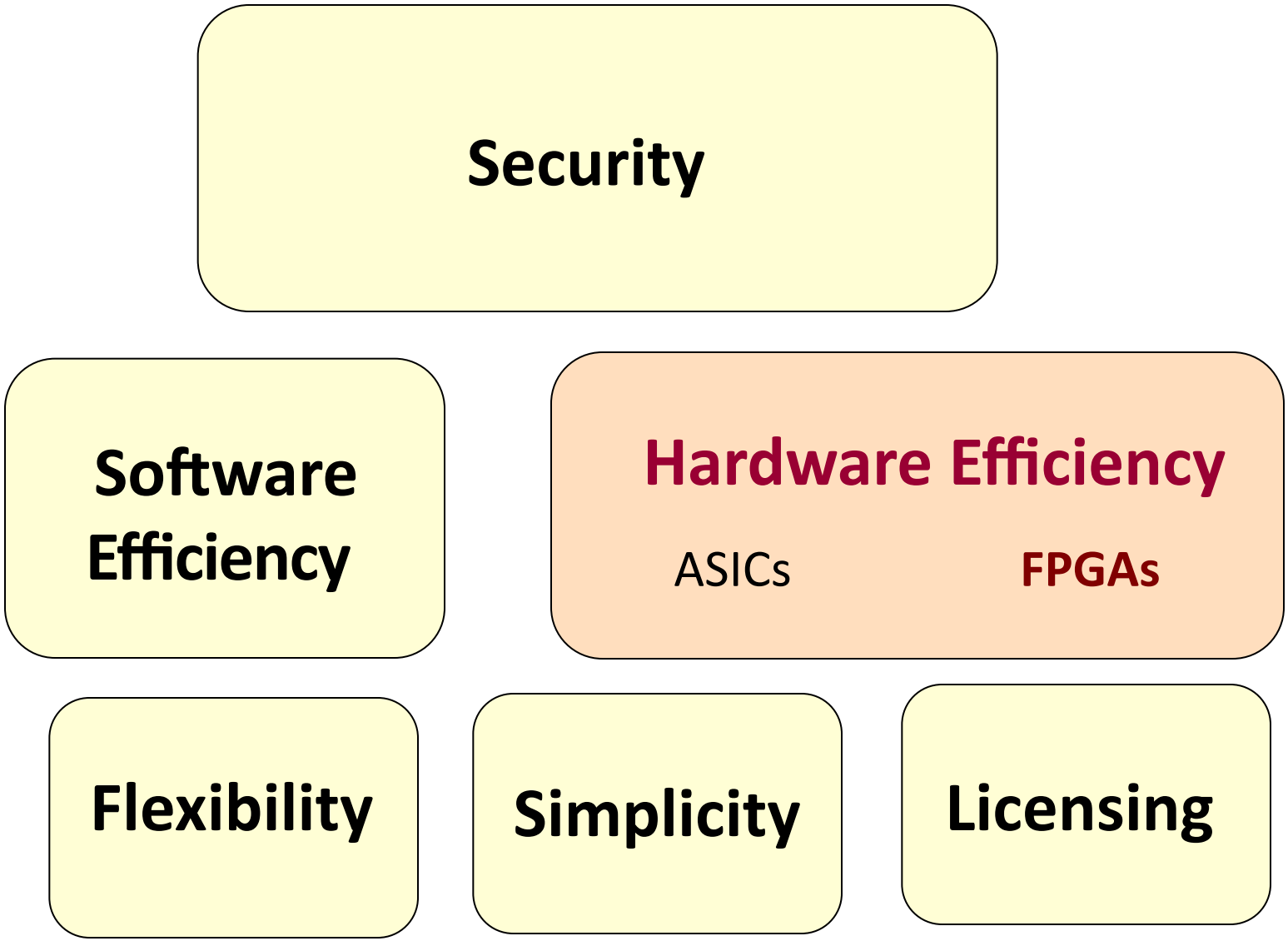
Comparing Hardware Performance of SHA-3 Candidates Using FPGAs

Krzysztof (Kris) Gaj
George Mason University
U.S.A.


Motivation
&
Goals

NIST Evaluation Criteria



Results of Security Evaluation
SHA-3 Zoo Page

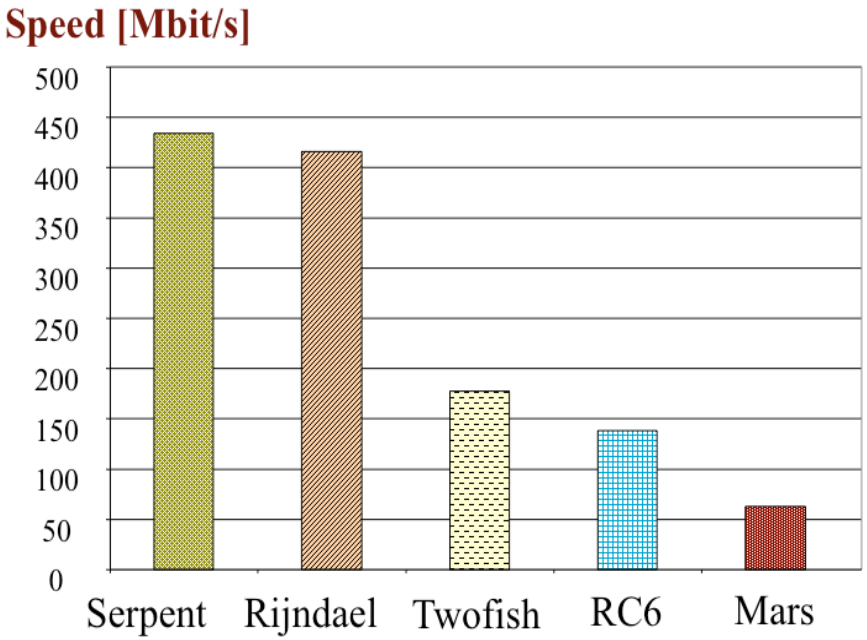
Hash Name	Principal Submitter	Best Attack on Main NIST Requirements	Best Attack on other Hash Requirements
BLAKE	Jean-Philippe Aumasson		
Blue Midnight Wish	Svein Johan Knapskog		
CubeHash	Daniel J. Bernstein	preimage	
ECHO	Henri Gilbert		
Fugue	Charanjit S. Jutla		
Grøstl	Lars R. Knudsen		
Hamsi	Özgül Küçük		
JH	Hongjun Wu	preimage	
Keccak	The Keccak Team		
Luffa	Dai Watanabe		
Shabal	Jean-François Misarsky		
SHAvite-3	Orr Dunkelman		
SIMD	Gaëtan Leurent		
Skein	Bruce Schneier		

Lessons from the Past: AES Contest

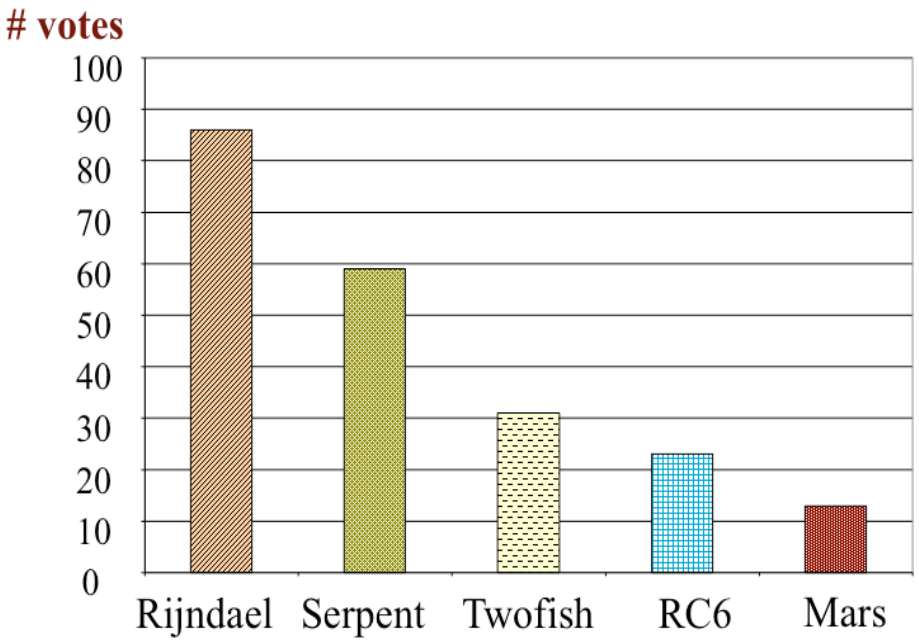
AES Contest – 1997-2000

Round 2 of AES Contest, 2000

Speed in FPGAs

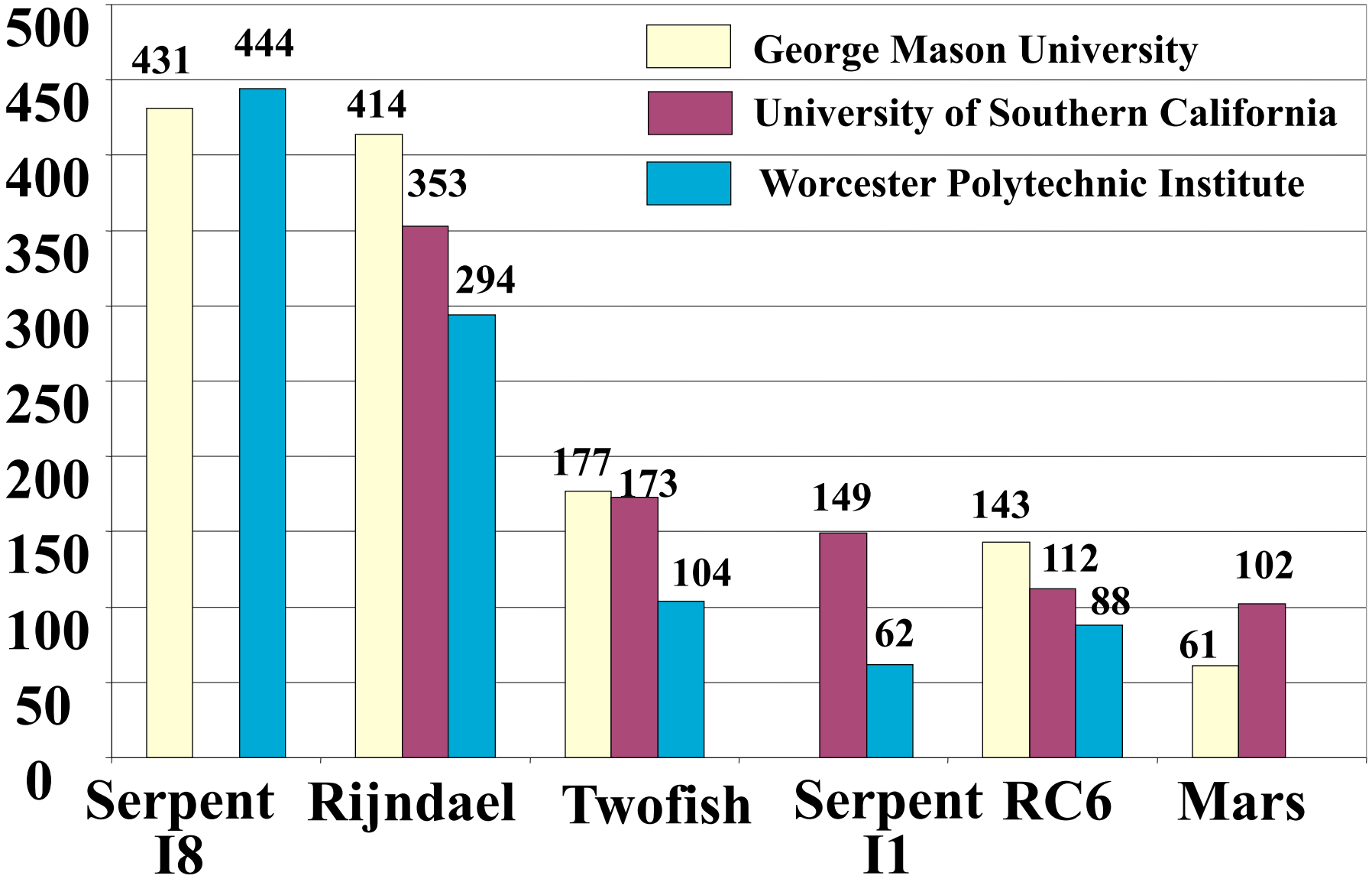


Votes at the AES 3 conference

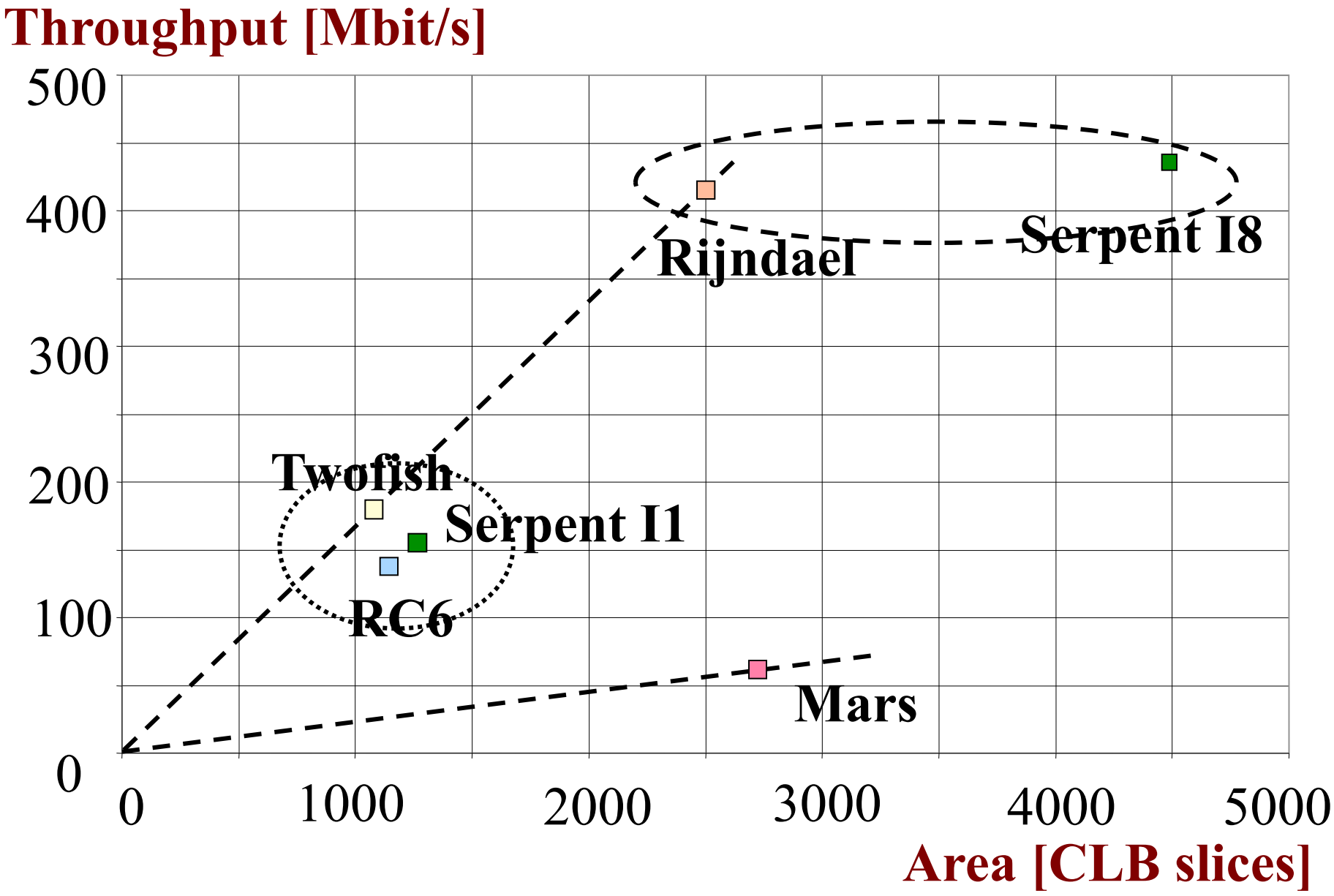


AES Final Candidates: FPGA Virtex 1000: Speed

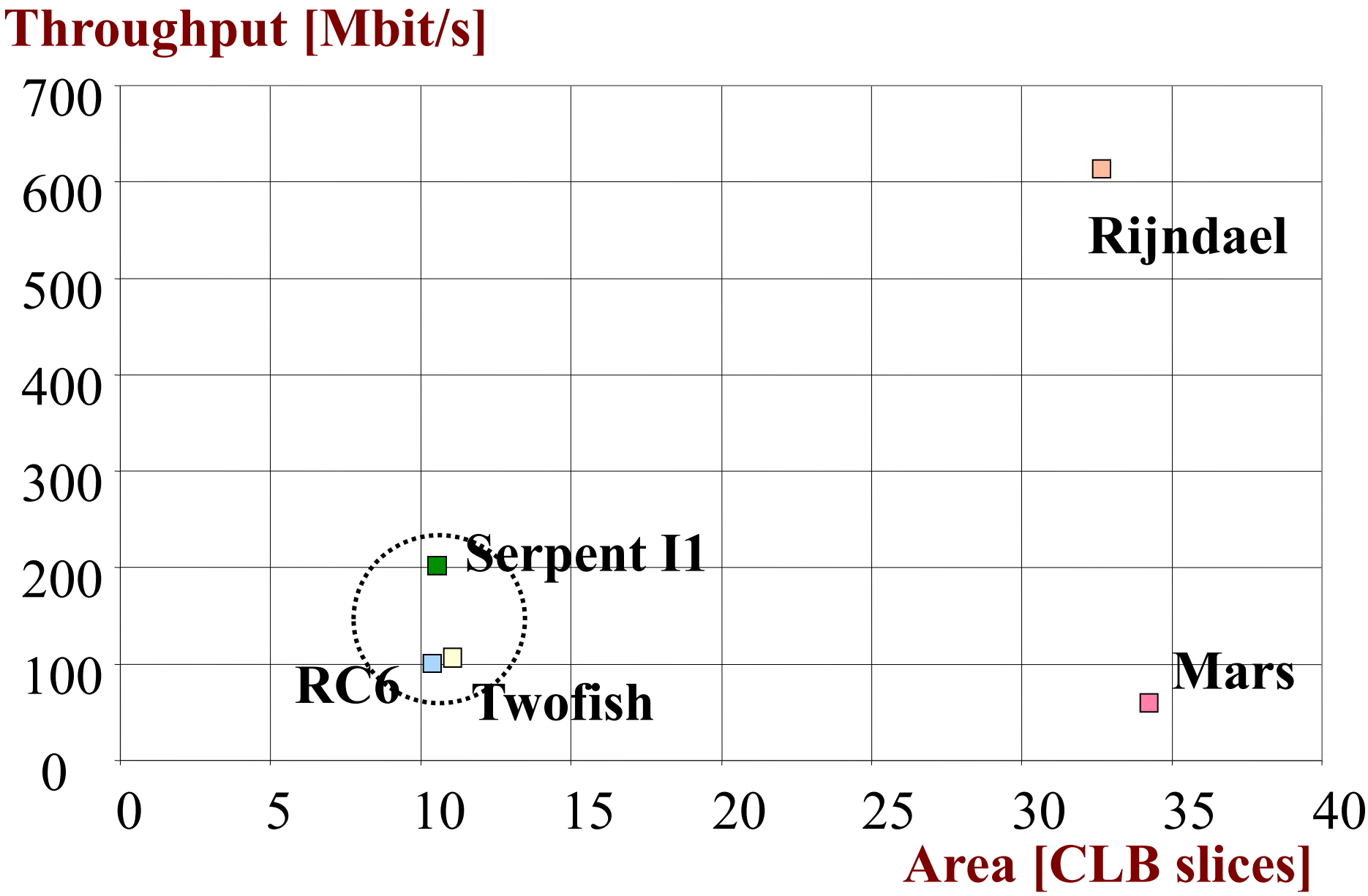
Throughput [Mbit/s]



