

Throughput vs. Area Trade-offs in High-Speed Architectures of Five Round 3 SHA-3 Candidates Implemented Using Xilinx and Altera FPGAs



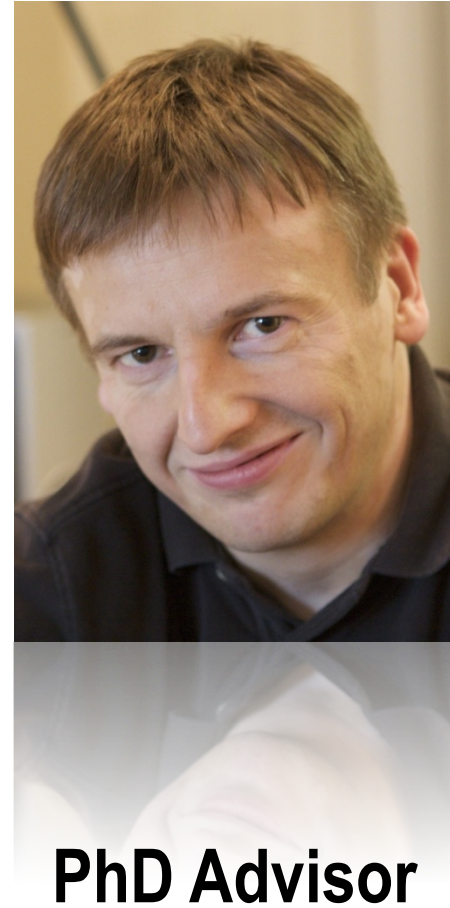
**Ekawat Homsirikamol,
Marcin Rogawski, and
Kris Gaj
George Mason University
U.S.A.**

Co-Authors

Marcin Rogawski



Kris Gaj

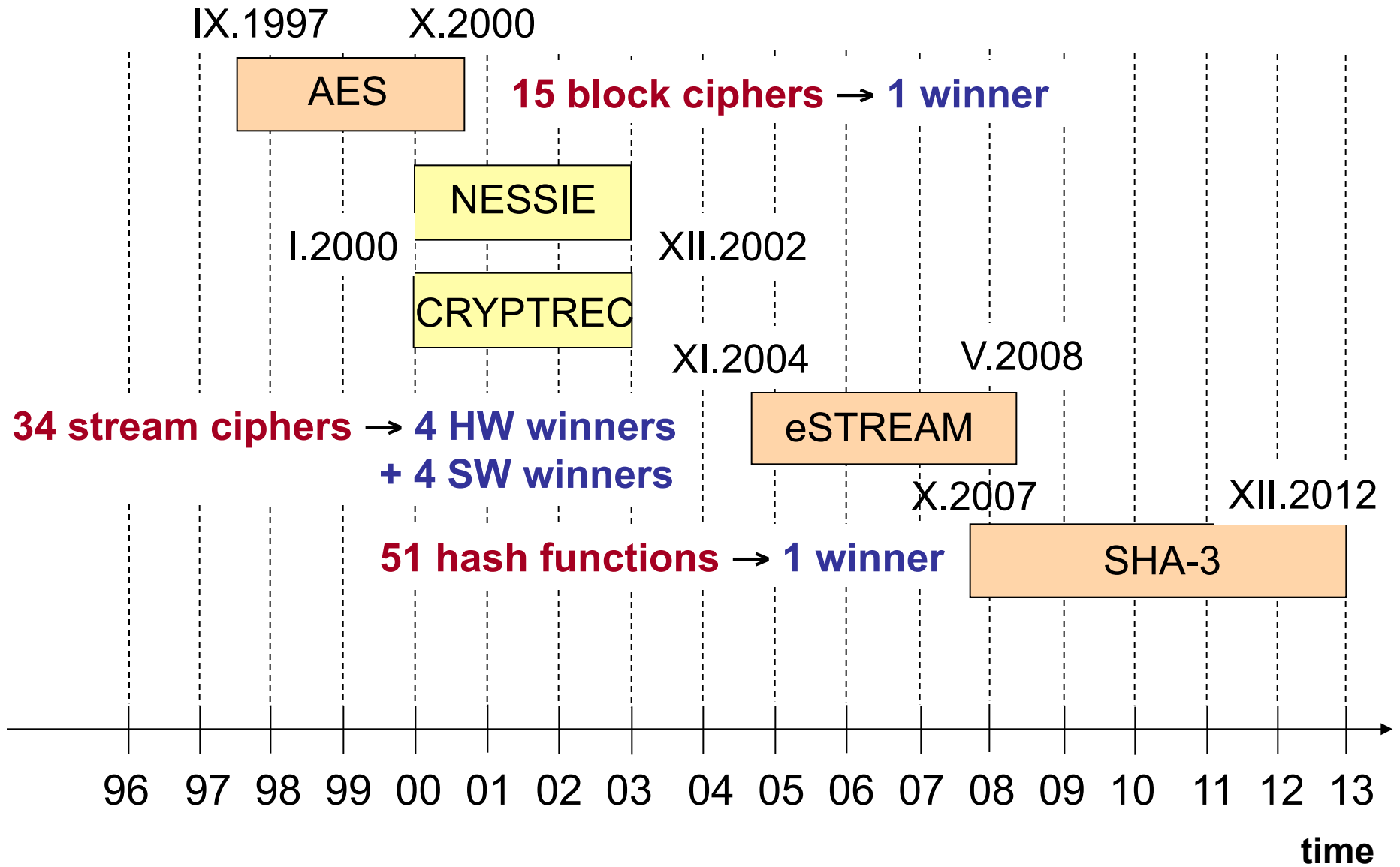


Outline

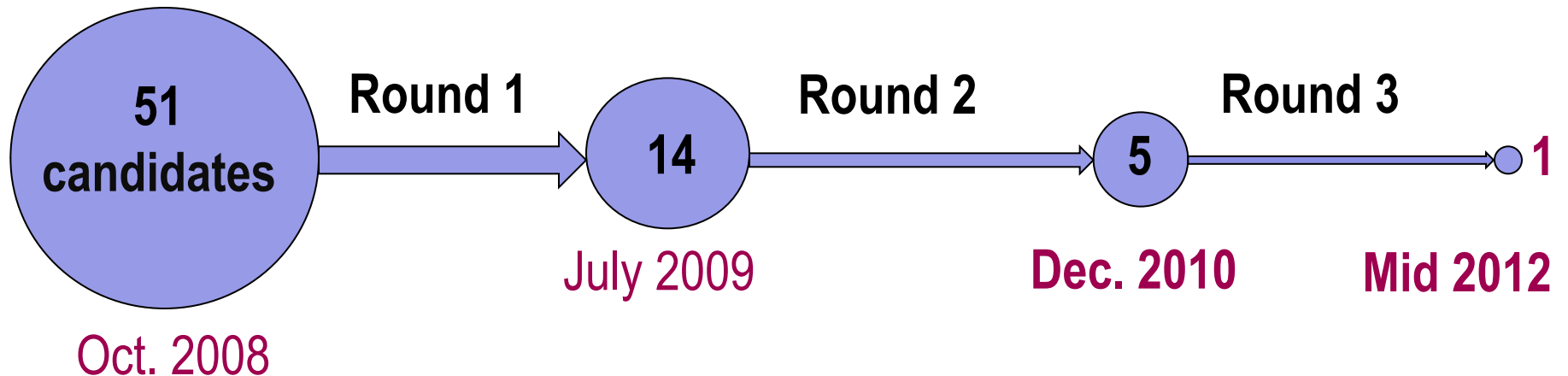
- **SHA-3 Contest**
- **Hardware Architectures of SHA-3 Candidates**
 - **basic**
 - **folded/unrolled**
 - **pipelined**
- **Results**
- **Ranking & Conclusions**
- **Reproducibility of Results & Future Work**



Cryptographic Standard Contests



NIST SHA-3 Contest - Timeline



Remaining steps:

Jan 2011-Mar 2012: Evaluation Period for Round 3 Candidates

22-23 Mar 2012: 3rd SHA-3 Candidate Conference, Washington D.C.

Summer 2012: **Announcement of a winner**

Beginning of 2013: Publication of the new FIPS standard

NIST Evaluation Criteria

Security

Software Efficiency

μProcessors

μControllers

Hardware Efficiency

FPGAs

ASICs

Flexibility

Simplicity

Licensing

Previous Work

SHA-3 Zoo Hardware Implementations

http://ehash.iaik.tugraz.at/wiki/SHA-3_Hardware_Implementations

	FPGA	ASIC
High-speed		
Low-area		

Most Related Previous Work

Comprehensive Evaluations of 14 Round 2 Candidates (FPGA):

- AIST-RCIS & UEC Japan; COSIC, Belgium; Virginia Tech, USA: 2nd SHA-3 Conf
- University College Cork, Ireland; Queens University Belfast, UK; RMIT University, Australia: 2nd SHA-3 Conf, FPL 2010
- George Mason University, USA: CHES 2010, 2nd SHA-3 Conf.

First Pipelined Architectures:

- Sabancı University, Turkey: 2nd SHA-3 Conf
- Skein Team: 2nd SHA-3 Conf

Limitations of the SHA-3 Round 2 Evaluations in FPGAs

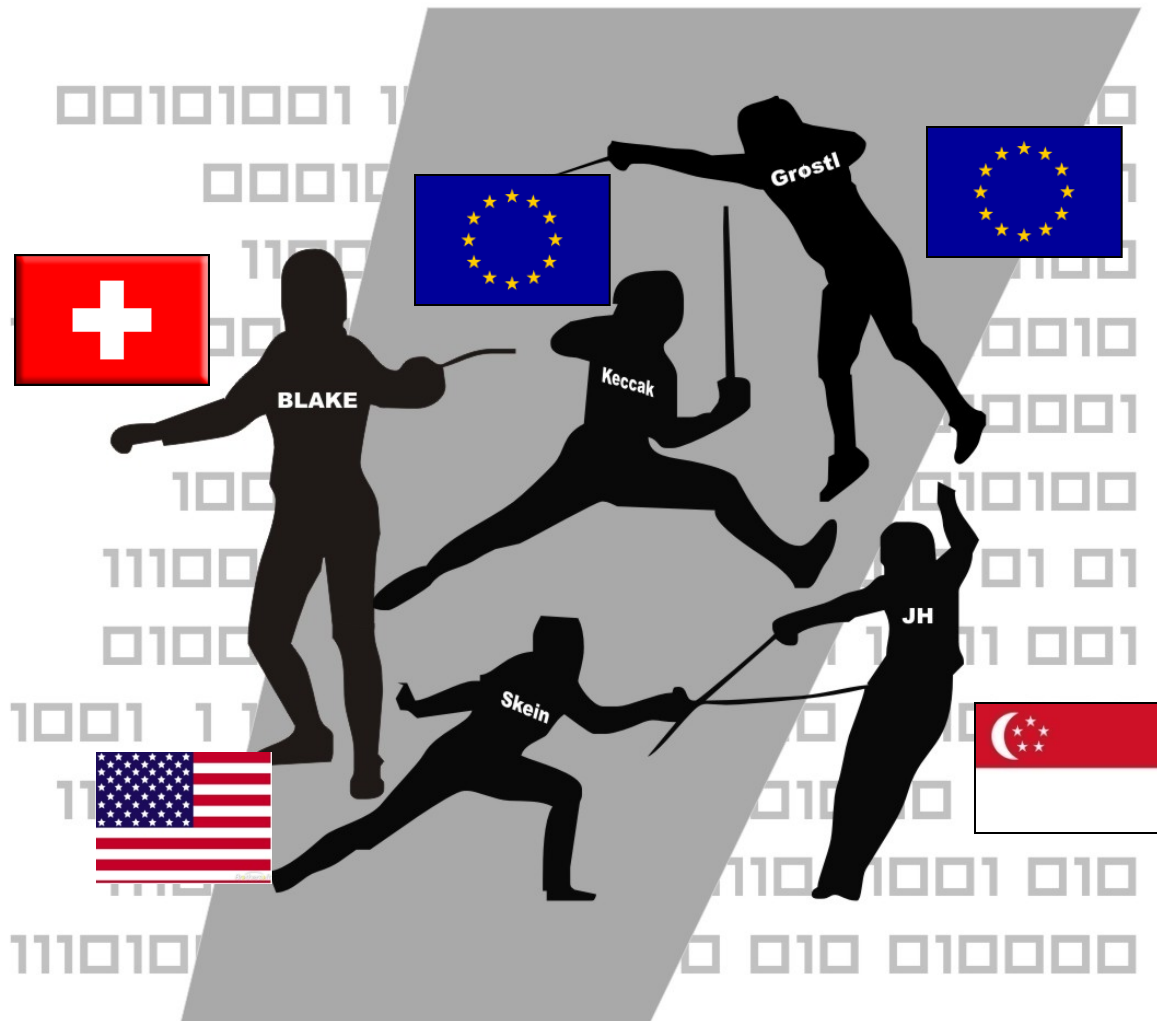
- **Single** high-speed **architecture** per candidate
- **No** comprehensive investigation of **pipelined architectures**
- **No use of embedded resources** of FPGAs (Block RAMs, dedicated multipliers)
- **No** comprehensive comparison of **low-area implementations**

