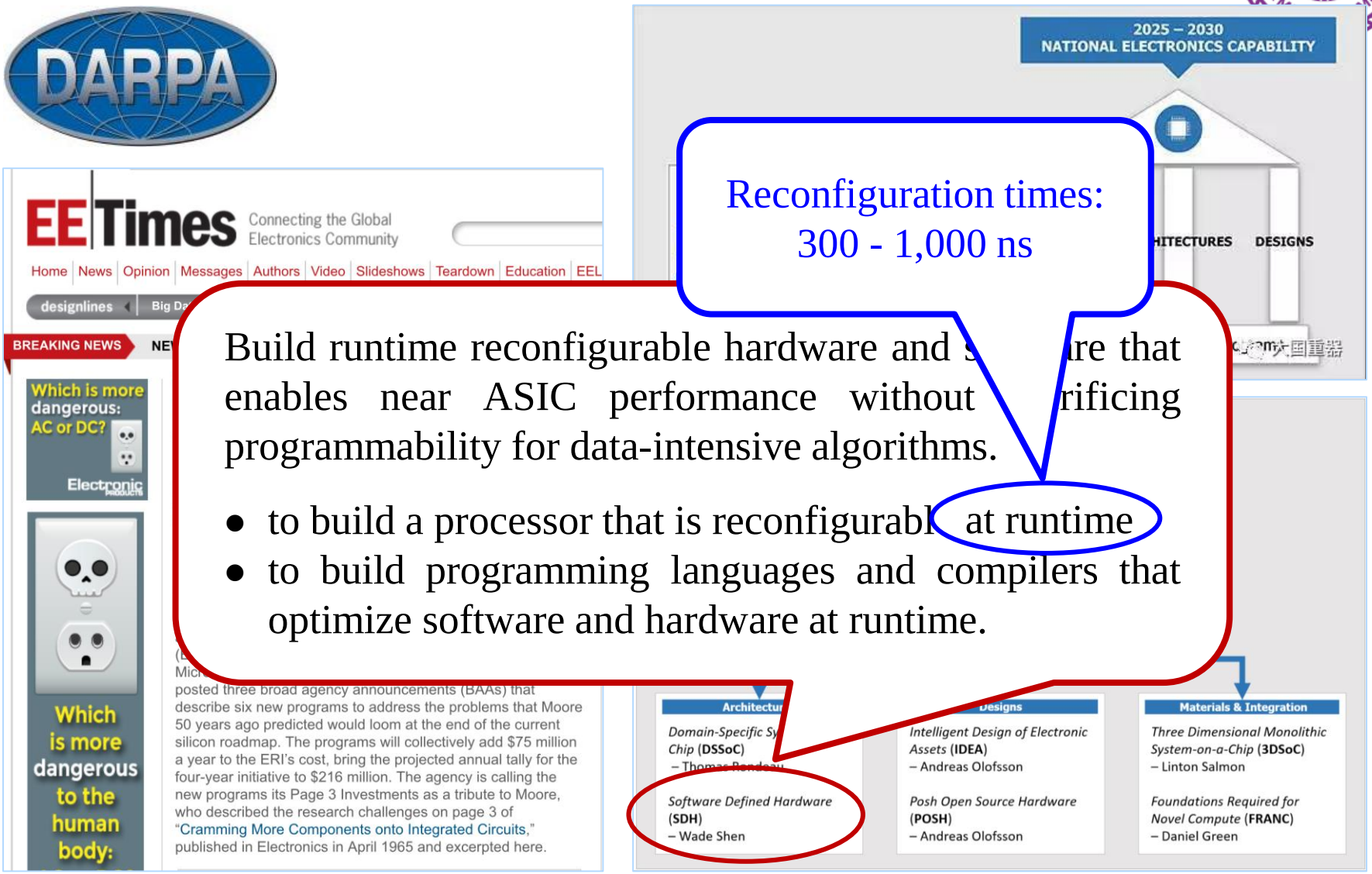
Reconfigurable Computing and Hardware Security

Dr. Shaojun WEI, Professor
Dean, Department of Microelecronics and Nanoelectronics
Tsinghua University, China
October 19, 2017

Software defined chip：Reconfigurability



Reconfiguration times:
300 - 1,000 ns

Build runtime reconfigurable hardware and s ure that enables near ASIC performance without rificing programmability for data-intensive algorithms.

- to build a processor that is reconfigurabl at runtime
- to build programming languages and compilers that optimize software and hardware at runtime.

Outline

1. Background and Motivation
2. Reconfigurable Computing: Architecture
3. Technologies and Challenges
4. Opportunities and Applications
5. Reconfigurability and Hardware Security
6. Summary

Outline

1. **Background and Motivation**
2. Reconfigurable Computing: Architecture
3. Technologies and Challenges
4. Opportunities and Applications
5. Reconfigurability and Hardware Security
6. Summary

Integrated circuits becoming expensive

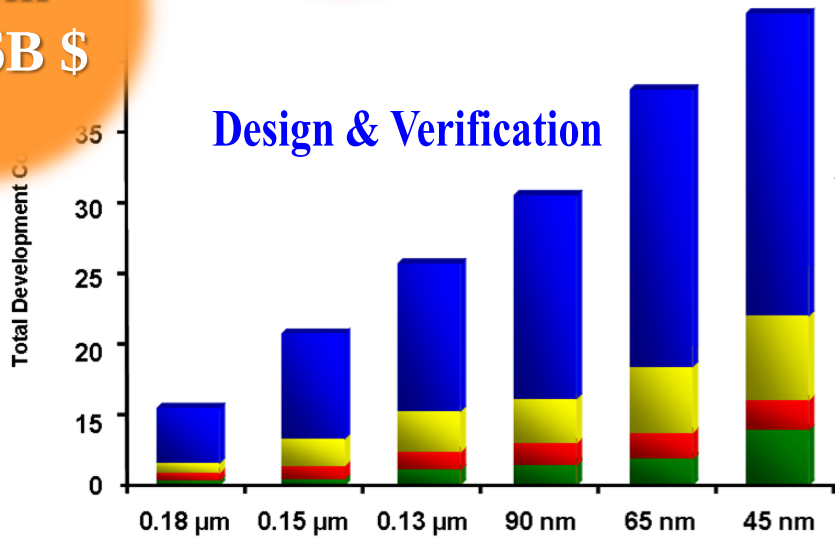
16nm
～8-10B $

22nm
～6-8B $

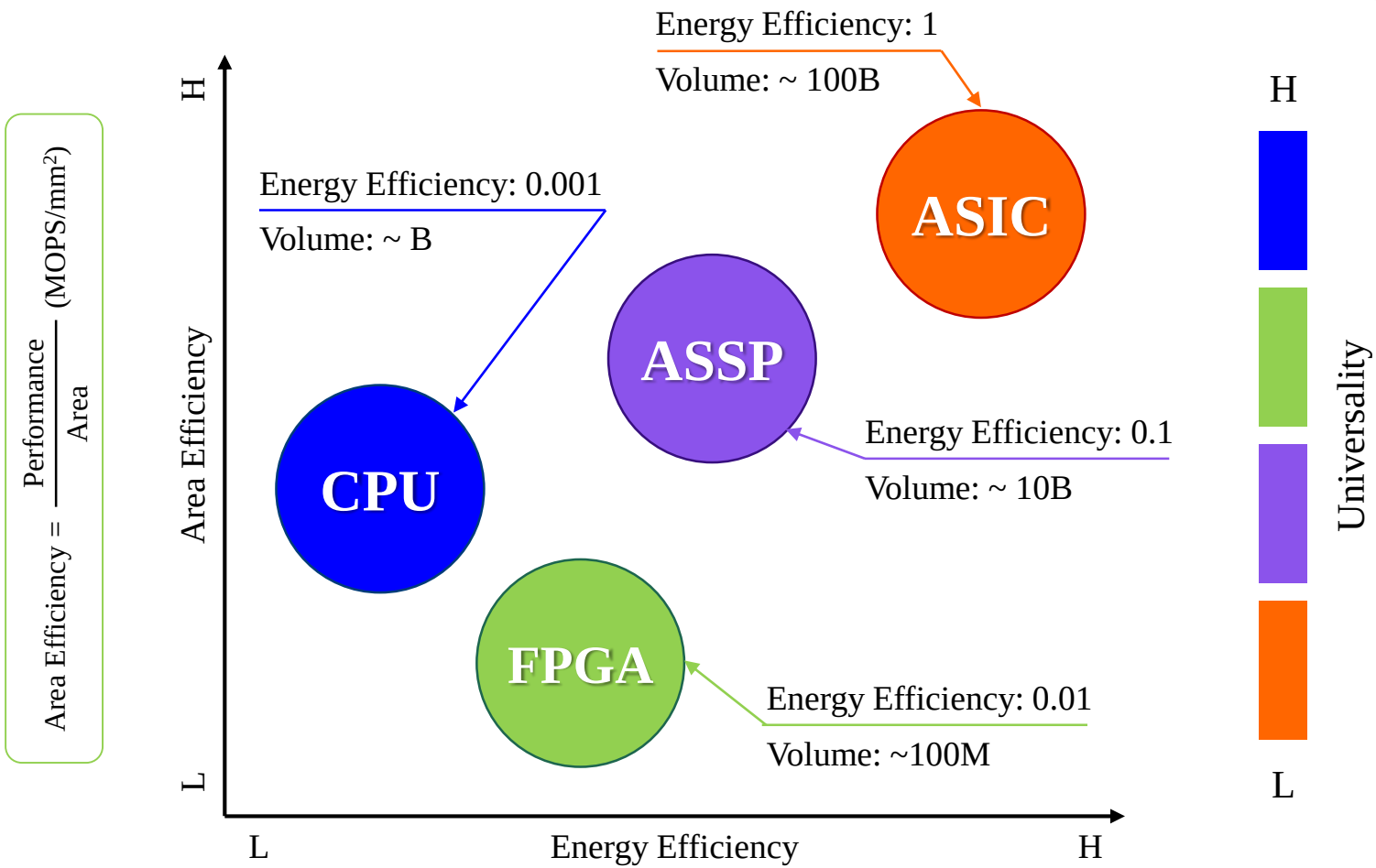
32nm
～4-6B $

45nm
～3.5-5B $

65nm
～2.5-3B $



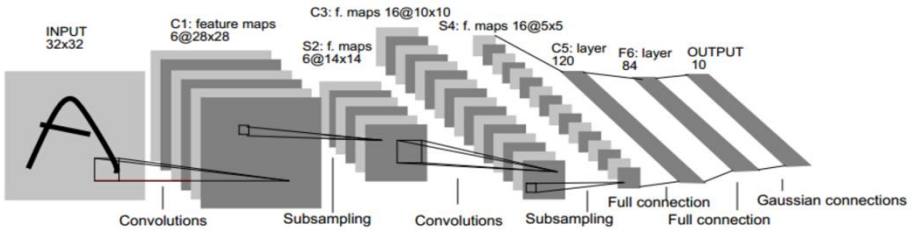
Energy efficiency & volume



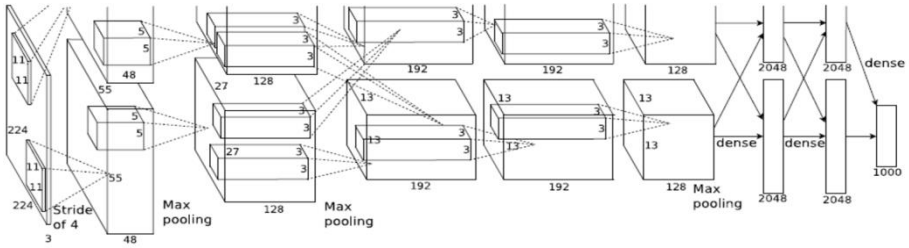
$$\text{Energy Efficiency} = \frac{\text{Performance}}{\text{Power}} \text{ (MOPS/mW)}$$

Deep learning needs new computing device

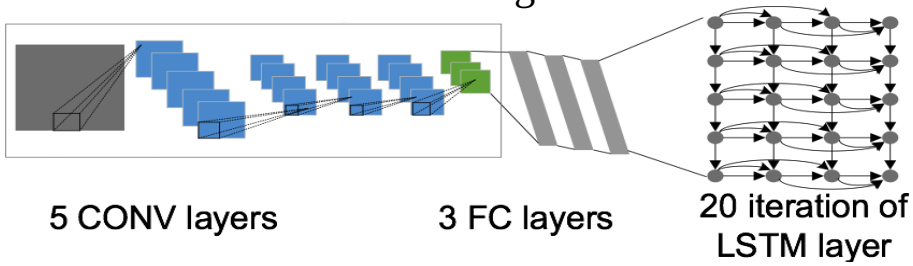
LeNet for handwriting recognition



AlexNet for image classification



LRCN for video understanding



Different AI applications use various type of neural networks.

- Number of neurons
- Topology of network
- Kind of interconnection

Require highly energy efficient and flexible computing chip (~TOPS/W)

Can FPGA replace ASIC?

Typical features of FPGA device:

No.	Features	Evaluation	Explanation
1	Grain	Fine	bit level, not suitable for computing
2	Configuration context	Large	megabyte ~ dozens of megabyte
3	Configuration time	Long	ms ~ dozens of ms
4	Configuration	Static	power down or stop running
5	Logic multiplexing	No	complete circuit
6	Area efficiency	Low	10M gates FPGA for hundreds gates circuits
7	Energy efficiency	Low	power consumption for configuration logic
8	Process technology	High	state-of-art
9	Knowledge required	Circuit design	circuit design engineer
10	Cost	High	Expensive

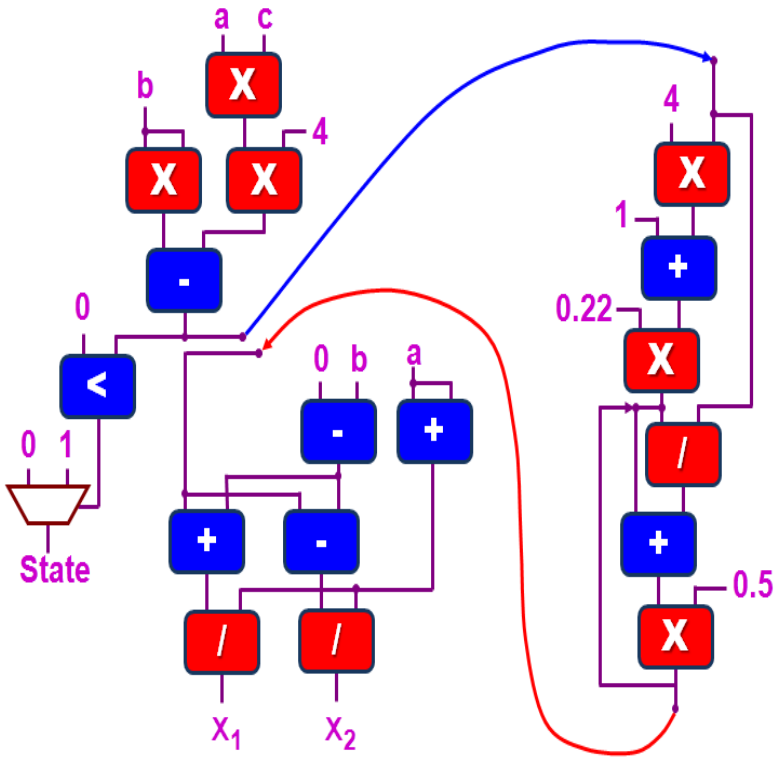
Ideal computing architecture

Software

```
r1 := b * b;
r2 := a * c;
r2 := r2 * 4;
r1 := r1 - r2;
IF (r1 < 0) THEN
   state <=  '1' ;
ELSE
   state <=  '0' ;
   SQR(r1);
   r1 := 0 - b;
   r2 := a + a;
   r4 := r1 + r3;
   x1 <= r4 / r2;
   r5 := r1 - r3;
   x2 <= r5 / r2;
END IF;
```

```
SQR(r1) is
BEGIN
   r3 := r1 * 4;
   r3 := r3 + 1;
   r3 := r3 * 0.2;
   FOR i = 1 TO 3 DO
      r4 := r1 / r3;
      r3 := r3 + r4;
      r3 := r3 / 2;
   END DO;
END;
```

Hardware



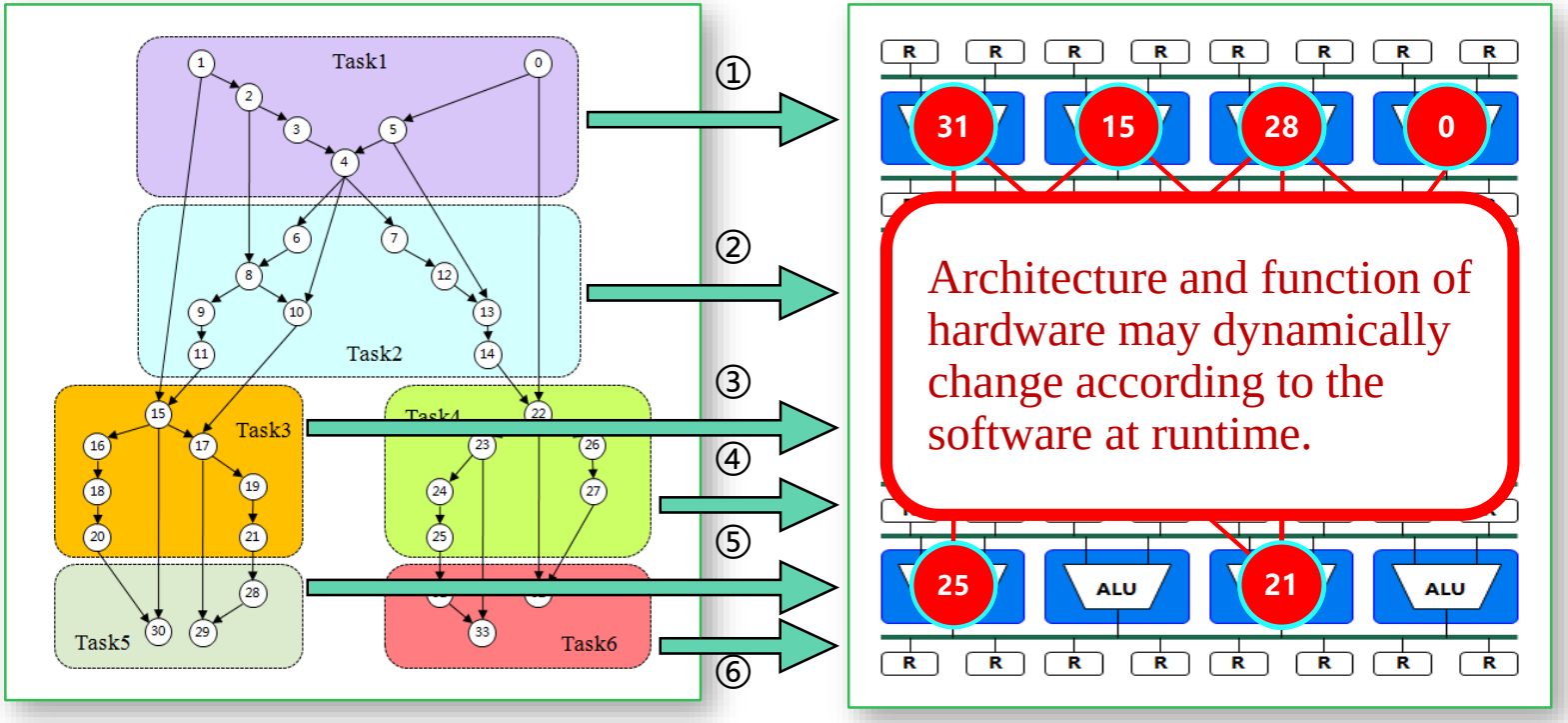
Without considering resource constrain, an ideal computing hardware should contain all the computing elements that corresponding to the operations in software and has the same topology with software.