GMU Results: Encryption in cipher feedback modes (CBC, CFB, OFB) - Virtex FPGA



NSA Results: Encryption in cipher feedback modes (CBC, CFB, OFB) - ASIC, 0.5 μm CMOS



Limitations of the AES Evaluation

- Optimization for maximum throughput in the feedback modes of operatation
- Single high-speed architecture per candidate
- No use of embedded resources of FPGAs (Block RAMs, dedicated multipliers)
- Single FPGA family from a single vendor – Xilinx Virtex


SHA-3 Round 2

Features of the SHA-3 Round 2 Evaluation

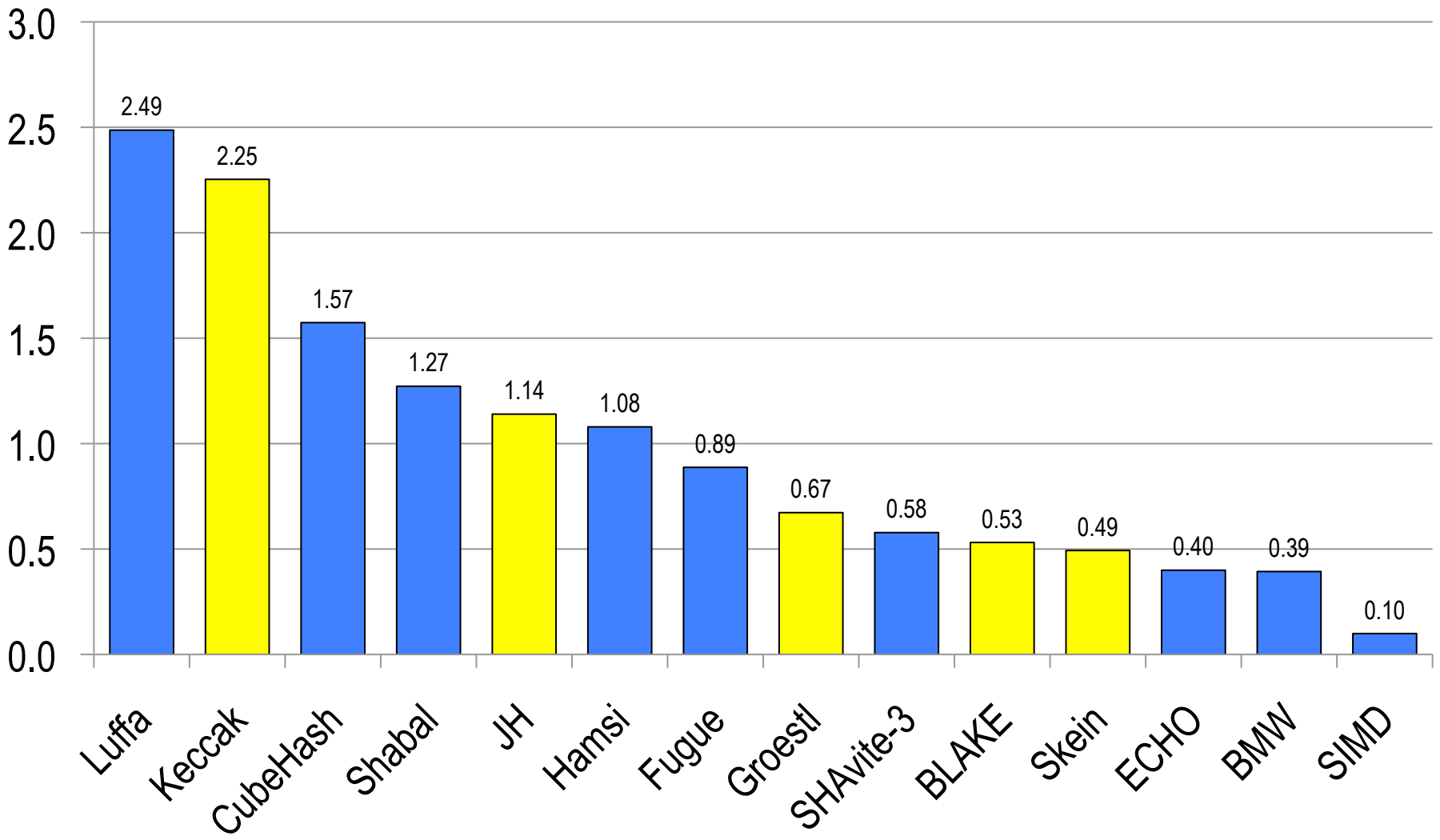
- Optimization for maximum throughput to area ratio
- 10 FPGA families from two major vendors :
  Xilinx and Altera

But still…

- Single high-speed architecture per candidate
- No use of embedded resources of FPGAs (Block RAMs, dedicated multipliers, DSP units)

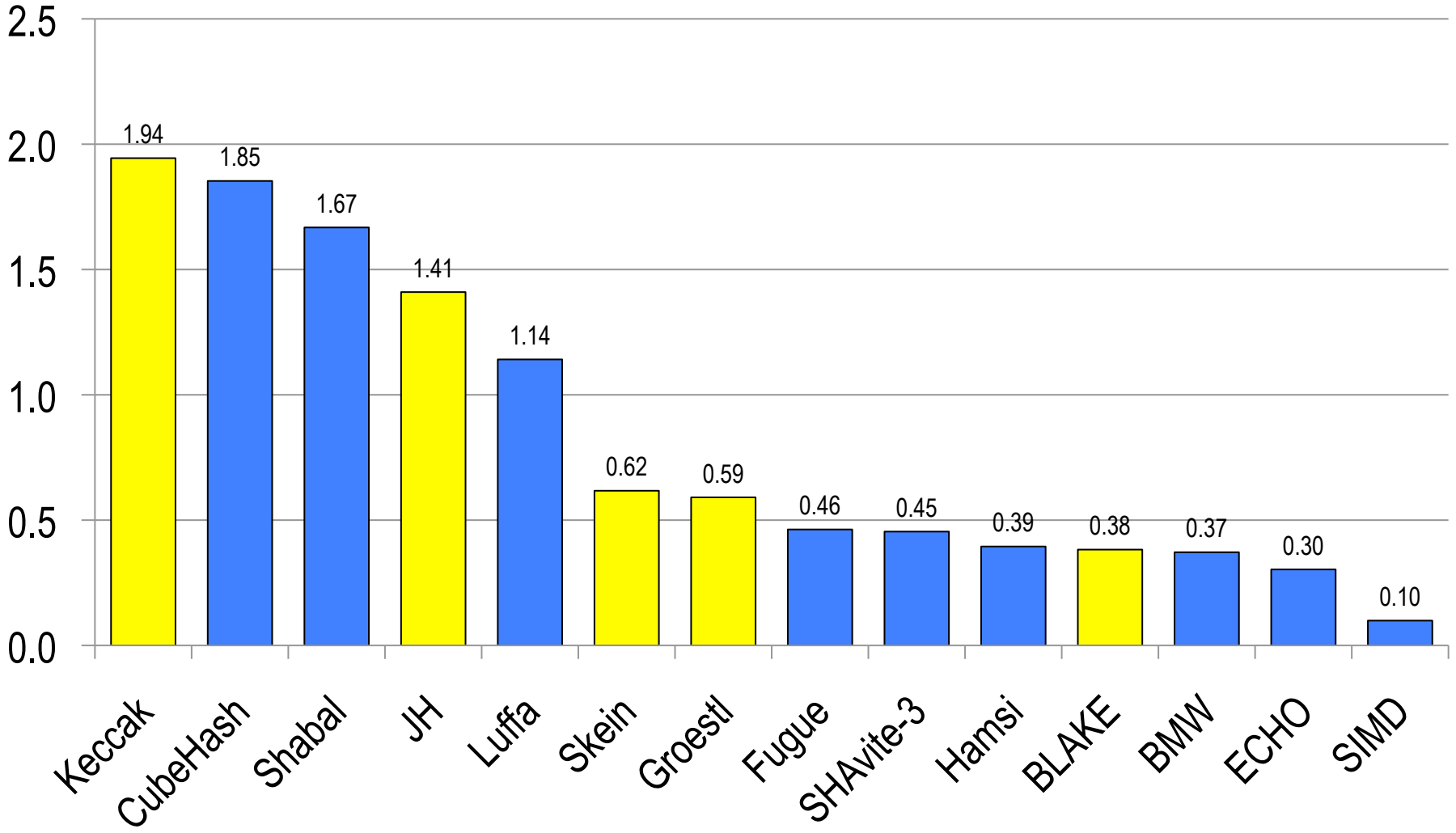
Overall Normalized Throughput/Area: 256-bit variants

Normalized to SHA-256, Averaged over 10 FPGA families

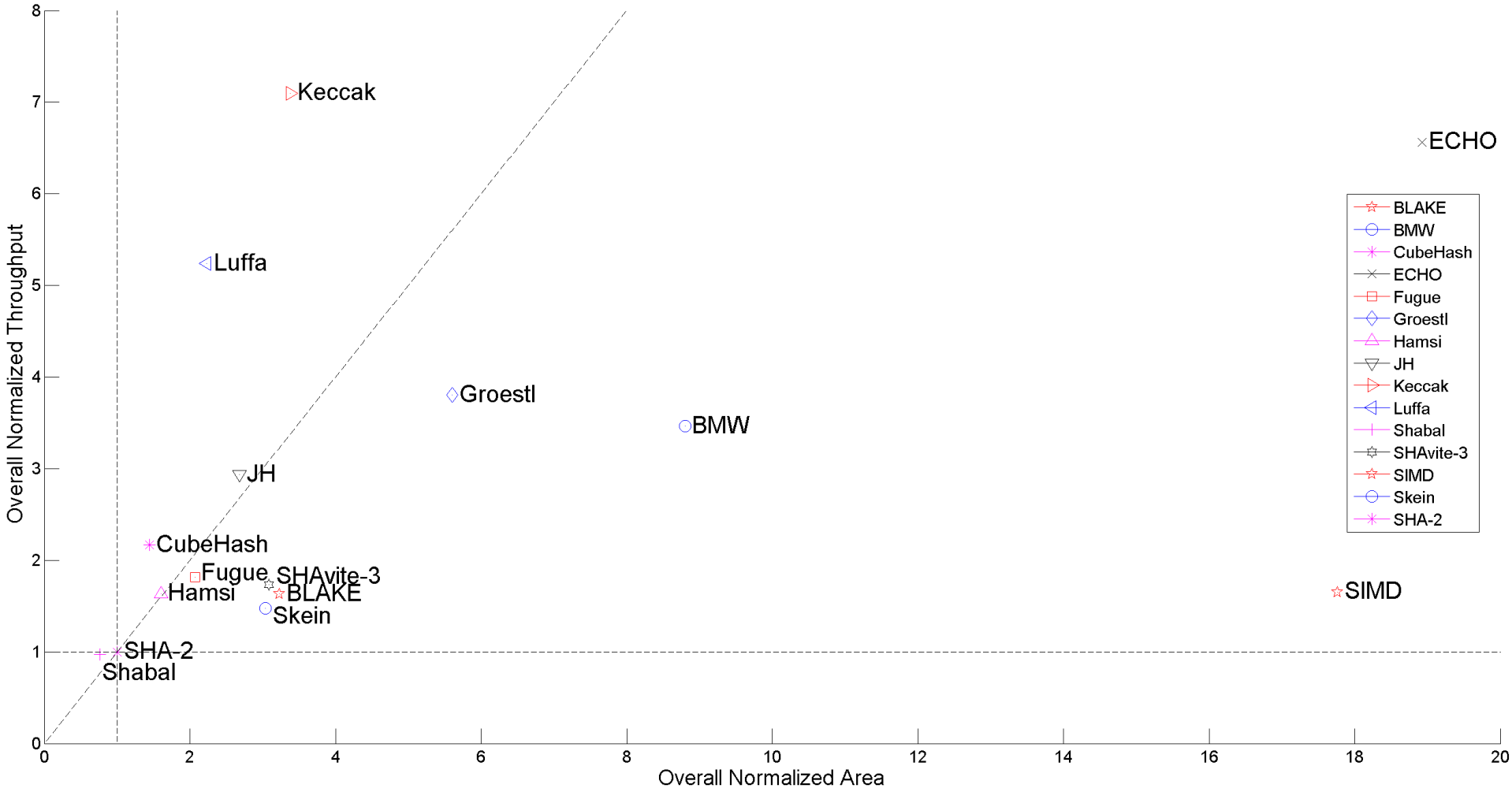


Overall Normalized Throughput/Area: 512-bit variants

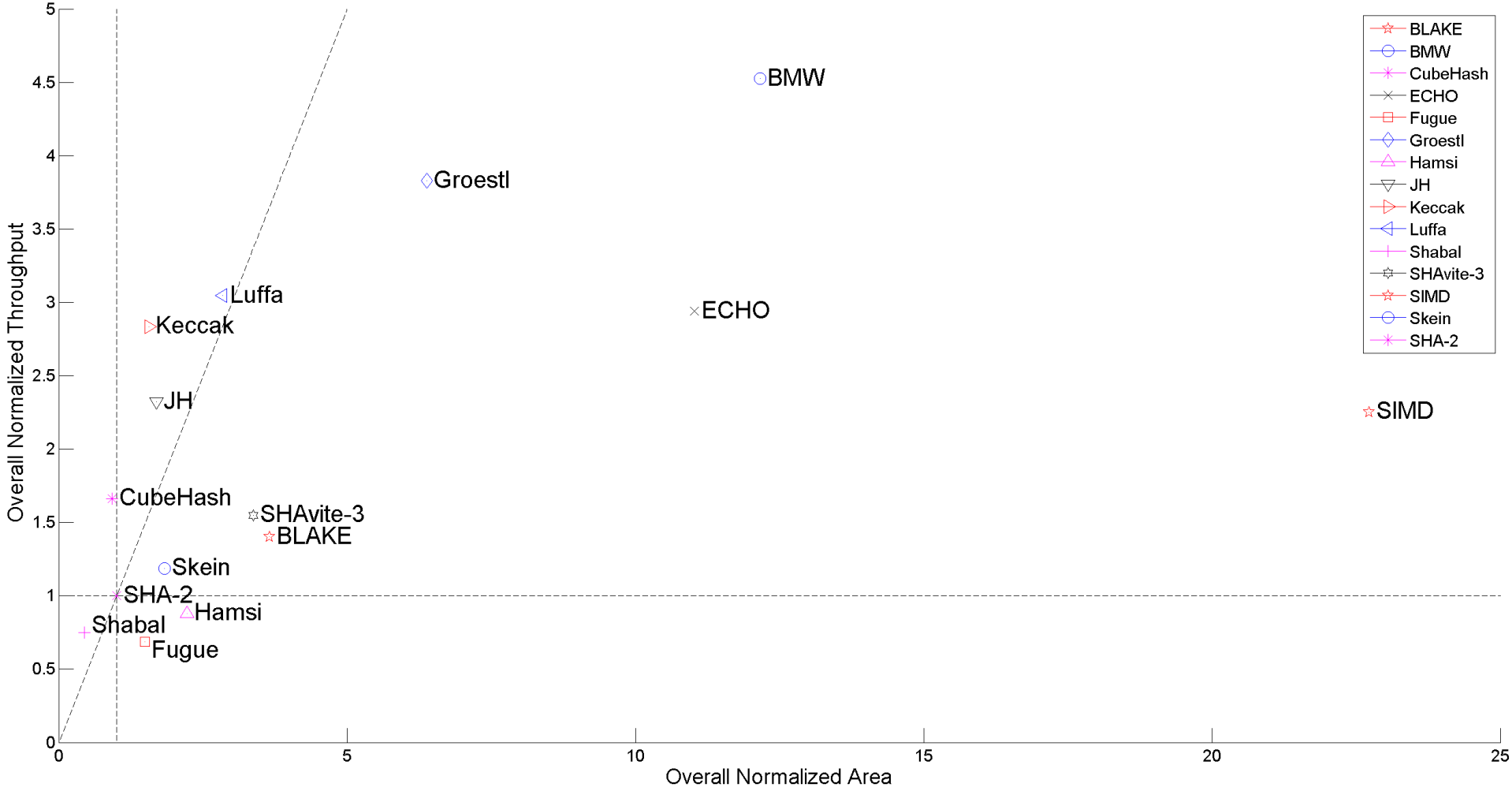
Normalized to SHA-512, Averaged over 10 FPGA families



Throughput vs. Area Normalized to Results for SHA-256 and Averaged over 11 FPGA Families – 256-bit variants



Throughput vs. Area Normalized to Results for SHA-512 and Averaged over 11 FPGA Families – 512-bit variants



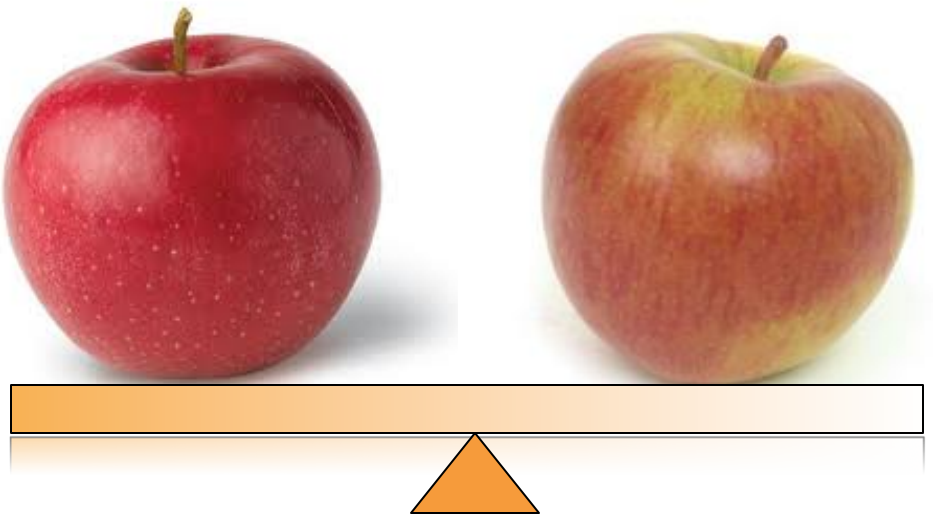
New in Round 3

- Multiple Hardware Architectures
- Effect of the Use of Embedded Resources
- Low-Area Implementations

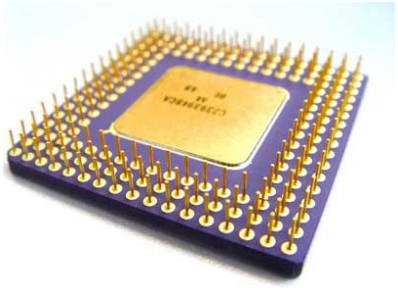

Common
Background

Uniform Evaluation

- Language: VHDL
- Tools: FPGA vendor tools
- Interface
- Benchmarking



Why Interface Matters?



