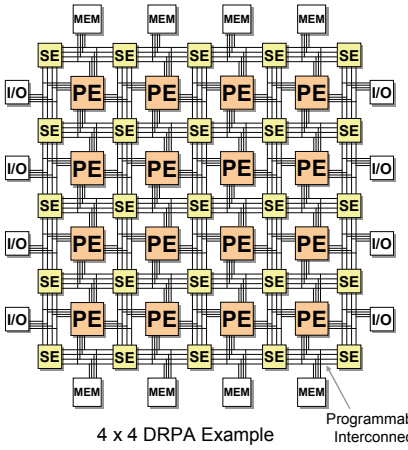
**Yohei Hasegawa, Satoshi Tsutsumi, Vasutan Tunbunheng,
Takuro Nakamura, Takashi Nishimura, and Hideharu Amano**

Department of Information & Computer Science, Keio University

E-Mail: muccra@am.ics.keio.ac.jp

1. DRPA Overview

Dynamically Reconfigurable Processor Array (DRPA) is a high-performance and low-cost programmable device for versatile System-on-Chips (SoCs)

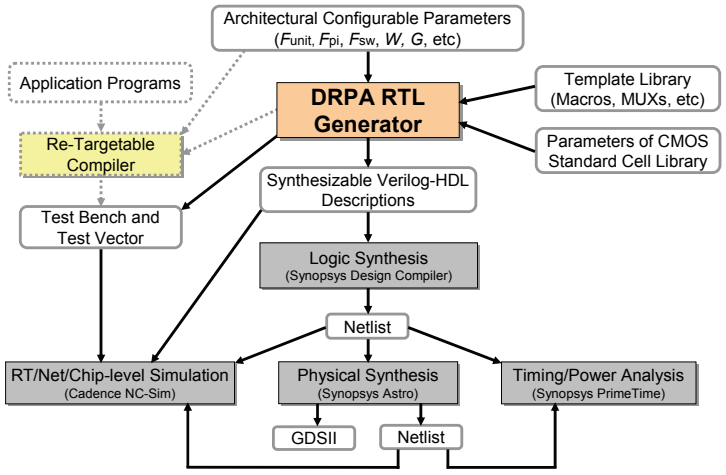


4 x 4 DRPA Example

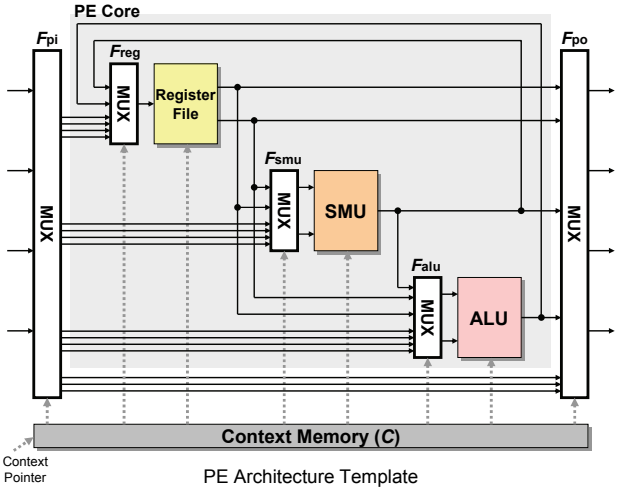
- **Coarse-grained Processing Element (PE) Array**
  - Consists of many PEs which provide a 4-bit to 32-bit word-level datapath
  - Exploits Instruction-level and data-level parallelisms for streaming applications
- **Dynamic Reconfigurability**
  - PE operations and inter-PE connections (called a ***context***) can be dynamically reconfigured in cycle-by-cycle manner
  - Each PE has its own context memory
- **C-based Programming Methodology**
  - Algorithmic behaviors can be described in C-like languages
  - Applying High-level Synthesis Technology for back-end compiler
  - Various parallel algorithms can be mapped (ex: SIMD, VLIW, Pipelining)

2. DRPA Design Environment

Our DRPA design environment offers an efficient design space exploration for highly parameterized DRPA architecture



3. Target DRPA Architecture Template



PE Architecture Template

Our DRPA architecture is highly parameterized, and the DRPA Generator can generate various types of DRPAs by controlling the configurable parameters for a target application domain