Concept of reconfigurable computing

Software may be very large but hardware is always limited. So

(1) Software should be partitioned into modules and executed one by one;

(2) Hardware should be able to dynamically change in order to adapt the modules.



Task partitioning, dependency analysis and task graph generation

Reconfigurable PE array that meets the requirement of performance and cost

Outline

1. Background and Motivation
2. **Reconfigurable Computing: Architecture**
3. Technologies and Challenges
4. Opportunities and Applications
5. Reconfigurability and Hardware Security
6. Summary

Control and computing implied in software

```
IF (condition = 1) THEN
   r1 = b * b;
   r2 = a * c;
   r2 = r2 * 4;
   r1 = r1 - r2;
   IF (r1 < 0) THEN
      state = '1';
   ELSE
      state = '0';
      r3 = r1 * 4;
      r3 = r3 + 1;
      r3 = r3 * 0.2;
      FOR i = 1 to 3 LOOP
         r4 = r1 / r3;
         r3 = r3 + r4;
         r3 = r3 / 2;
      END LOOP;
      r1 = 0 - b;
      r2 = a + a;
      r4 = r1 + r3;
      x1 = r4 / r2;
      r5 = r1 - r3;
      x2 = r5 / r2;
   END IF;
END IF;
```

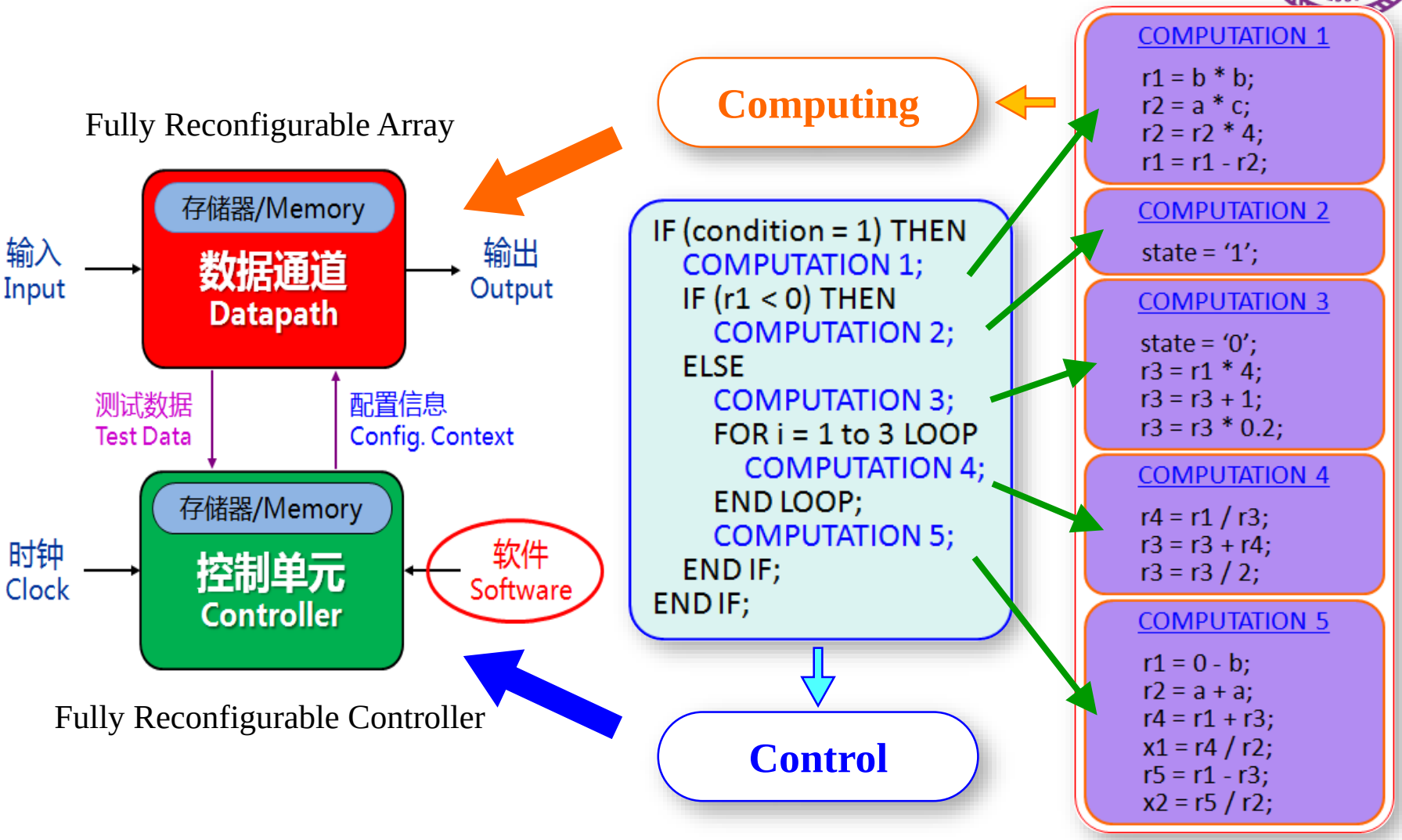
Computing

```
IF (condition = 1) THEN
   COMPUTATION 1;
   IF (r1 < 0) THEN
      COMPUTATION 2;
   ELSE
      COMPUTATION 3;
      FOR i = 1 to 3 LOOP
         COMPUTATION 4;
      END LOOP;
      COMPUTATION 5;
   END IF;
END IF;
```

Control

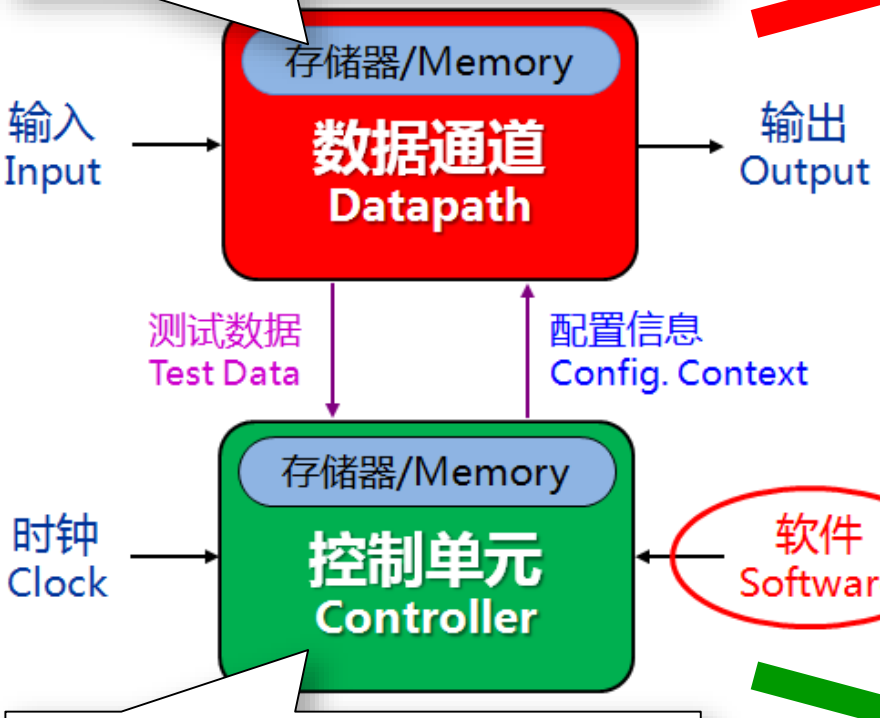
```
COMPUTATION 1
r1 = b * b;
r2 = a * c;
r2 = r2 * 4;
r1 = r1 - r2;

COMPUTATION 2
state = '1';

COMPUTATION 3
state = '0';
r3 = r1 * 4;
r3 = r3 + 1;
r3 = r3 * 0.2;

COMPUTATION 4
r4 = r1 / r3;
r3 = r3 + r4;
r3 = r3 / 2;

COMPUTATION 5
r1 = 0 - b;
r2 = a + a;
r4 = r1 + r3;
x1 = r4 / r2;
r5 = r1 - r3;
x2 = r5 / r2;
```

Mapping

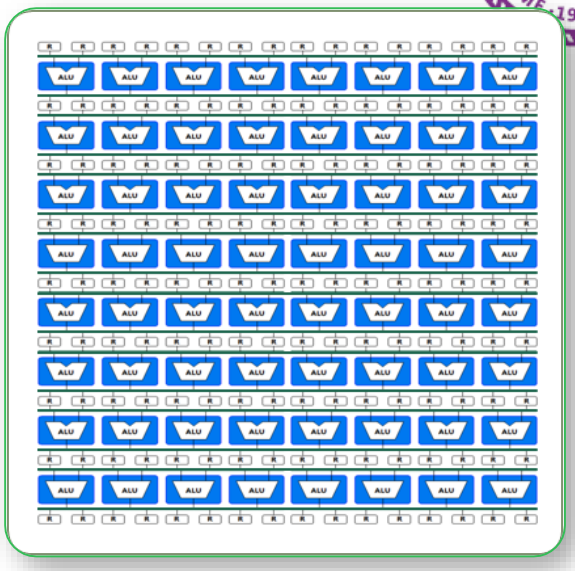

Fully Reconfigurable Array
存储器/Memory
数据通道
Datapath
输入
Input
输出
Output
测试数据
Test Data
配置信息
Config. Context
存储器/Memory
控制单元
Controller
时钟
Clock
软件
Software
Fully Reconfigurable Controller
Computing
Control
IF (condition = 1) THEN
COMPUTATION 1;
IF (r1 < 0) THEN
COMPUTATION 2;
ELSE
COMPUTATION 3;
FOR i = 1 to 3 LOOP
COMPUTATION 4;
END LOOP;
COMPUTATION 5;
END IF;
END IF;
COMPUTATION 1
r1 = b * b;
r2 = a * c;
r2 = r2 * 4;
r1 = r1 - r2;
COMPUTATION 2
state = '1';
COMPUTATION 3
state = '0';
r3 = r1 * 4;
r3 = r3 + 1;
r3 = r3 * 0.2;
COMPUTATION 4
r4 = r1 / r3;
r3 = r3 + r4;
r3 = r3 / 2;
COMPUTATION 5
r1 = 0 - b;
r2 = a + a;
r4 = r1 + r3;
x1 = r4 / r2;
r5 = r1 - r3;
x2 = r5 / r2;

Reconfigurable architecture

Configuring and execution under the control of Controller.

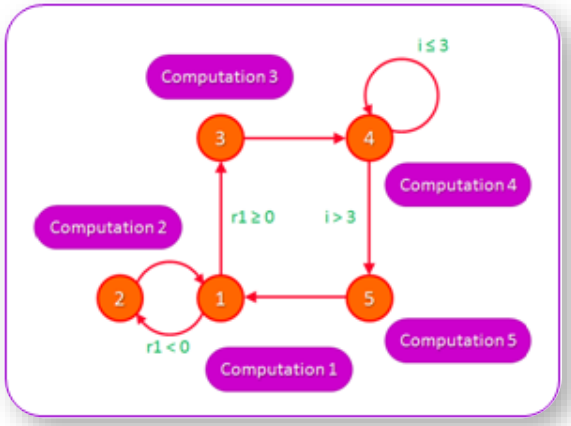


Sending the modules to Datapath according to partitioned software



Fully reconfigurable Datapath

Fully reconfigurable Controller



Reconfigurable computing system

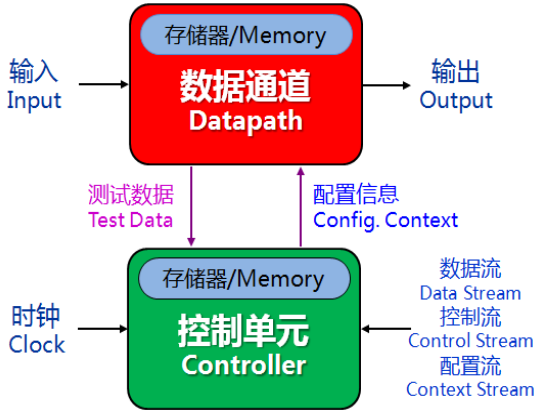
C Program

```
Int main(void)
{
......
func(..., ...)
......
dct(..., ,,,)
......
......
}
```

```
func(..., ...)
{
......
}
```

```
dct(..., ...)
{
......
}
```

Software defined Chip



Data Stream
Control Stream
Context Stream

Compiler

- Syntax Check
- Code Analysis
- Code Transformation
- Code Optimization
- DFG Generation
- Task Partitioning
- Task Scheduling
- Resource Allocation
- Interconnection Generation
- Mapping
- Evaluation
- Context Generation

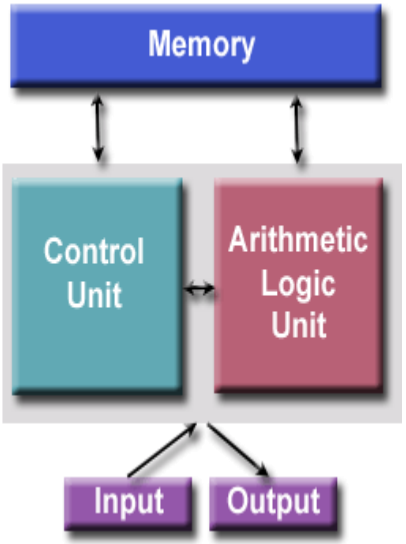
Von Neumann Architecture

The Von Neumann architecture is a design model for a stored-program digital computer that uses a central processing unit (CPU) and a single separate storage structure (memory) to hold both instructions and data.

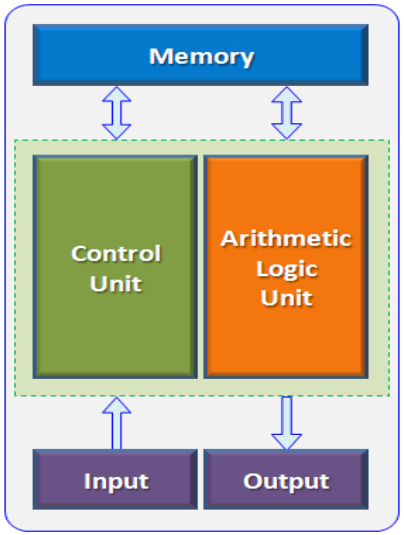
Characteristics of the Von Neumann's architecture:

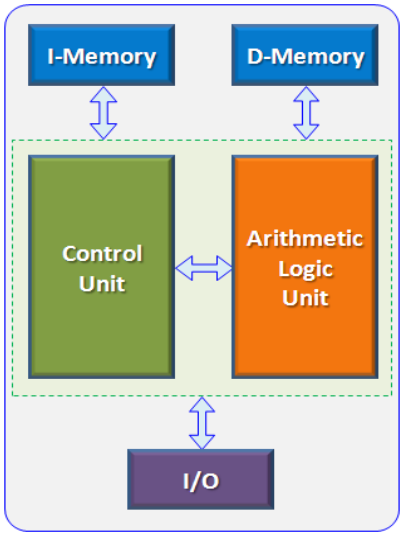
a) memory;
b) control unit;
c) arithmetic logic unit;
d) input / output interface.

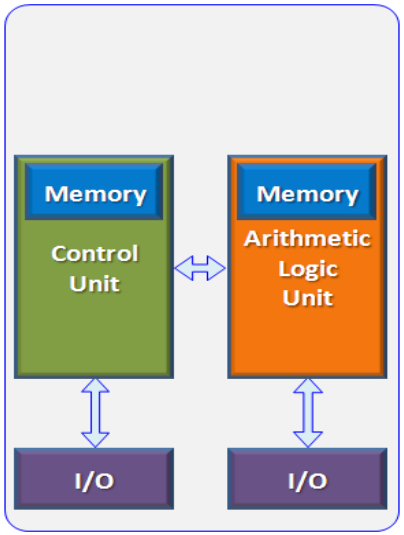
The disadvantage of Von Neumann architecture: shared memory for instructions and data with one data bus and one address bus between processor and memory. Instructions and data have to be fetched in sequential order (known as the Von Neumann Bottleneck), limiting the operation bandwidth.

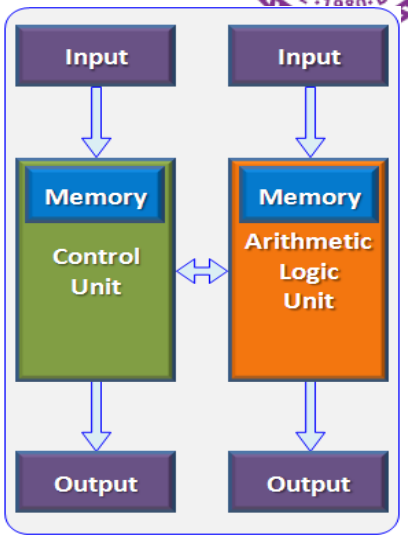


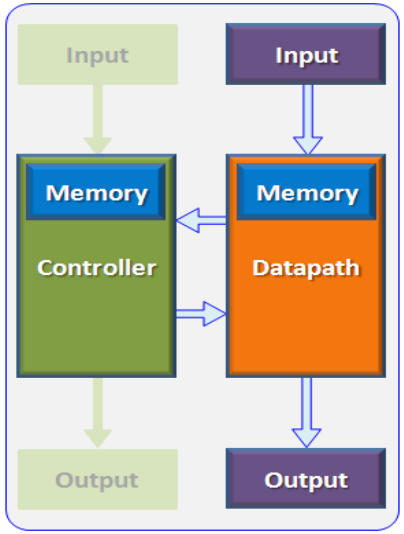
Transformation of Von Neumann Architecture