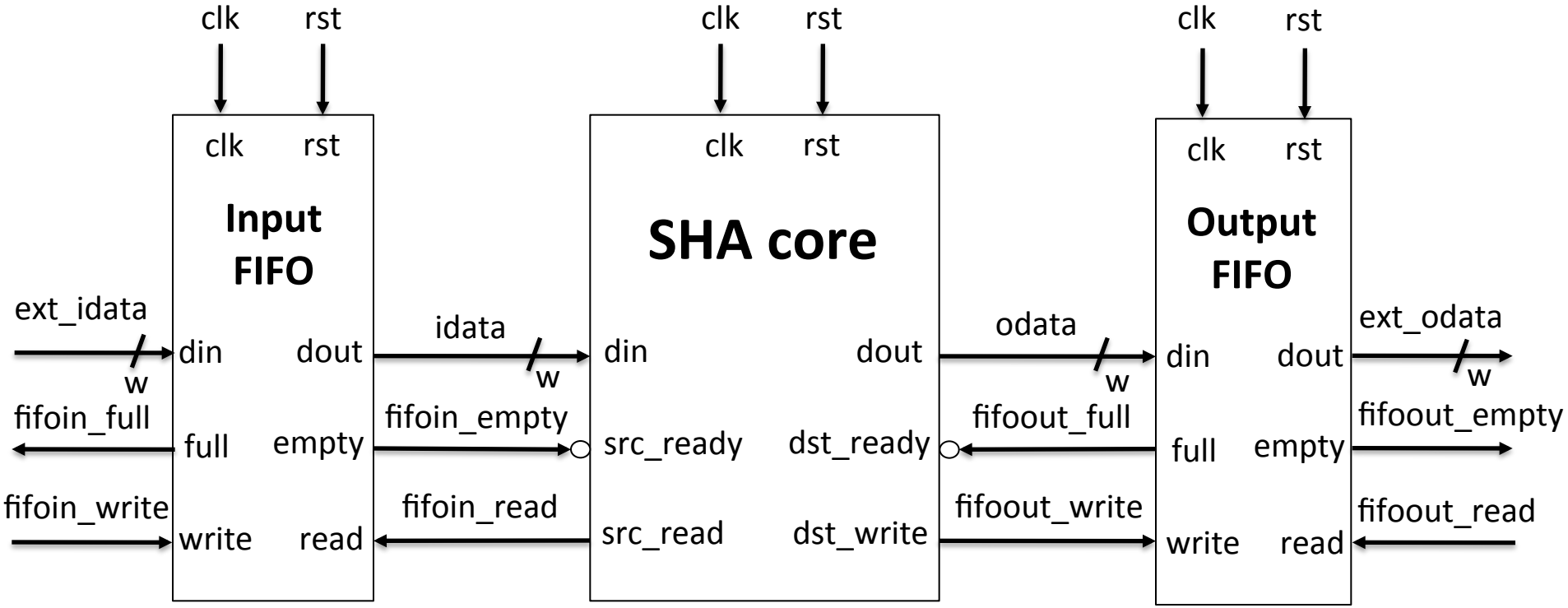
- Pin limit

Total number of i/o ports ≤ Total number of an FPGA i/o pins

- Support for the maximum throughput

Time to load the next message block ≤ Time to process previous block

SHA Core: Interface & Typical Configuration



- SHA core is an active component; surrounding FIFOs are passive and widely available
- Input interface is separate from an output interface
- Processing a current block, reading the next block, and storing a result for the previous message can be all done in parallel

Benchmarking platforms

- two major vendors: Altera and Xilinx (~90% of the market)
- 11 FPGA 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

ATHENa – **A**utomated **T**ool for **H**ardware **E**valuatio**N**

http://cryptography.gmu.edu/athena

Benchmarking open-source tool, written in Perl, aimed at an

AUTOMATED generation of
OPTIMIZED results for
MULTIPLE FPGA platforms

Under development at
George Mason University.

Why Athena?

"The Greek goddess Athena was frequently called upon to settle disputes between the gods or various mortals. Athena Goddess of Wisdom was known for her superb logic and intellect. Her decisions were usually well-considered, highly ethical, and seldom motivated by self-interest."

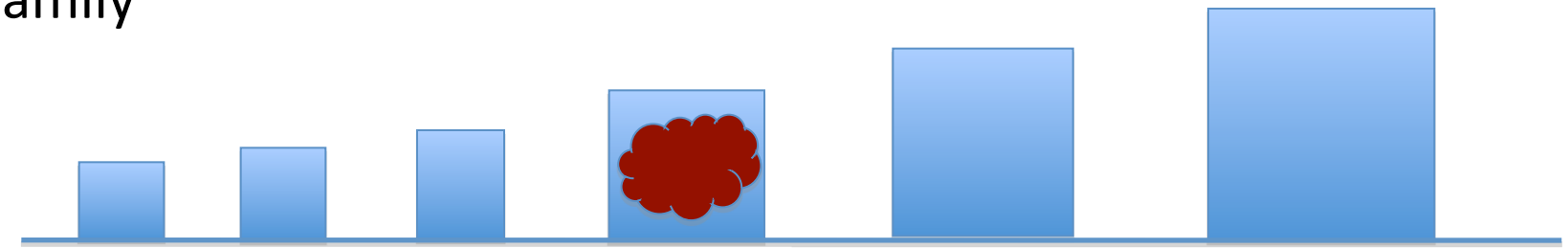
from "Athena, Greek Goddess of Wisdom and Craftsmanship"

ATHENa Major Features (1)

- synthesis, implementation, and timing analysis in **batch mode**
- support for devices and tools of **multiple FPGA vendors**:

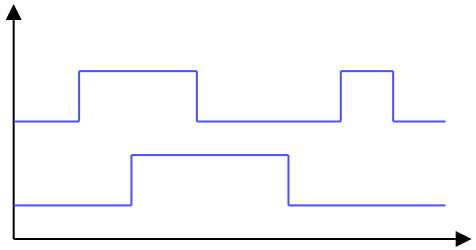
- generation of results for **multiple families** of FPGAs of a given vendor

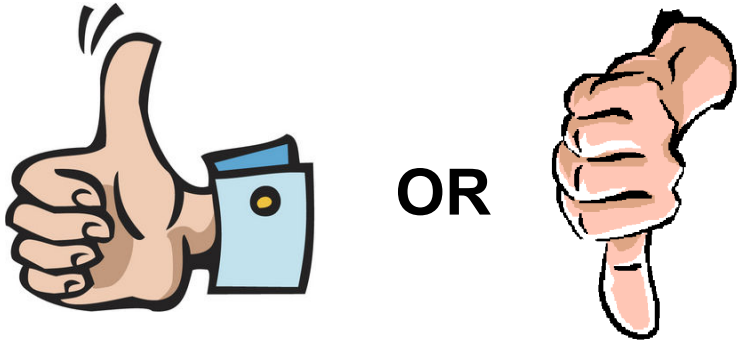
- automated choice of a **best-matching device** within a given family



ATHENa Major Features (2)

- **automated verification** of designs through simulation in batch mode





- support for **multi-core processing**
- automated **extraction and tabulation of results**
- several **optimization strategies** aimed at finding
  - optimum options of tools
  - best target clock frequency
  - best starting point of placement

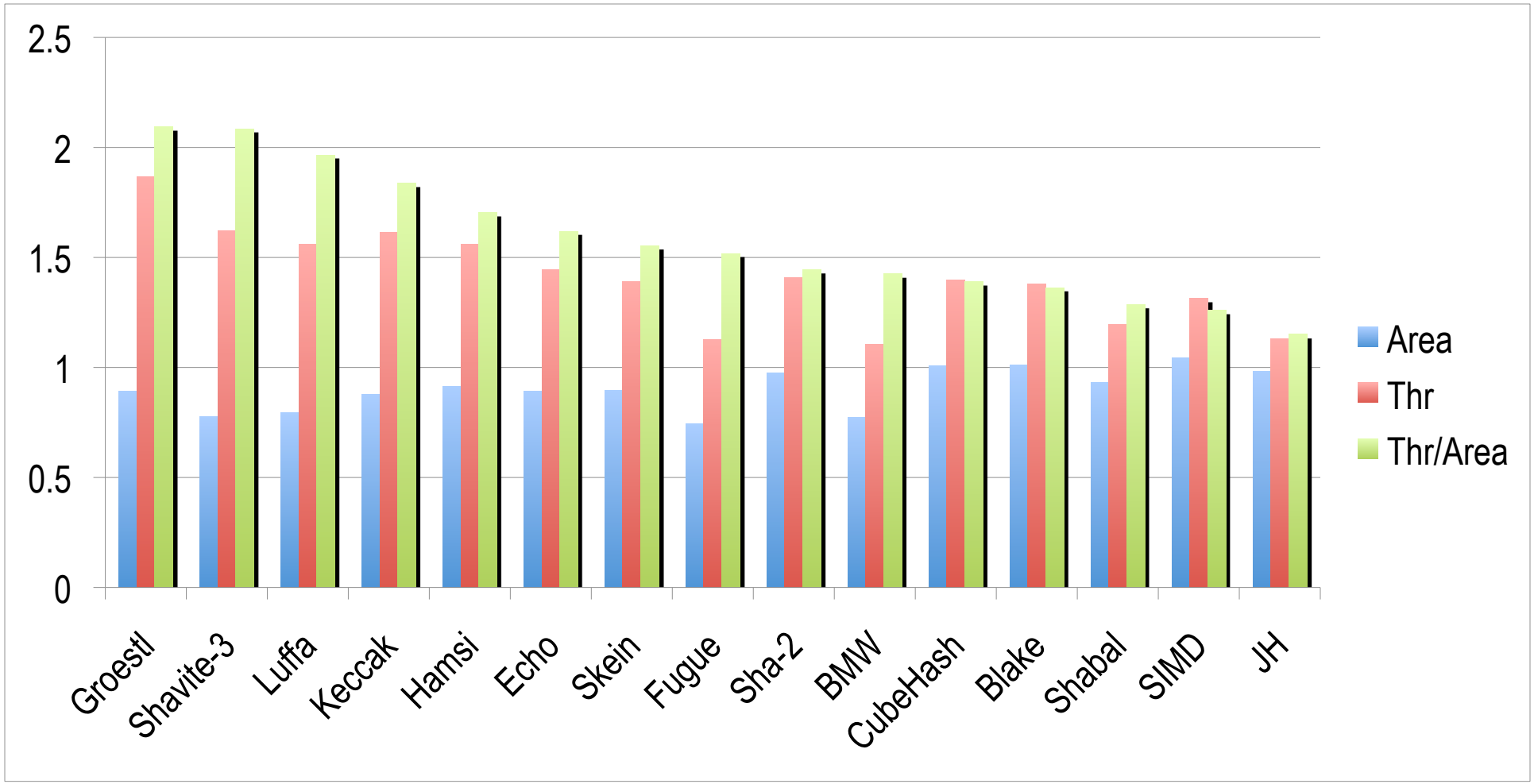
Generation of Results Facilitated by ATHENa

- batch mode of FPGA tools

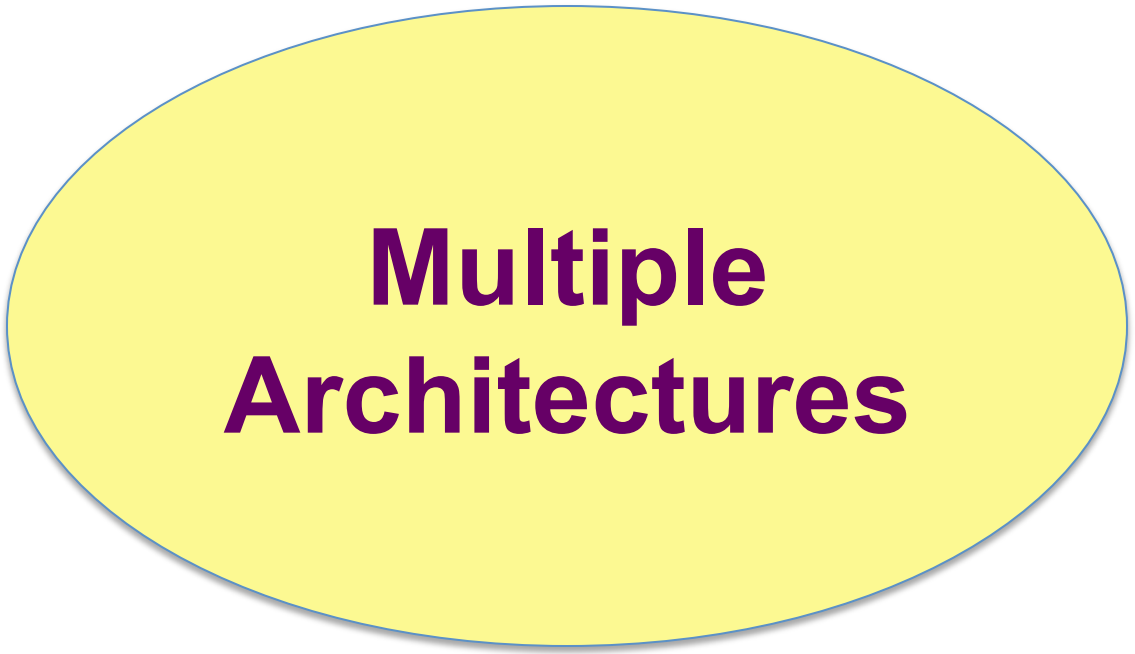
vs.

