Transparent Binary Acceleration Using Reconfigurable Array

Jongkyung Paek, Yangsoo Kim,
and Kiyoung Choi

School of EECS
Seoul National University

**Jongeun Lee**

School of ECE
UNIST

Outline

- **Introduction**
- **FloRA**
- **Configurable Range Memory**
- **Experiments**
- **Conclusion**

Introduction

- **Binary acceleration**
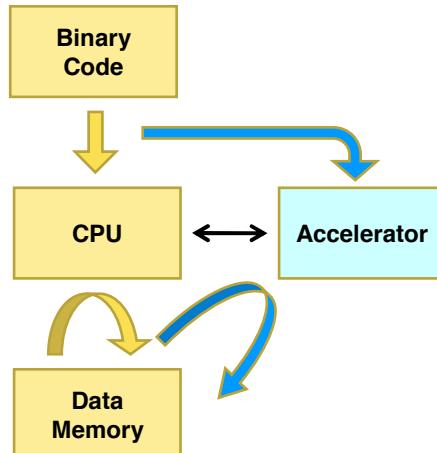
Introduction

- **Accelerator**
  - **FloRA: coarse-grained reconfigurable array (CGRA)**
  - **Wide range of application kernels**
 - **Low run-time configuration overhead**
- **Transparency**
  - **Automatic insertion of memory management code**
    - **For memory-centric communication**
    - **Configurable range memory (CRM) allows efficient and transparent data communication between the processor and the CGRA**
  - **Need static analysis of binary code**
    - **Currently not 100% transparent**