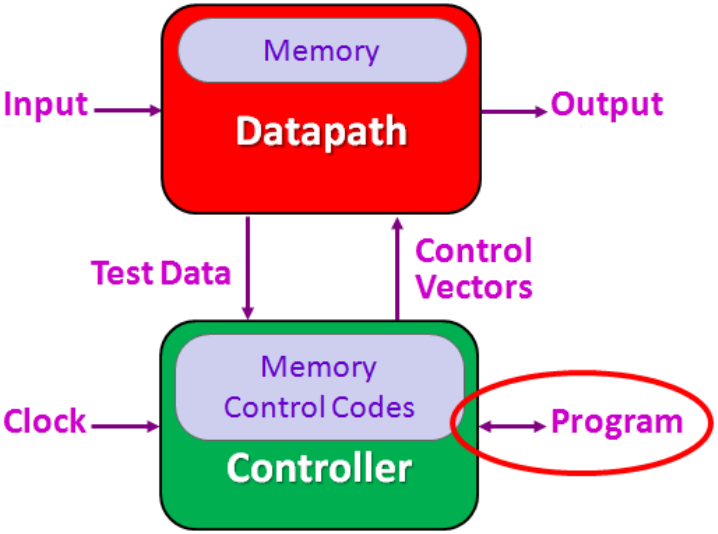




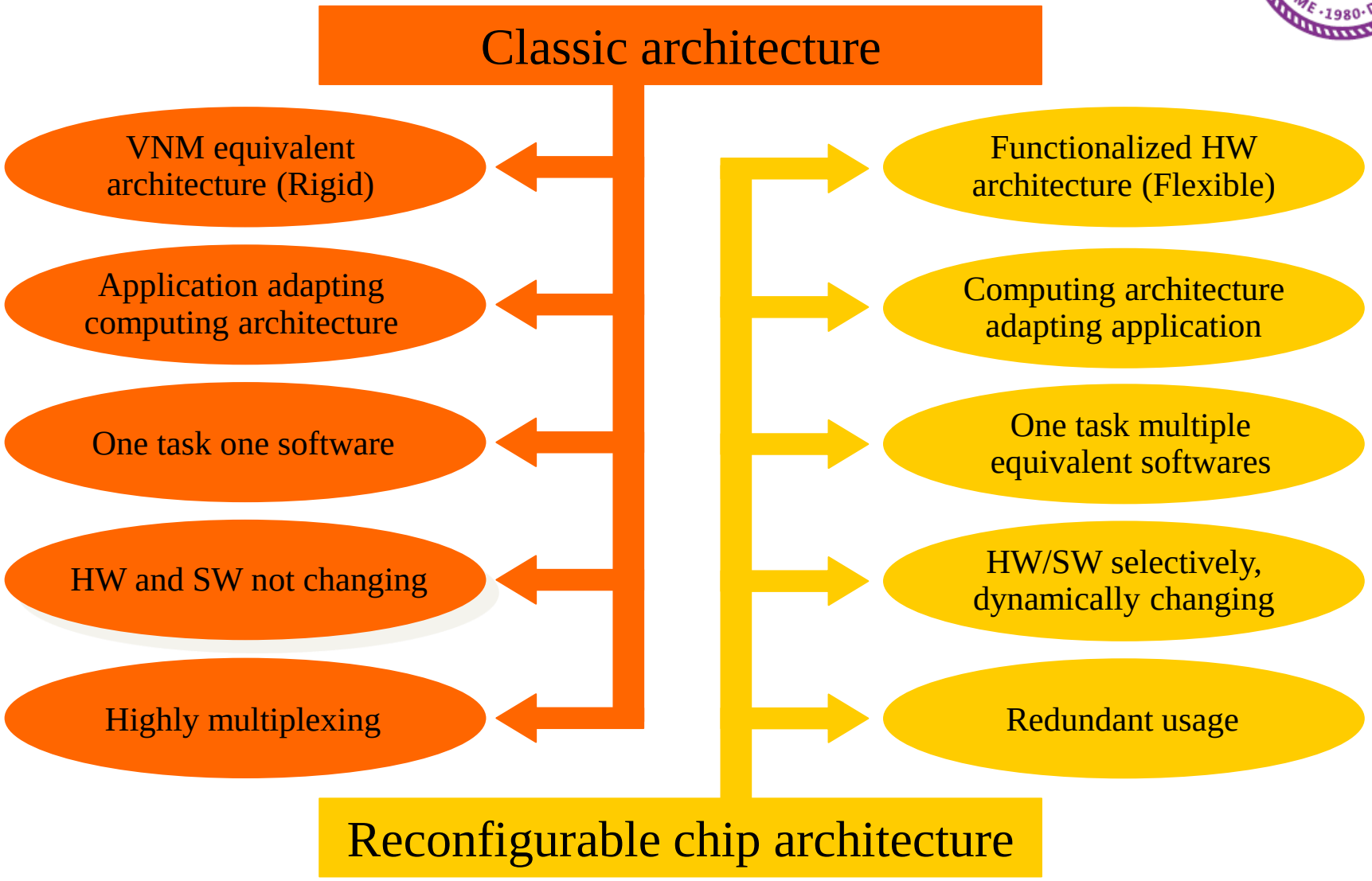








Comparison of RC to classic architecture

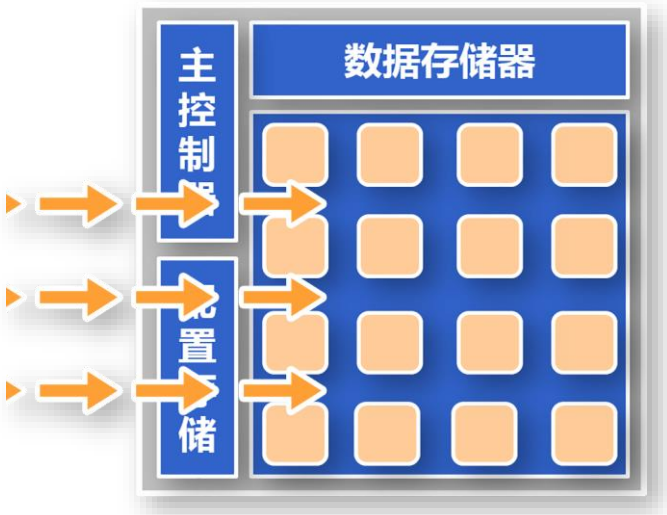


Outline

1. Background and Motivation
2. Reconfigurable Computing: Architecture
3. **Technologies and Challenges**
4. Opportunities and Applications
5. Reconfigurability and Hardware Security
6. Summary

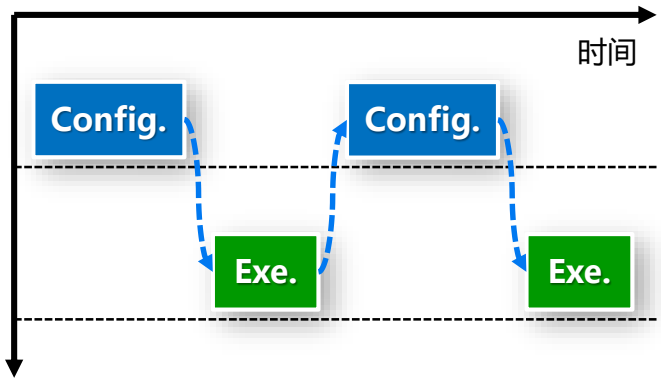
Computing mode

- Under “Dynamic configuration vs. Data Driven” mode, “Configuration – Execution” alternating raises a difficulty to store and load large configuration context.
- Configuration takes most time, PE array has low coefficient of utilization.



Configuration takes most time

Computing: Configuration + Execution.

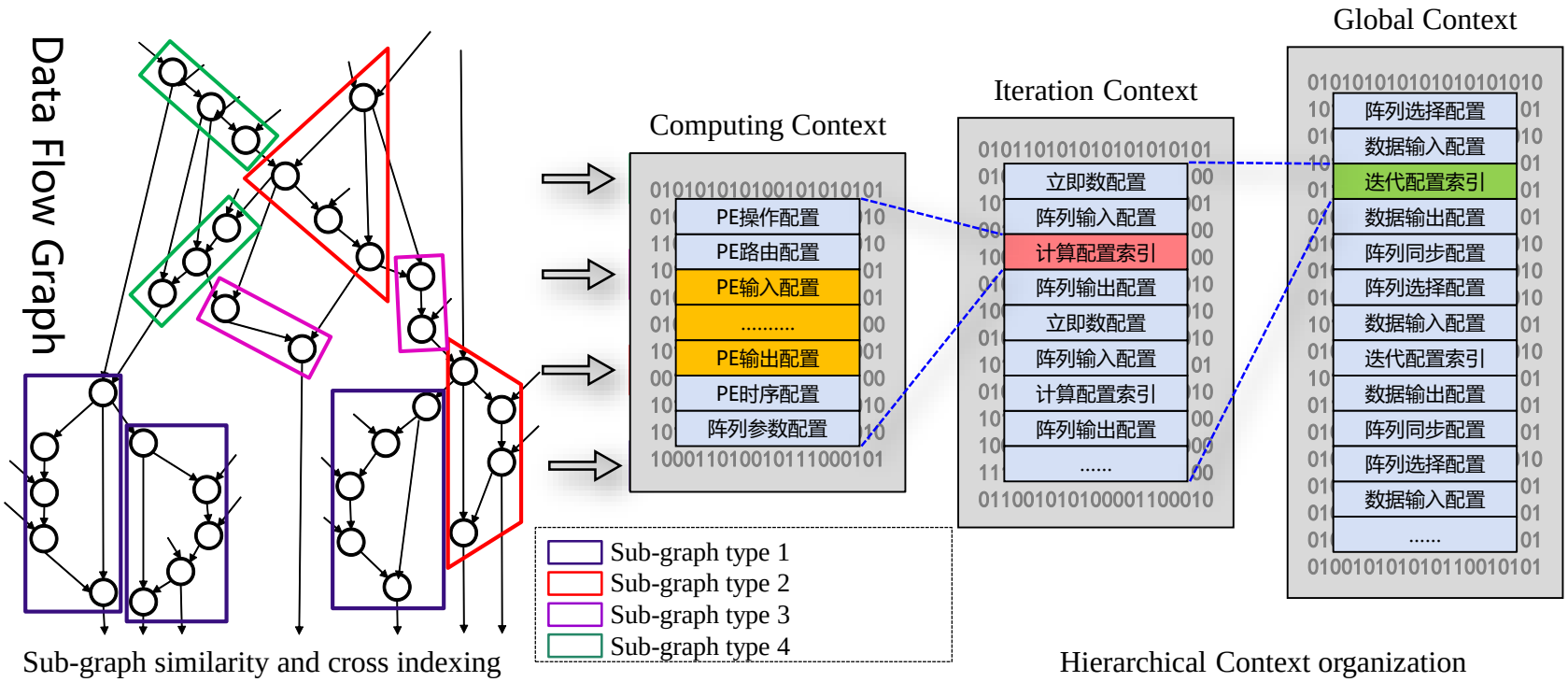


“Configuration-Execution” alternating

- Reducing configuration context by using hierarchical, re-organization and compress strategy, developing a high efficient context loading mechanism and highly optimized on-chip memory architecture.

Computing mode

- Hierarchical context generation based on sub-graph similarity, cross indexing.
- Hierarchical Configuration Context (HCC) reduces 77% of the amount of context [1].

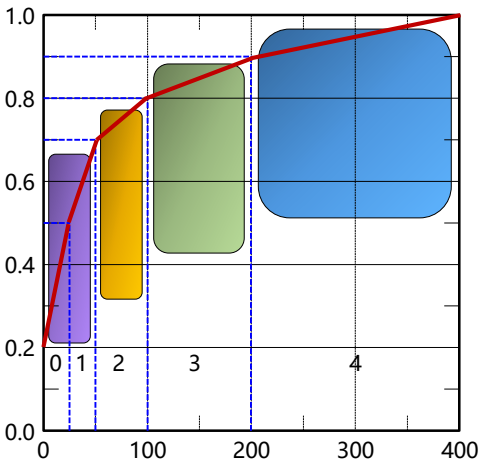


Patents: filed 18, granted 9, ZL200910090401, ZL201110099347, ZL200910090403, ….
Papers: 9 (SCI: 7), IEEE TVLSI, FPGA, …

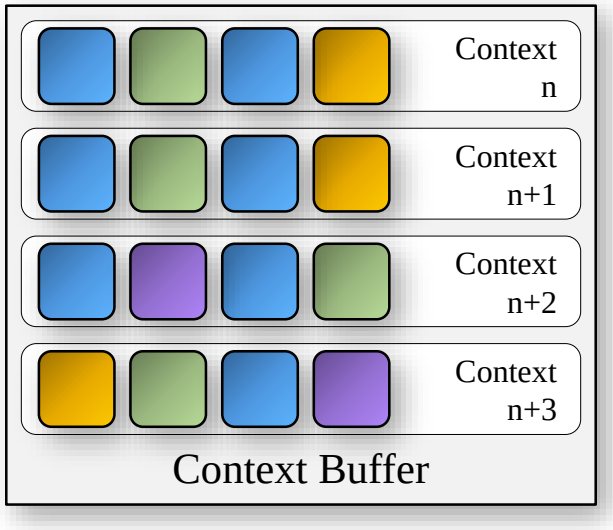
[1] Yansheng Wang, Leibo Liu, Shaojun Wei, et al. "On-Chip Memory Hierarchy in One Coarse-Grained Reconfigurable Architecture to Compress Memory Space and to Reduce Reconfiguration Time and Data-Reference Time," ***IEEE Transactions on Very Large Scale Integration Systems***, vol. 22, no. 5, pp. 983-994, 2014

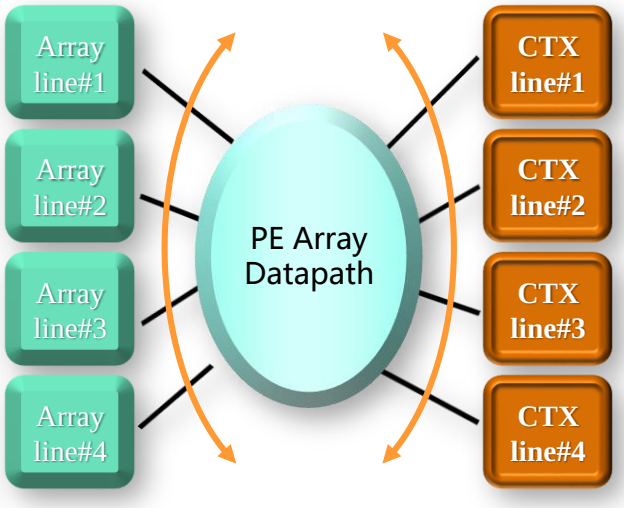
Computing mode

- Frequentness prioritized context buffering strategy and dynamic pendulum-style context loading mechanism.
- Context reading and loading speed has been accelerated 12X [1].



Frequentness prioritized context buffering Strategy





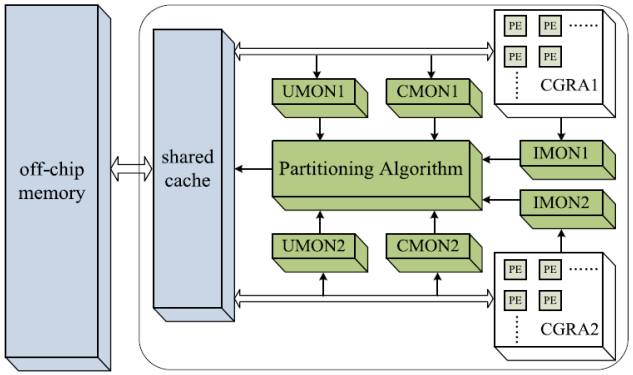
Dynamic pendulum-style context loading mechanism

Patents: filed 15, granted 5, ZL201110140942, ZL201110140266, ZL201110159626, ….
Papers: 5 (SCI: 3), IEEE TVLSI、CICC/FPGA

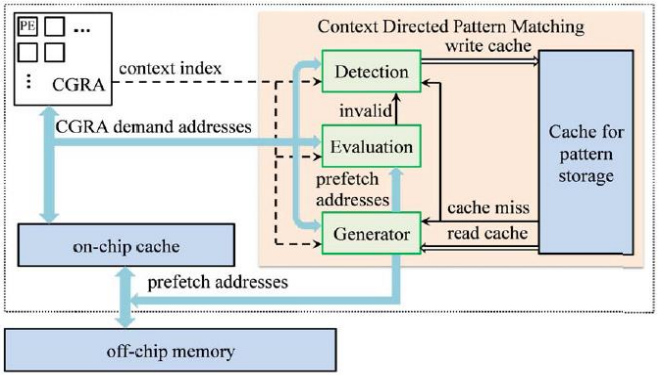
[1] L.B. Liu, S.J. Wei et al., Cost-Effective Memory Architecture … ACM Symposium on FPGA, 2015

Computing mode

- Optimization of on-chip memory is so important that it impacts not only the chip size and cost, but also the performance and utilization of PE array.
- Thanks to HCC and LDO, the normalized on-chip data memory size is reduced up to 85%[1]. In addition, CIACP outperformed the state-of-the-art utility-based cache partitioning techniques by 16% in performance[2]. CDP averagely improved the performance by 31.1% compared to tests without any prefetching and by 7.7% compared to state-of-the-art prefetching techniques [3]



(a) Hardware framework for Correlation- and Iteration- Aware Cache Partitioning (CIACP)



(b) Framework for Context Directed Pattern Matching (CDPM)

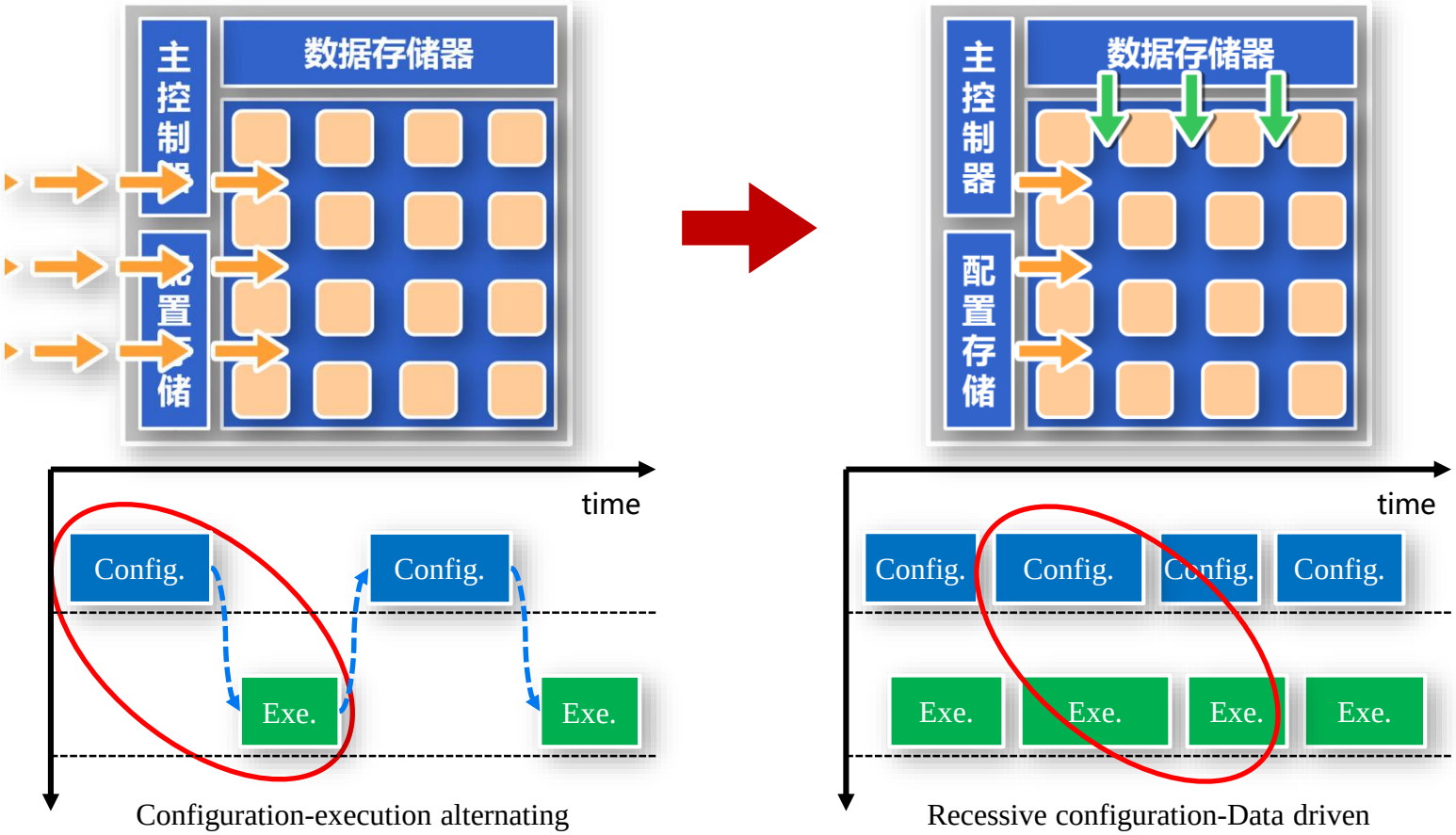
[1] Yansheng Wang, Leibo Liu, Shaojun Wei, et al. "On-Chip Memory Hierarchy in One Coarse-Grained Reconfigurable Architecture to Compress Memory Space and to Reduce Reconfiguration Time and Data-Reference Time," ***IEEE Transactions on Very Large Scale Integration Systems***, vol. 22, no. 5, pp. 983-994, 2014

[2] Chen Yang, Leibo Liu, Shaojun Wei, et al. "CIACP: A Correlation- and Iteration- Aware Cache Partitioning Mechanism to Improve Performance of Multiple Coarse-Grained Reconfigurable Arrays," ***IEEE Transactions on Parallel and Distributed Systems,*** vol. 28, no. 1, pp. 29-43, 2017.