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

Relative Improvement of Results from Using ATHENa Virtex 5, 256-bit Variants of Hash Functions



Ratios of results obtained using ATHENa suggested options vs. default options of FPGA tools


Multiple
Architectures

Study of Multiple Architectures

- Analysis of multiple hardware architectures per each finalist, based on the known design techniques, such as
  - Folding
  - Unrolling
  - Pipelining
- Identifying the best architecture in terms of the throughput to area ratio
- Analyzing the flexibility of all algorithms in terms of the speed vs. area trade-offs

Performance Metrics - Area

$$Resource\ Utilization_{\text{Spartan3}} = (\#CLB\ slices,\ \#BRAMs,\ \#MULs)$$

$$Resource\ Utilization_{\text{Cyclone III}} = (\#LE,\ \#memory\_bits,\ \#MULs).$$

We force these vectors to look as follows through the synthesis and implementation options:

$$Resource\ Utilization_{\text{Spartan3}} = (\#CLB\ slices,\ 0,\ 0)$$

$$Resource\ Utilization_{\text{Cyclone III}} = (\#LE,\ 0,\ 0).$$

Area

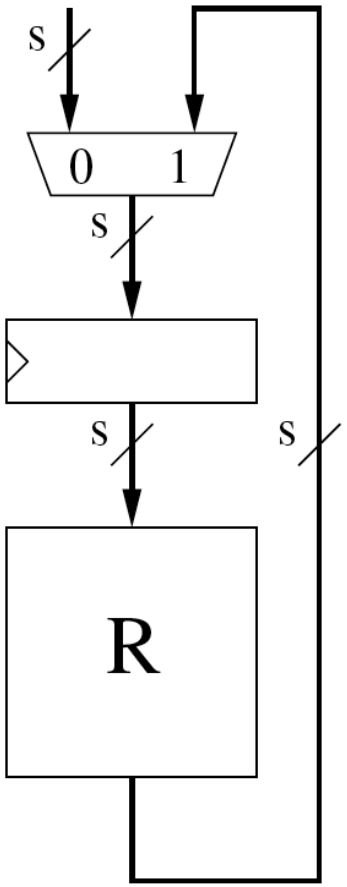
Primary Designers

Ekawat Homsirikamol
a.k.a "Ice"

Marcin Rogawski

Developed optimized VHDL implementations of
14 Round 2 SHA-3 candidates + SHA-2
in two variants each (256 & 512-bit output),
using several alternative architectures per each variant

Starting Point: Basic Iterative Architecture



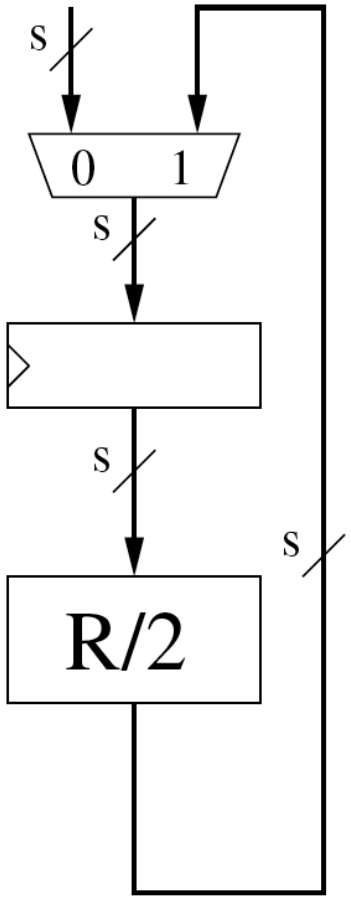
- datapath width = state size
- one clock cycle per one round/step

Block processing time = #R · T

#R = number of rounds/steps
T = clock period

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

Block processing time = (2·#R) * T'

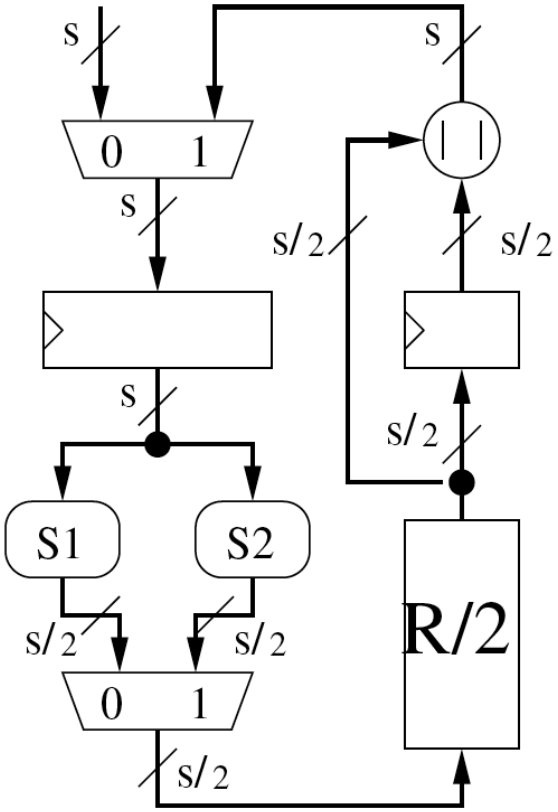
$T/2 < T' < T$

typically $T' \approx T/2$

$Area/2 < Area' < Area$

Typically Throughput/Area ratio increases

Vertical Folding - /2(v)



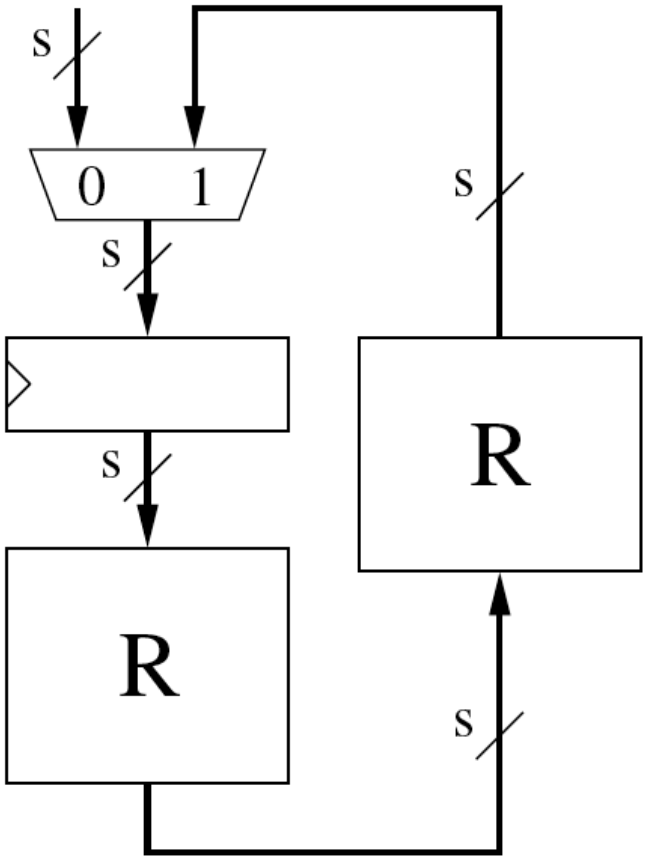
- datapath width = state size/2
- two clock cycles per one round/step

Block processing time = (2·#R) * T'

typically T' ≈ T

Area/2 < Area' < Area

Unrolling - x2



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

Block processing time = (#R/2) * T'

$T < T' < 2 \cdot T$

typically $T' \approx 2 \cdot T$

$Area/2 < Area' < 2 \cdot Area$

Typically $Area' \approx 2 \cdot Area$

Typically Throughput/Area ratio decreases

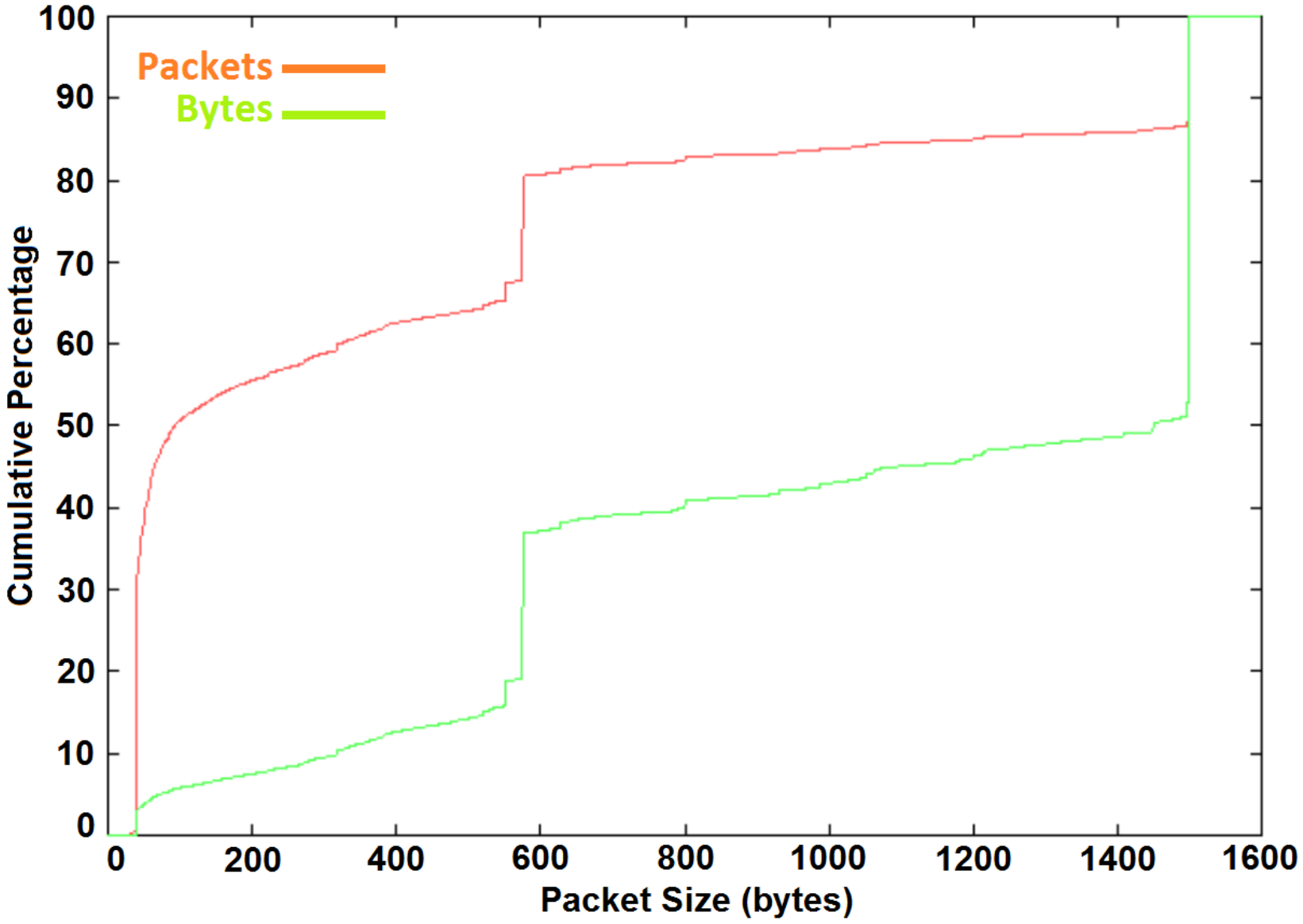
How to Increase the Speed? : The case for pipelining and parallel processing

- Protocols: IPSec, SSL, WLAN (802.11)
- Minimum Required Throughput Range: 100 Mbit/s - 40 Gbit/s (based on the specs of Security Processors from Cavium Networks, HiFn, and Broadcom)
- Supported sizes of packets: 40B - 1500B

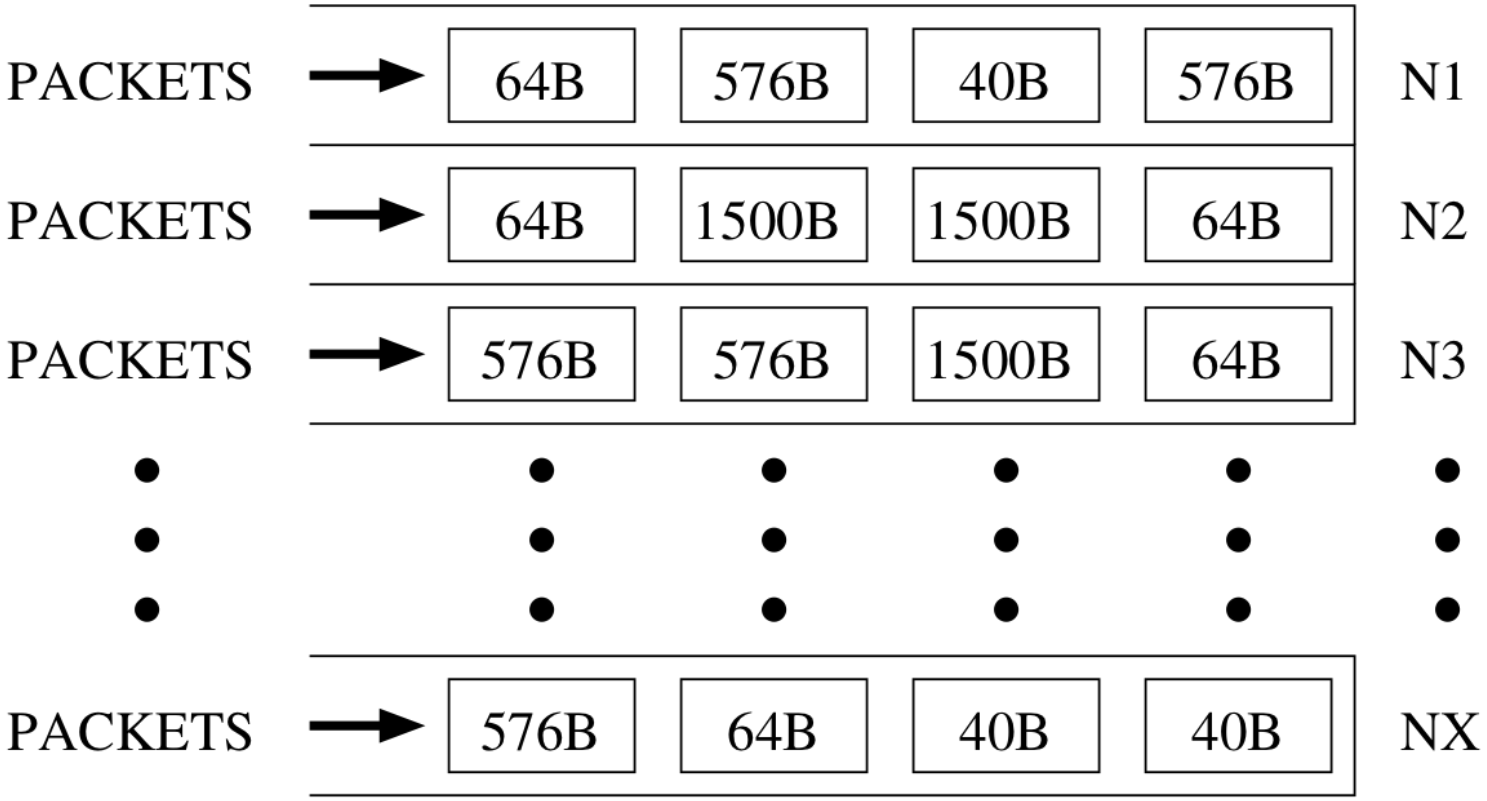
  1500 B = Maximum Transmission Unit (MTU) for Ethernet v2

  576 B = Maximum Transmission Unit (MTU) for Internet IPv4 Path
- Most Common Operation Involving Hashing: HMAC

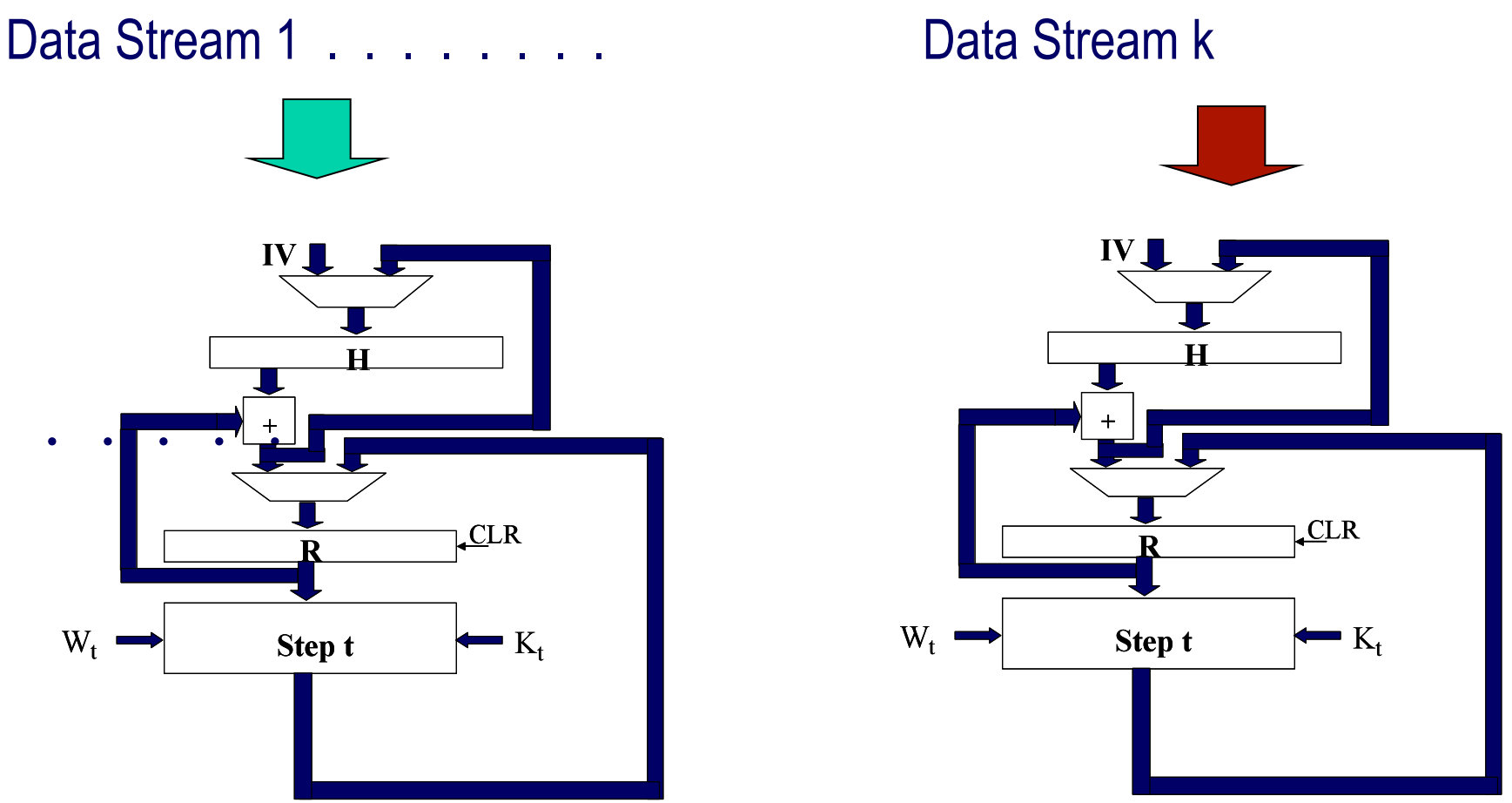
Cumulative Distribution of Packet Sizes



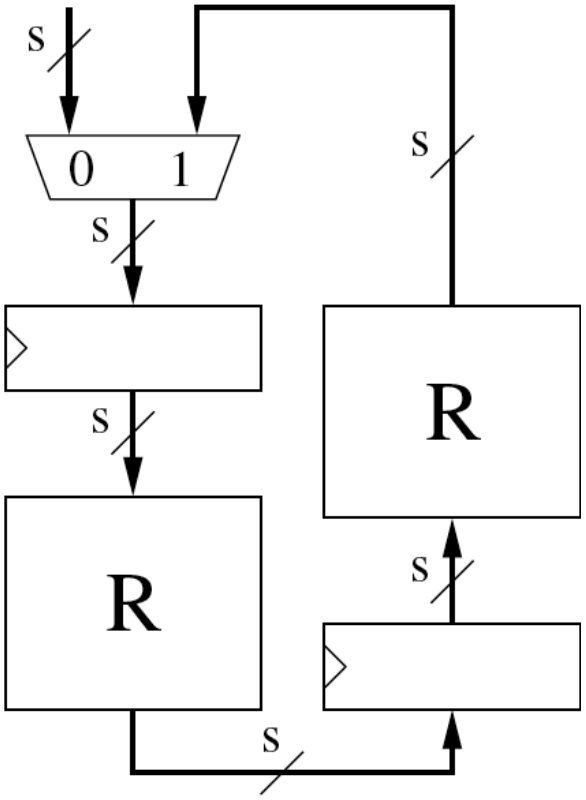
Multiple Packets Available for Parallel Processing