I. PE Architecture
- Programmable PE Core
  - Register File
  - Shift and Mask Unit (SMU)
  - Arithmetic Logic Unit (ALU) including a multiplier
- Connection blocks (MUXs) for intra-PE interconnect
- Context memory

II. Array Architecture
- *M* x *N* two-dimensional PE array
- Island-style inter-PE connection network
- Switching Element (SE) at a channel intersection
  - A set of programmable switches
  - Context memory
- A certain number of distributed memory elements

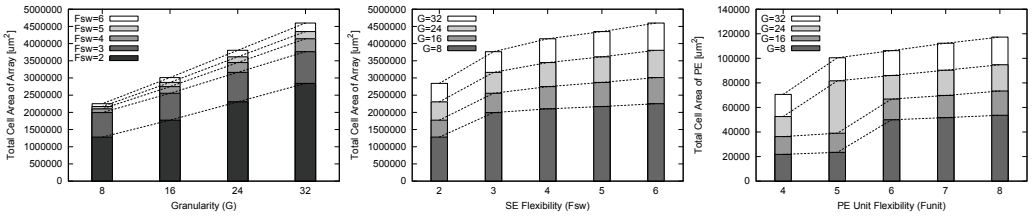
III. Context Switching Controller
- Multi-Context scheme for dynamic reconfiguration
- Broadcasting the context pointer to each PE and SE
- Cycle-by-cycle context switching

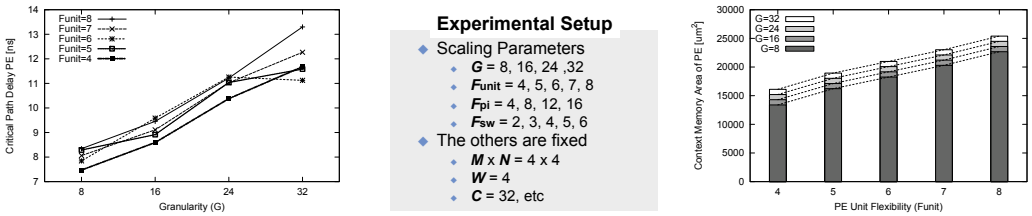
Configurable Parameters

Parameter	Description	Typical Range
G	PE Granularity	4, 8, 16, 24, 32
M x *N*	PE array size	4 x 4, 8 x 8, …
W	Number of channels	2, 3, 4
C	Number of contexts	4, 8, 16, 32, 64
N_{mem}	Number of memories	4, 8, 16, …
N_{reg}	Number of registers	4, 8, 16, 24
F_{reg}, F_{smu}, F_{alu} (= F_{unit})	PE unit flexibility	4, 5, 6, 7, 8
F_{pi}, F_{po}	PE I/O flexibility	4, 8, 12, 16
F_{sw}	SE flexibility	2, 3, 4, 5, 6

4. Experimental Results

Our DRPA design environment enables SoC designers to rapidly evaluate architectural trade-offs, and finally provides a chip-layout of the preferable DRPA architecture





Experimental Setup
- Scaling Parameters
  - *G* = 8, 16, 24 ,32
  - F_{unit} = 4, 5, 6, 7, 8
  - F_{pi} = 4, 8, 12, 16
  - F_{sw} = 2, 3, 4, 5, 6
- The others are fixed
  - *M* x *N* = 4 x 4
  - *W* = 4
  - *C* = 32, etc

5. Case Study: MuCCRA-1

We fabricated a real DRPA chip in 180nm CMOS tech.

MuCCRA = **M**ulti-**C**ore **C**onfigurable **R**econfigurable **A**rchitecture

MuCCRA-1 Configuration
- *G* = 24 for efficiently treating RGB streaming data
- *M* x *N* = 4 x 4
- *W* = 2
- *C* = 64
- F_{unit} = 4
- F_{pi} = 4
- F_{sw} = 2

Technology: **Rohm 180nm 6-layer CMOS**
Die Size: **5.18mm²**
Voltage: **1.8V** for core and **3.3V** for I/O
Package: **QFP304**

	Block Size [bit]	Clocks	Delay [ns]	Exec. Time [us]	Power [mW]
DCT	1024	195	40.0	7.8	85.1
α-Blender	8192	644	24.0	15.5	103.3
SHA-1	512	418	50.0	20.9	50.6
Viterbi	16	600	42.0	25.2	45.9

MuCCRA-1 is up to 2 times faster than TI's 225-MHz DSP concurrently with lower power dissipation