New in Round 3

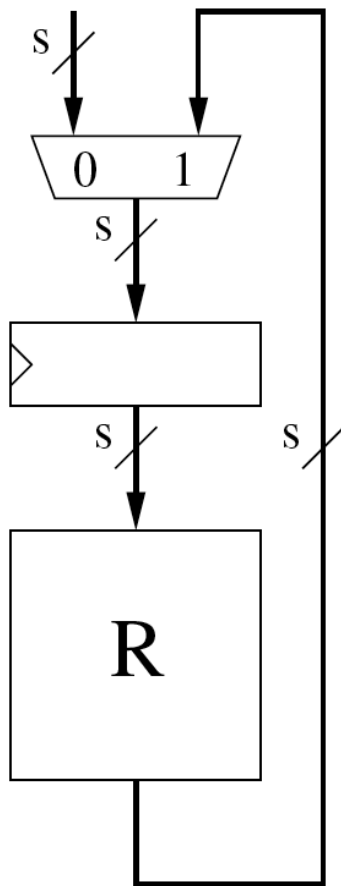
- **Multiple hardware architectures**
- **Pipelining**
- **Flexibility**



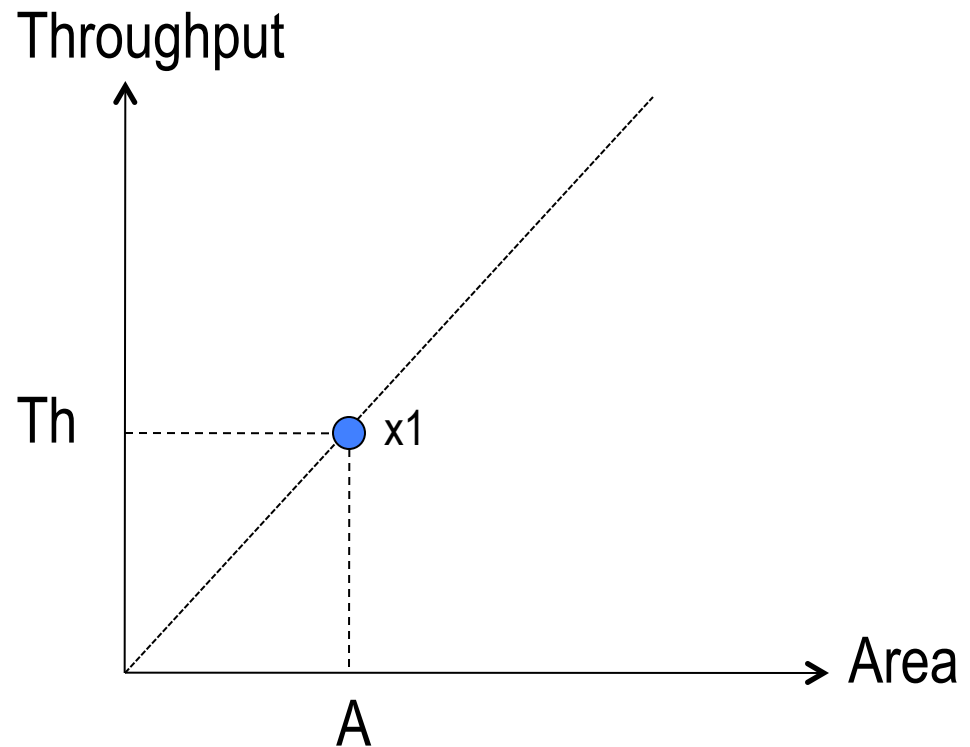
SHA-3 Contest Finalists



Starting Point: Basic Iterative Architecture

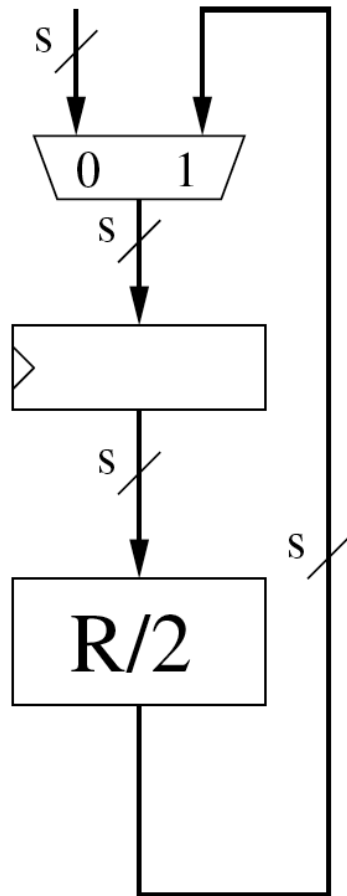


- datapath width = state size
- one clock cycle per one round/step

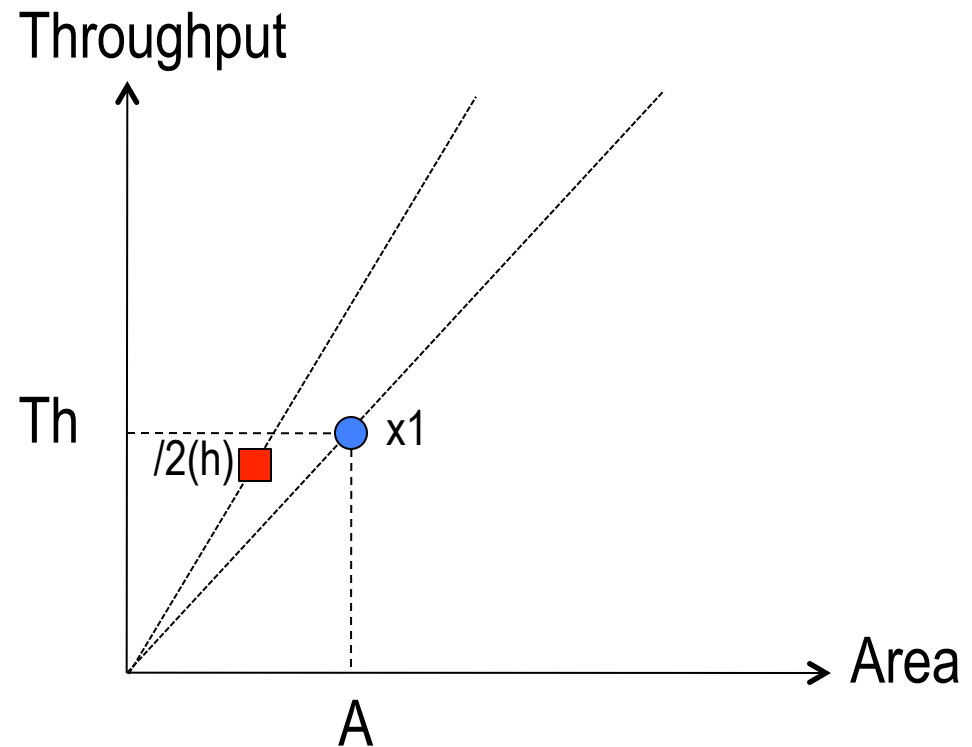


Currently, most common architecture used to implement SHA-1, SHA-2, and many other hash functions.