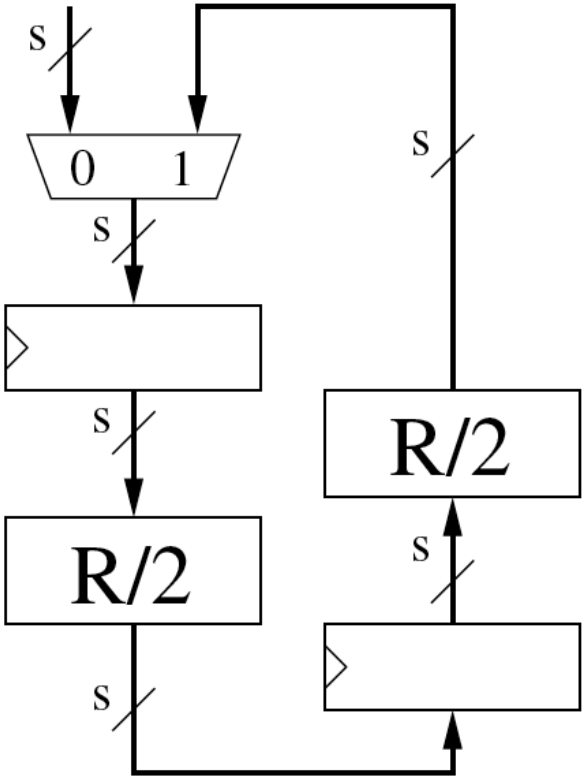


Parallel Processing Using Multi-Unit Architecture



Pipelining - x2-PPL2, x1-PPL2





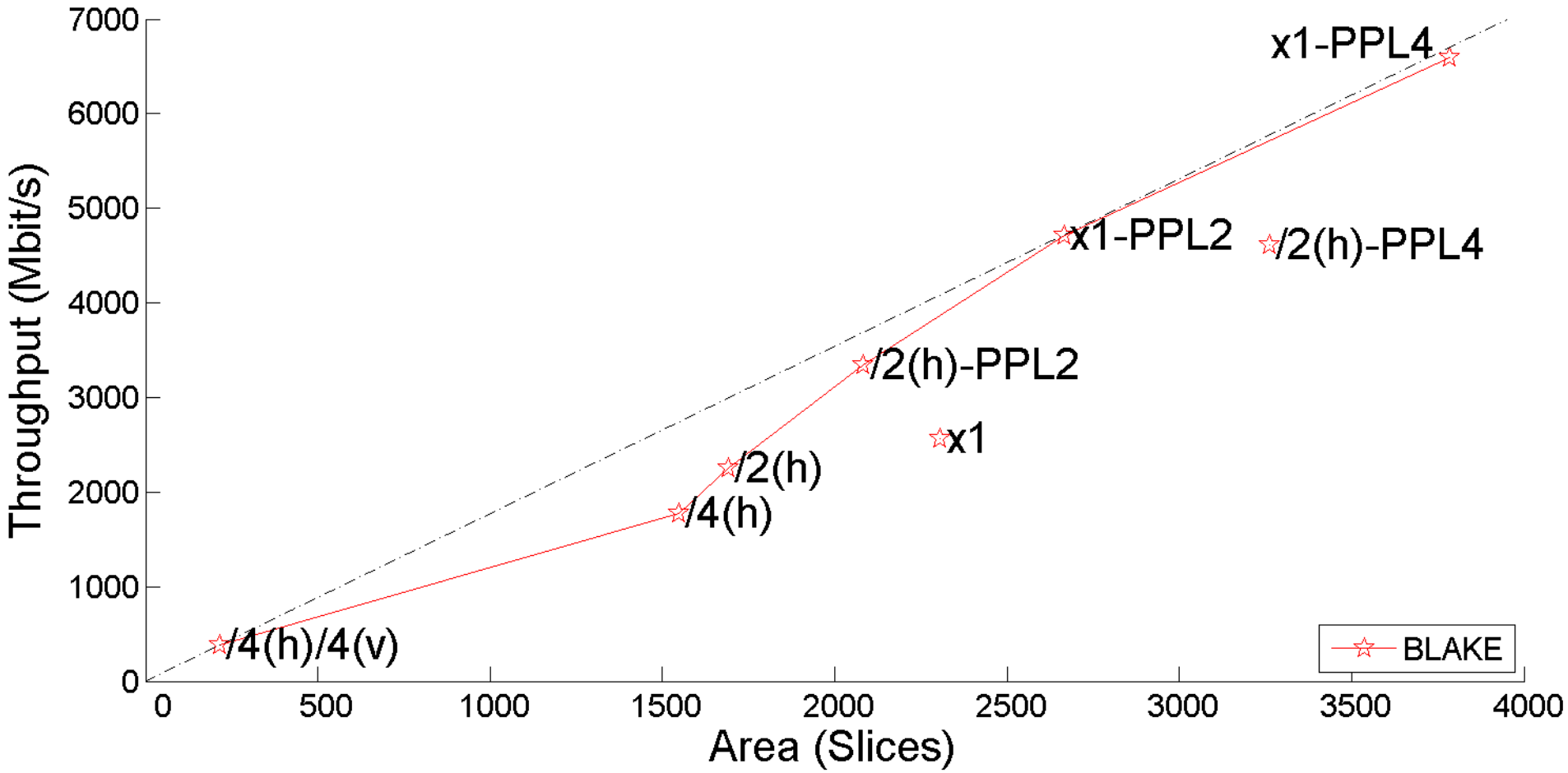
Results For Multiple Architectures

Comprehensive Evaluation

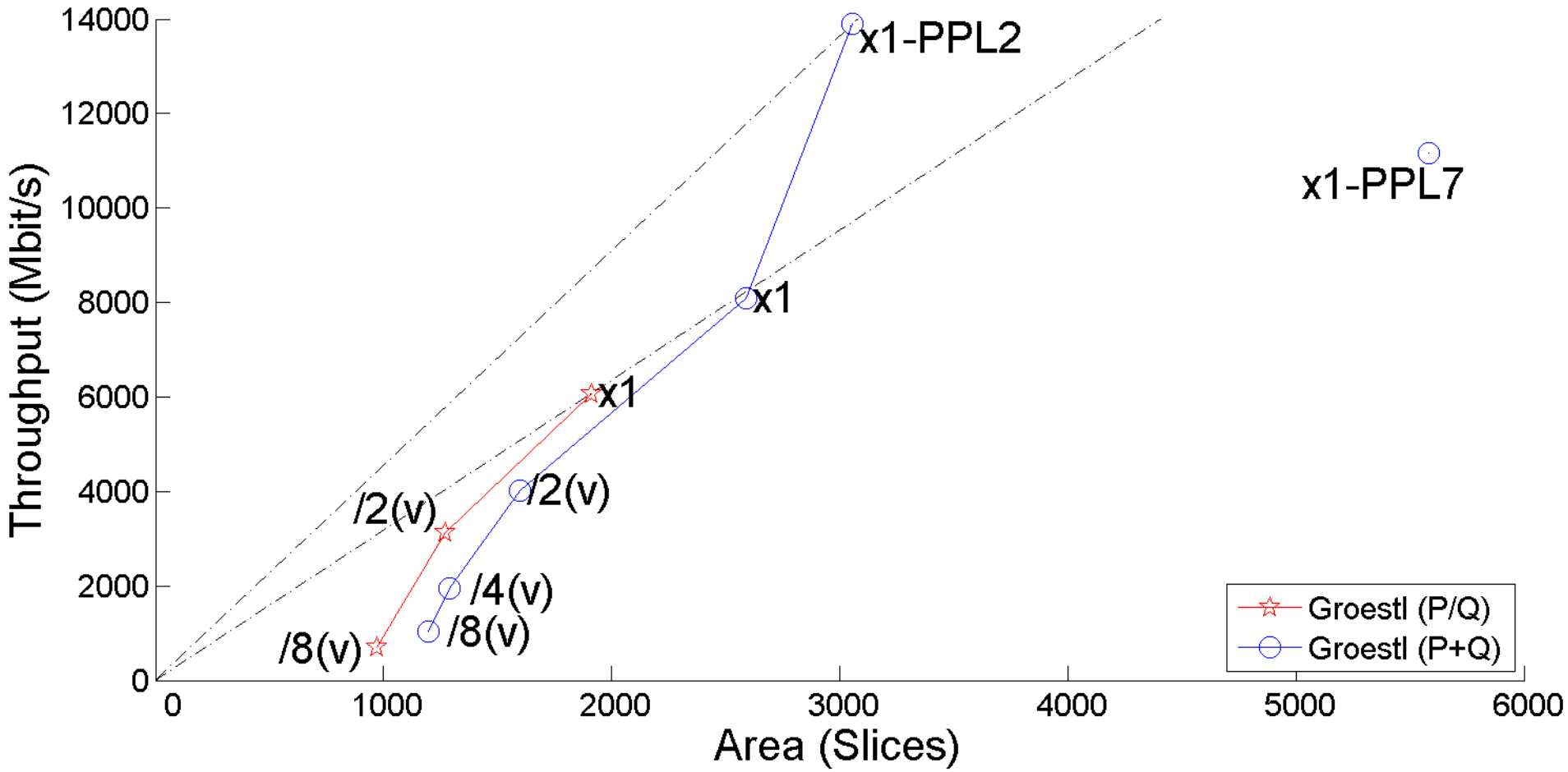
- two major vendors: Altera and Xilinx (~90% of the market)
- two most recent high-performance families

	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

BLAKE-256 in Virtex 5



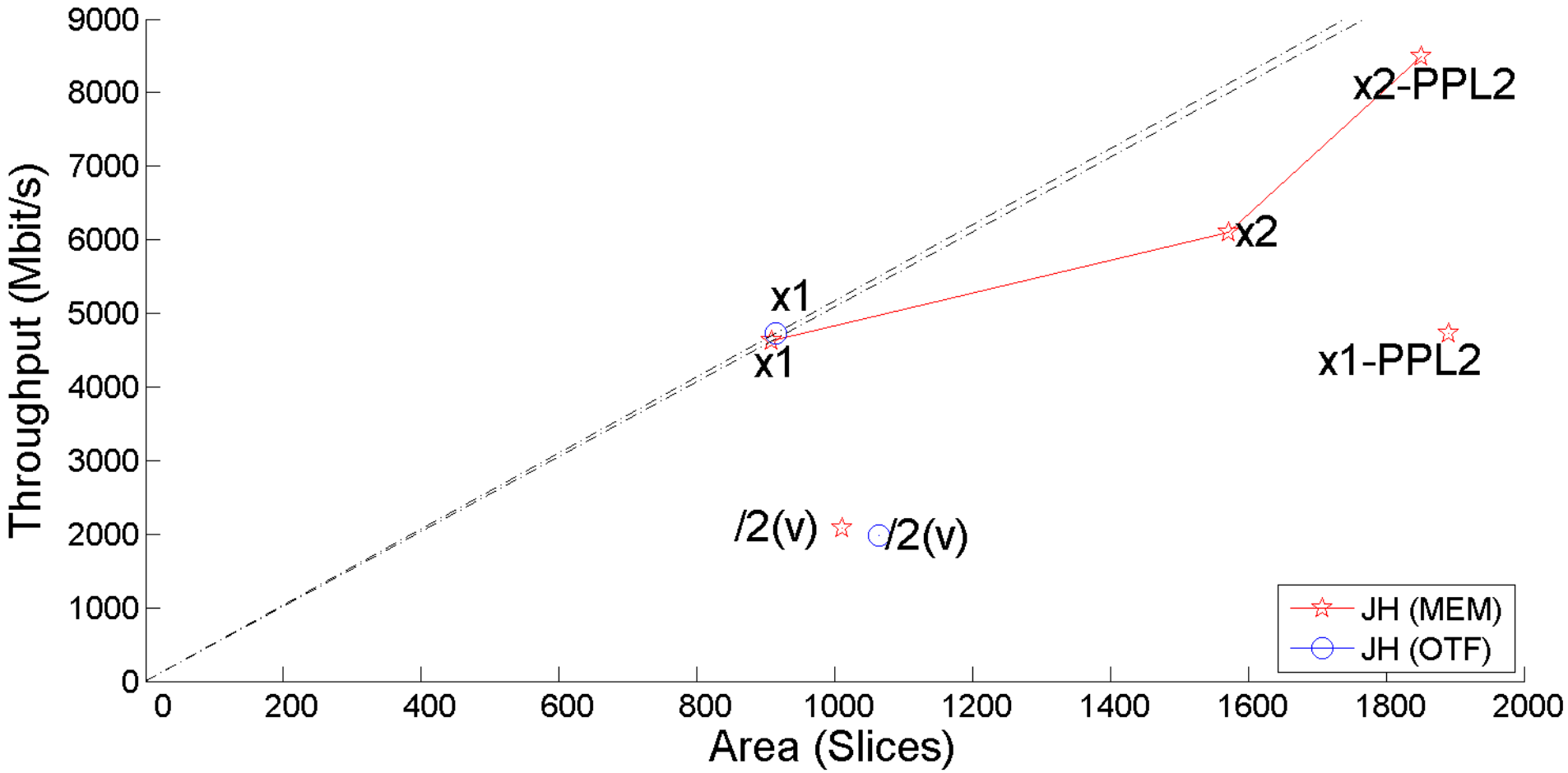
Groestl-256 in Virtex 5



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

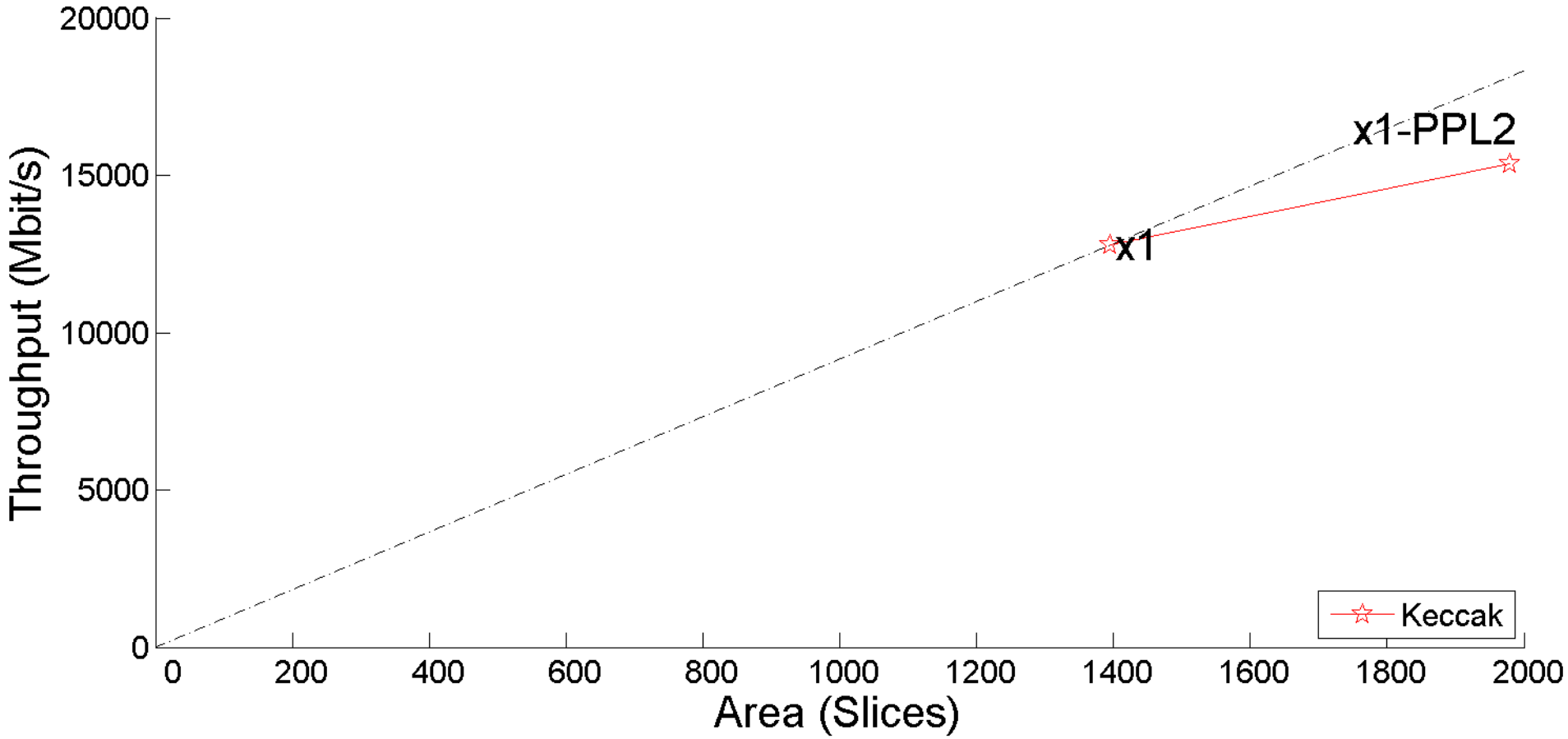
JH-256 in Virtex 5



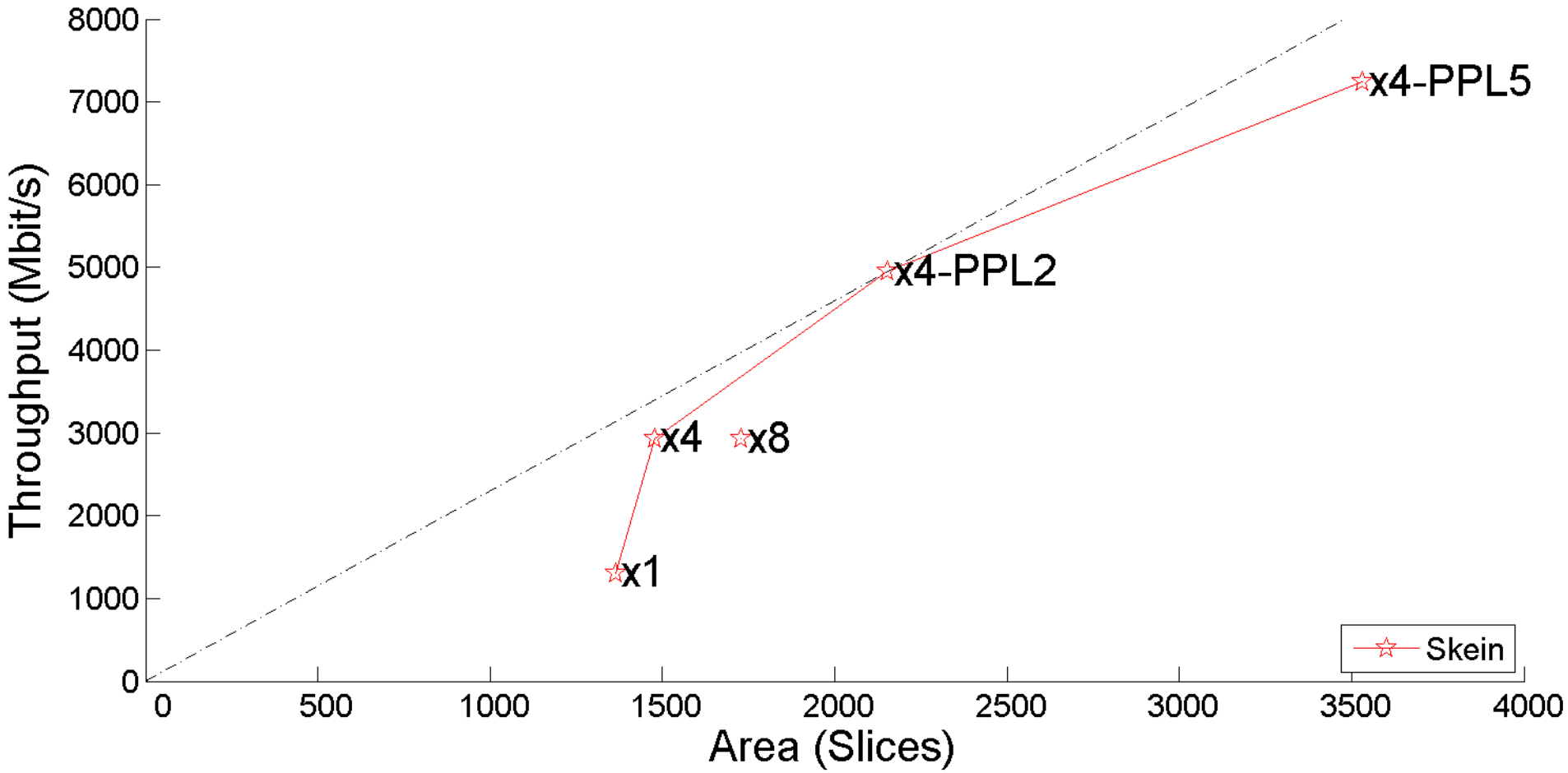
JH MEM – round constants stored in memory

