FFT implementation using monoinstruction set computer (MISC) architecture

#### Hiroki Shinba and Minoru Watanabe

Department of Electrical and Electronic Engineering Shizuoka University, Japan E-mail: watanabe.minoru@shizuoka.ac.jp

## **Background**

- 20 years ago, FPGA's performances were much lower than those of ASICs.
- However, currently, the performance difference between FPGAs and ASICs became smaller
  - ✓ ASICs cannot use the latest VLSI technologies due to an initial cost issue
  - ✓ FPGAs can use the latest VLSI technologies
- However, FPGA's performances are still not good compared with ASICs
- Therefore, we are aiming at increasing the performances of programmable gate arrays

#### **Overview of an optically reconfigurable gate array**



11:20 AM – 12:00 AM: New/Exploratory paradigms , 25 February, WPPP2018

#### Storage capacity of a holographic memory



In the case of fine-grained gate array, 1 gate programming requires 3 bits.

If 3 cm<sup>3</sup> memory can be implemented onto an ORGA, 1 Tera gates can be achieved.



Holographic memory

Current VLSI's gate array is less than one billion gates.

The gate array is at least 1000 times larger than that of current VLSI

#### **Optically reconfigurable gate array VLSI**

# Table 1: Specification of a fabricated ORGA-VLSI chip.

Technology	0.18 $\mu m$ double-poly			
	5-metal standard CMOS process			
Die size	$5.0 \times 5.0 \ mm^2$			
Supply voltage	Core	1.8V		
	I/O	3.3V		
Photodiode	Junction area	$4.40 \times 4.45 \ \mu m^2$		
	Switching energy	$2.12 \times 10^{-14} \text{ J}$		
	Horizontal interval	30.08 $\mu m$		
	Vertical interval	30.24 $\mu m$		
	Num. of photodiodes	17,664		
Gate array	Num. of logic blocks	128		
	Num. of switching matrices	144		
	Num. of Wires in a wiring channel	8		
	Num. of I/O blocks	16 (64 bit)		
	Gate count	8,704		



Photograph of a fabricated ORGA-VLSI chip.

## **Optically Reconfigurable Logic Block**



11:20 AM – 12:00 AM: New/Exploratory paradigms , 25 February, WPPP2018

#### **Optically Reconfigurable Switching matrix**



Block diagram of a configurable switching matrix.

	<b>L</b> ÉS		i finish					
	<b>677</b>		668			1738		653
		RON						
					163		67	
		KARA				<b>1</b> 533		
	b:			63		6	63	m
Ę		- 673	1273	ল্লেন্স	673	63	<b>F</b>	<b>1</b>
<u>,</u>		<b>Sten</b>			. <u>CÚ</u>			
<b>←→</b> 30.08 μm								
	←			236.8	8 սm			→

# Photograph of a configurable switching matrix.

#### 16 configuration context ORGA system



## **Performance of ORGAs**

#### We have demonstrated

- ✓ over 100 MHz reconfiguration
- ✓ 256 reconfiguration contexts in ORGA

# Such dynamic reconfiguration can increase the gate array performance!

#### Mono-Instruction Set Computer (MISC) concept

Microprocessor history

• Famous change is from CISC to RISC

Complex instruction => single step instruction

A number of instructions => a small number of instructions

Operating clock frequency is increased, power consumption is decreased, and die size is also decreased.

• This success : simplest circuit is the best

 $\bigstar$  Such success can be adopted into programmable devices



## An example of MISC implementation



- While reconfiguration is executed, values on all registers are kept
- Since an optical reconfiguration operation can be executed as a background operation, overhead of the reconfiguration can be out of consideration

# **MISC implementation result**

Implementation results of mono-instruction set computers (MISCs) of 11 kinds. The bottom line shows a conventional RISC soft-core processor including all instructions of the MISC processors described above, which is a comparison target under the same conditions. Here, the target process technology is a 40 nm process on VTR.

Instruction	Area	Num. of units	Operating frequency	Frequency ratio	Total Performance
(32bit)	$[mm^2]$	(RISC/MISC)	[MHz]	(MISC/RISC)	
MISC Adder	0.27	59.1	181.89	45.23	2672.2
MISC Subtractor	0.27	59.1	181.25	45.07	2662.7
MISC Multiplier	5.33	3.0	43.12	10.72	32.1
MISC Divider	9.05	1.8	4.23	1.05	1.9
MISC AND	0.11	145.0	795.22	197.76	28675.4
MISC OR	0.11	145.0	795.22	197.76	28675.4
MISC EXOR	0.11	145.0	795.22	197.76	28675.4
MISC NOT	0.11	145.0	853.67	212.30	30783.0
MISC Barrel Shifter(Left,Zero)	1.51	10.6	359.27	89.35	943.8
MISC Barrel Shifter(Left,sign)	1.40	11.4	368.35	91.60	1043.6
MISC Barrel Shifter(Right,Zero)	1.56	10.2	374.24	93.07	951.6
MISC Barrel Shifter(Right, sign)	1.56	10.2	354.94	88.27	902.5
RISC ALU	15.95	1.0	4.02	1.00	1.0

# FFT implementation



We have implemented a **16-point Fast Fourier Transform (FFT)** onto a 40 nm process technology using the MISC architecture. For this MISC implementation for the 16-point FFT, a **16-bit fixed point calculation** was applied. The number of digits after the decimal point is 8 bits.

#### 8-step FFT operation using MISCs



11:20 AM – 12:00 AM: New/Exploratory paradigms , 25 February, WPPP2018

# Performance comparison of 16 point FFT in each part by MISC implementation

Instruction	Area	Number	Operating	Processing
	$[mm^2]$	FFT/	Frequency	Time $[\mu s]$
		MISC	[MHz]	
MISC-FFT-1	3.40	11.33	245.78	0.0041
MISC-FFT-2	16.60	2.32	73.21	0.0137
MISC-FFT-3	3.40	11.33	265.12	0.0038
MISC-FFT-4	7.65	5.04	75.87	0.0132
MISC-FFT-5	3.40	11.33	271.82	0.0037
MISC-FFT-6	1.99	19.36	296.93	0.0034
MISC-FFT-7	3.40	11.33	221.86	0.0045
MISC-FFT-8	2.05	18.80	278.04	0.0036
Total		2.32		0.0498
Full hardware FFT	38.53	1.00	42.29	0.0473
Corei7-4790			3600.00	225.0000

## Conclusion

- ORGA architecture allows a very high-speed dynamic reconfiguration
- By using the capability, we can implement MISC processors inside the programmable gate array
- Its operation on a programmable gate array can be accelerated to about 2-3,000 times faster than static implementations
- The total performance of the MISC FFT was estimated as about twice performance compared with a full-hardware implementation without any reconfiguration