[3] Chen Yang, Leibo Liu, Shaojun Wei, et al. "Data cache prefetching via context directed pattern matching for coarse-grained reconfigurable arrays," in Proceedings of the 2016 53rd Annual ***Design Automation Conference on - DAC '16***, Austin, TX, United States, Aug. 2016, pp. 1–6.

Computing mode

Parallelizing the configuration and execution by "hiding" the configuration into task execution.
The coefficient of utilization of PE array is improved from 25% to 87% [1][2].



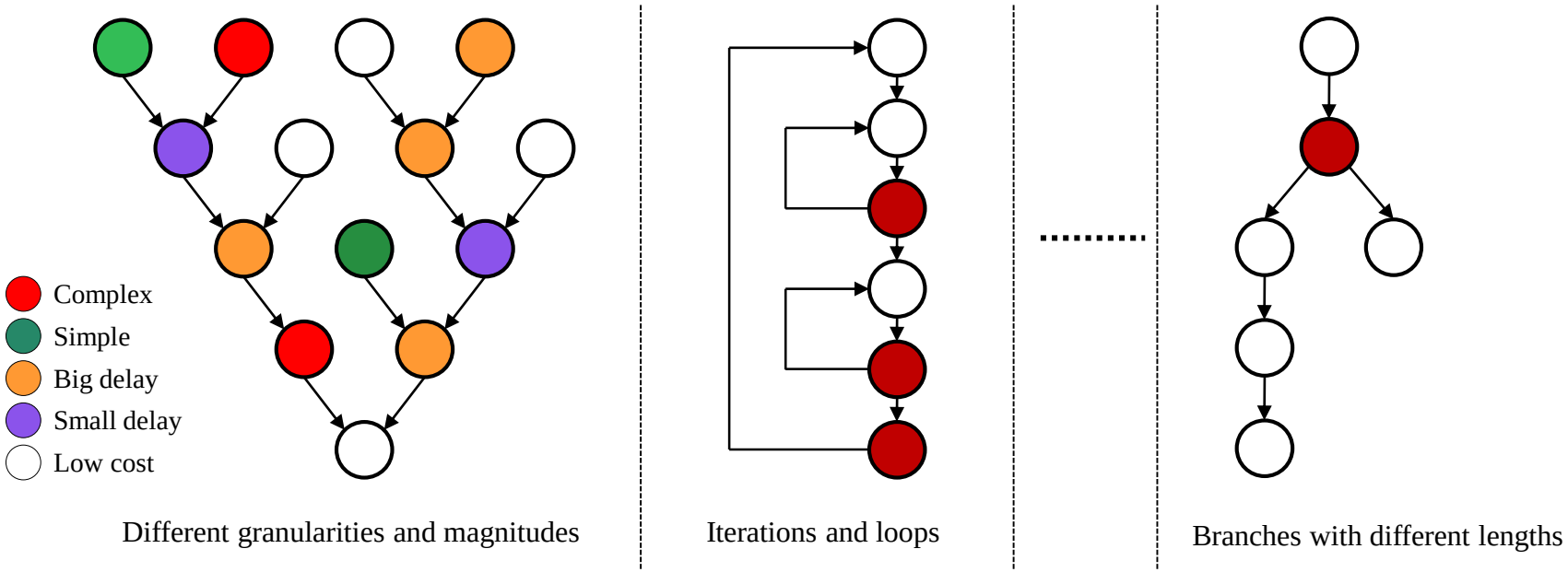
Configuration-execution alternating

Recessive configuration-Data driven

[1] Leibo Liu, ShaoJun Wei et al., An energy-efficient coarse-grained … IEEE Conference CICC, 2013
[2] Leibo Liu, ShaoJun Wei et al., SimRPU: A Simulation Environment for… ," ***IEEE Transactions on VLSI Systems***, 2013

Reconfigurable PE array

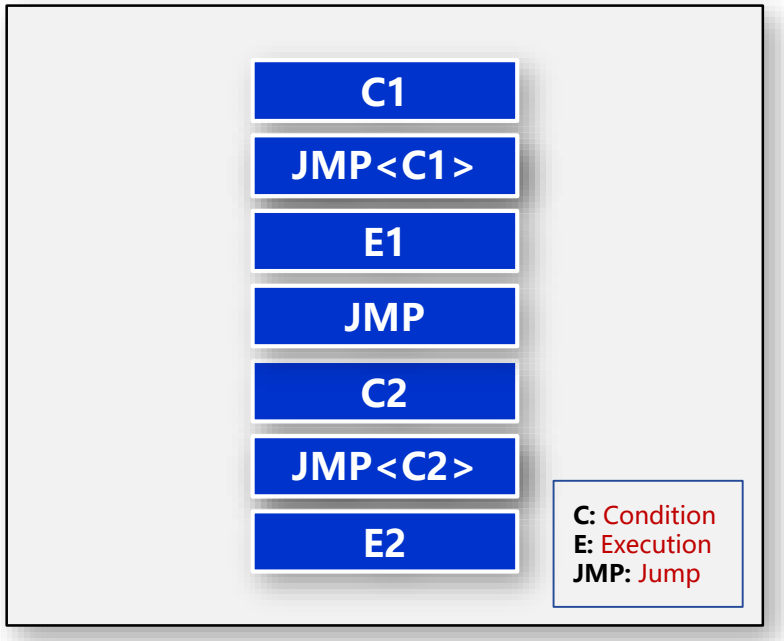
- Regular PE array and irregular task.
- Different applications involve different PEs and different interconnections. Frequently transforming PE array will lead to a low energy efficiency.



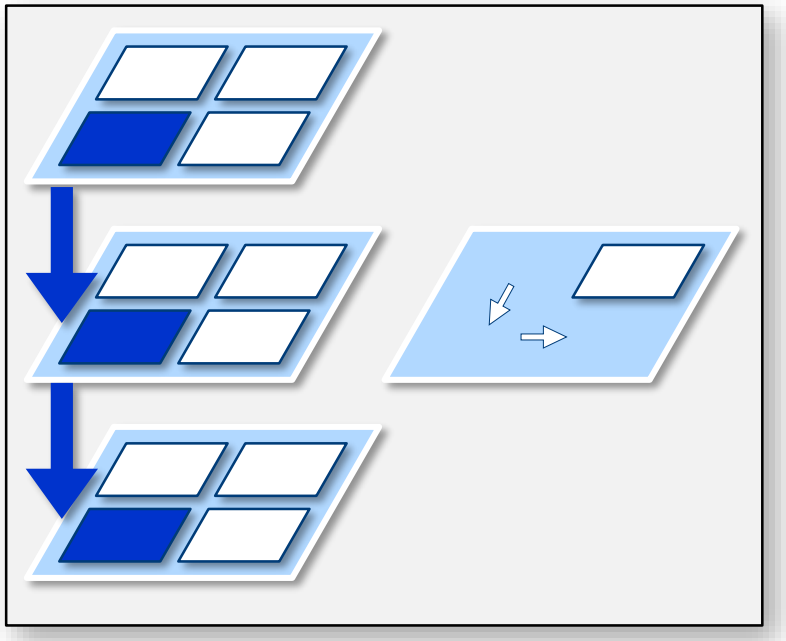
- Exploring task parallelization, optimizing control flow, developing a PE array with reconfigurable function, interconnection and driven voltage.

Reconfigurable PE array

- Parallelizing the execution of control intensive task on PE array and merging the contexts of execution and condition calculation.
- The performance of control intensive task has been improved 40%[1][2][3].



Combining execution body and condition calculation



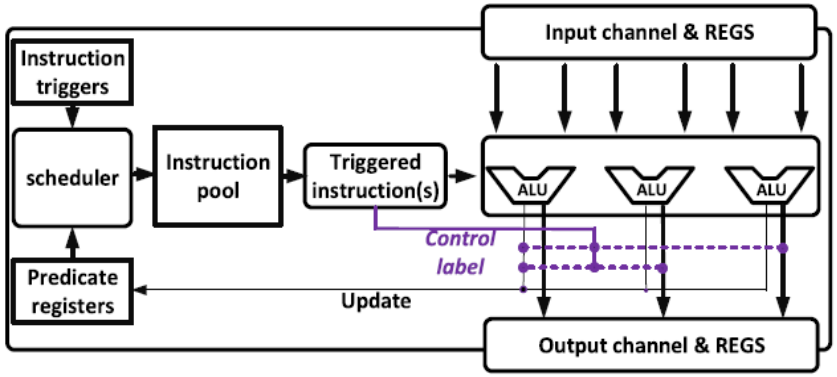
Context Merge

Patents: filed 10, granted 9, ZL201110152239, ZL201110159481, ZL201110140357, ….
Papers: 12 (SCI: 8), IEEE TVLSI, DAC, FPGA, …

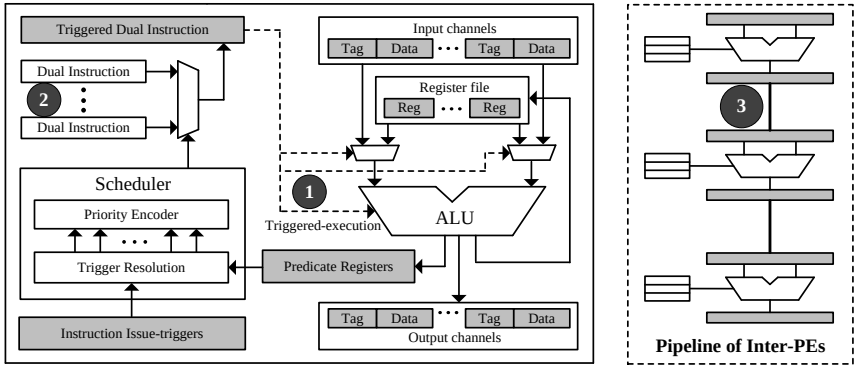
[1] Leibo Liu, Shaojun Wei et al., Reconfigurable…Control-Intensive ... ***IEEE Transactions on VLSI Systems***, 2014
[2] Leibo Liu, Shaojun Wei et al., Acceleration of Control Flows … Design Automation Conference, 2015
[3] Jianfeng Zhu, Leibo Liu, Shaojun Wei, et al. "A Hybrid Reconfigurable Architecture and Design Methods Aiming at Control-Intensive Kernels," ***IEEE Transactions on VLSI Systems***, vol. 23, no. 9, pp. 1700-1709, 2015.

Reconfigurable PE array

- Different control flows may lead to different results which impact the system performance.
- Optimization of parallel condition, configuration branch and compound configuration can improve the performance by over 40% compared with conventional techniques [1]. TLIA improves the performance by 20.9% to 140.0% over the equivalent Triggered Instruction Architecture (TIA) [2]. TITE improves the performance, energy efficiency and area efficiency by 21%, 17% and 12%, respectively, when compared with TIA[3].



(a) Triggered-Long-Instruction Architecture (TLIA)



(b) Triggered-Issue and Triggered-Execution (TITE)

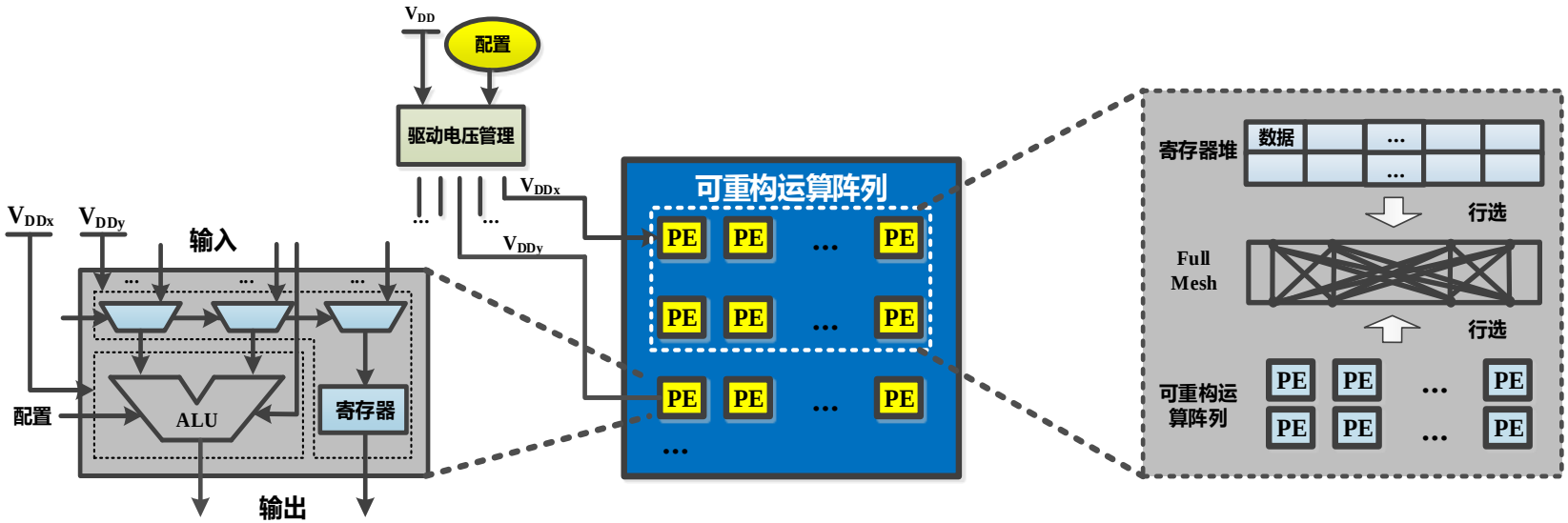
[1] Jianfeng Zhu, Leibo Liu, Shaojun Wei, et al. "A Hybrid Reconfigurable Architecture and Design Methods Aiming at Control-Intensive Kernels," ***IEEE Transactions on Very Large Scale Integration Systems,*** vol. 23, no. 9, pp. 1700-1709, 2015.

[2] Leibo Liu, Junbin Wang, Shaojun Wei, et al. "TLIA: Efficient Reconfigurable Architecture for Control-Intensive Kernels with Triggered-Long-Instructions," ***IEEE Transactions on Parallel and Distributed Systems***, vol. 27, no. 7, pp. 2143-2154, 2016.

[3] Yanan Lu, Leibo Liu, Shaojun Wei, et al. "Minimizing Pipeline Stalls in Distributed-Controlled Coarse-Grained Reconfigurable Arrays with Triggered Instruction Issue and Execution", to appear in the 54th Annual ***Design Automation Conference (DAC '17),*** June 18-22, 2017, Austin, TX, USA (to appear).

Reconfigurable PE array

- PE array architecture reconfigurable in operation, interconnection and voltage, driven voltage dynamically reconfigurable and Interconnection network based on line switching.
- Power consumption of PE array is reduced 30% [1].



Driven voltage dynamically reconfigurable

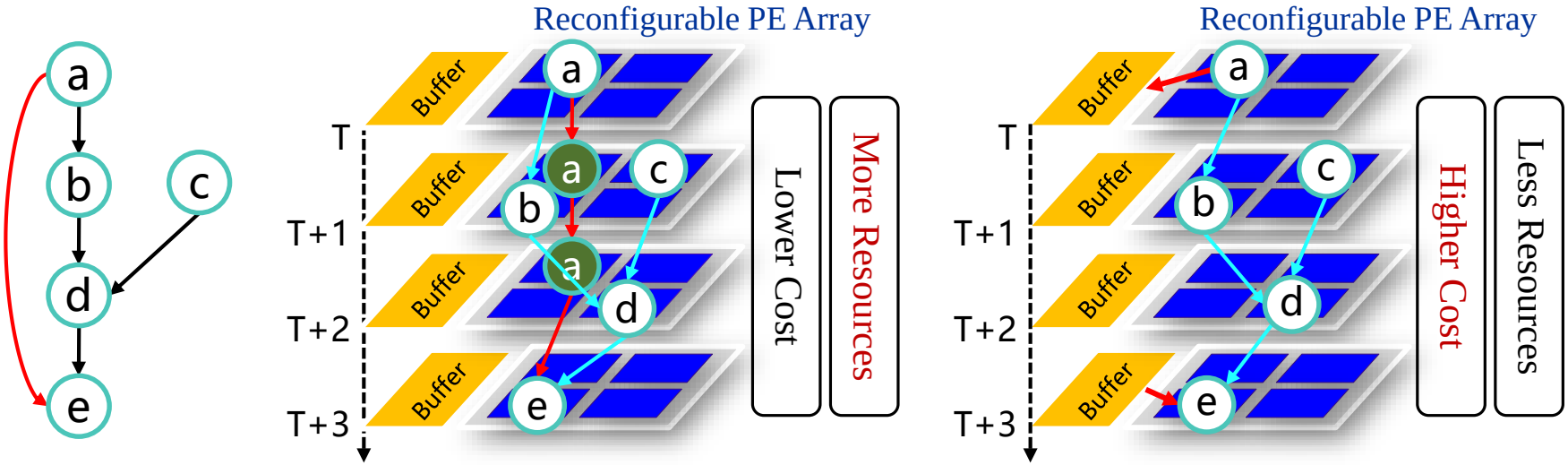
Interconnection network based on line switching

Patents: filed 8, granted 5, ZL201110140356, ZL201110140953, ZL201110152238, ….
Papers: 6 (SCI: 5), IEEE TVLSI/TED/TPDS, FPGA, …

[1] Leibo Liu, Shaojun Wei et al., Low-Power Reconfigurable Processor… ***IEEE Transactions on CAS-II***, 2013

Multi-parameters mapping

- Mutually coupling and restricting among multi-parameters such as PE array reconfiguration, computing, routing and buffering etc. deeply impact the system performance and energy consumption.
- It is hard to have a optimized mapping.