FloRA

- ## Architecture
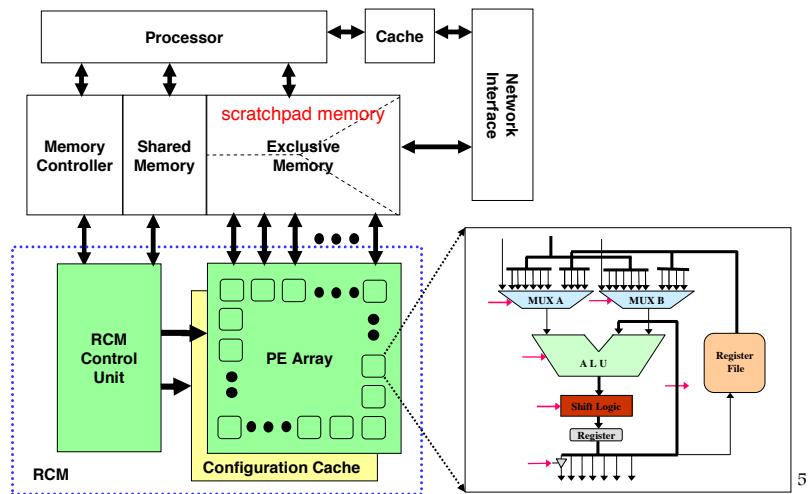
5

# FloRA

- ## Binary acceleration on FloRA
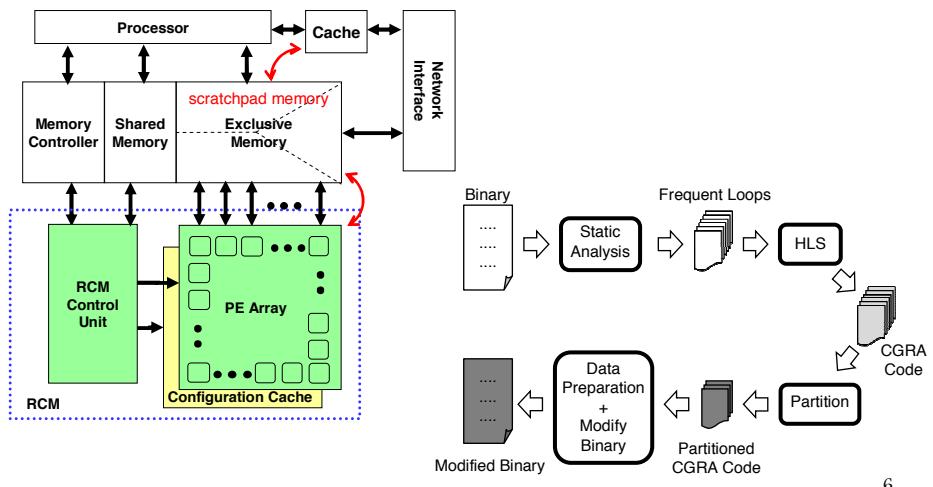
6

Configurable Range Memory

- **Memory-centric communication**
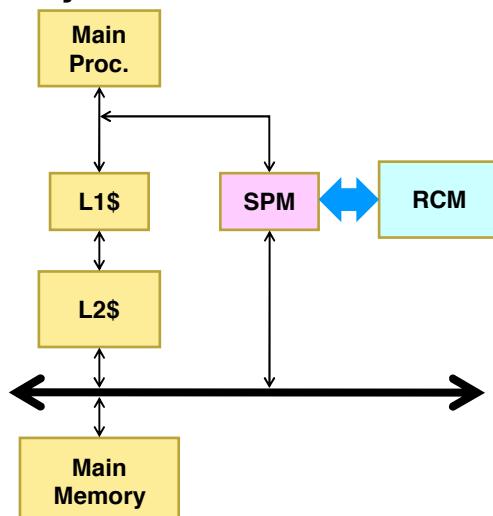
7

Configurable Range Memory

- **Memory-centric communication**
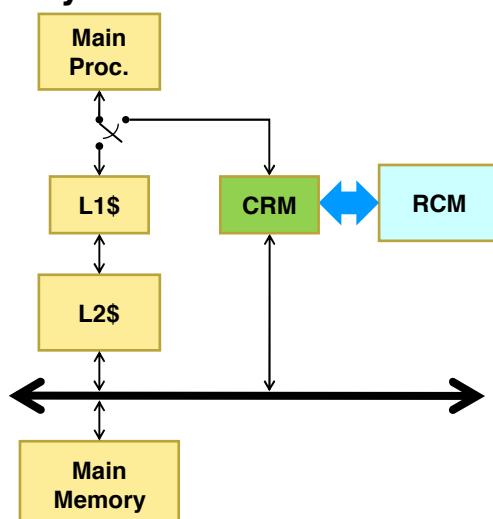
8

Configurable Range Memory

- ## CRM
  - ### Range recognizer

Address

Tag	Index	Blk Offset

=? → **Tag Match**

Tag Reg

| V | | D |

Valid Bit **Dirty Bit**

BValid Bits

Address Space

→ A block (Index=0)

→ All addresses here have the same tag (=0x4000)

→ Valid blocks

Example:
Addr=0x4000_30_00
Tag=0x4000
Index=0x30
Hit if Tag Reg=0x4000
and BValid[0x30]=1