Horizontal Folding - $/2(h)$

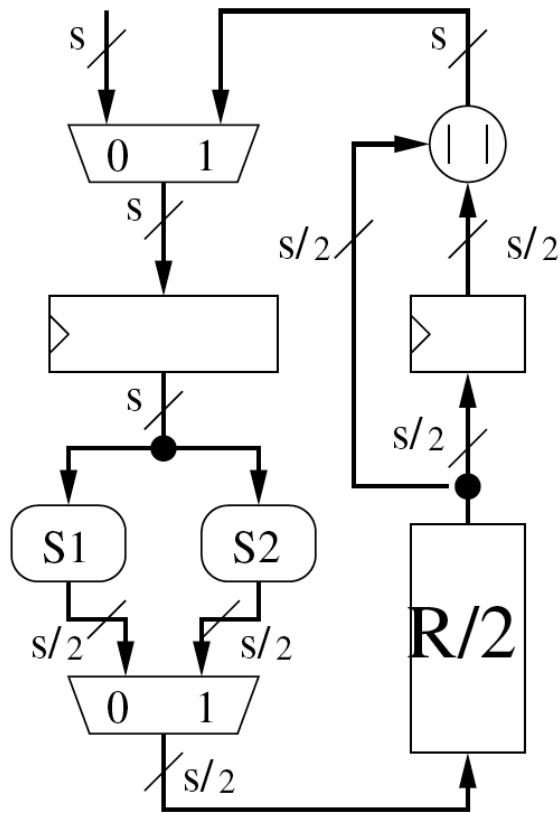


- datapath width = state size
- two clock cycles per one round/step

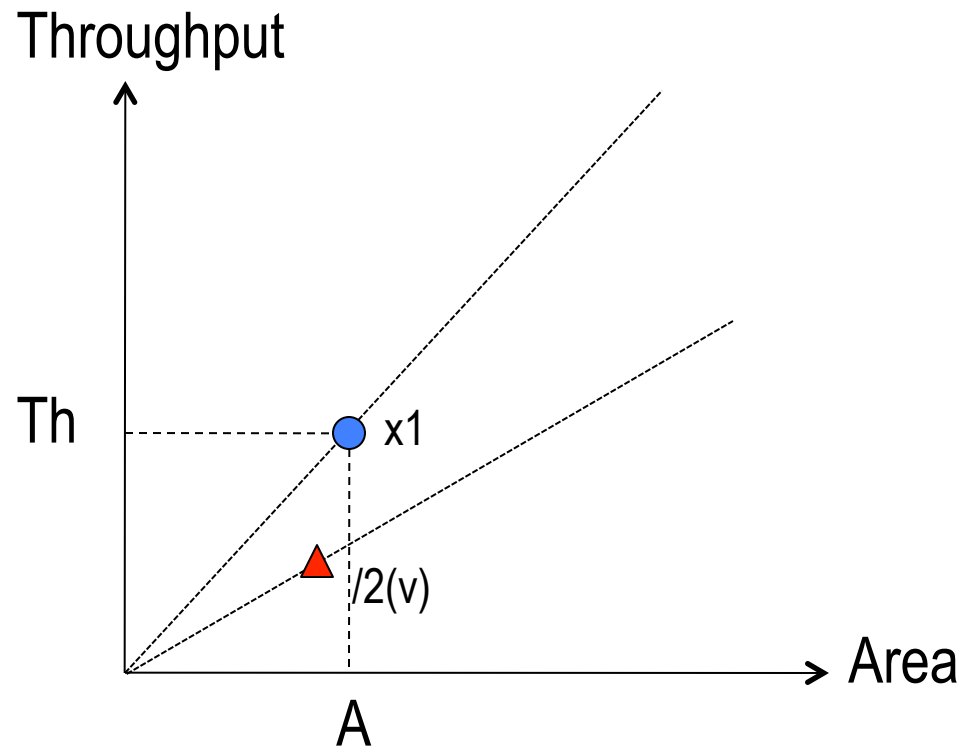


Typically Throughput/Area ratio increases

Vertical Folding - $/2(v)$

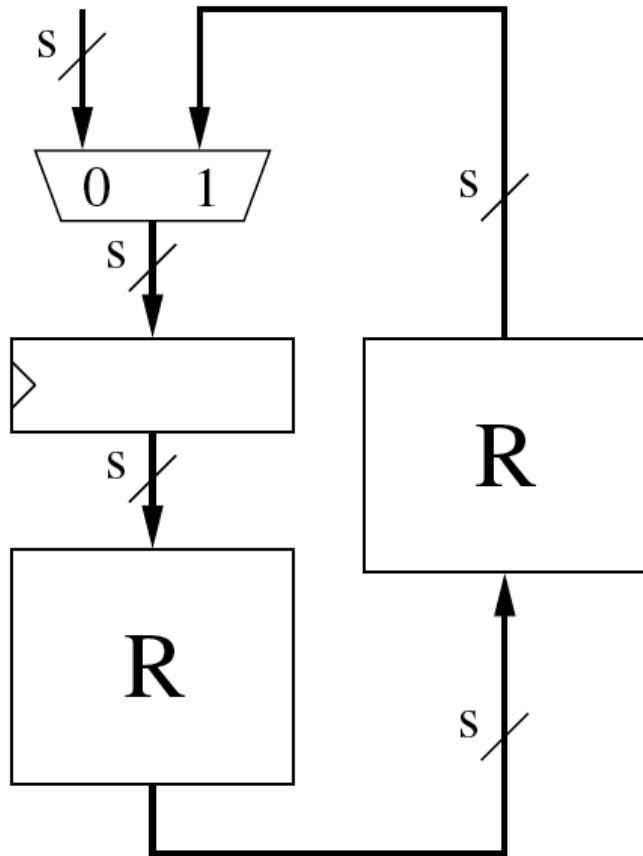


- datapath width = state size/2
- two clock cycles per one round/step

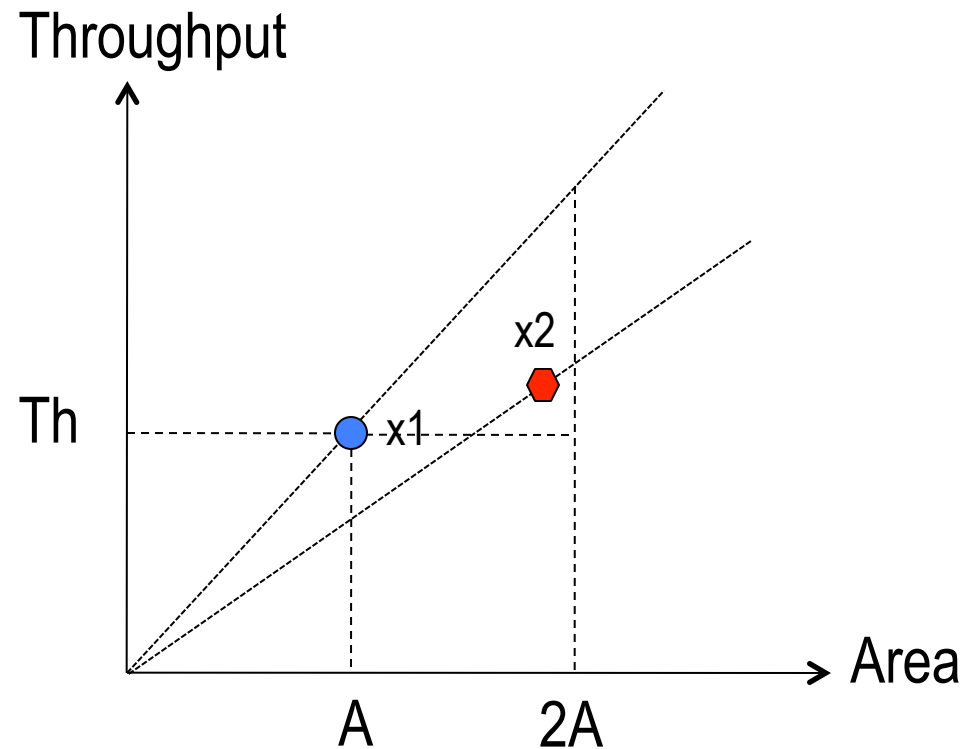


Typically Throughput/Area ratio decreases

Unrolling - x2

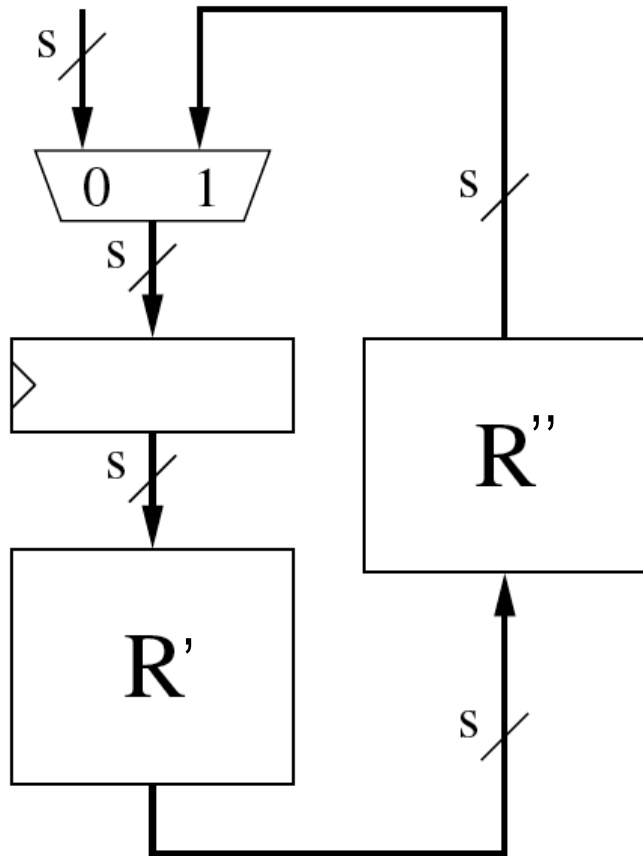


- datapath width = state size
- one clock cycle per two rounds