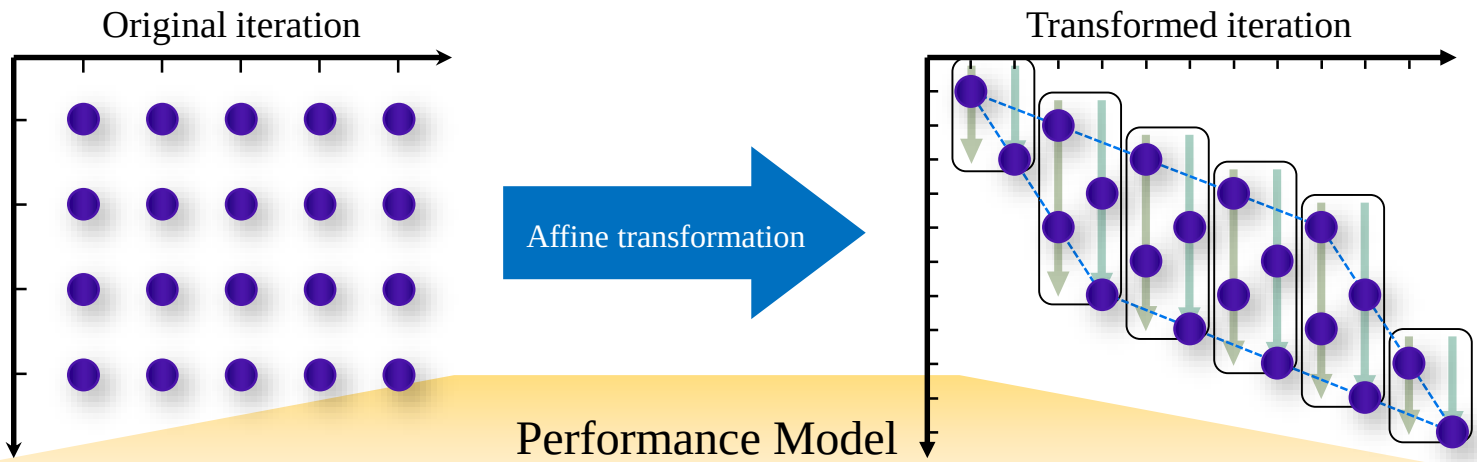


Original DFG Route prior mapping Buffer prior mapping

- Performance and energy oriented multi-parameters modeling, combining with code transformation in order to find optimal solution, and exploring the innovative programming model.

Multi-parameters mapping

- Performance oriented polyhedral modeling, mapping and affine transformation.
- Task execution time is reduced 20% [1][2].



Performance Model

$$TET = \frac{1}{f} N_{cfg} \cdot CFC + \frac{1}{f} N_{tp} \cdot CPC + \frac{2}{f \cdot BW}(N_{rp} + \epsilon N_{ip}) \sum_{e \in E} (\eta \delta_e(\Theta) + \zeta \delta_e(\Pi) - \delta_e(\Theta)\delta_e(\Pi))$$

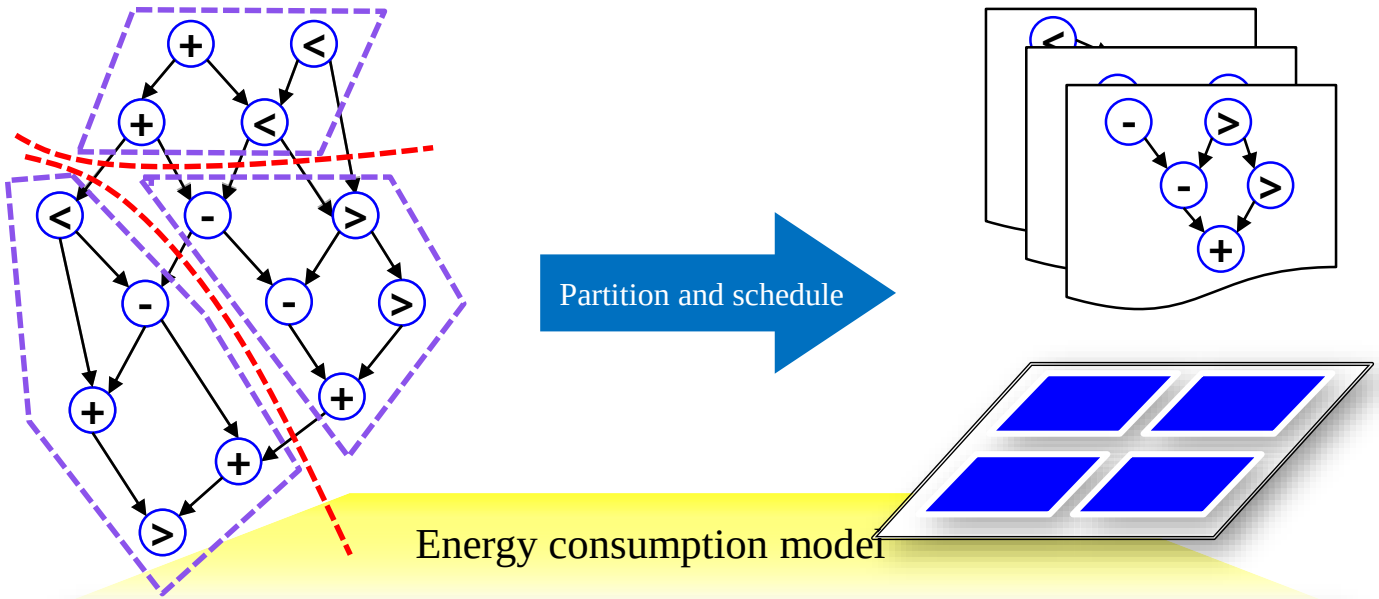
Exe. time | No of config. | No of calculation | Mem. bandwidth | Data through-put

Patents: filed 12, granted 10, ZL201010593805, ZL201110099347, ZL201010525777, ….
Papers: 13 (SCI: 9), IEEE TVLSI, DAC, …

[1] Shouyi Yin, Shaojun Wei et al., Polyhedral model based mapping… Design Automation Conference, 2013
[2] Shouyi Yin, Shaojun Wei et al., Optimizing Spatial Mapping of … ***IEEE Transactions on VLSI Systems***, 2014

Multi-parameters mapping

- Energy consumption oriented partitioning and scheduling based on a system level energy modeling.
- Energy consumption is reduced 26% [1].



$$E = \sum_{i=1}^{T}\sum_{1=1}^{Q} o_{ijq}E(D_i) - \sum_{i=1}^{T}\left(1-\sum_{q=1}^{Q} o_{ijk}\right) E_r^{ave}(s_{ij,k_{ij}})$$

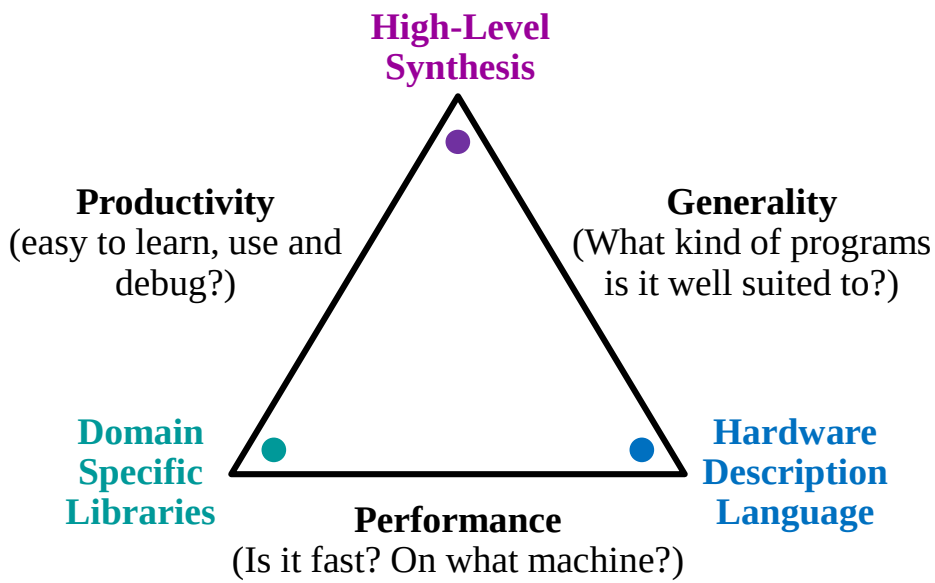
E : Energy of array o : Task state
Q : No. of task s : Power state

Patents: filed 10, granted 7, ZL200710099600, ZL200710100320, ZL200910090402, ….
Papers: 8 (SCI: 6), IEEE TVLSI、DAC, …

[1] Shouyi Yin, Shaojun Wei et al., Energy Management on Battery … ***IEEE Transactions on VLSI Systems***, 2014

Programmability

- Road to programmability: a trade-off among performance, productivity & generality
  - Popular solutions for reconfigurable computing
    - Hardware-description language
    - Domain-specific libraries
    - High-level synthesis (HLS)
  - Our Solutions
    - Novel programming model for previously intractable applications,
    - e.g. graph analytics[1] Extending design space for HLS, e.g. imperfect nested loops[2]

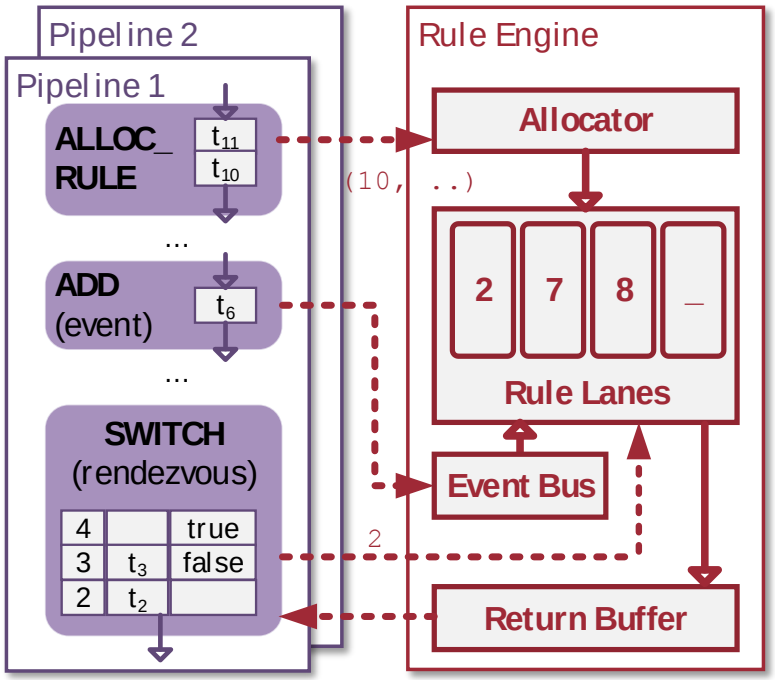


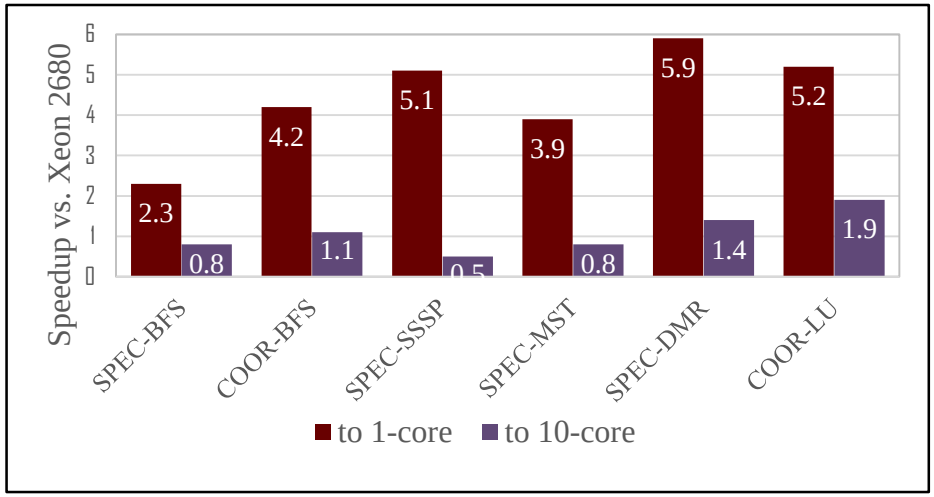
[1] Shouyi Yin, Xinghan Lin, Shaojun Wei, et al., "Exploiting Parallelism of Imperfect Nested Loops on Coarse-Grained Reconfigurable Architectures," ***IEEE Transactions on Parallel and Distributed Systems***, 2016.

[2] Dajiang Liu, Shouyi Yin, Shaojun Wei, et al., "Optimizing Spatial Mapping of Nested Loop for Coarse-Grained Reconfigurable Architectures," ***IEEE transactions on Very Large Scale Integration Systems***, 2014.

Filling the gap between ….

- Current high-level synthesis on FPGA: rely on **compile-time** analysis
- Irregular applications: dependencies have to be determined at **run-time**
- Our approach[1]:
  - Specify dependencies as rules at **programming**
  - Schedule tasks aggressively at **compile-time**
  - Evaluate rules to ensure correctness at **run-time**
- Over 10x speed-up compared to Altera OpenCL; 0.5x~1.9x performance compared to a 10-core Xeon.





[1] Zhaoshi Li, Leibo Liu, Shaojun Wei, et al., "Aggressive Pipelining of Irregular Applications on Reconfigurable Hardware", the 44th International Symposium on Computer Architecture (**ISCA**), 2017, to appear.

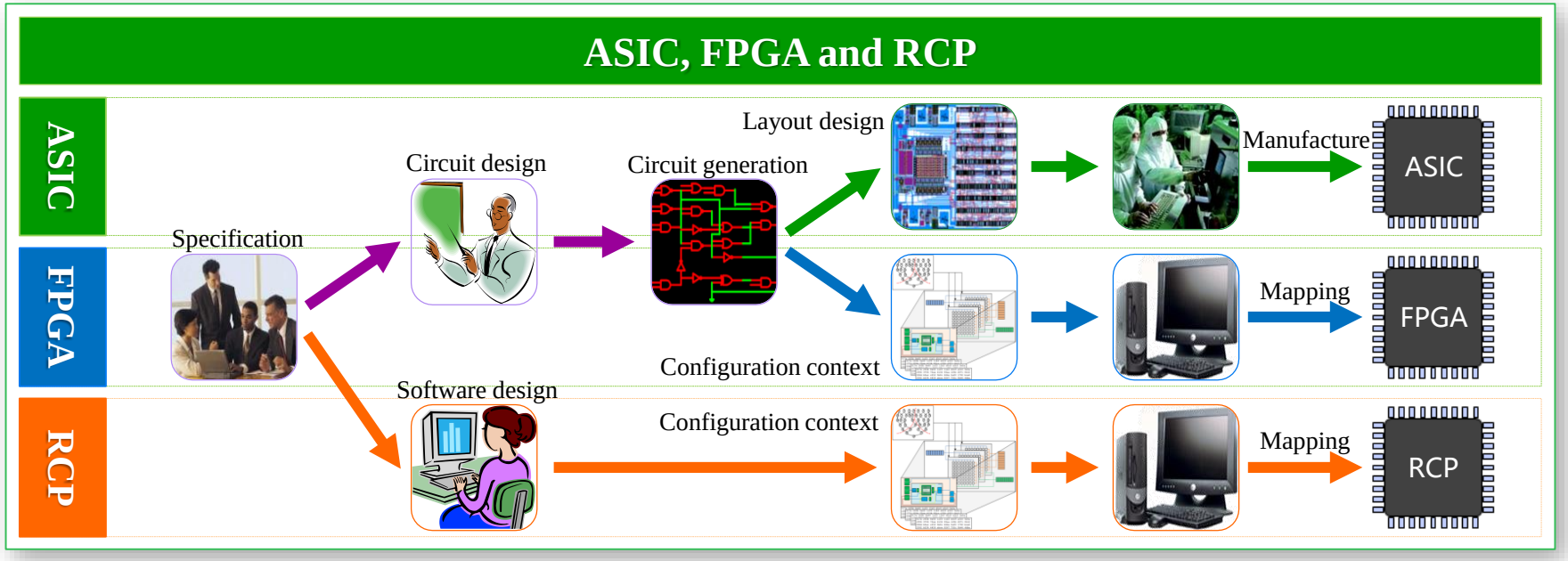
Outline

1. Background and Motivation
2. Reconfigurable Computing: Architecture
3. Technologies and Challenges
4. **Opportunities and Applications**
5. Reconfigurability and Hardware Security
6. Summary

Comparison of different products

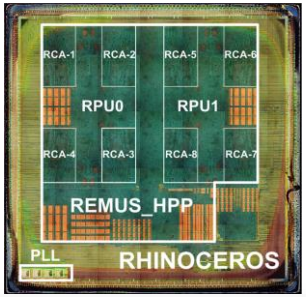
Product	Cost	Perf.	Power	Energy Efficiency	Programmability	Flexibility	Grain	Area Efficiency	Application
CPU	high	medium	high	0.001	software	very good	coarse	<100	numeric
FPGA[2]	higher	high	higher	0.01	hardware	good	fine	<100	logic
ASIC	low[1]	very high	low	1	non	non	find	>1000	logic
SoC	Lower	high	lower	0.1	software	good	mixed	>500	numeric/logic
RCP[3]	**low**	**high**	**low**	**~1**	**hardware/software**	**very good**	**mixed**	**~1000**	**numeric/logic**

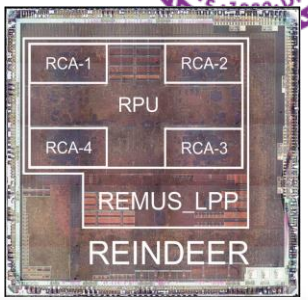
[1] Low cost while selling sufficient quantity.
[2] Do not support software. User should have experience of circuit design.
[3] RCP: Reconfigurable Computing Processor, programmable both in hardware and software.



Reconfigurable multimedia systems

- Application domains: video processing (H.264, HEVC, AVS…), baseband processing (GPS), cryptography (AES, DES, SMS4…) etc.
- REMUS_HPP: high performance.
- REMUS_LPP: low power.





	Reconfigurable				Multicore [2012, TVLSI]	GPGPU [2013, ISSCC]	DSP [2008, TCE]
	REMUS_HPP [2013,CICC]	XPP-III [2013,TVLSI]	REMUS_LPP [2013,CICC]	ADRES [2008, Micro]			
Technology (nm)	65	90	65	90	40	32	130
Area (mm^2)	48.9	75	21.6	3.6	210	N/A	529
Frequency (MHz)	200	450	75	300	333	166	600
Resolution	1920x1080	1920x1080	720x480	720x480	1920×1080	1920×1080	720x576
Performance (fps)	30	24	35	30	30	30	25
Power (mW)	280	3420	24.5	105.5	500	100	1900
Normalized Performance (MBs/s/MHz)	1224	435	630	135	735	1474	67.5
Energy Efficiency (MBs/s/mW)	874	57	1904	384	490	2448	21.3
Energy Efficiency (Scale to 65nm)	874	109	1904	760	186	593	81

[1] Leibo Liu, Dong Wang, Min Zhu, et al., "An Energy-Efficient Coarse-Grained Reconfigurable Processing Unit for Multiple-Standard Video Decoding," ***IEEE Transactions on Multimedia,*** 2015.

[2] Leibo Liu, Chenchen Deng, Dong Wang, et al., "An energy-efficient coarse-grained dynamically reconfigurable fabric for multiple-standard video decoding applications," in ***2013 IEEE Custom Integrated Circuits Conference (CICC)***, 2013.