JH OTF – round constants computed on-the-fly

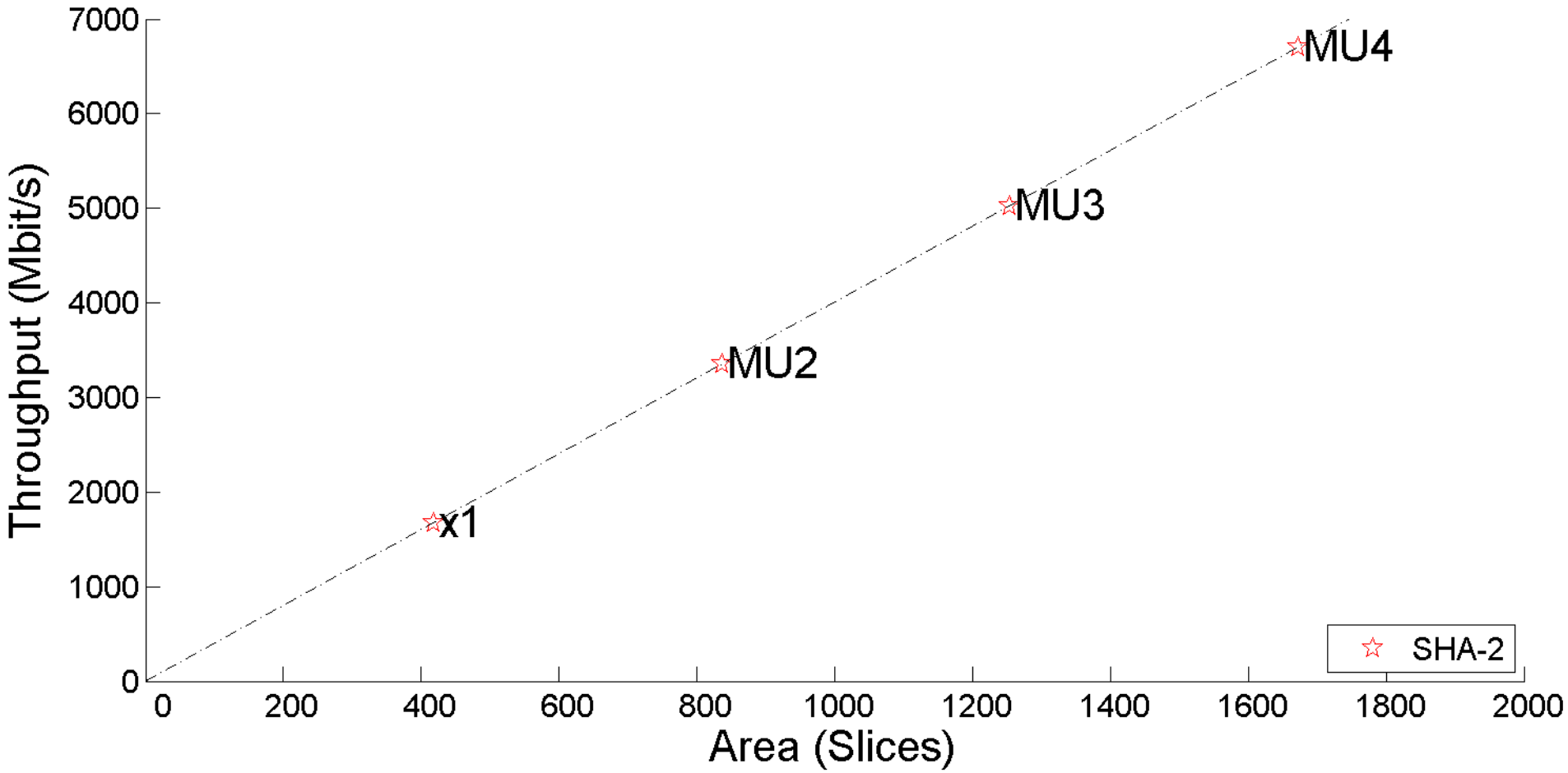
Keccak-256 in Virtex 5



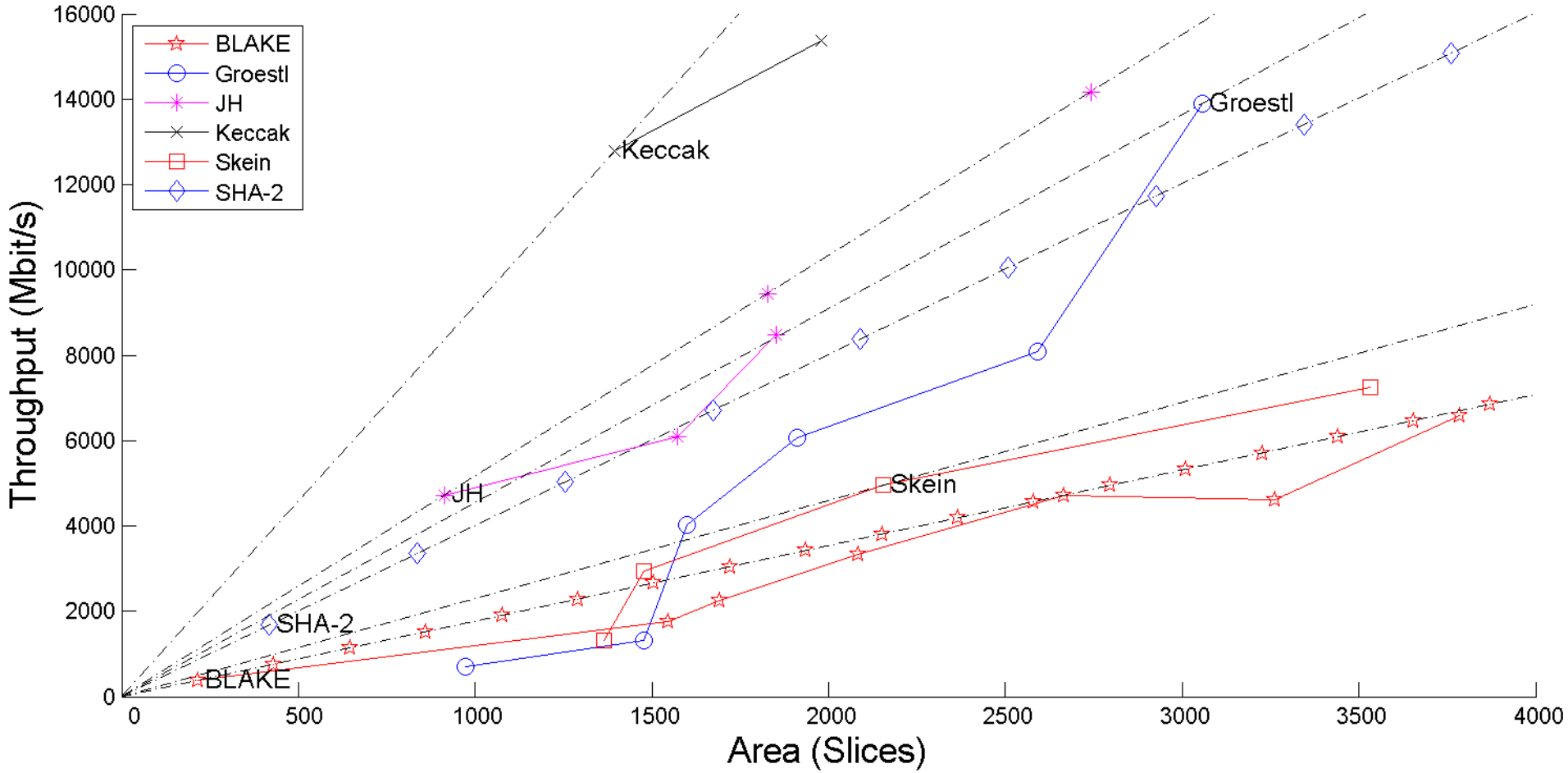
Skein-256 in Virtex 5



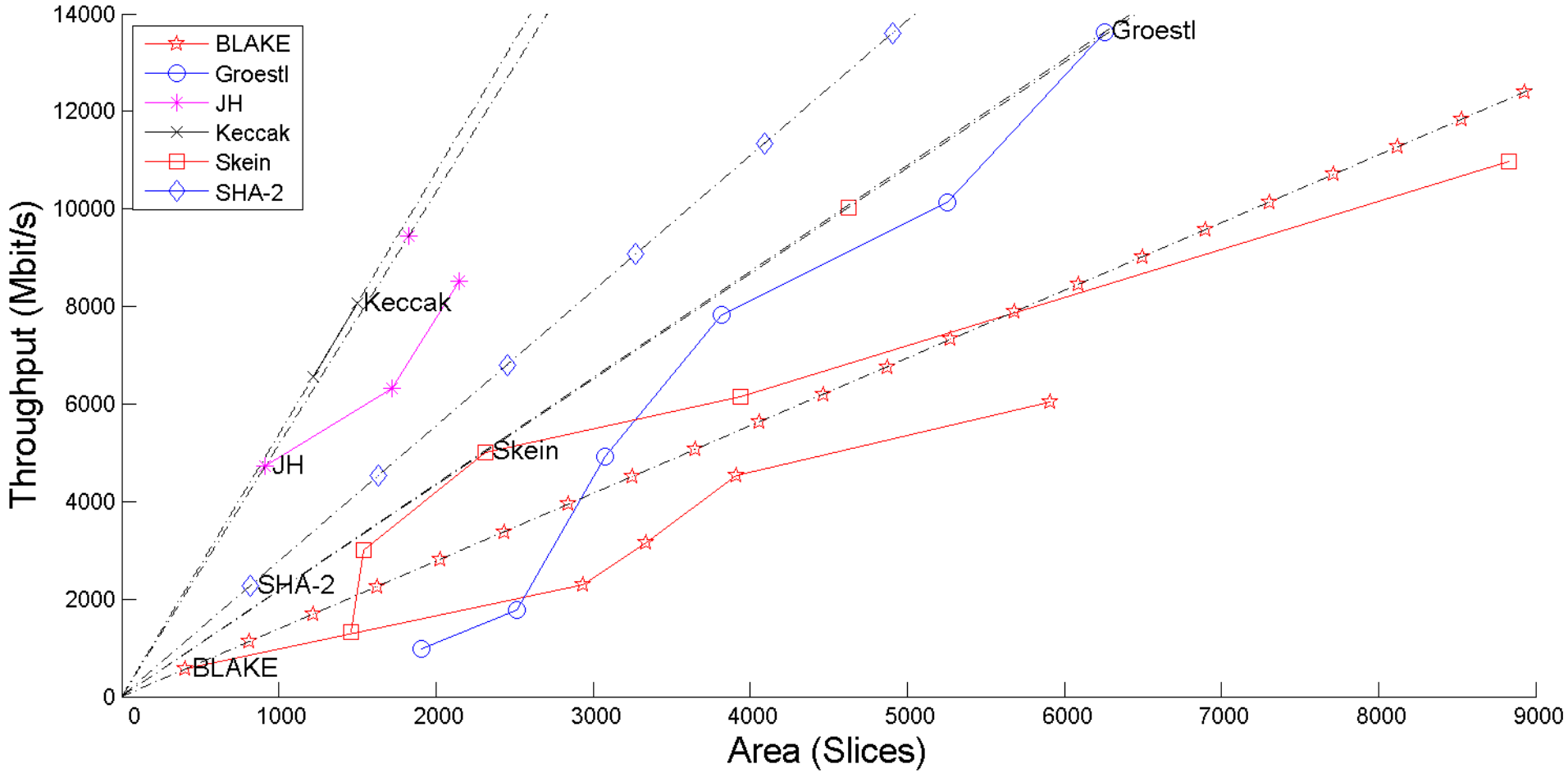
SHA-256 in Virtex 5



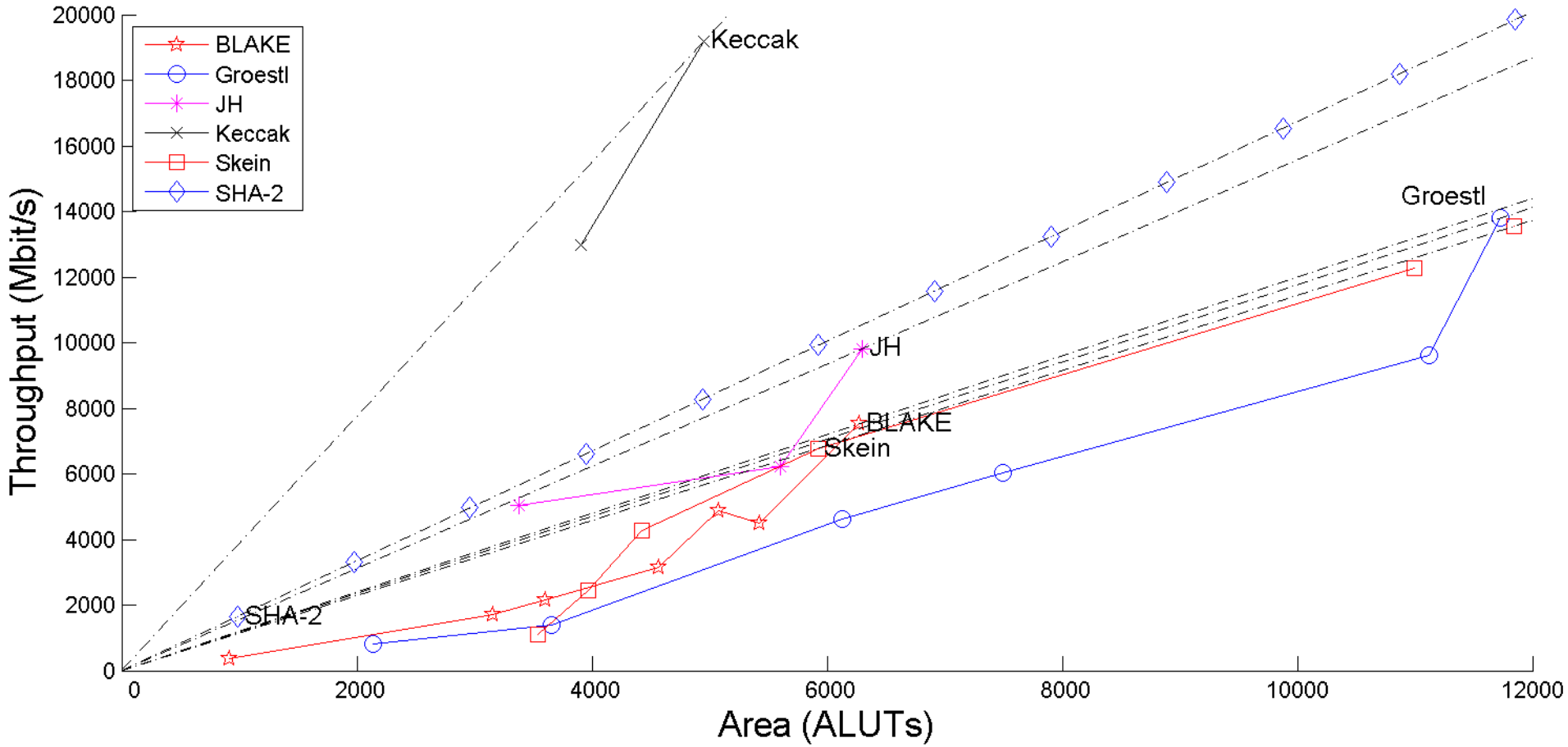
256-bit variants in Virtex 5



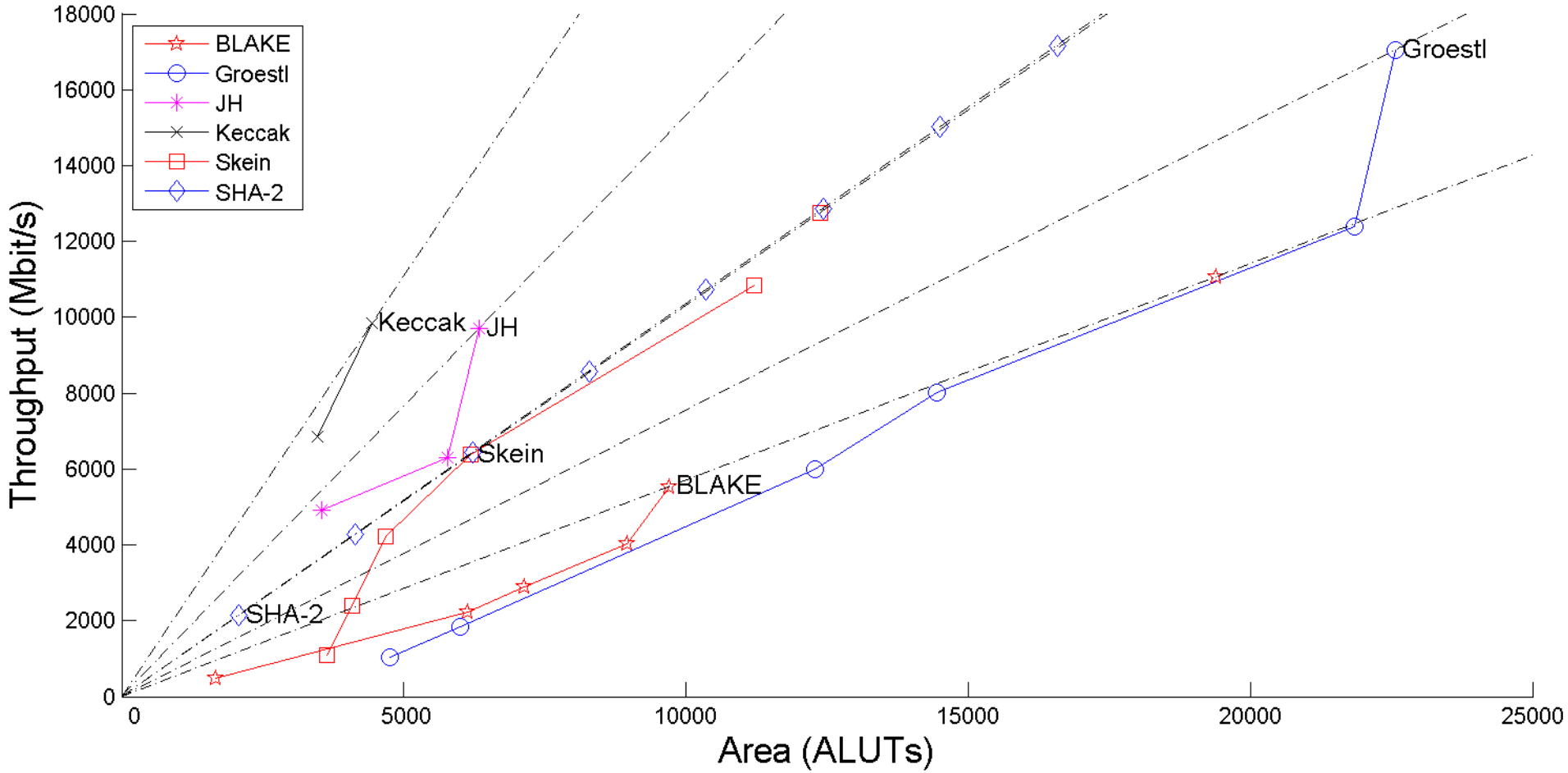
512-bit variants in Virtex 5



256-bit variants in Stratix III



512-bit variants in Stratix III



Conclusions for Multiple Architectures

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 folding or inner-round pipelining

Groestl – better than SHA-2 only for one out of four FPGA families,
but 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
- Pipelined architectures the best in terms of the throughput to area ratio for 4 out of 5 candidates
- Two front-runners: Keccak, JH

Use of Embedded FPGA Resources in Implementations of Five Round Three SHA-3 Candidates

Malik Umar Sharif, Rabia Shahid,
Marcin Rogawski and Kris Gaj

George Mason University, USA

Agenda

- SHA-3 High Speed Implementations – results summary
- SHA-3 candidates high speed architectures with embedded resources methodology
- Hardwired resources in FPGAs
- Results
- Conclusions

SHA-3 High Speed Architectures on Xilinx Virtex 5 (single stream of data)

Round 2: [SHA3 ZOO and ATHENaDB]

	Area	Bram	Throughput	Throughput/Area	Source
Blake	1623	0	3176	1.96	GMU
Groestl-0	1722	N/A	10276	5.97	Gauravaram et al.
Groestl-0	1381	17	7552	5.46	Jungk et al.
Groestl-0	1597	0	7885	4.94	GMU
Keccak	1272	0	12817	10.08	GMU
JH	1056	0	5874	5.56	GMU
Skein	1306	0	2565	1.96	GMU

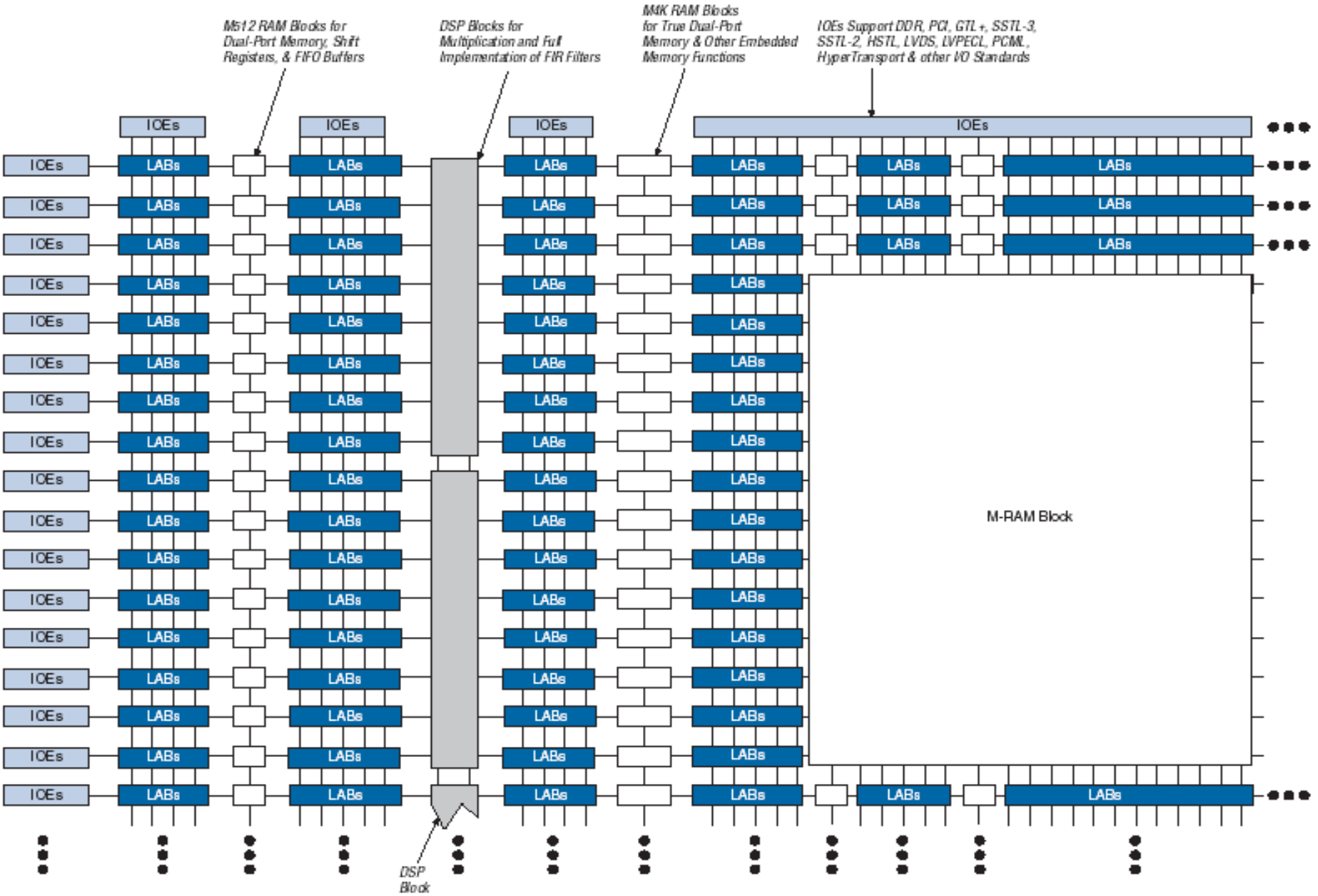
Round 3:

	Area	Bram	Throughput	Throughput/Area	Source
Blake	1702	0	2275	1.32	GMU
Groestl	1852	0	6083	3.28	GMU
Keccak	1272	0	12817	10.08	GMU
JH	1056	0	4917	4.65	GMU
Skein	1306	0	2565	1.96	GMU

Methodology

- Top level architectures from GMU basic designs,
- http://eprint.iacr.org/2010/445
- Uniform and practical Interface,
- Use of multiple FPGAs: 90nm low cost: Altera Cyclone II, Xilinx Spartan 3 and 65nm high performance: Altera Stratix III and Xilinx Virtex5,
- Clear performance metrics,
- Uniform optimization criteria,
- Use of ATHENa for generation, optimization and comparative analysis of results,

Altera Stratix II



Performance metrics

Area:

Vendor	Family	Resource Utilization Vector
Xilinx	Spartan 3	(#CLB_slices, #BRAMs, #multipliers)
	Virtex 5	(#CLB_slices, #BRAMs, #DSP48s)
Altera	Cyclone II	(#LEs, #Mem-bits, #multipliers)
	Stratix III	(#ALUTs, #Mem-bits, #DSP_18s)

$$Throughput = \frac{block\_size}{T \cdot (HTime(N+1) - HTime(N))}$$

SHA-3 candidates operations and FPGA embedded resources

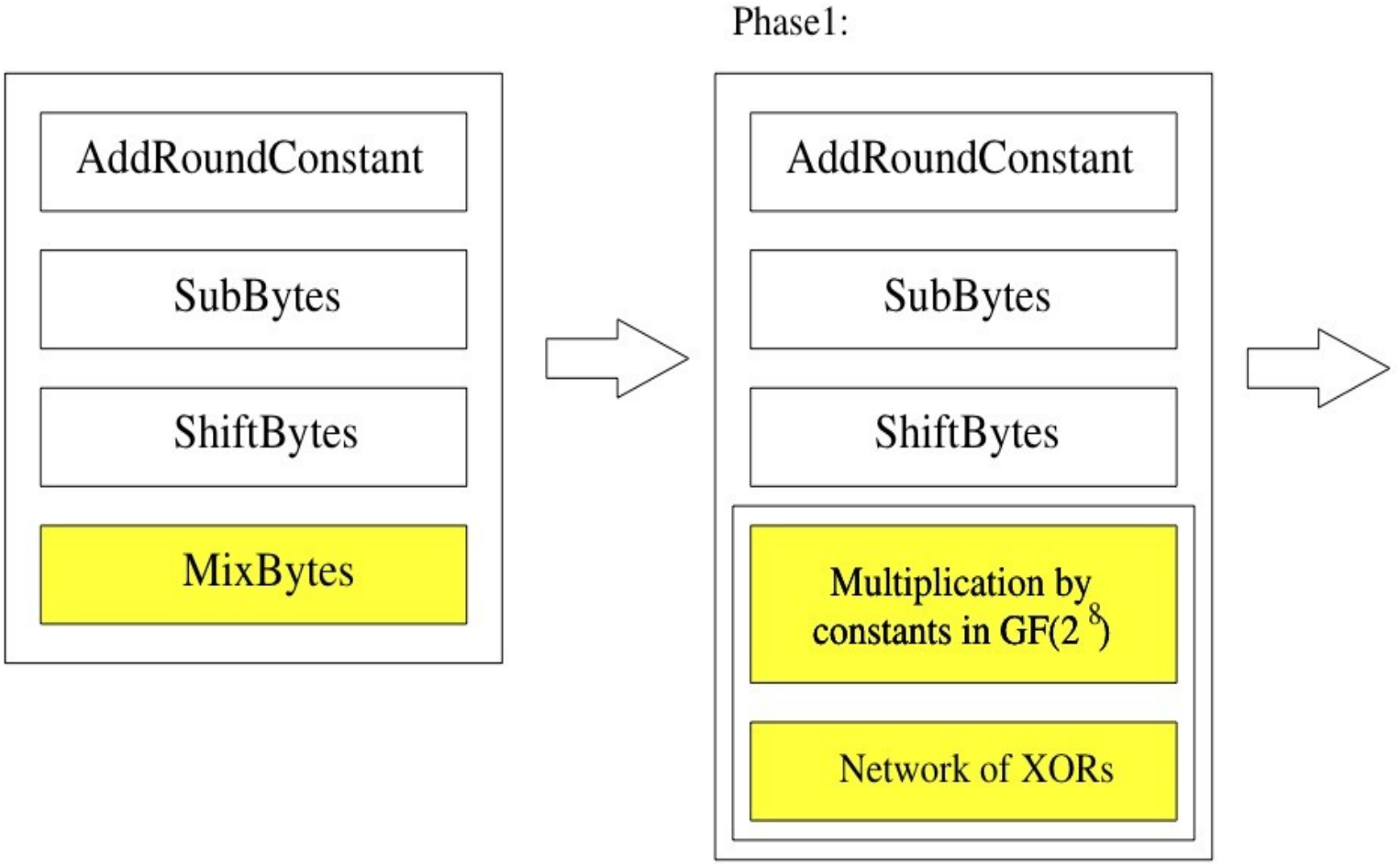
	S-box	GF Mul	mADD	ADD/SUB	Additional Memories
Blake			mADD3	ADD	Message expansion tables
Groestl	AES 8x8	x02-x07			
JH	4x4	x02, x05			Round constants
Keccak					Round constants
Skein				ADD-64	

SHA-3 candidates operations and FPGA embedded resources

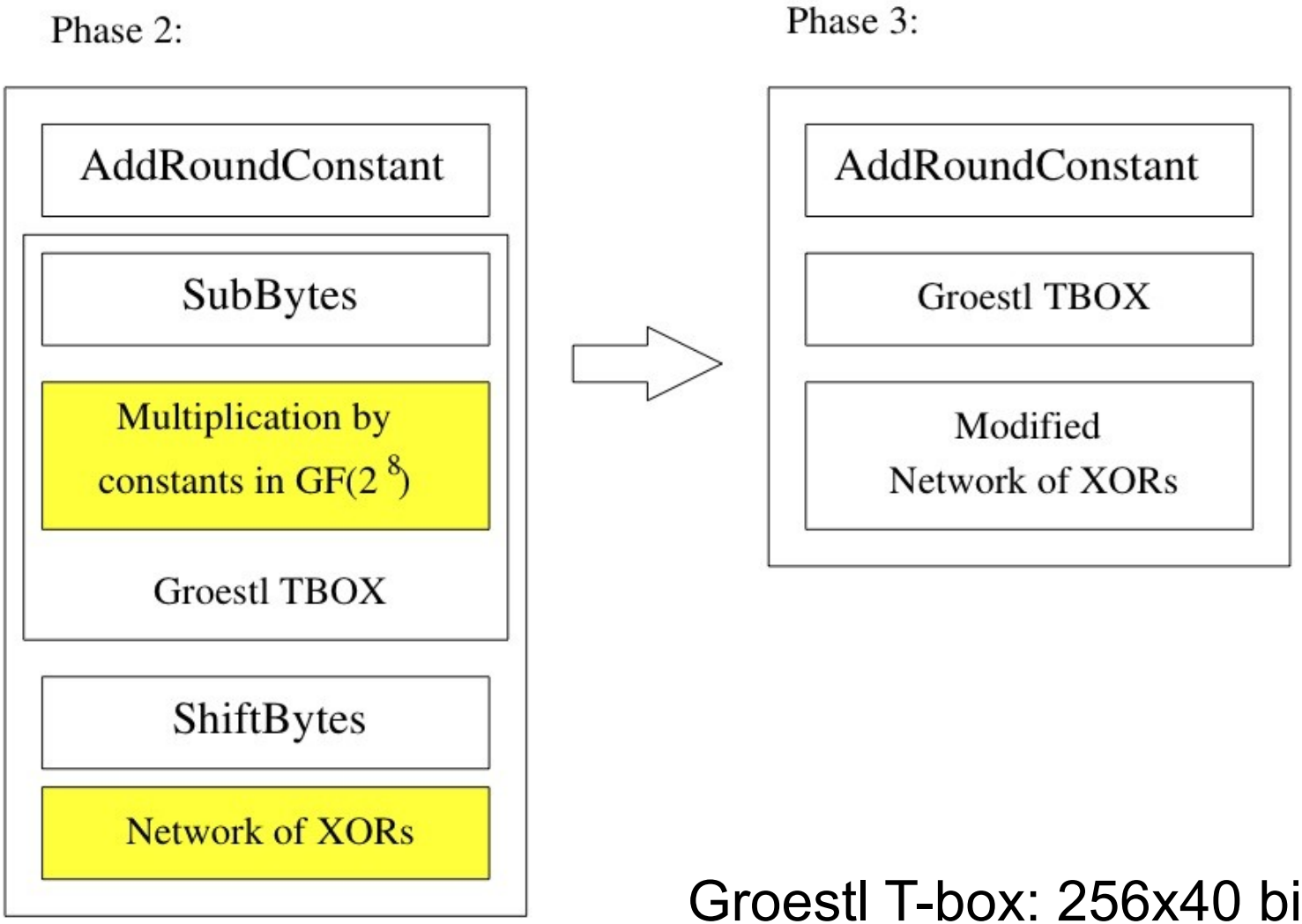
	S-box	GF Mul	mADD	ADD/SUB	Additional Memories
Blake			mADD3	ADD	Message expansion tables
Groestl	AES 8x8	x02-x07			
JH	4x4	x02, x05			Round constants
Keccak					Round constants
Skein				ADD-64	

	S-box	GF Mul	mADD	ADD/SUB	Additional Memories
Blake			DSP	DSP	BRAM
Groestl	BRAM				
JH	BRAM				BRAM
Keccak					BRAM
Skein				DSP	