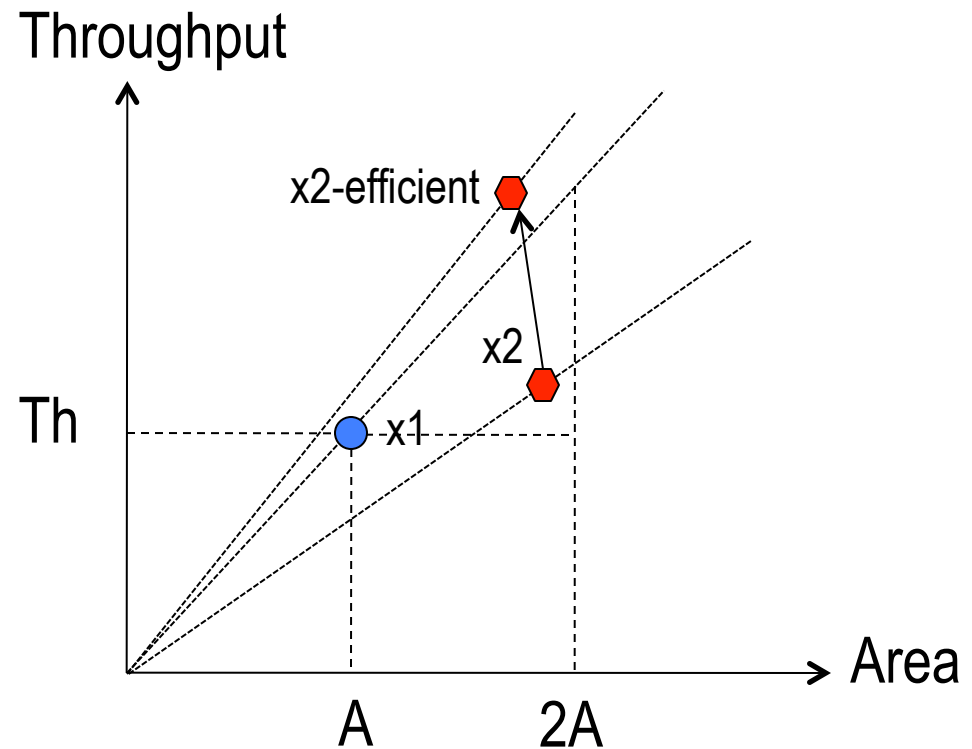


Typically Throughput/Area ratio decreases

Efficient Unrolling - x2

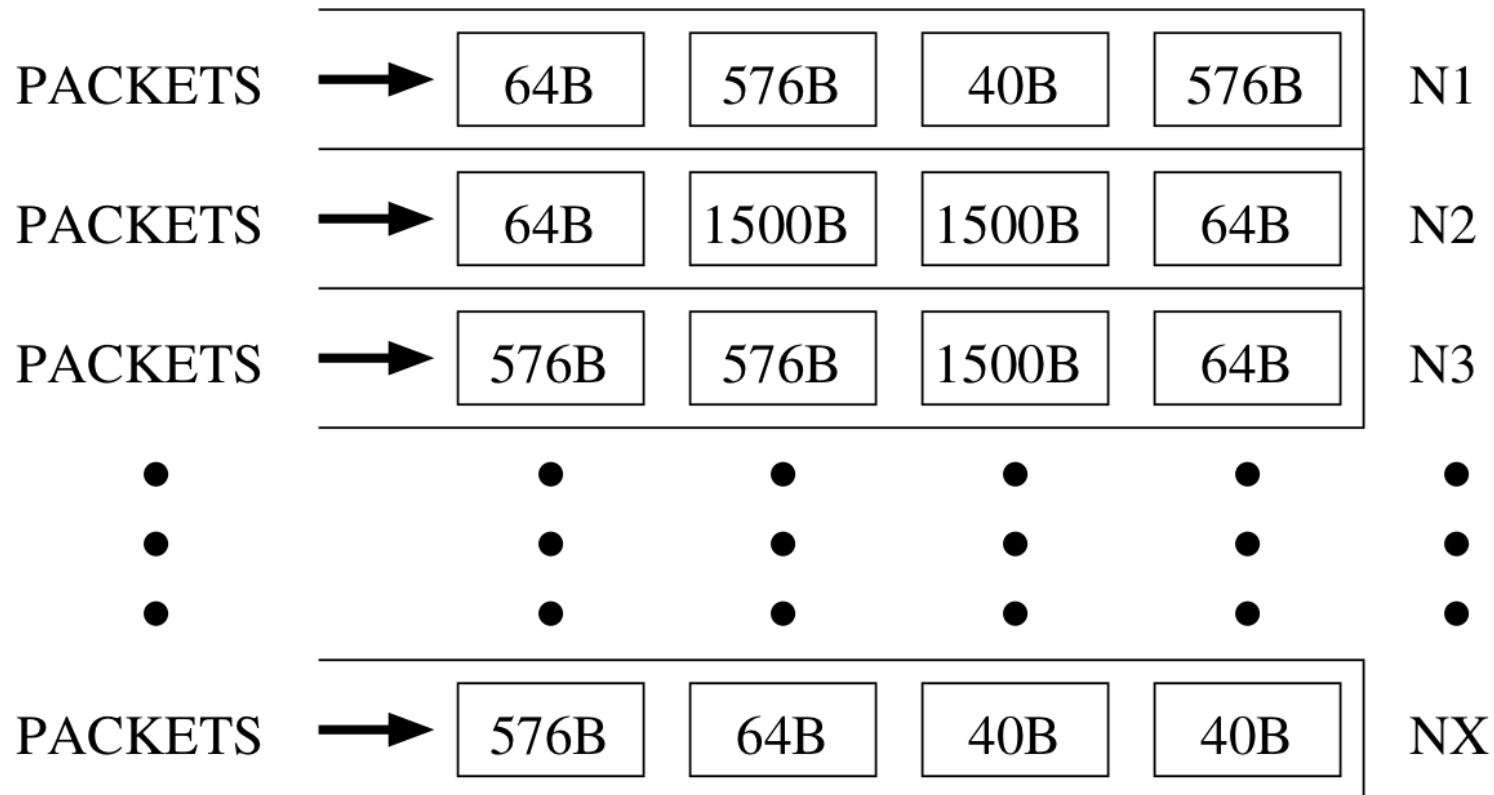


- datapath width = state size
- one clock cycle per two rounds



Sometimes Throughput/Area ratio increases

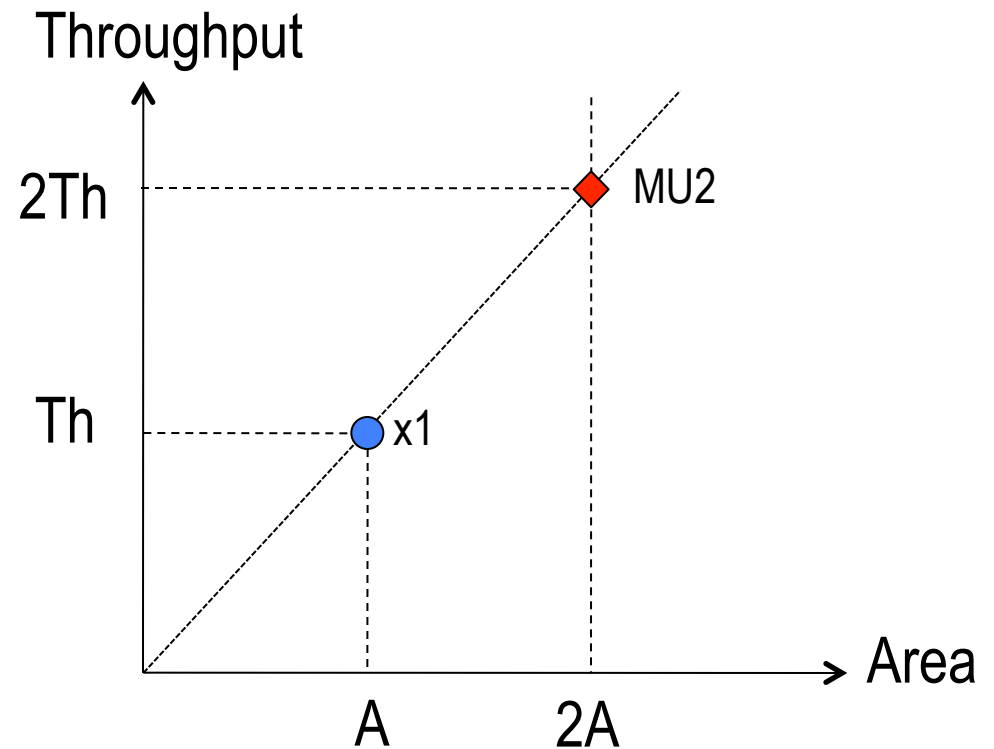
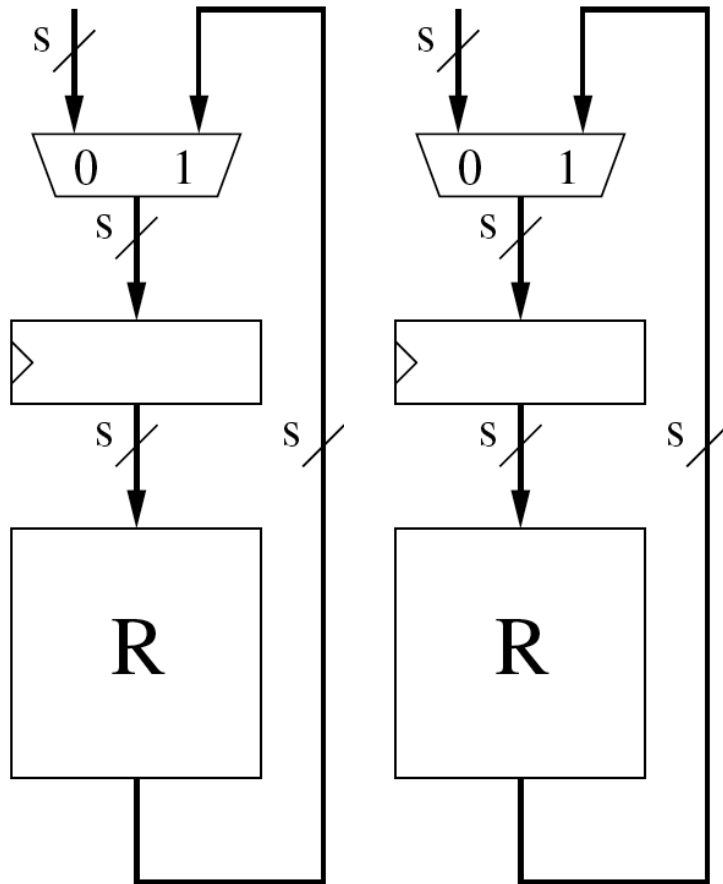
Multiple Packets Available for Parallel Processing



Typical sizes of packets: 40B – 1500B

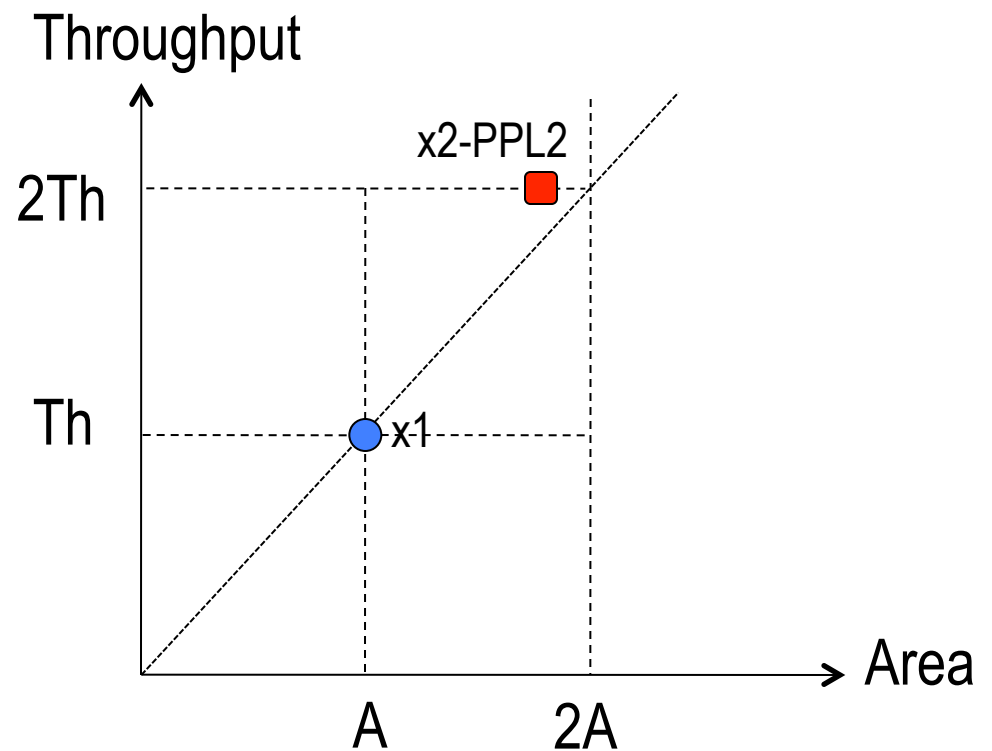
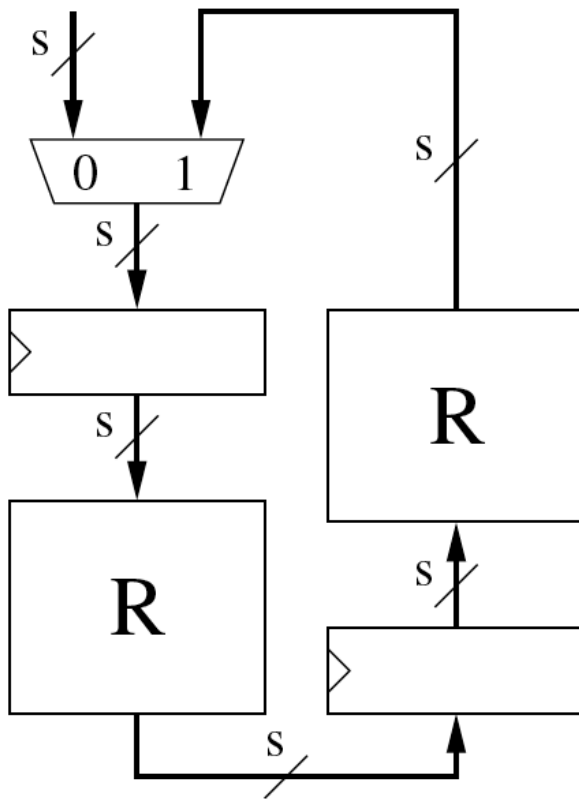
1500 B = Maximum Transmission Unit (MTU) for Ethernet v2