General-purpose reconfigurable processor

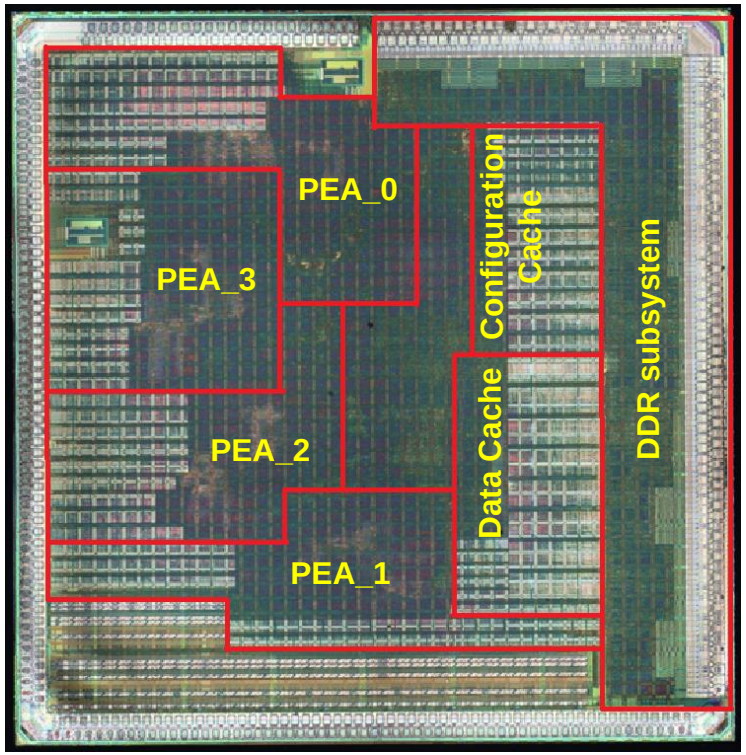
- Application domains: from Domain-specific (multi-media, cryptography) to General-Purpose (Berkeley 13 Dwarves)
- HReA: 25.26x more energy-efficient to Atom N550, 8.55x more energy-efficient than cortex A15.

Physical Implementation of HReA

Chip Name	HReA
Technology	TSMC 65nm LP1P8M
Frequency	280 MHz
Area	23.8 mm^2
Power	296.95 mW

Comparison of Different Architectures

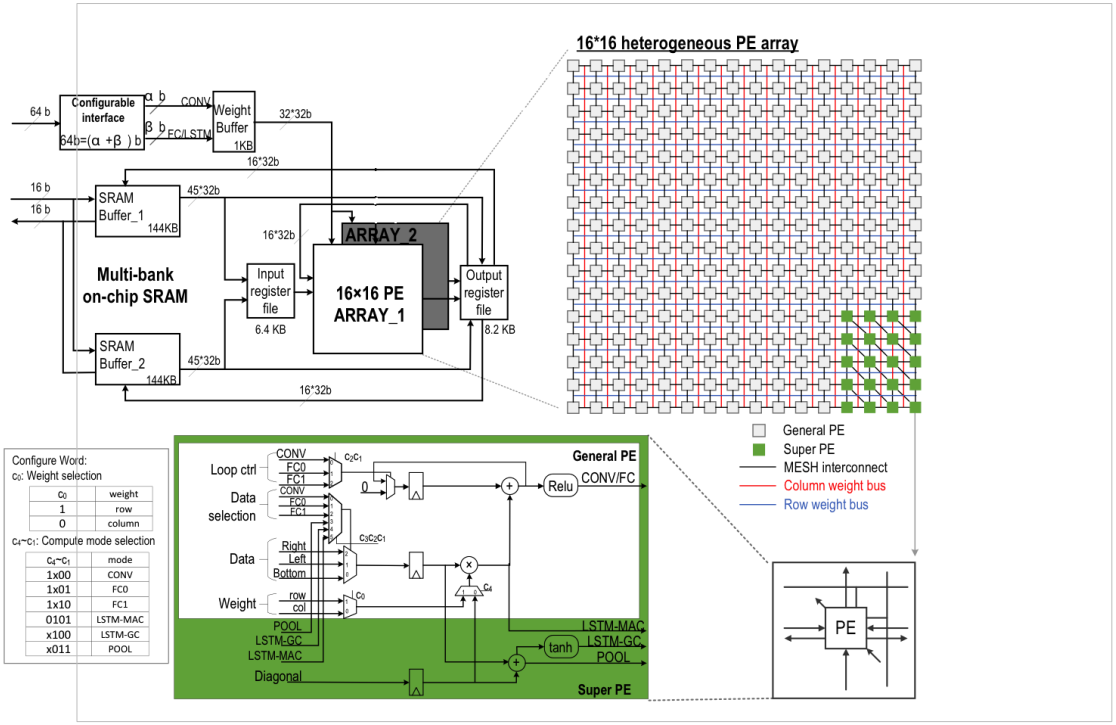
Architectures	Atom N550	Cortex A15	**HReA**
Technology (nm)	45	28	**65**
Cores/PEAs	2 cores	4 cores	**4 PEAs**
Area (mm^2)	87	10.8	**23.8**
Frequency (MHz)	1500	1600	**280**
Power (W)	8.5	6.57	**0.297**
Performance Ratio[a]	1.09×	2.32×	**1.00×**
Energy Efficiency Ratio[a]	0.038×	0.105×	**1.00×**



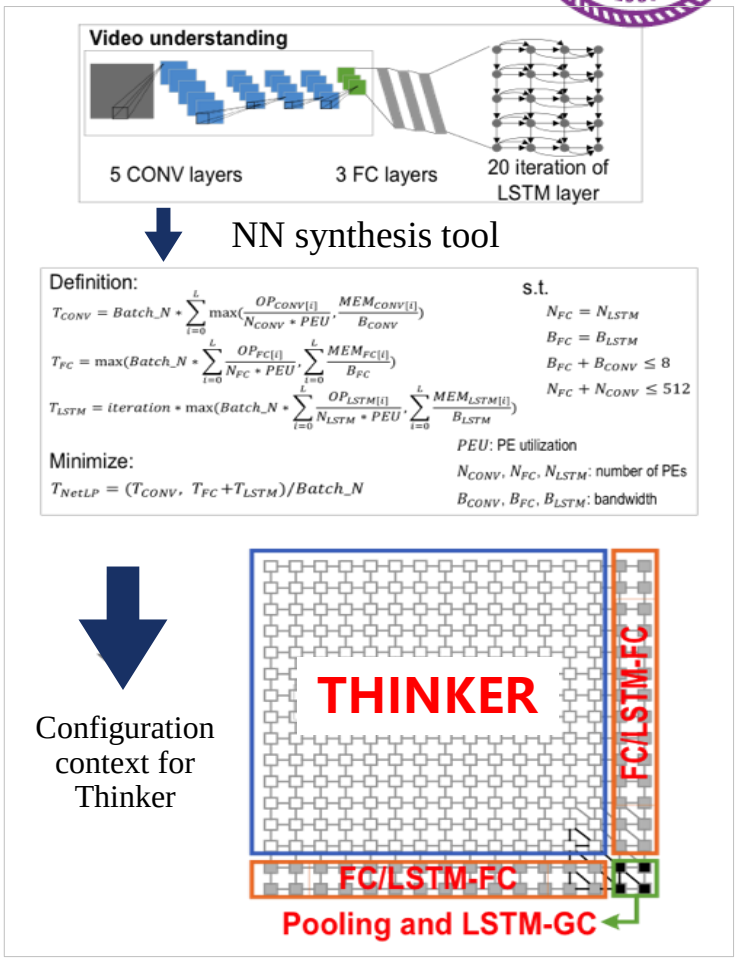
[1] Jianfeng Zhu, Leibo Liu, Shaojun Wei, et al., "A Hybrid Reconfigurable Architecture and Design Methods Aiming at Control-Intensive Kernels," ***IEEE Transactions on Very Large Scale Integration Systems***, 2015

[2] Chen Wu, Chenchen Deng, Leibo Liu, et al., "A Multi-Objective Model Oriented Mapping Approach for NoC-based Computing Systems," ***IEEE Transactions on Parallel and Distributed Systems***, 2017.

Reconfigurable neural computing device



1. Heterogeneous PE array supports CNN/RNN/FCN
2. Reconfigurable PE and programmable interconnections
3. Array based data reuse pattern improves energy efficiency



AI application → Define → Neural networks → Define → Chip functions

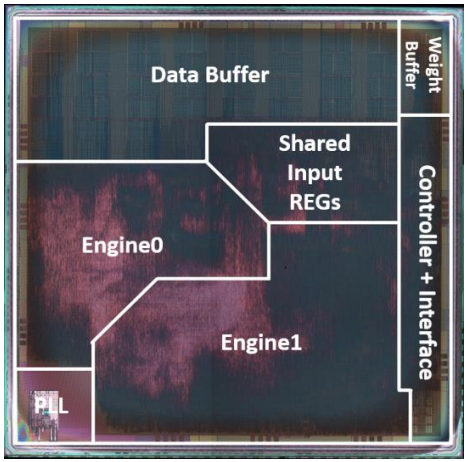
[1] P. Ouyang, S.Y. Yin*, S.J. Wei. A Fast and Power Efficient Architecture to Parallelize LSTM based RNN for Cognitive Intelligence Applications, ***DAC 2017***

[2] S.Y. Yin, S.J. Wei et al. Exploiting Parallelism of Imperfect Nested Loops on Coarse-Grained Reconfigurable Architectures, ***IEEE Transactions on PDS* 2016**

[3] P. Ouyang, S.Y. Yin*, S.J. Wei et al. A Fast and Power-Efficient Memory- Centric Architecture for Affine Computation, ***IEEE Transactions on CAS II* 2016**

Specifications and comparison

Technology	TSMC 65nm LP
Supply voltage	0.67V~1.2 V
Area	4.4mm*4.4mm
SRAM	348KB
Frequency	10 ~ 200MHz
Peak performance	409.6GOPS
Power	4mW ~ 447mW
Energy efficiency	1.06TOSP/W~5.09TOPS/W



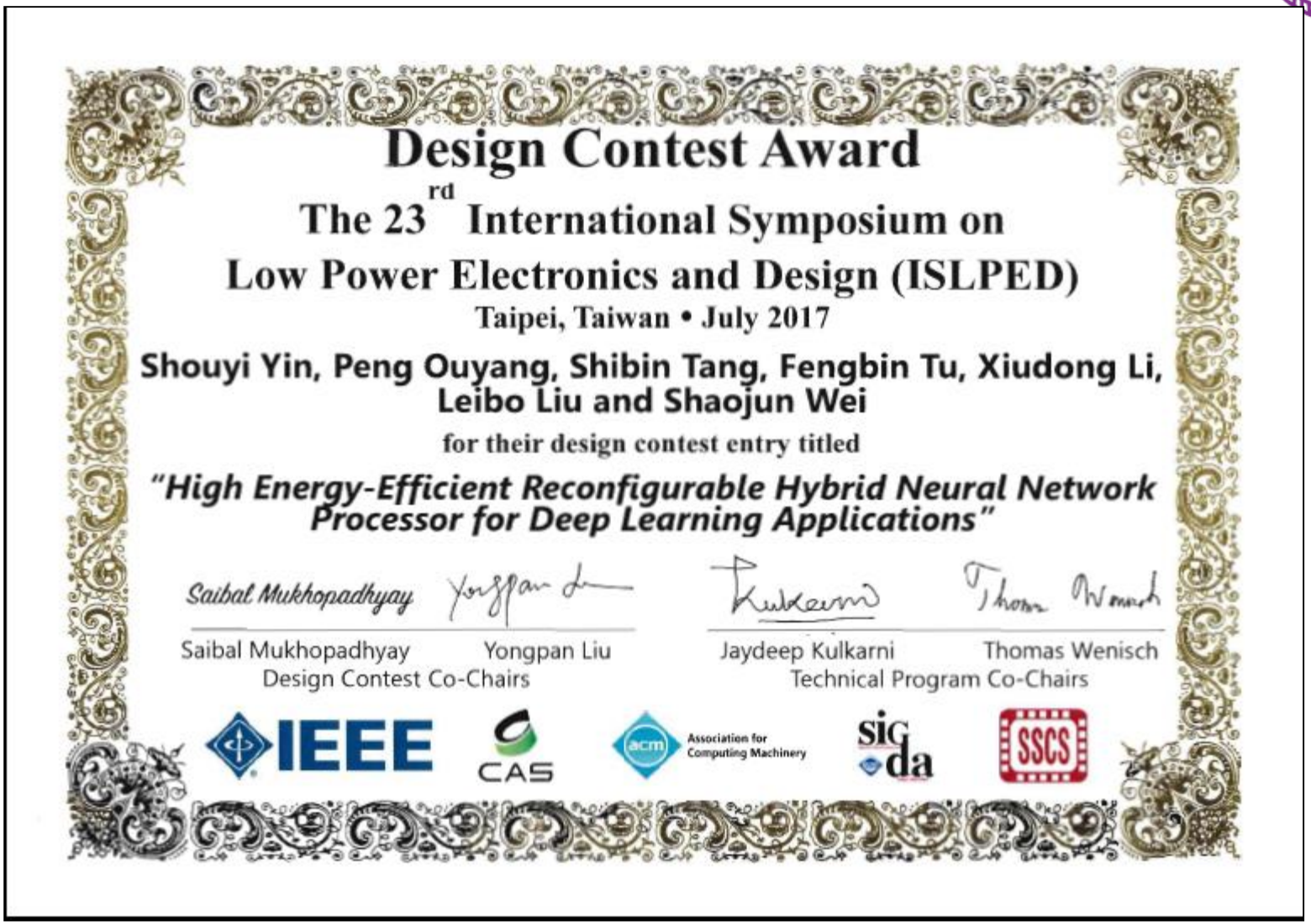
	Thinker		ISSCC'16	VLSI'16
Frequency	200MHz		200MHz	204MHz
Area	4.4mm*4.4mm		3.5mm*3.5mm	1.2mm*2mm
On-chip SRAM	348KB		108KB	144KB
Technology	65nm/1.2V		65nm/(0.82-1.17)V	40nm/1.1V
Benchmark	AlexNet (CNN+FCN)	LRCN (CNN+FCN+LSTM)	AlexNet (Conv Only)	AlexNet (Conv Only)
Throughput	368.4GOPS	381.2GOPS	67.6GOPS	71.4GOPS
Power	290mW	297.8mW	278mW	76mW
Energy efficiency	1.27TOPS/W	1.28TOPS/W	243GOPS/W	0.94TOPS/W

[1] S. Y. Yin, S. J. Wei, et al. A 1.06-to-5.09 TOPS/W Reconfigurable Hybrid-Neural-Network Processor for Deep Learning Applications. ***2017 Symposia on VLSI***.

[2] F. B. Tu, S. Y. Yin, S. J. Wei et al. A Reconfigurable Multi-modal Neural Processor for Cognitive Intelligence Applications, ***ISSCC SRP 2017***

[3] F. B. Tu, S. Y. Yin, S. J. Wei et al. Deep Convolutional Neural Network Architecture with Reconfigurable Computation Patterns, ***IEEE Transactions on VLSI***, 2017

Design contest award at ISLPED



Design Contest Award

The 23rd International Symposium on
Low Power Electronics and Design (ISLPED)

Taipei, Taiwan • July 2017

Shouyi Yin, Peng Ouyang, Shibin Tang, Fengbin Tu, Xiudong Li, Leibo Liu and Shaojun Wei

for their design contest entry titled

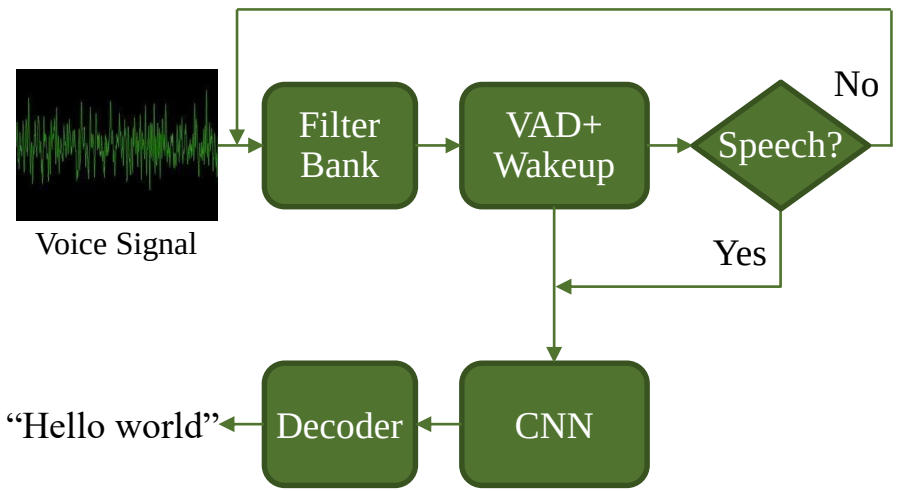
"High Energy-Efficient Reconfigurable Hybrid Neural Network Processor for Deep Learning Applications"

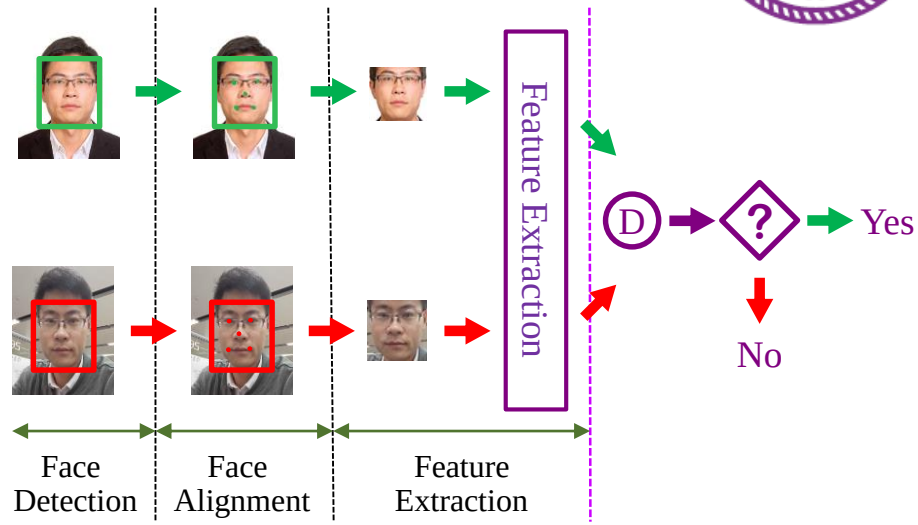
Saibal Mukhopadhyay Yongpan Liu
Design Contest Co-Chairs