9
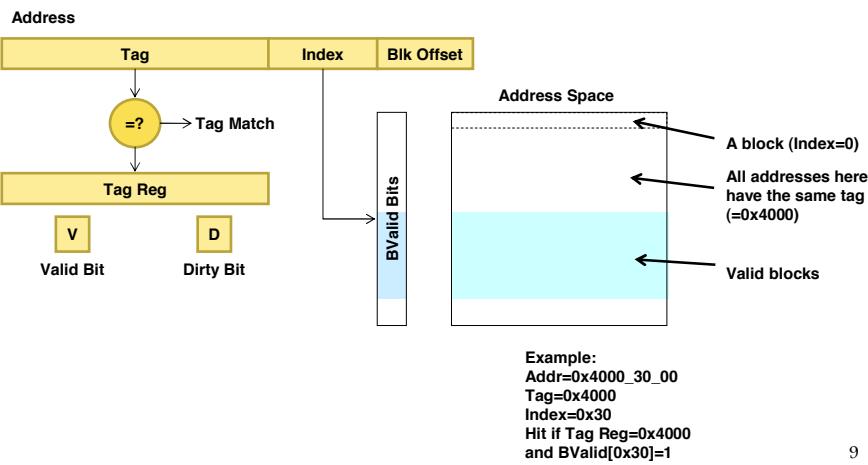
Configurable Range Memory

- ## CRM
  - ### Multiple range recognizers per bank
  - ### Multiple banks

Tag Reg0

Tag Reg1

Tag Reg2

Tag Reg3

Bank0

Bank1

BValid Bits0

BValid Bits1

BValid Bits2

BValid Bits3

Array_C
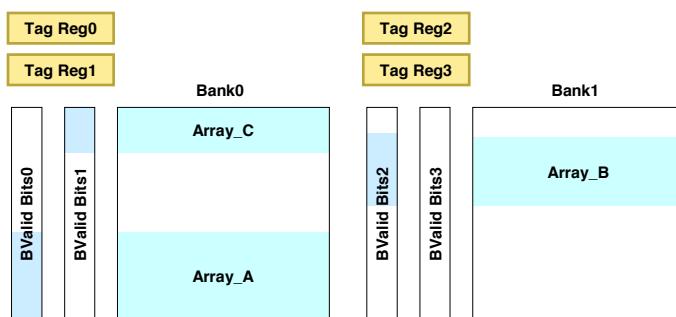
**Array_A**

Array_B

10

Experiments

- **Experimental setup**
  - **Main processor: ARM926EJ-S**
  - **D cache has 32KB**
  - **RCM: 4x4 array of PEs**
  - **4 banks of CRM**
  - **4KB (8KB) for each bank**
  - **2 range recognizers per bank**
  - **AHB is used for the bus**
  - **Access to CRM takes 1 cycle**

11

Experiments
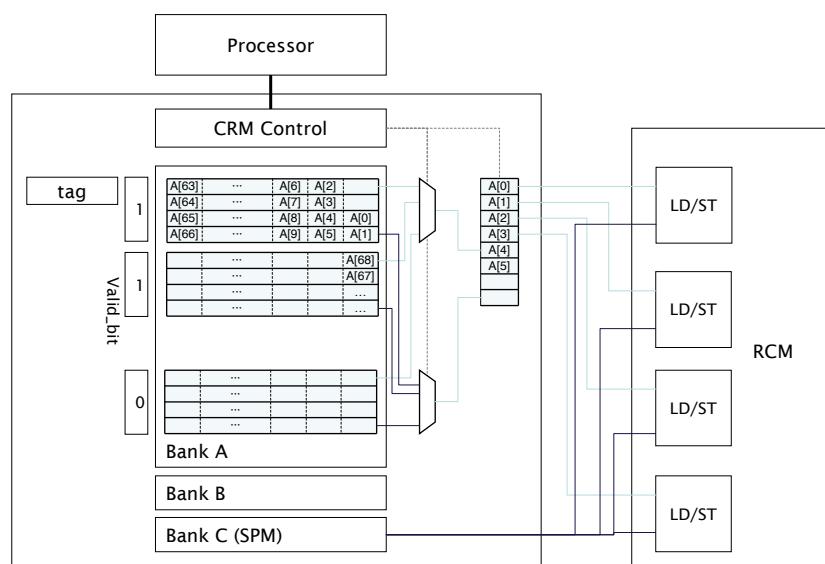
Experiments

- **DSPstone benchmark result**

	SW	RCM		Speedup	
		Comp.	Comm.	Conv.	CRM
Complex multiply (N=16)	2928	24	1199	2.39	26.38
Complex updates (N=16)	3807	24	1744	2.15	27.19
Convolution (16-point)	939	16	373	2.41	9.12
Dot product (N=16)	1429	16	601	2.32	13.87
FIR (16 taps)	1492	17	319	4.44	11.22
FIR 2D (3x3 coeff)	5782	40	651	8.37	45.53
IIR biquad (4 sections)	949	16	67	11.43	7.19
LMS (16 taps)	1810	25	275	6.03	12.84
Matrix mult (4x4)	3813	36	1715	2.18	31.00
N real updates (N=16)	1950	12	1250	1.55	15.23
Average(geometric mean)	-	-	-	3.46	17.00

Experiments

- **JPEG decoder result**

	SW	Conventional	CRM
Computation	-	355,947	355,947
Communication	-	4,793,885	547,501
Management	-	-	189,996
Total	7,511,315	5,149,832	1,093,444
Kernels Speedup	1.00	1.46	6.87

Array	Size (Bytes)	Scope	Address	in/out
A	256	global	fixed	in
B	64	global	sweeping	in
C	128	global	sweeping	in+out
D	128	local	fixed	in+out
E	128	local	fixed	in+out
F	64	local	fixed	in+out

Experiments

- **MPEG-2 encoder result**

		SW	Conventional	CRM
	Computation	-	90M	90M
Kernels	Overhead	-	547M	267M
	Total	3,322M	638M	357M
	Speedup	**1.00**	**5.21**	**9.29**
Others	Total	523M	523M	523M
MPEG-2	Cycles	3,846M	1,161M	881M
	Speedup	**1.00**	**3.31**	**4.36**

Conclusion

- **Binary acceleration with FloRA**
  - **Tried to make it transparent with CRM architecture**
  - **Less communication overhead between the processor and the accelerator**
- **Future work**
  - **Improving CRM architecture for efficient access from RCM**
  - **Automatic code generation**
  - **Application to other kinds of accelerators**

Thank you!