Parallel Processing Using Multi-Unit Architecture – MU2



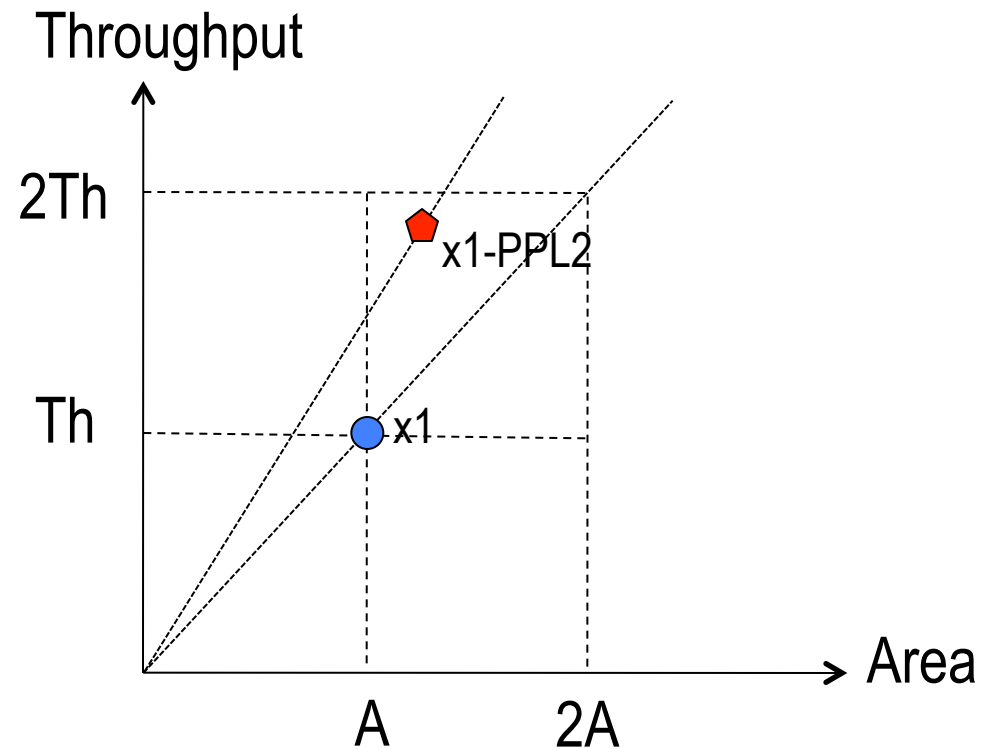
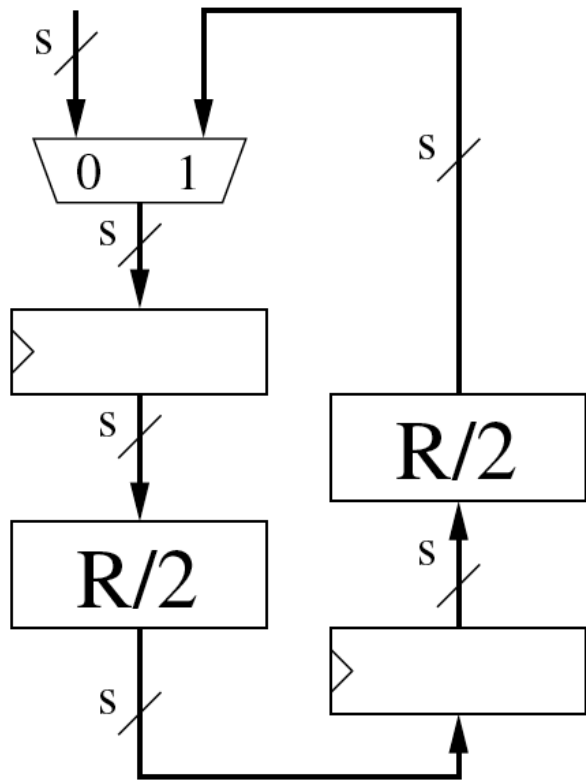
Typically Throughput/Area ratio stays the same

Unrolled Architecture with Pipelining - x2-PPL2



Typically Throughput/Area ratio stays almost the same

Basic Architecture with Pipelining - x1-PPL2



Typically Throughput/Area ratio increases

Results

Comprehensive Evaluation

- two major vendors: Altera and Xilinx (~90% of the market)
- two most recent high-performance families

	Altera		Xilinx	
Technology	Low-cost	High-performance	Low-cost	High-performance
90 nm	Cyclone II	Stratix II	Spartan 3	Virtex 4
65 nm	Cyclone III	Stratix III		Virtex 5
40-60 nm	Cyclone IV	Stratix IV	Spartan 6	Virtex 6

Generation of Results Facilitated by ATHENa



ATHENa – Automated Tool for Hardware Evaluation
Benchmarking tool developed at GMU since 2009

- batch mode of FPGA tools

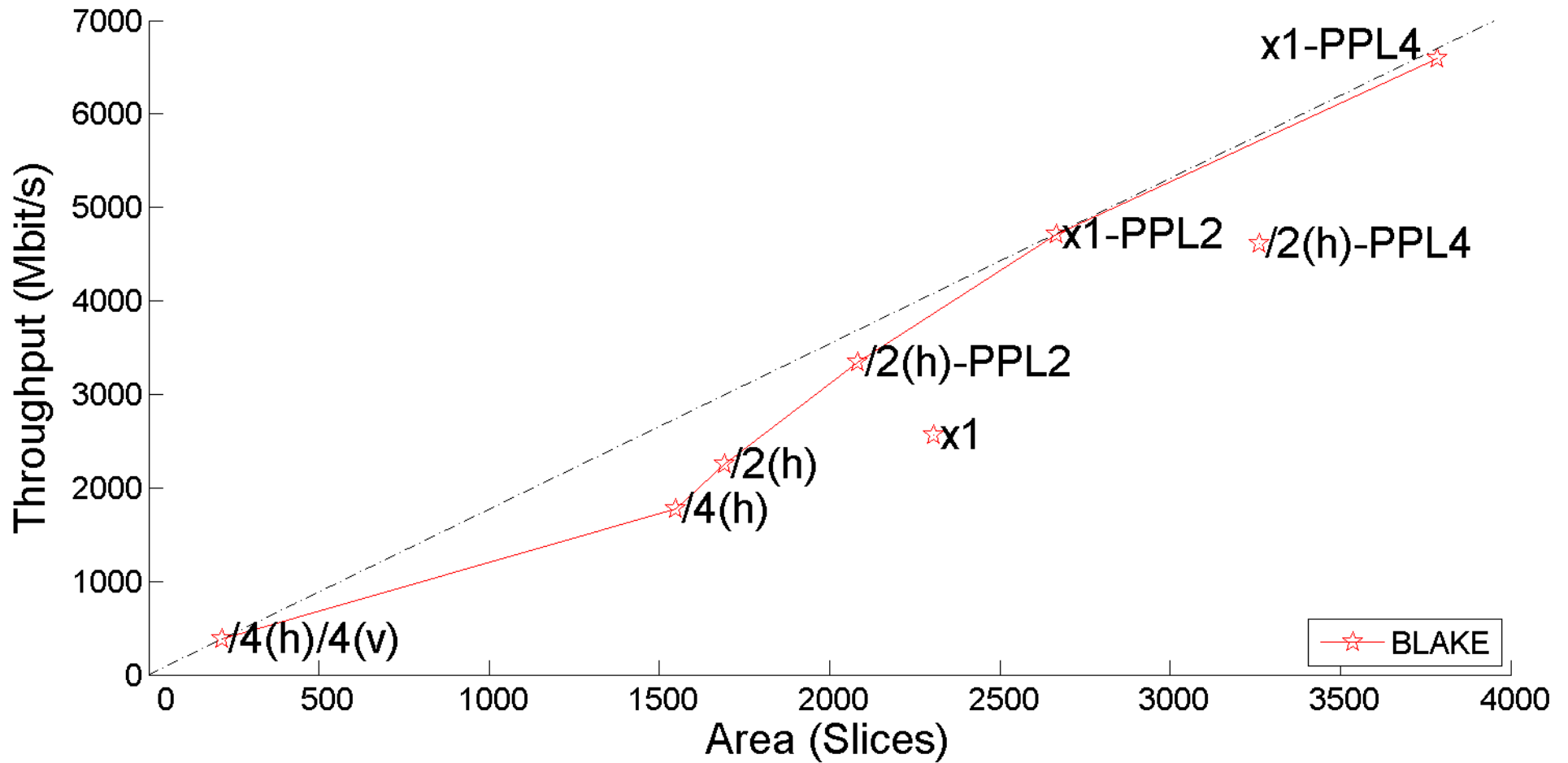


vs.

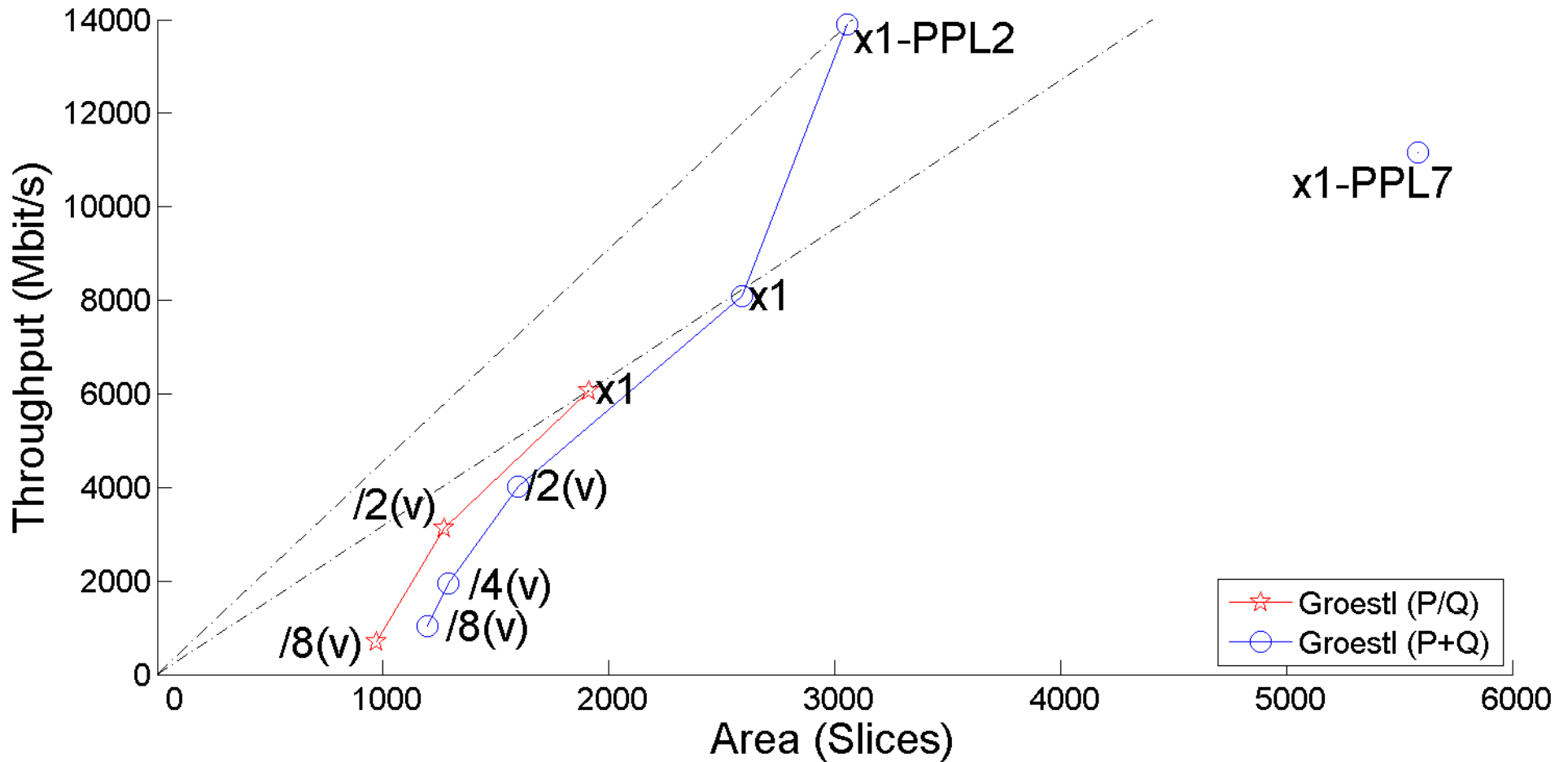


- ease of extraction and tabulation of results (Excel, CSV)
- optimized choice of tool options