Jaydeep Kulkarni Thomas Wenisch
Technical Program Co-Chairs

IEEE CAS acm Association for Computing Machinery sigda SSCS

Speech recognition and face recognition





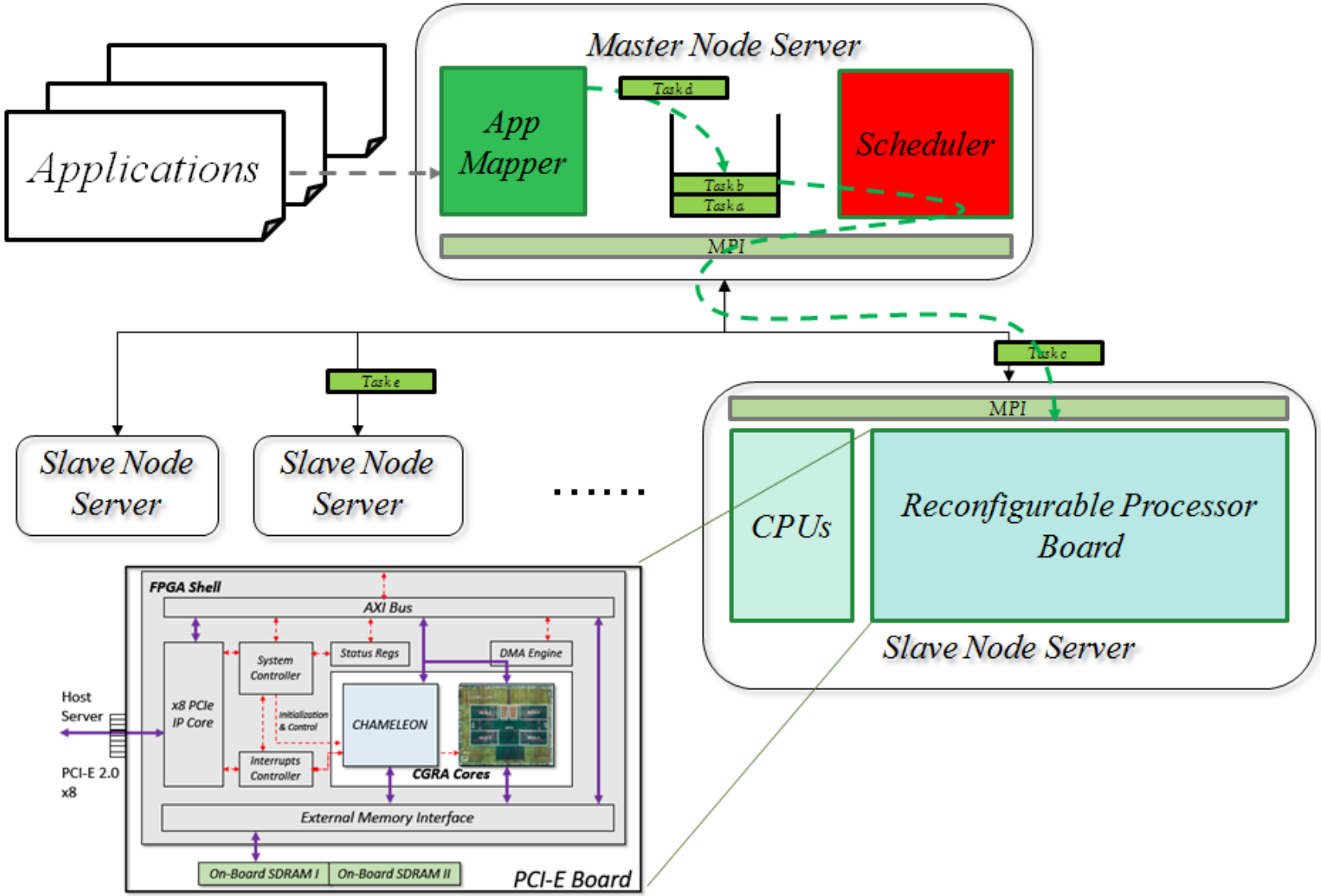
Test on 200-hour speech dataset

VAD Accuracy	91.8%
Wake Up Accuracy	90%
Keyword Spotting Accuracy	85%
Latency	< 25ms
Power	~ 1mW

Face Recognition Performance on LFW dataset

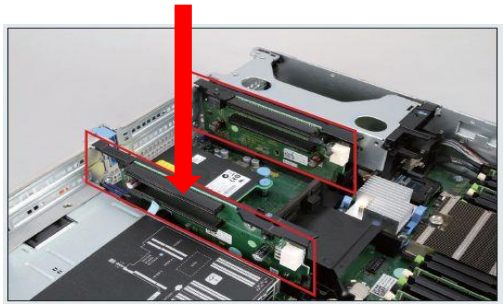
Human Accuracy	97.53%
Thinker Accuracy	98.8%
Speed	6.22ms@200MHz
Power	~ 20mW

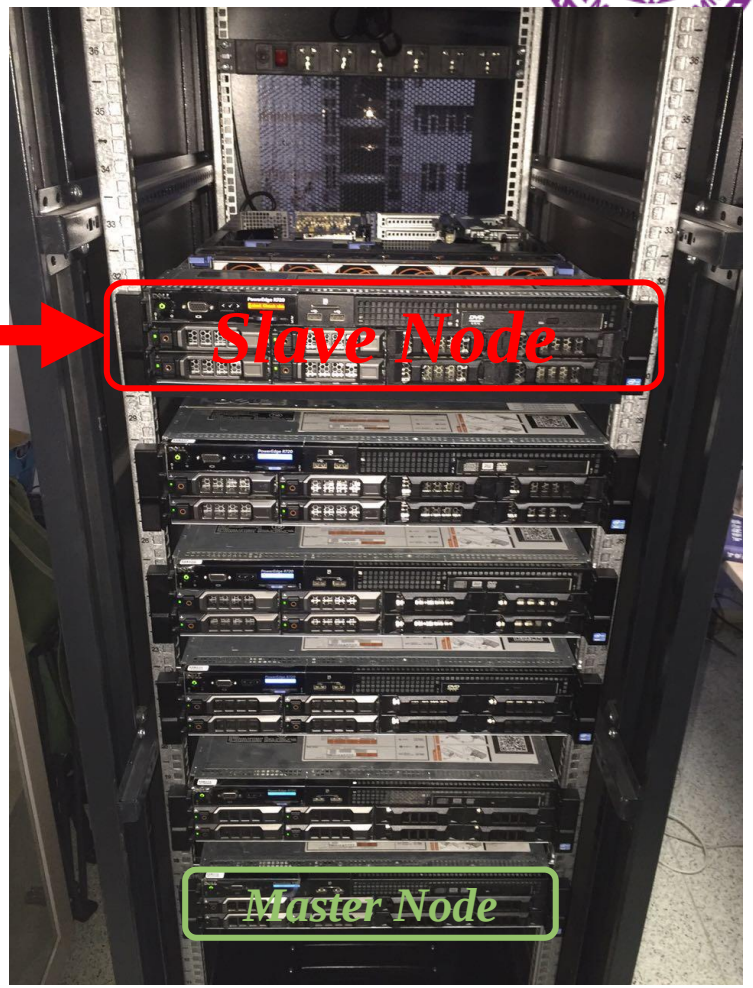
Elastic cloud computing based on RCP



Elastic cloud computing based on RCP

Reconfigurable Processor Board





Server	DELL R720 High performance server
Operating System	Windows Server 2008 R2
Network	1000Mbps Ethernet
Protocol	MPICH 2

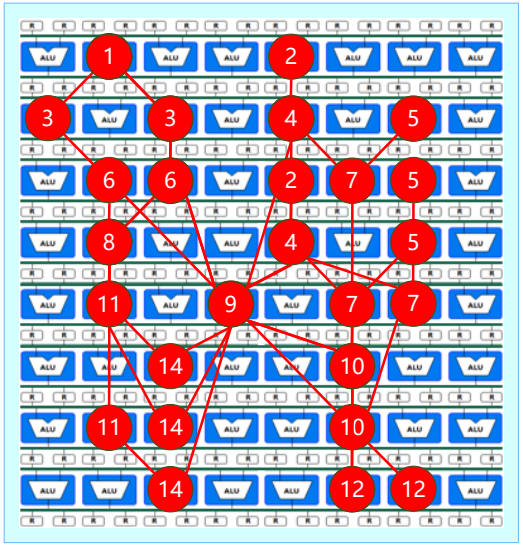
[1] Yin S, Liu D, Peng Y Wei S, Improving Nested Loop Pipelining on Coarse-Grained Reconfigurable Architectures, ***IEEE Transactions on VLSI***, 2014.
[2] Yin S, Zhou P, Liu L, Wei S, Trigger-Centric Loop Mapping on CGRAs, ***IEEE Transactions on VLSI***, 2015
[3] Yin S, Yao X, Liu D, Wei S, Memory-Aware Loop Mapping on Coarse-Grained Reconfigurable Architectures, ***IEEE Transactions on VLSI***, 2015
[4] Yin S, Gu J, Wei S, Joint Modulo Scheduling and Vdd Assignment for Loop Mapping on Dual-Vdd CGRAs, ***IEEE Transactions on CAD***, 2016

Outline

1. Background and Motivation
2. Reconfigurable Computing: Architecture
3. Technologies and Challenges
4. Opportunities and Applications
5. **Reconfigurability and Hardware Security**
6. Summary

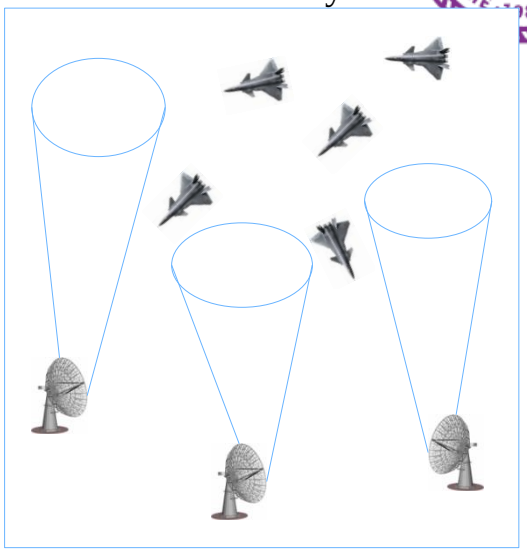
High security features

Randomness



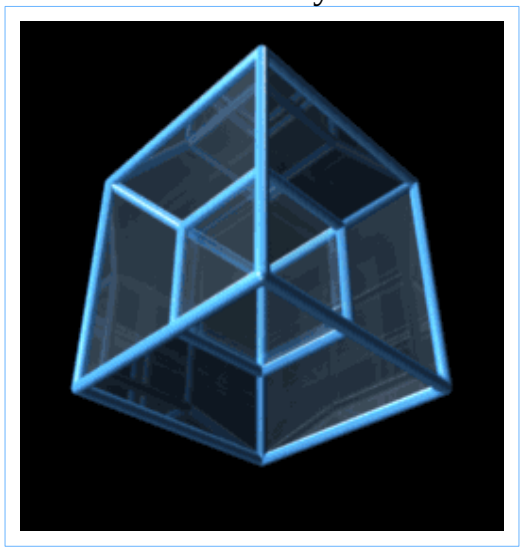
Fragmentation

Invisibility

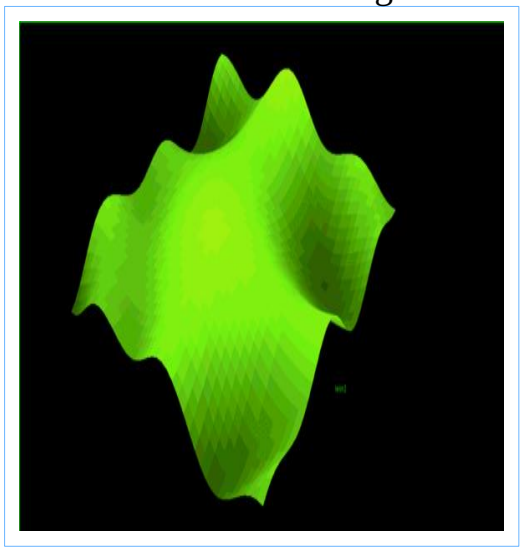


Active defense

Diversity

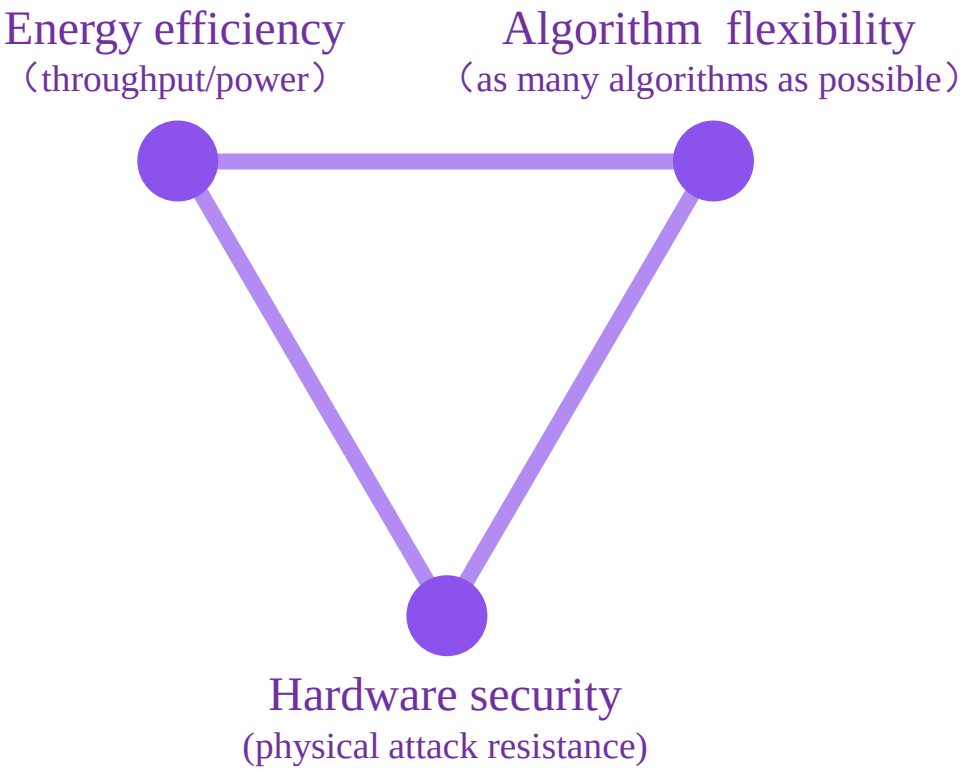


Fast transforming



Reconfigurable cryptographic architecture

- Maintaining algorithm flexibility, energy efficiency and hardware security at the same time [1]
- Resistance against novel threatening physical attacks utilizing the characteristics of reconfigurable architecture
  - Spatial and time randomization countermeasures based on dynamic reconfiguration
  - Countermeasures based on array processing resource and special function module



[1] Leibo Liu, Bo Wang, Shaojun Wei, “Reconfigurable Computing Processor for Cryptographic Algorithms”, China Science Press, September 2017. ISBN: 9787030542441

Randomized reconfiguration against DFA

Spatial and time randomization countermeasures are designed utilizing randomized dynamic and partial reconfiguration to resist against threatening double fault attacks.

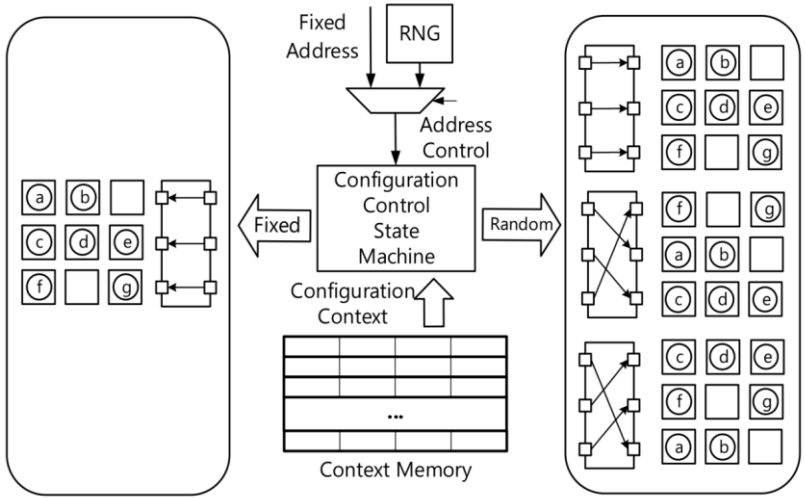


Fig.1 Hardware support for Round Based Relocation: configuration control logic driven by random numbers.

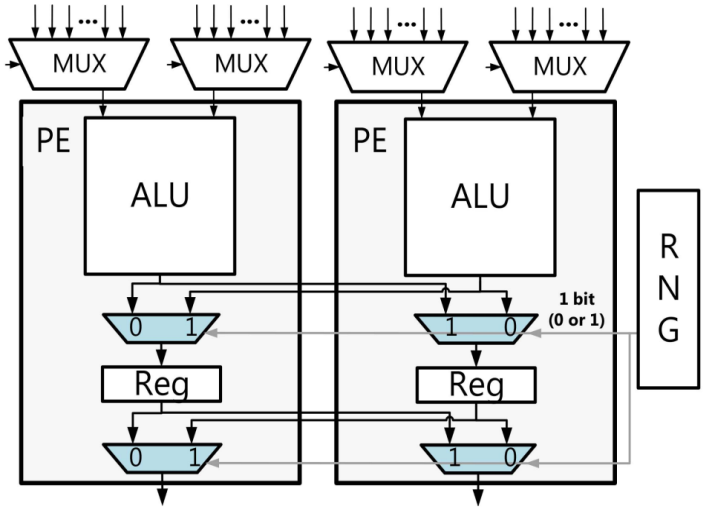
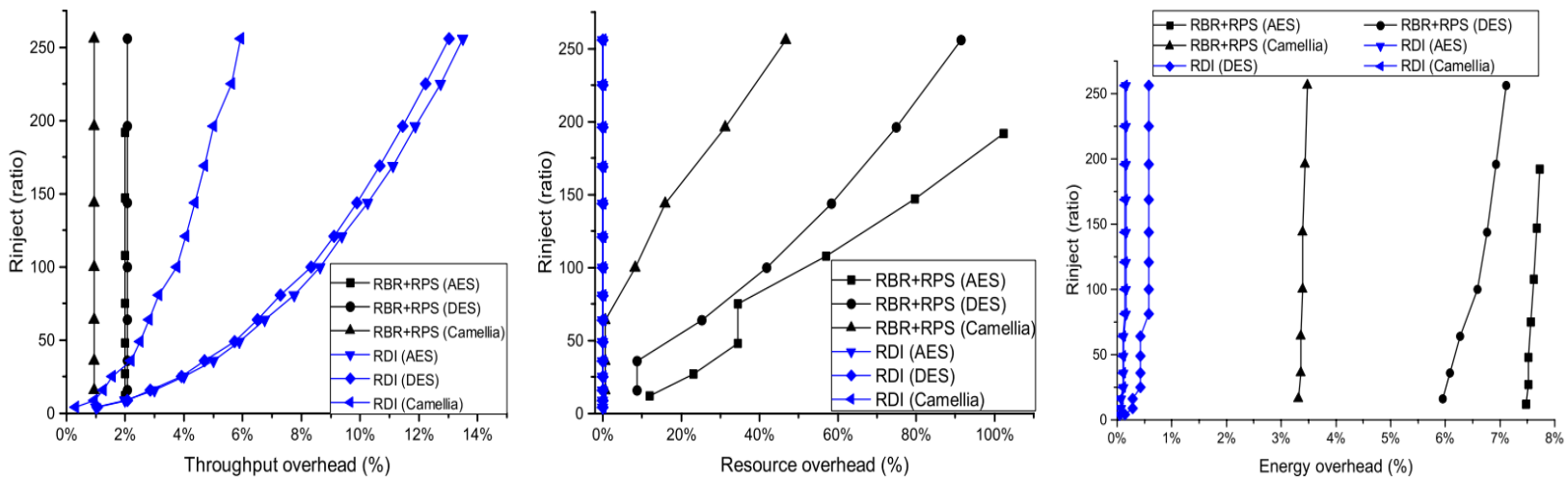


Fig.2 Hardware support for Register Pair Swap: MUXs controlled by random bit.

[1] Bo Wang, Leibo Liu, Shaojun Wei et.al. Against Double Fault Attacks: Injection Effort Model, Space and Time Randomization Based Countermeasures for Reconfigurable Array Architecture [J]. IEEE Transactions on Information Forensics and Security (IF:4.3), 2016, 11(6): 1151-1164.

Randomized reconfiguration against DFA

When the overhead constraints in terms of throughput, hardware resources, and energy are 5%, 35%, and 10% respectively, the double fault resistance (represented by Rinject in Fig.1) can increase by two to four orders of magnitude for different algorithms.