BLAKE-256 in Virtex 5



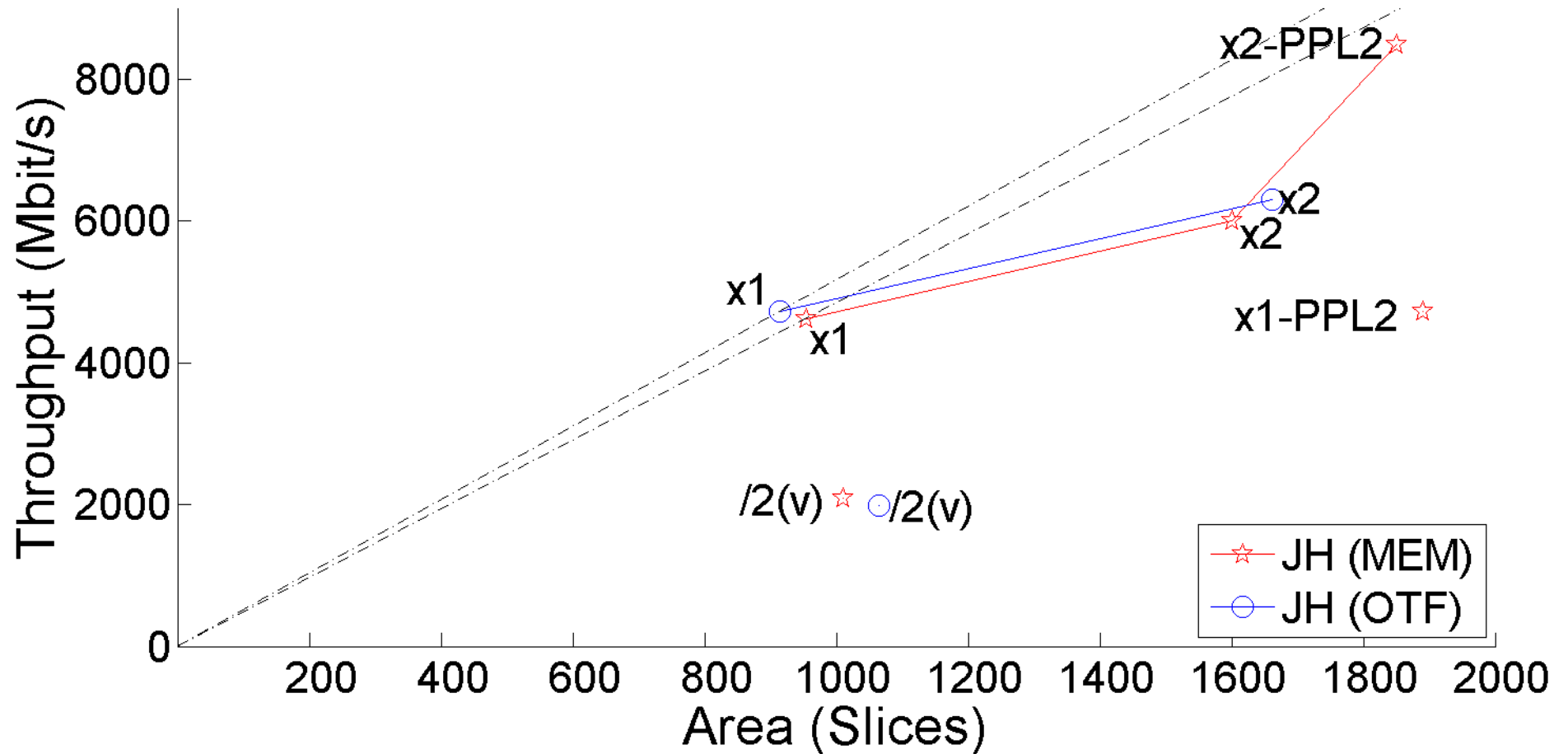
Groestl-256 in Virtex 5



Groestl P/Q – quasi-pipelined architecture; one unit shared between P and Q

Groestl P+Q – parallel architecture; two independent units for P and Q

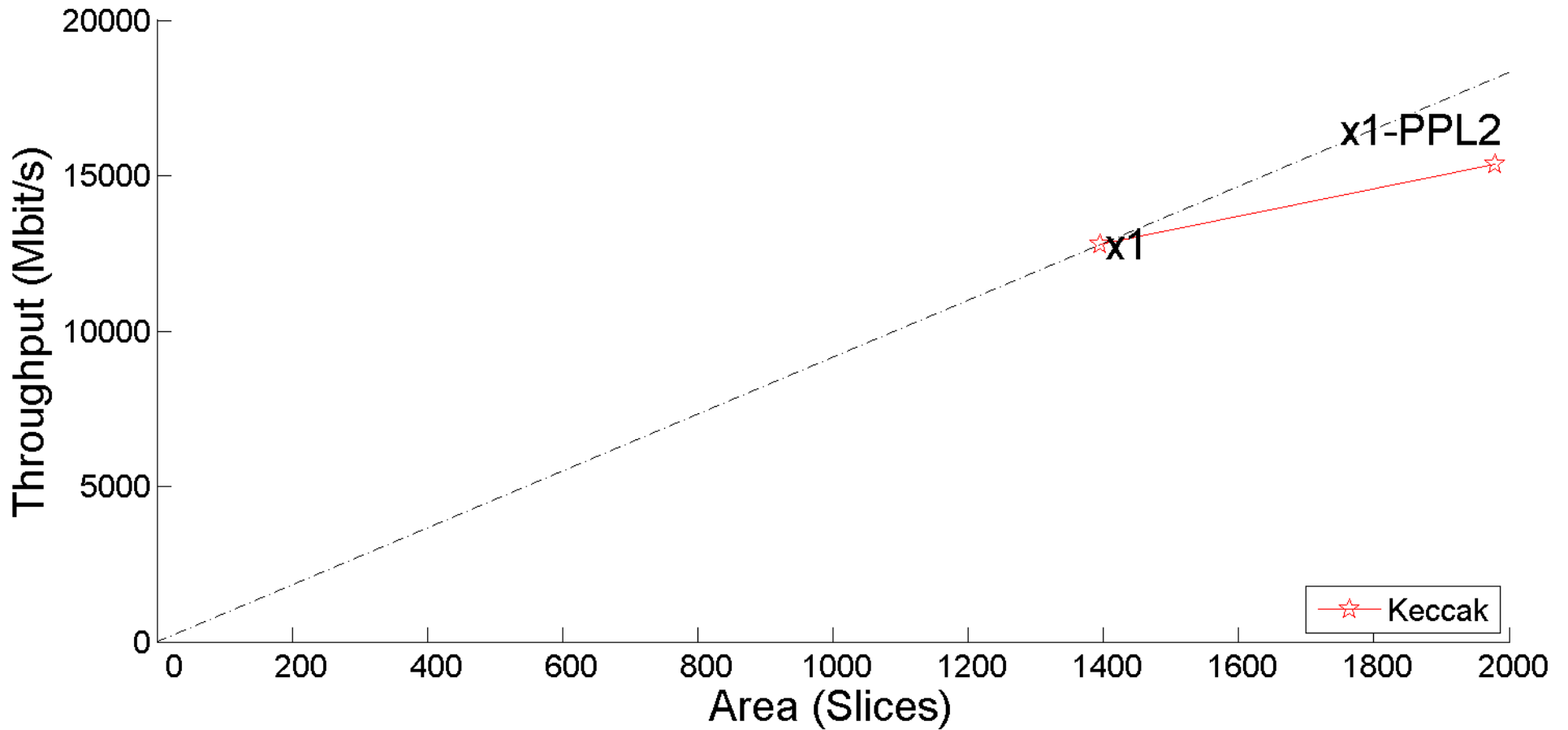
JH-256 in Virtex 5



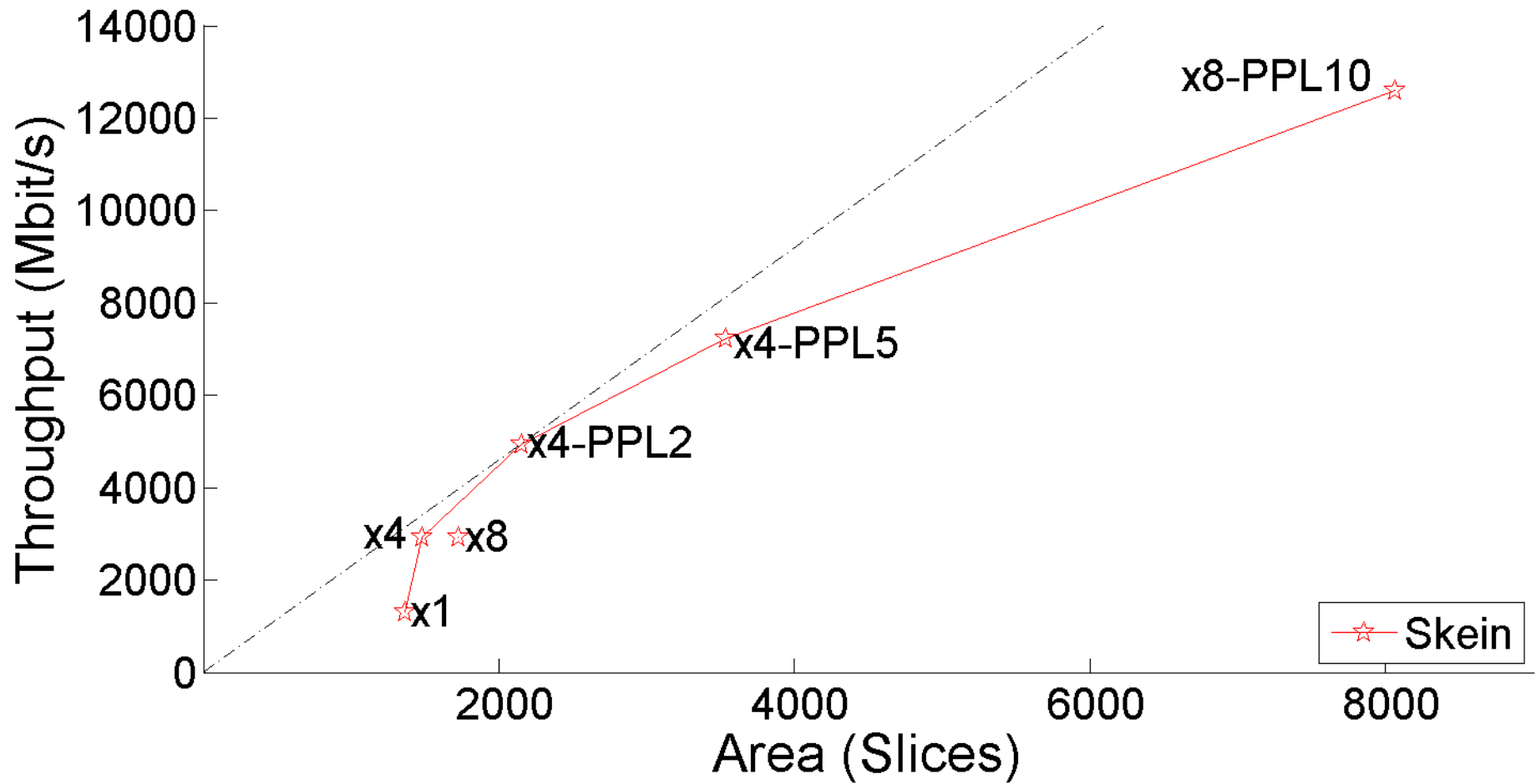
JH MEM – round constants stored in memory

JH OTF – round constants computed on-the-fly

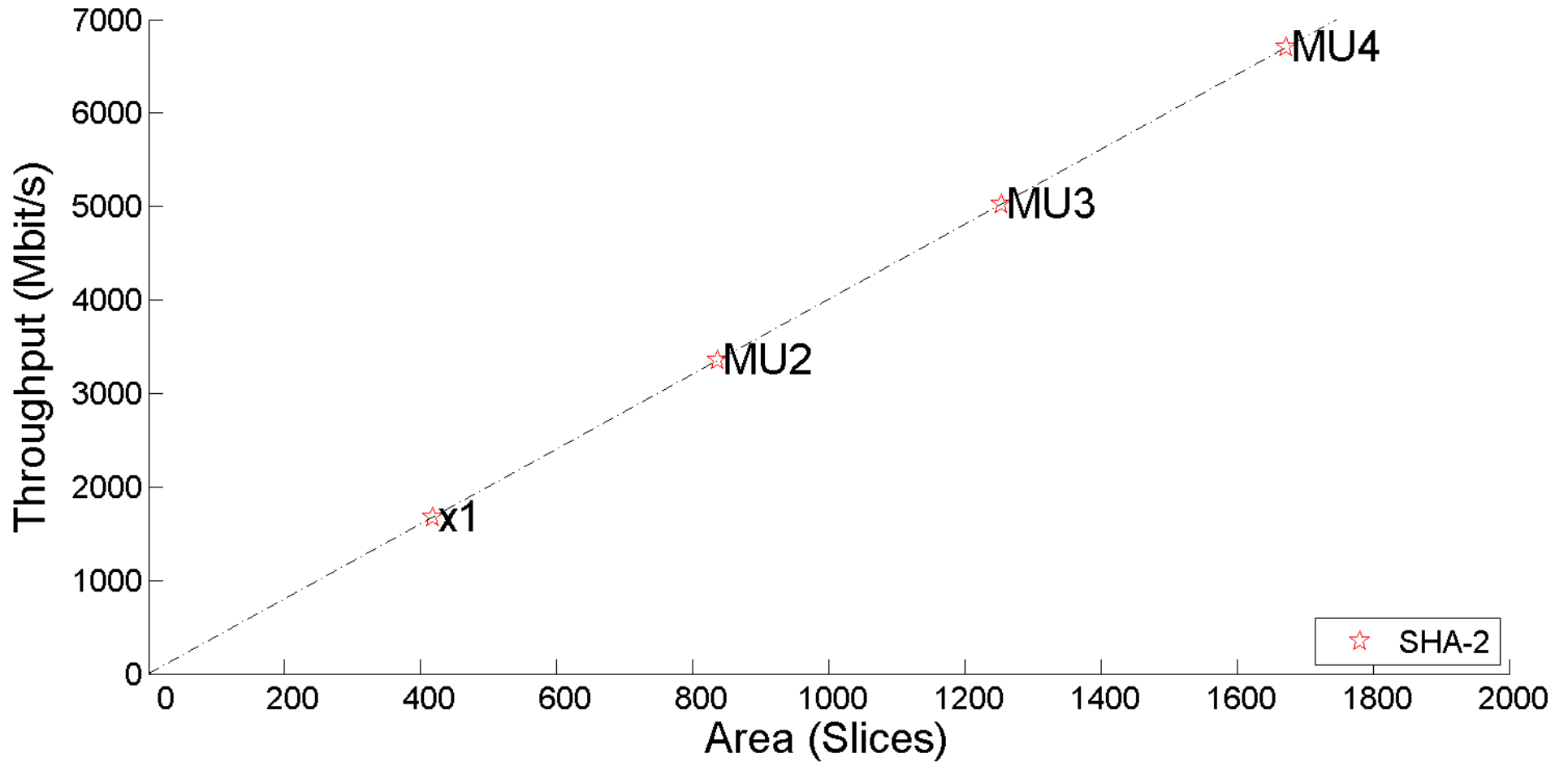
Keccak-256 in Virtex 5



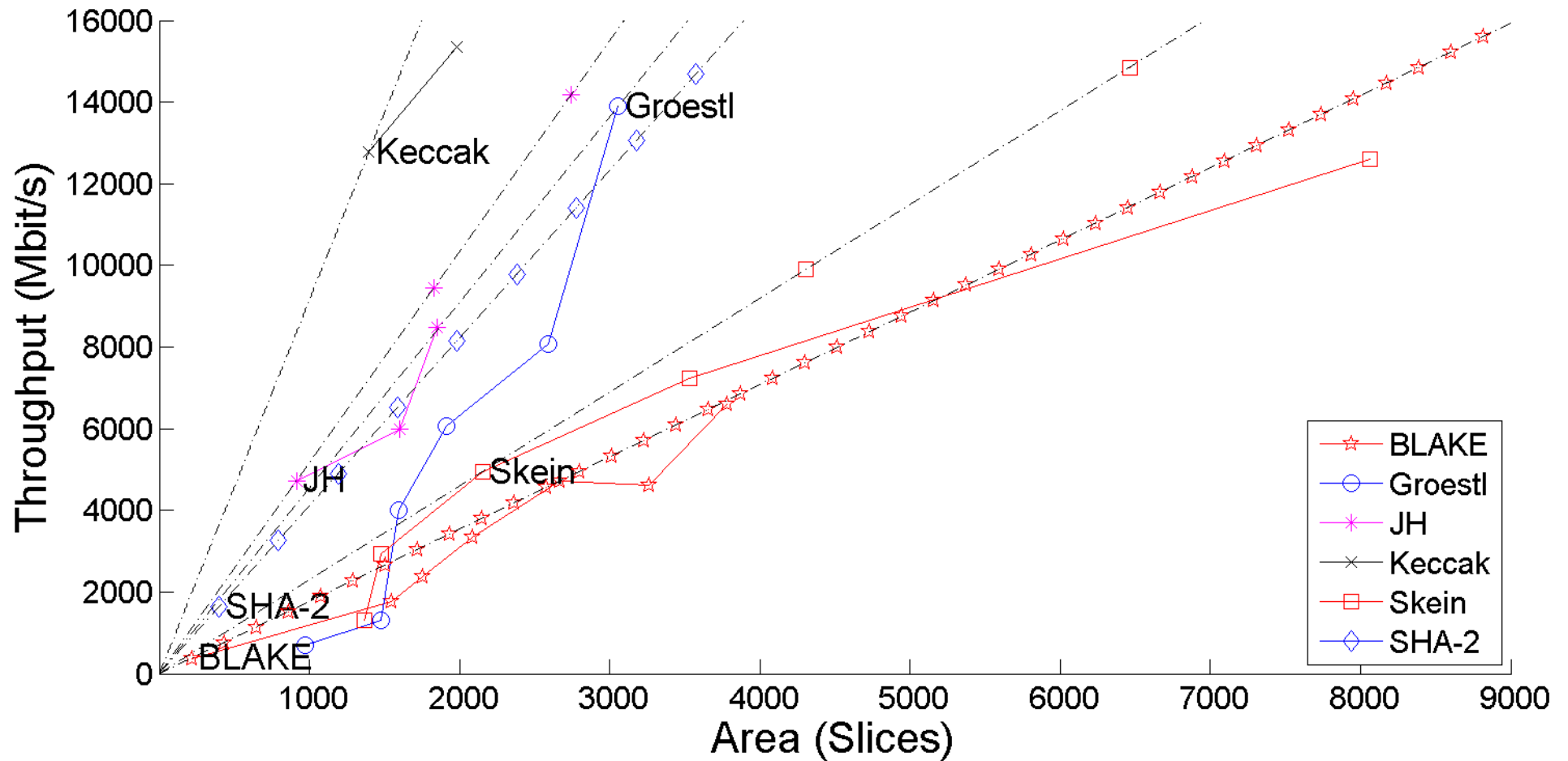
Skein-256 in Virtex 5



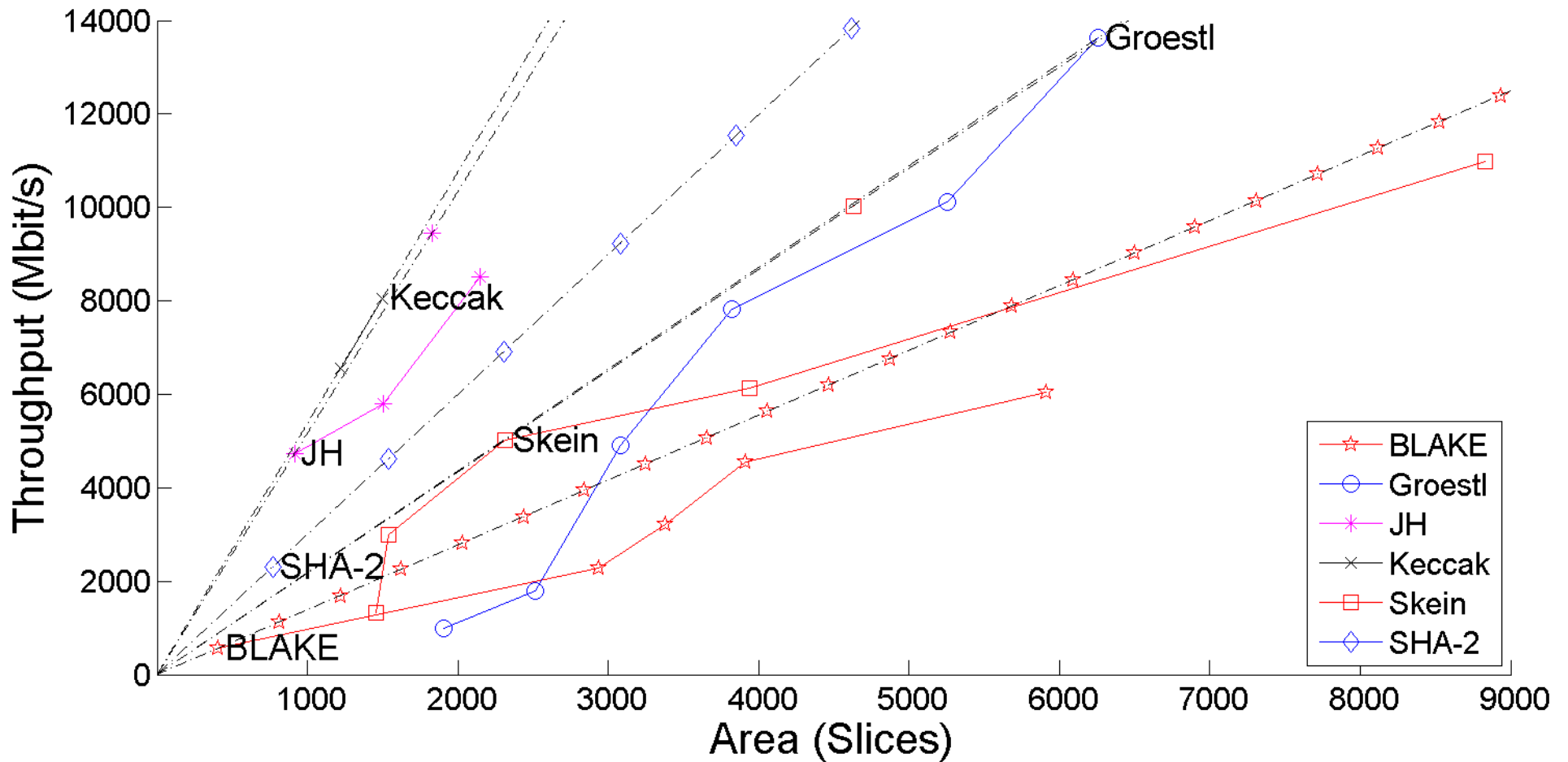
SHA-256 in Virtex 5



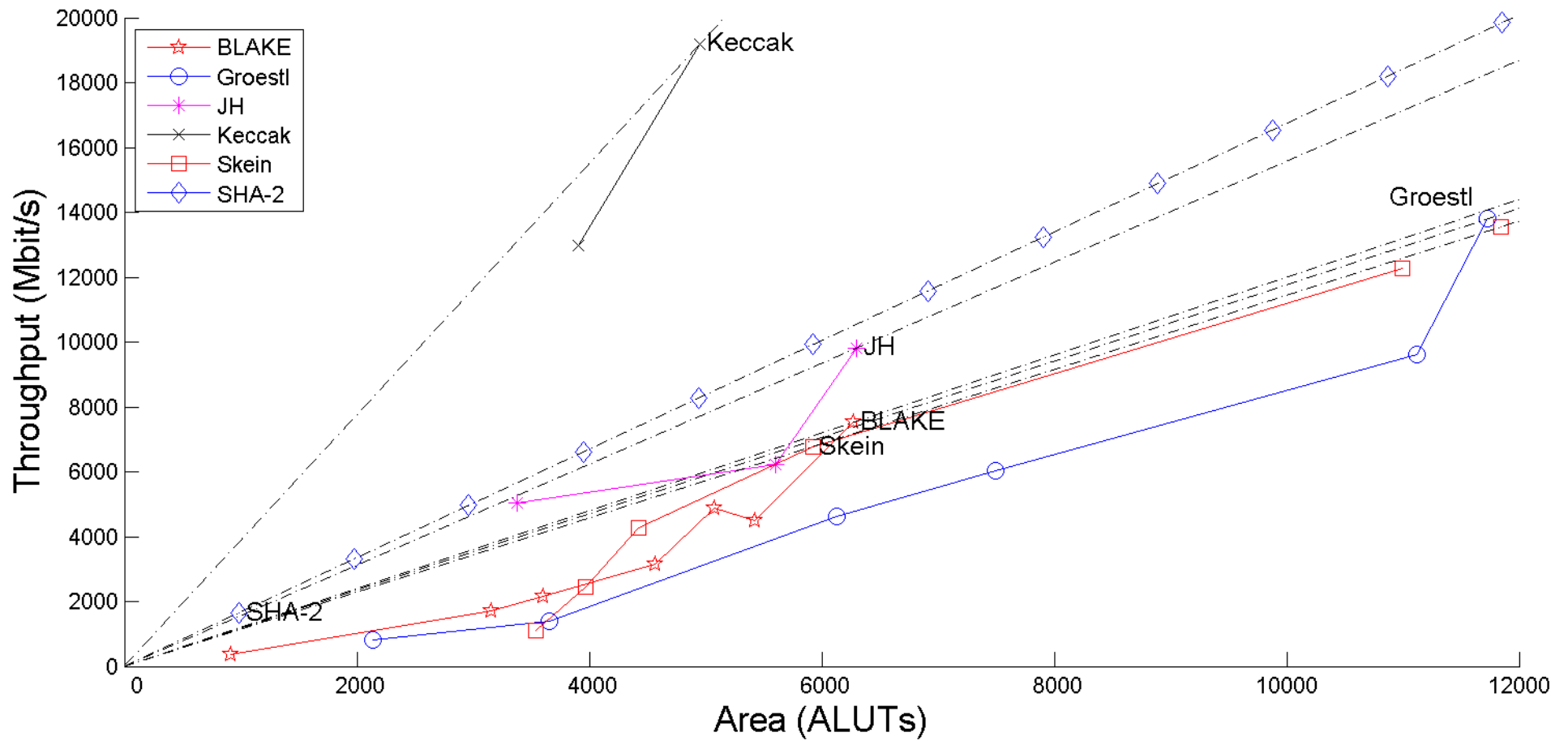
256-bit variants in Virtex 5



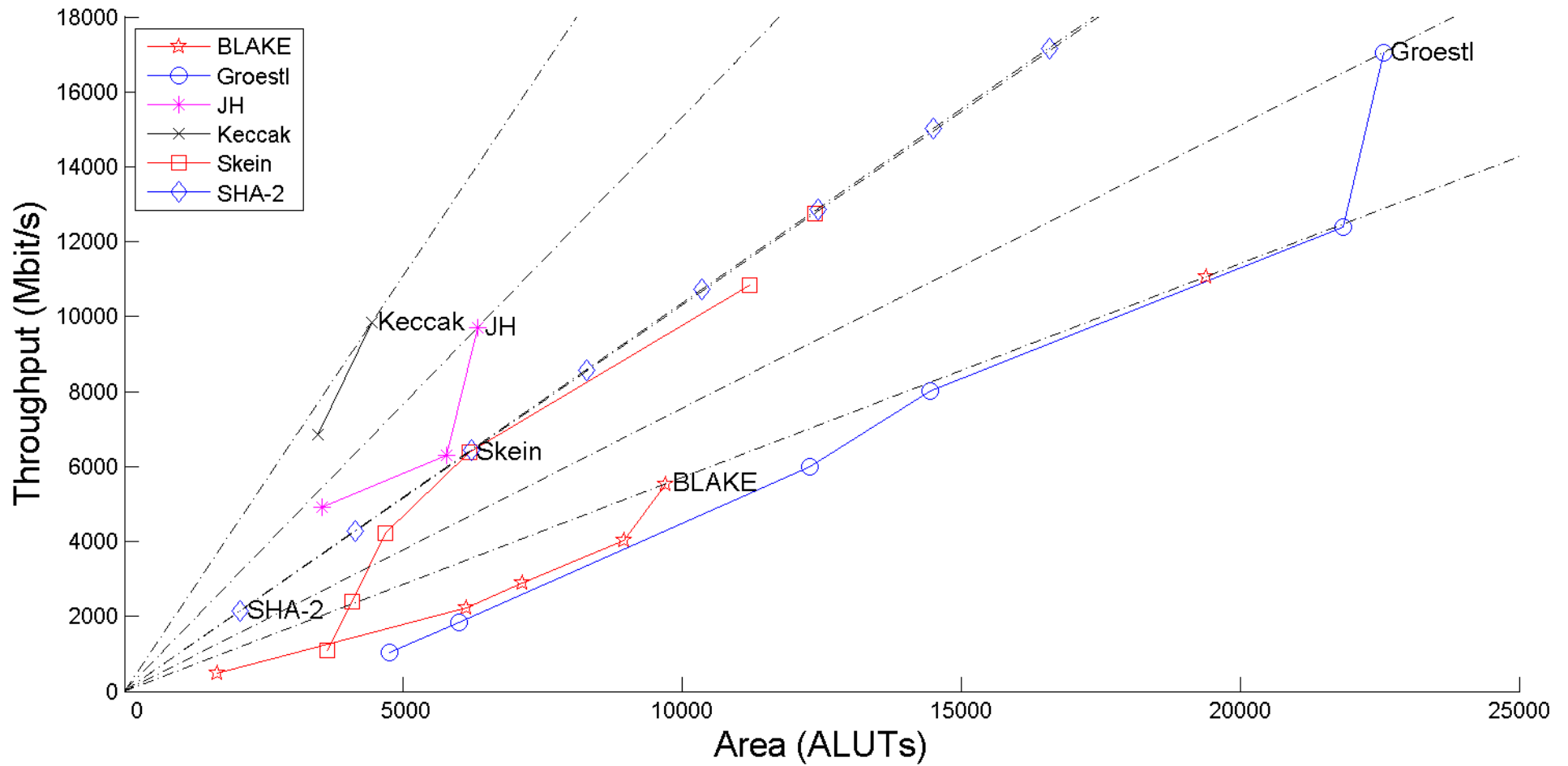
512-bit variants in Virtex 5



256-bit variants in Stratix III



512-bit variants in Stratix III



Selected Most Efficient Architectures

Algorithm	Iterative	Folded			Pipelined			Efficient Unrolled
		Horizontally	Vertically	Mixed	Unrolled	Basic	Folded	
BLAKE	x1	/2(h), /4(h)		/4(h)/4(v)*		x1-PPL2, x1-PPL4	/2(h)-PPL2, /2(h)-PPL4	
Groestl	x1*		/2(v), /4(v), /8(v)			x1-PPL2, x1-PPL7		
JH	x1*		/2(v)		x2-PPL2			
Keccak	x1*					x1-PPL2		
Skein	x1				x4-PPL2, x4-PPL5			x4*

ARCH_SYMBOL* - the best non-pipelined architecture

BLAKE – most flexible, **Keccak** – least flexible

Best Architectures for 256-bit Variants

Algorithm	Xilinx		Altera	
	Virtex 5	Virtex 6	Stratix III	Stratix IV
Best non-pipelined architecture				
BLAKE	/4(h)/4(v)	/4(h)/4(v)	/2(h)	/2(h)
Groestl	x1 (P/Q)	x1 (P+Q)	x1 (P+Q)	x1 (P+Q)
JH	x1 (OTF)	x1 (MEM)	x1 (MEM)	x1 (MEM)
Keccak	x1	x1	x1	x1
Skein	x4	x4	x4	x4
Overall best architecture				
BLAKE	/4(h)/4(v)	x1-PPL4	x1-PPL4	x1-PPL4
Groestl	x1-PPL2 (P+Q)	x1-PPL2 (P+Q)	x1-PPL2 (P+Q)	x1-PPL2 (P+Q)
JH	x1 (OTF)	x1 (MEM)	x1 (MEM)	x1 (MEM)
Keccak	x1	x1-PPL2	x1-PPL2	x1-PPL2
Skein	x4-PPL2	x4-PPL2	x4-PPL5	x4-PPL5

Best Architectures for 512-bit Variants


Algorithm	Xilinx		Altera	
	Virtex 5	Virtex 6	Stratix III	Stratix IV
Best non-pipelined architecture				
BLAKE	/4(h)/4(v)	/4(h)/4(v)	/2(h)	/2(h)
Groestl	x1 (P+Q)	x1 (P/Q)	x1 (P+Q)	x1 (P+Q)
JH	x1 (OTF)	x1 (MEM)	x1 (MEM)	x1 (MEM)
Keccak	x1	x1	x1	x1
Skein	x4	x4	x4	x4
Overall best architecture				
BLAKE	/4(h)/4(v)	/4(h)/4(v)	/2(h)-PPL4	/2(h)-PPL4
Groestl	x1-PPL2 (P+Q)	x1 (P/Q)	x1-PPL2 (P+Q)	x1-PPL2 (P+Q)
JH	x1 (OTF)	x1 (MEM)	x1 (MEM)	x1 (MEM)
Keccak	x1-PPL2	x1-PPL2	x1-PPL2	x1-PPL2
Skein	x4-PPL2	x4-PPL2	x4-PPL5	x4-PPL5

Summary

- Keccak** – consistently outperforms SHA-2; front runner for high-speed implementations, but not suitable for folding
- JH** – performs better than SHA-2 most of the time, not suitable for inner-round pipelining