(a) (b) (c)

Fig.1 Overhead-Resistance trade-offs for the countermeasures. (a) Rinject-Throughput trade-offs. (b) Rinject-Resource trade-offs. (c) Rinject-Energy trade-offs.

[1] Bo Wang, Leibo Liu, Shaojun Wei et.al. Against Double Fault Attacks: Injection Effort Model, Space and Time Randomization Based Countermeasures for Reconfigurable Array Architecture [J]. IEEE Transactions on Information Forensics and Security (IF:4.3), 2016, 11(6): 1151-1164.

Random infection countermeasure

This work presents an infection method that resists fault attacks by slightly modifying the existing Benes network module in high-performance crypto processors.

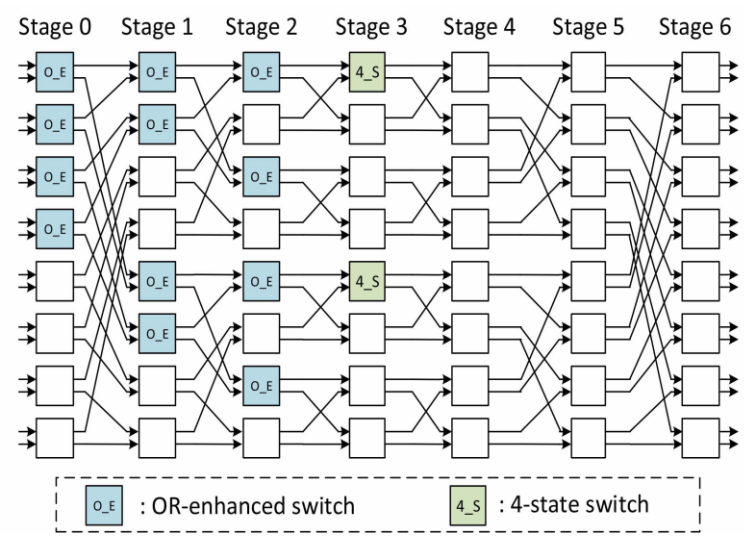


Fig.1 The infection-oriented network structure (INS) (N = 16).

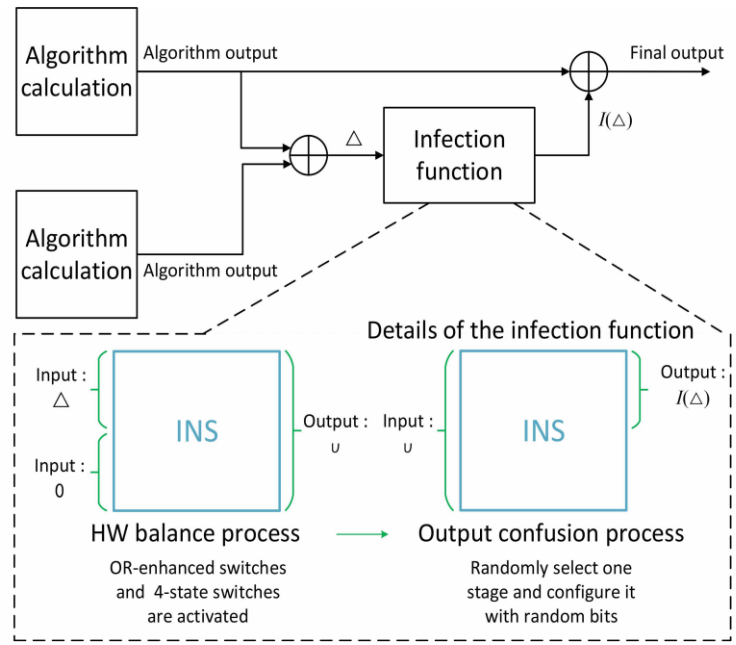


Fig.2 The overall execution flow of the countermeasure.

[1] Bo Wang, Leibo Liu, Shaojun Wei et.al. Exploration of Benes Network in Cryptographic Processors: A Random Infection Countermeasure for Block Ciphers against Fault Attacks [J]. IEEE Transactions on Information Forensics and Security (IF:4.3), 2017, 12(2): 309-322.

Random infection countermeasure

Under statistical evaluation, the proposed countermeasure improves the fault resistance by over four orders of magnitude compared with the unprotected case. Also, the performance and the area overhead are within 10% compared with the original Benes network.

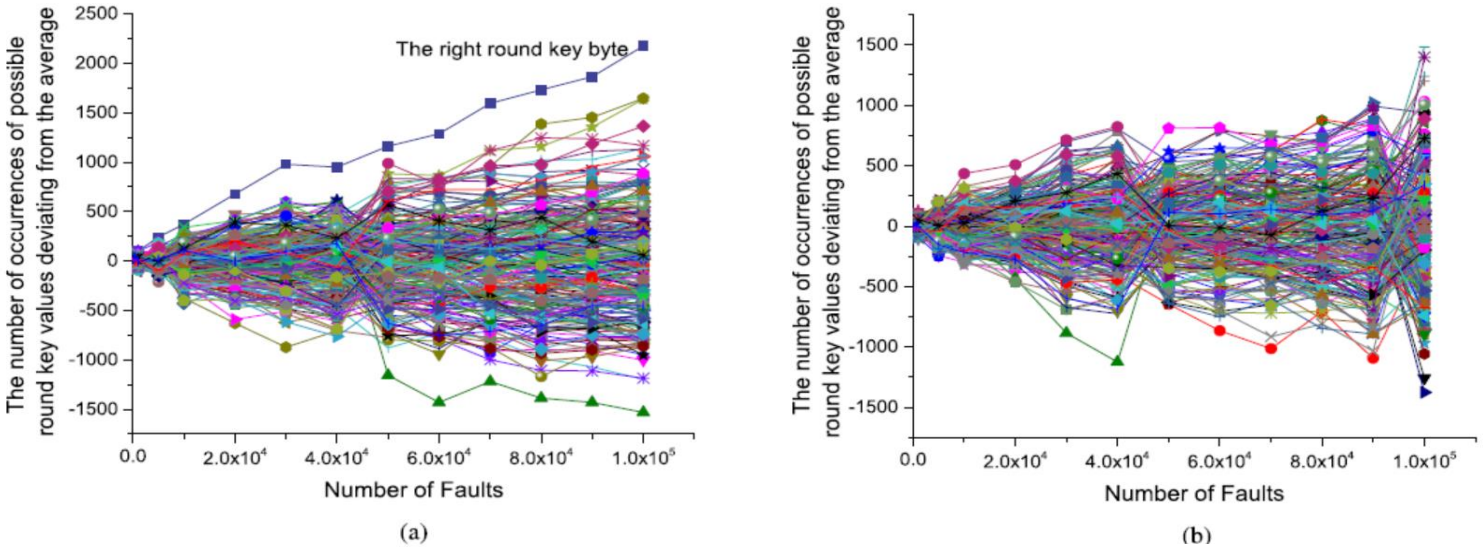


Fig.1 Two instances of the improved attack (i.e. the statistical evaluation method) against the proposed countermeasure for the AES algorithm under a constraint of 100,000 faults. (a) A successful attack (with little success probability). (b) A failed attack.

[1] Bo Wang, Leibo Liu, Shaojun Wei et.al. Exploration of Benes Network in Cryptographic Processors: A Random Infection Countermeasure for Block Ciphers against Fault Attacks [J]. IEEE Transactions on Information Forensics and Security (IF:4.3), 2017, 12(2): 309-322.

New X86 CPU with reconfigurable computing

Classical general purpose processor is hard to meet the requirements of high energy efficiency and various applications for data center. January 2016, Intel had signed agreement with Tsinghua University in order to develop a new CPU architecture combining reconfigurable computing and X86 for Data center.

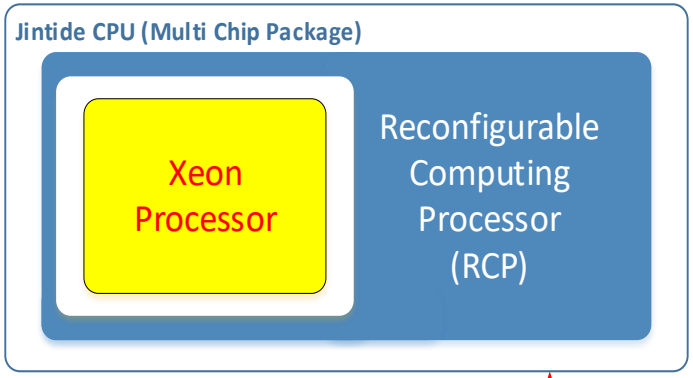
Tsinghua-Intel collaboration agreement signing Ceremony

Diane Bryant, Intel SVP

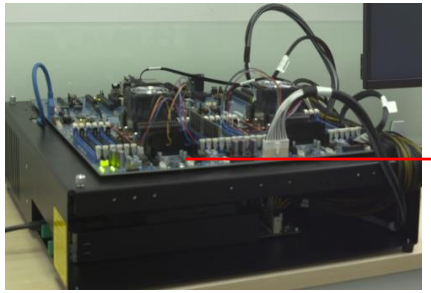
Tsinghua university has strong advantage in reconfigurable computing research. The collaboration between Tsinghua and Intel will certainly give a outstanding result with global impaction.

JinTide™: A hardware trustworthy CPU

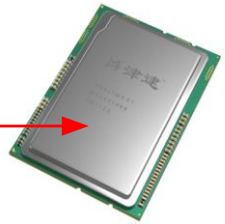
First hardware security solution for server CPU. Based on Xeon Skylake processor and reconfigurable technology Jintide combines security and high-performance.



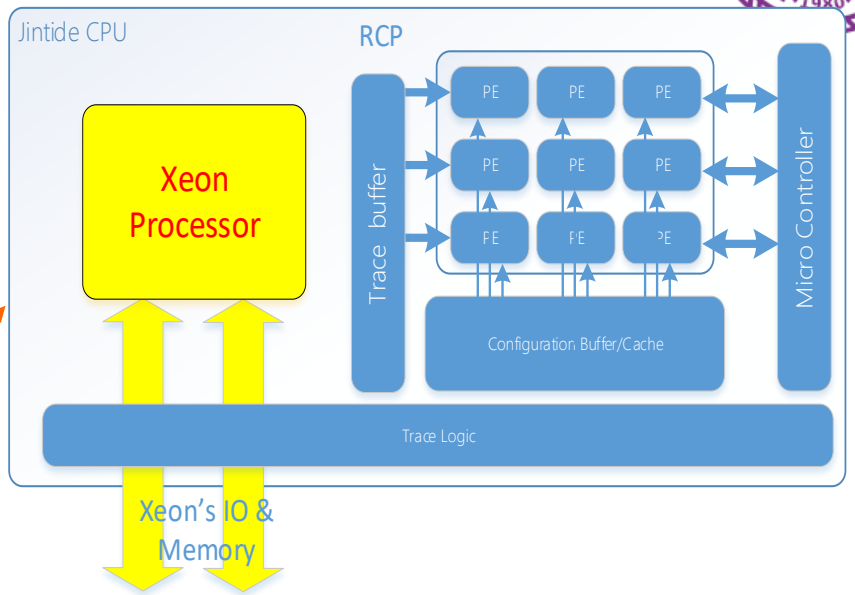
Jintide CPU Chip Package



Jintide Platform



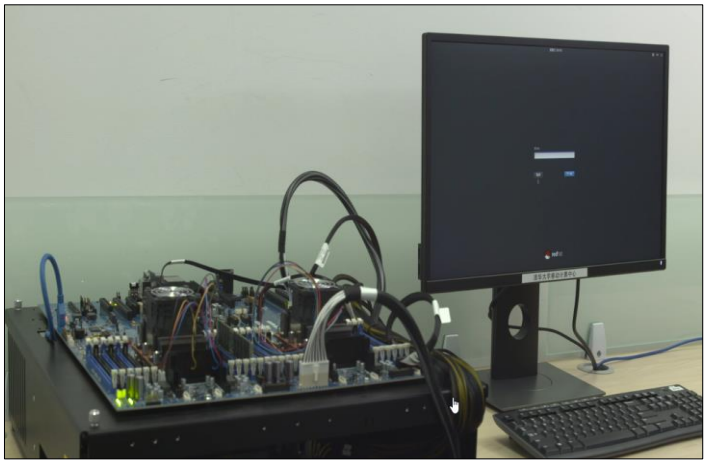
Jintide CPU Chip



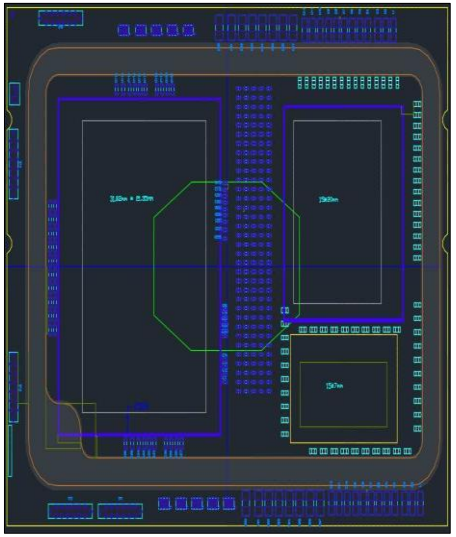
RCP Key features:

- "White chip": configuration defined hardware
- Offloading sensitive tasks, e.g. crypto
- Trace, Check and Control of Xeon processor
  - Detect any hardware trojan in Xeon and mitigate impact by IO control
- Flexible: work with multiple generation Xeon

JinTide™: A hardware trustworthy CPU



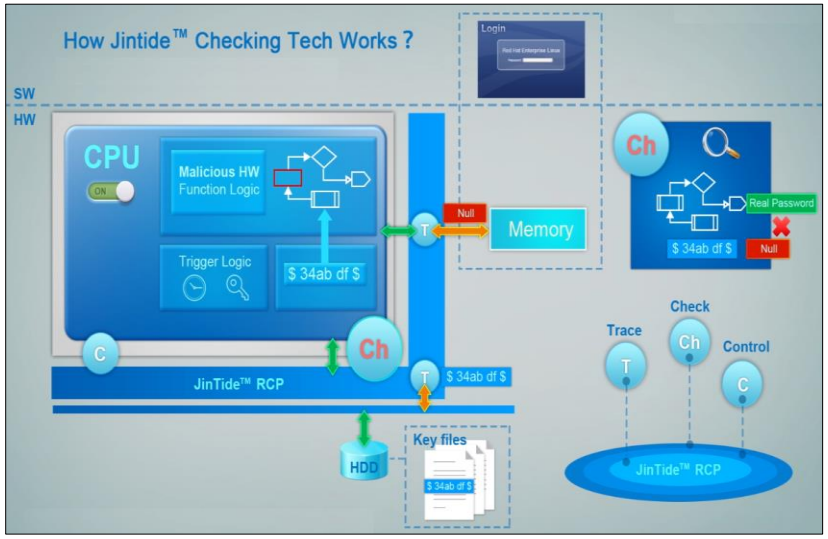
Demo Scenario



Jintide CPU Package

Hardware Threat and JinTide™ Security DEMO

An emulated hardware trojan that attacks Linux system login process, and detected by JinTide™ Dynamic CPU Checking:



Demo Explanation

JinTide™ CPU SDV and MCP Package

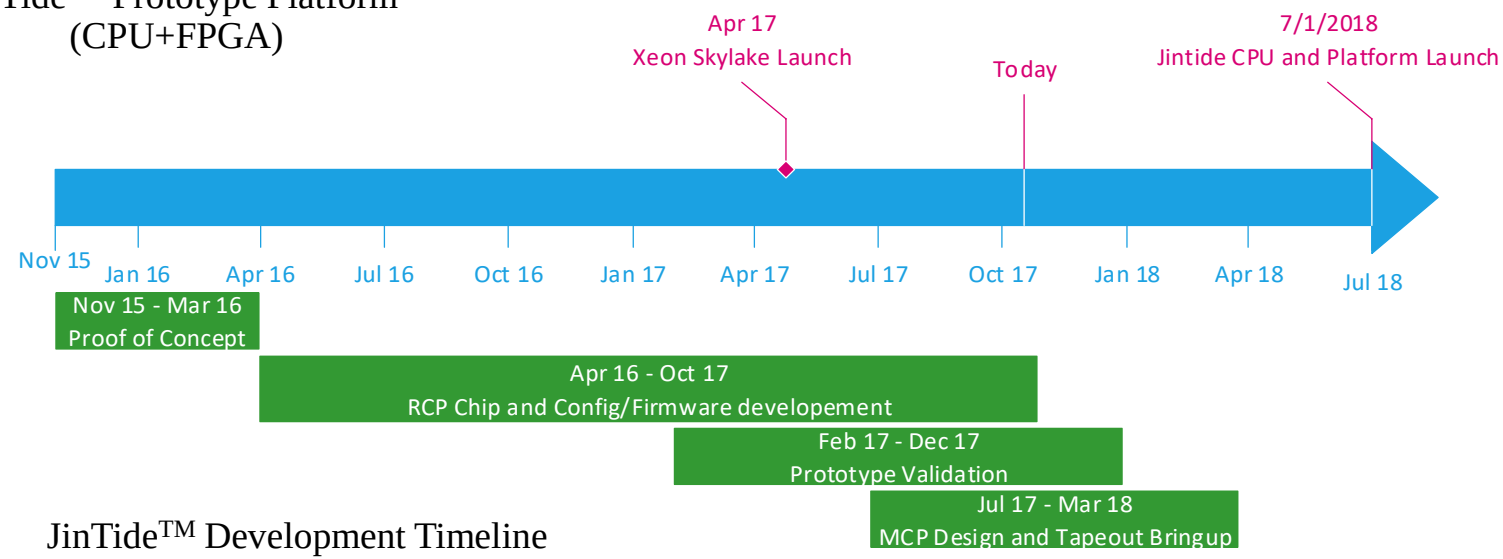
JinTide™ CPU use MCP to integrate RCP tracing circuit (15mm*20mm), checking circuit (15mm*7mm) and Xeon Skylake processor (32mm*21mm).

JinTide™: A hardware trustworthy CPU

JinTide™ Prototype Platform (CPU+FPGA)

JinTide™ CPU

- Compatible to Xeon processor
- Efficient to capture Xeon processor design bugs or "malicious hardware" at runtime
- Full trace/interpretation/control of Xeon's IO (sample based)
- Offloading of critical Tasks



JinTide™ Development Timeline

Outline

1. Background and Motivation
2. Reconfigurable Computing: Architecture
3. Technologies and Challenges
4. Opportunities and Applications
5. Reconfigurability and Hardware Security
6. **Summary**

- As process technology goes to 1Xnm, ASIC meet s serious challenge. The high NRE cost makes ASIC that are in general many varieties and small volume, become extremely expensive.
- Exploration of a new integrated circuit architecture having advantages of ASIC, such as high performance and high energy efficiency and CPU, such as highly flexible, is an urgent task.
- Dynamically reconfigurable architecture (Software defined Chip) allows user to redefine functionality of chip in real time according to requirement and multiplexes hardware to adapt different tasks so that it provides a very flexible chip with significantly high energy efficiency.
- Several different applications using the reconfigurable computing technologies had been shown, that cover multimedia, general-purpose processor, neural computing and elastic cloud computing.
- The hardware reconfigurability provides outstanding security features by not only creating a huge, randomized solution space but also making real-time, independent behavior checking become possible that may change today's game rule of processor security.

Thank you