- Groestl** – better than SHA-2 for only one out of four FPGA families, and only with relatively large area; suitable for vertical folding
- Skein** – the only candidate benefiting from unrolling; easy to pipeline after unrolling
- BLAKE** – most flexible; can be folded horizontally and vertically, can be effectively pipelined, however relatively slow compared to other candidates.

Conclusions

- Using multiple architectures provides a more comprehensive view of the algorithms
- Algorithms differ substantially in terms of their flexibility and suitability for folding, unrolling, and pipelining
- Optimum architecture (including an optimum number of pipeline stages) may depend on FPGA family
- For most families, pipelined architectures the best in terms of the throughput to area ratio for 4 out of 5 candidates
- **Two front-runners:** **Keccak, JH**



**Reproducibility
of
Results**

GMU Source Codes and Block Diagrams

- First batch of **GMU Source Codes** for
all Round 3 SHA-3 Candidates & SHA-2
made available at the ATHENa website at:
<http://cryprography.gmu.edu/athena>
- Included in this release:
 - **Basic architectures**
 - **Folded architectures**
 - **Unrolled architectures**
 - Each code supports **two variants**:
with **256-bit** and **512-bit** output.
 - Each source code accompanied by comprehensive
hierarchical block diagrams

Details of Results and Replication Scripts

- Currently available in the **ATHENa database** at <http://cryptography.gmu.edu/athena>
 - 600+ optimized results**
 - for**
 - 16 hash functions**
 - 50+ designs**
 - 11 FPGA families**
- **Scripts and configuration files sufficient to easily reproduce all results (without repeating optimizations)**
- **Automatically created by ATHENa and stored in ATHENa Database**

Future & Parallel Work

- Adding **padding units** to all architectures
- **Optimization of pipelined architectures**
- Extended analysis of performance for **short messages**
- **Experimental testing** using high-performance FPGA boards

In parallel at GMU:

- Study of **low-area architectures** – **Indocrypt 2011, 11-14 Dec**
- Evaluating influence of **embedded resources**
(DSP units, multipliers, block memories) - **FPT 2011, 12-14 Dec**

Thank you!

Questions?



Questions?

CERG: <http://cryptography.gmu.edu>

ATHENa: <http://cryptography.gmu.edu/athena>