Groestl-0/Groestl T-box



Groestl-0/Groestl T-box

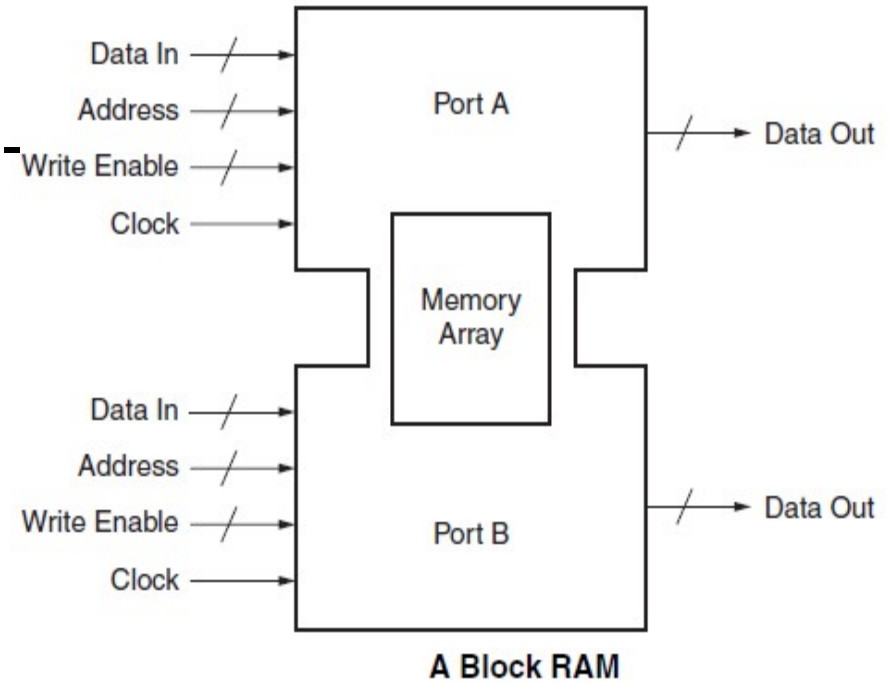


Groestl T-box: 256x40 bits

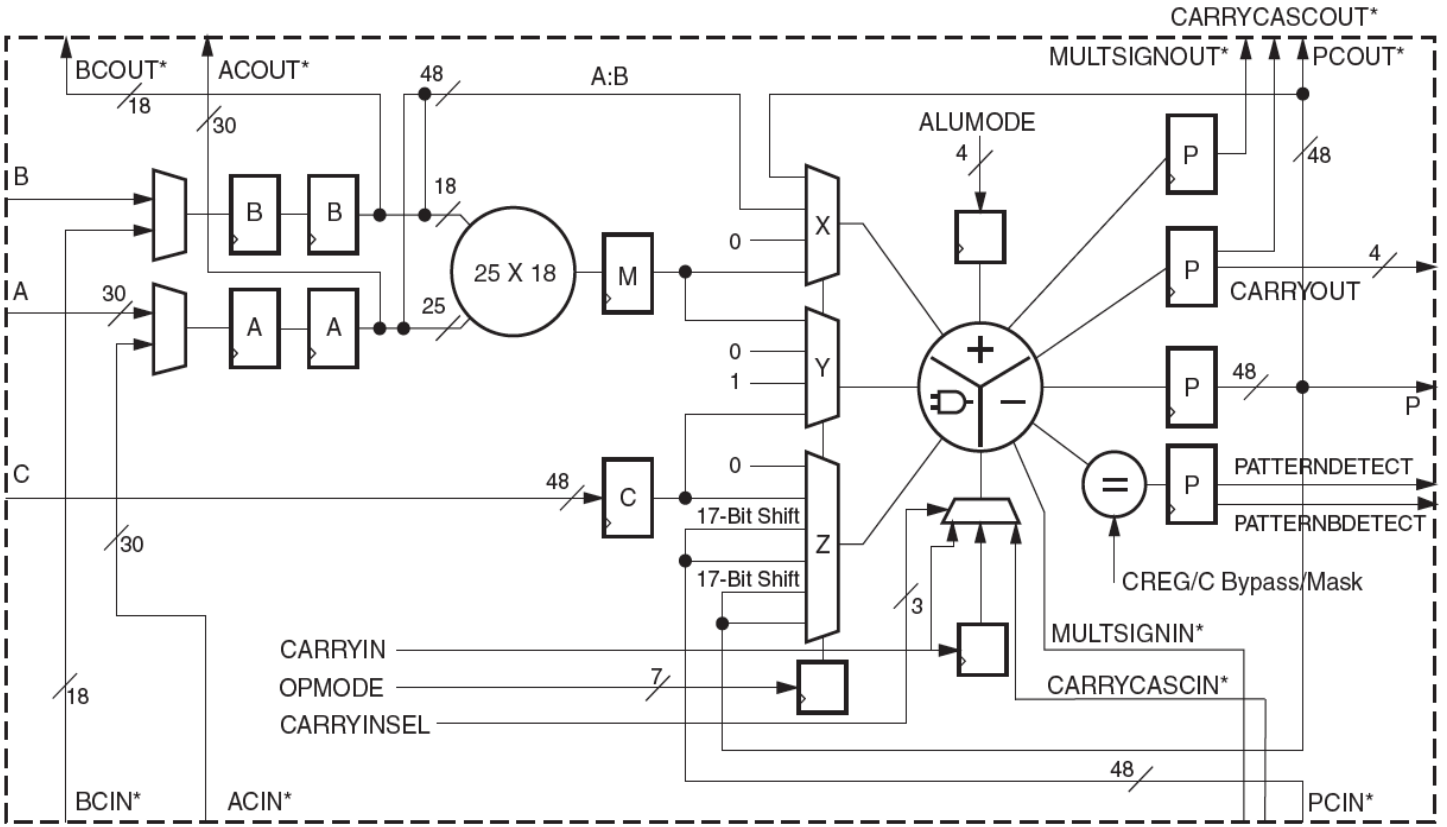
Block Memories

- Cyclone II (M4k), Stratix III (M9k),
- Spartan 3 (RAM18k), Virtex 5 (RAM36k)
- Aspect ratio (up to 32 bits words)
- 2xAES S-box/T-box: Spartan 3 BRAM -
- configured as dual port ROM
- (2kx8/512x32)
- 2xGroestl T-box: in 2xSpartan 3
- BRAMs - configured as
- dual port ROM (2kx8 and 512x32)

Xilinx Virtex 5 - RAM36k



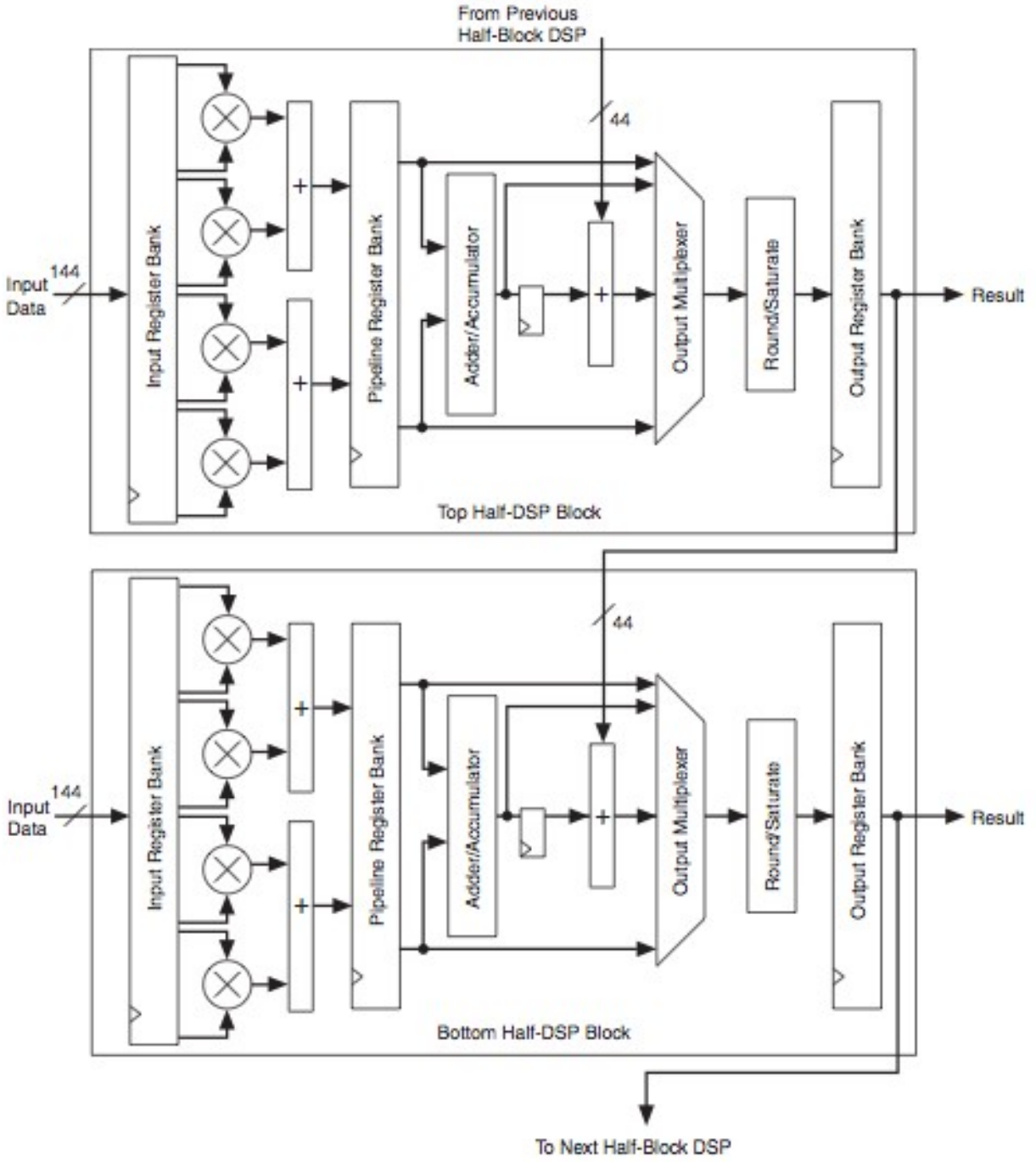
DSP48E Slice : Xilinx Virtex5



*These signals are dedicated routing paths internal to the DSP48E column. They are not accessible via fabric routing resources.

UG193_c1_01_032806

Full DSP Block
Altera Stratix III



Xilinx Virtex 5 results

Algorithm	Architecture	Throughput	Resource utilization	Tp/#CLB
		Mb/s	#CLB slice, #BRAMs, #DSPs	(Mb/s)/#CLB
	DSP Units			
Skein	Basic	2565	1306,0,0	1.96
	Embedded	2359	1264,0,32	1.86
DSP Units and Block RAMs				
Blake	Basic	2252	1702,0,0	1.32
	Embedded	1534	662,12,8	2.31
SHA-2	Basic	1504	418,0,0	3.6
	Embedded	1719	320,1,5	5.37
	Block RAMs			
Blake	Basic	2252	1854,0,0	1.32
	Embedded	1861	726,13,0	2.56
Groestl	Basic	6083	1852,0,0	3.28
	Embedded	5858	1255,50,0	4.67
JH	Basic	4917	1056,0,0	4.65
	Embedded	3120	1066,4,0	2.92
Keccak	Basic	13536	1352,0,0	10.01
	Embedded	11252	1338,1,0	8.41
SHA-2	Basic	1504	418,0,0	3.6
	Embedded	1591	381,1,0	4.17

Altera Stratix III results

Algorithm	Architecture	Throughput	Resource utilization	Tp/#ALUT
		Mb/s	#ALUT, #Mem_bits, #DSPs	(Mb/s)/#ALUT
DSP Units				
Skein	Basic	2426	4381,0,0	0.55
	Embedded	1472	5705,0,128	0.26
DSP Units and Block RAMs				
Blake	Basic	2086	4752,0,0	0.43
	Embedded	1073	1773,12k,32	0.6
SHA-2	Basic	1654	988,0,0	1.67
	Embedded	1621	795,2k,16	2.03
Block RAMs				
Blake	Basic	2086	4752,0,0	0.43
	Embedded	1808	1900,12k,0	0.94
Groestl	Basic	5649	7242,0,0	0.78
	Embedded	6082	4438,655k,0	2.18
JH	Basic	4654	3331,0,0	1.39
	Embedded	4651	2743,9k,0	1.7
Keccak	Basic	13746	4221,0,0	3.25
	Embedded	14103	4277,2k,0	3.3
SHA-2	Basic	1654	988,0,0	1.67
	Embedded	1661	956,2k,0	1.73

Throughput/area ranking changes after embedded resources used

Virtex 5

	basic	embedded	%
Blake	1.32	2.56	94
Groestl	3.28	4.67	42
JH	4.65	2.92	-37
Keccak	10.01	8.41	-16
Skein	1.96	1.86	-5

Stratix III

	basic	embedded	%
Blake	0.43	0.94	118
Groestl	0.78	2.18	179
JH	1.39	1.7	22
Keccak	3.25	3.3	2
Skein	0.55	0.26	-52

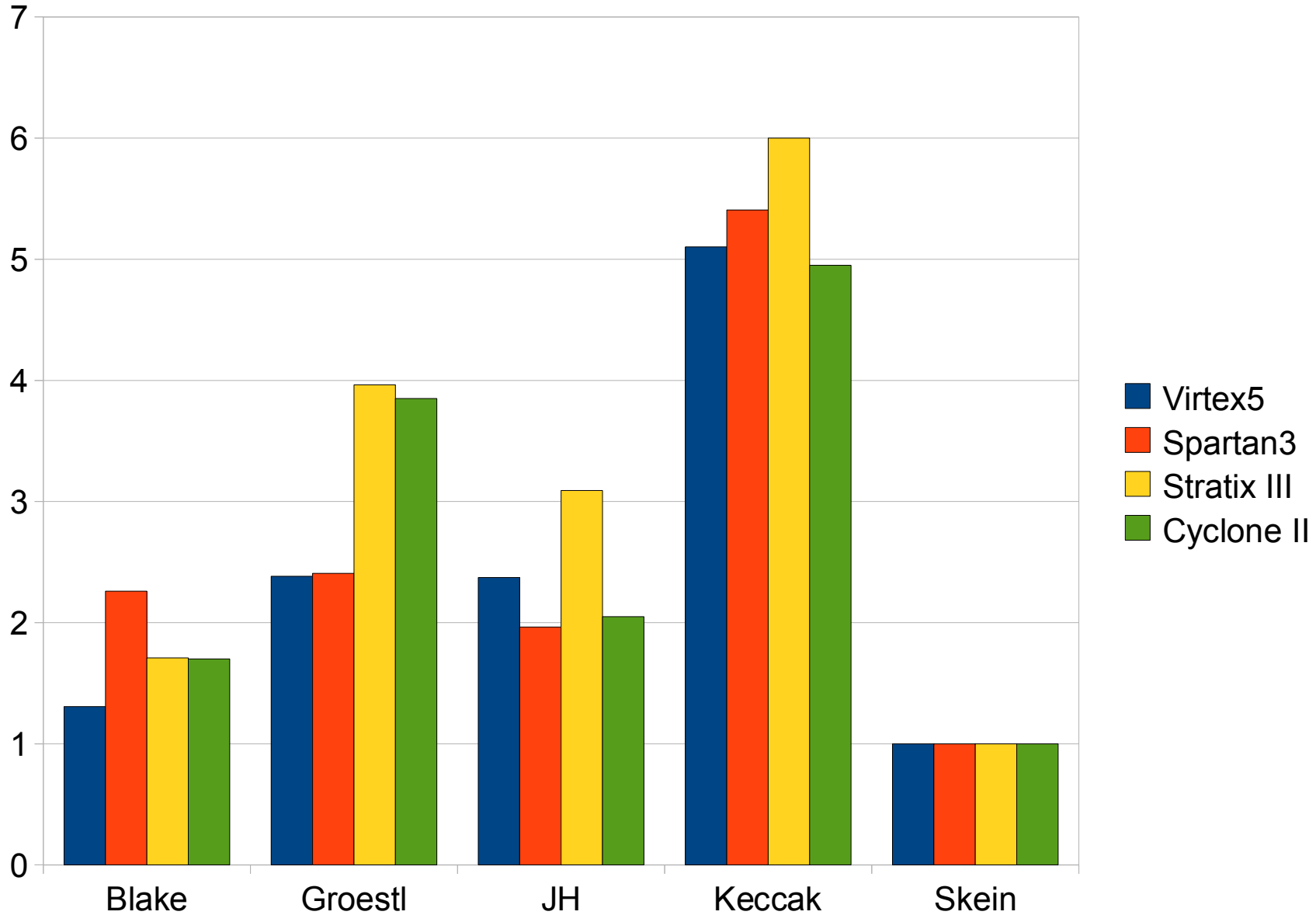
Spartan 3

	basic	embedded	%
Blake	0.25	0.61	144
Groestl	0.24	0.65	171
JH	0.41	0.53	29
Keccak	1.46	1.33	-9
Skein	0.27	N/A	N/A

Cyclone II

	basic	embedded	%
Blake	0.13	0.34	162
Groestl	0.15	0.77	413
JH	0.36	0.41	14
Keccak	0.99	0.98	-1
Skein	0.2	N/A	N/A

Normalized Throughput/Area of the Best Results out of Basic and Embedded Architectures



Conclusions

- Basic Architectures of SHA-3 candidates were enhanced by embedded resources – results were collected for both Altera and Xilinx low cost and high performance devices.
- The drop in frequency was caused by the interconnect delays between reconfigurable logic and embedded resources.
- DSP units and multipliers have limited importance for selected hash functions - majority of investigated algorithms use addition only.
- Except Skein on Altera Stratix III, significant portion of logic was shifted to embedded resources.
- The biggest improvement noted for Blake and Groestl FPGA architectures with hardwired components.
- SHA-3 Round 3 candidates ranking changes for High Speed implementations on FPGAs: 1. Keccak, 2. Groestl, 3. JH, 4. Blake, 5. Skein.


Low-Area
Implementations

Lightweight Implementations of the SHA-3 Finalists

Jens-Peter Kaps, Panasayya Yalla,
Kishore Kumar Surapathi, Bilal Habib,
Susheel Vadlamudi, and Smriti Gurung

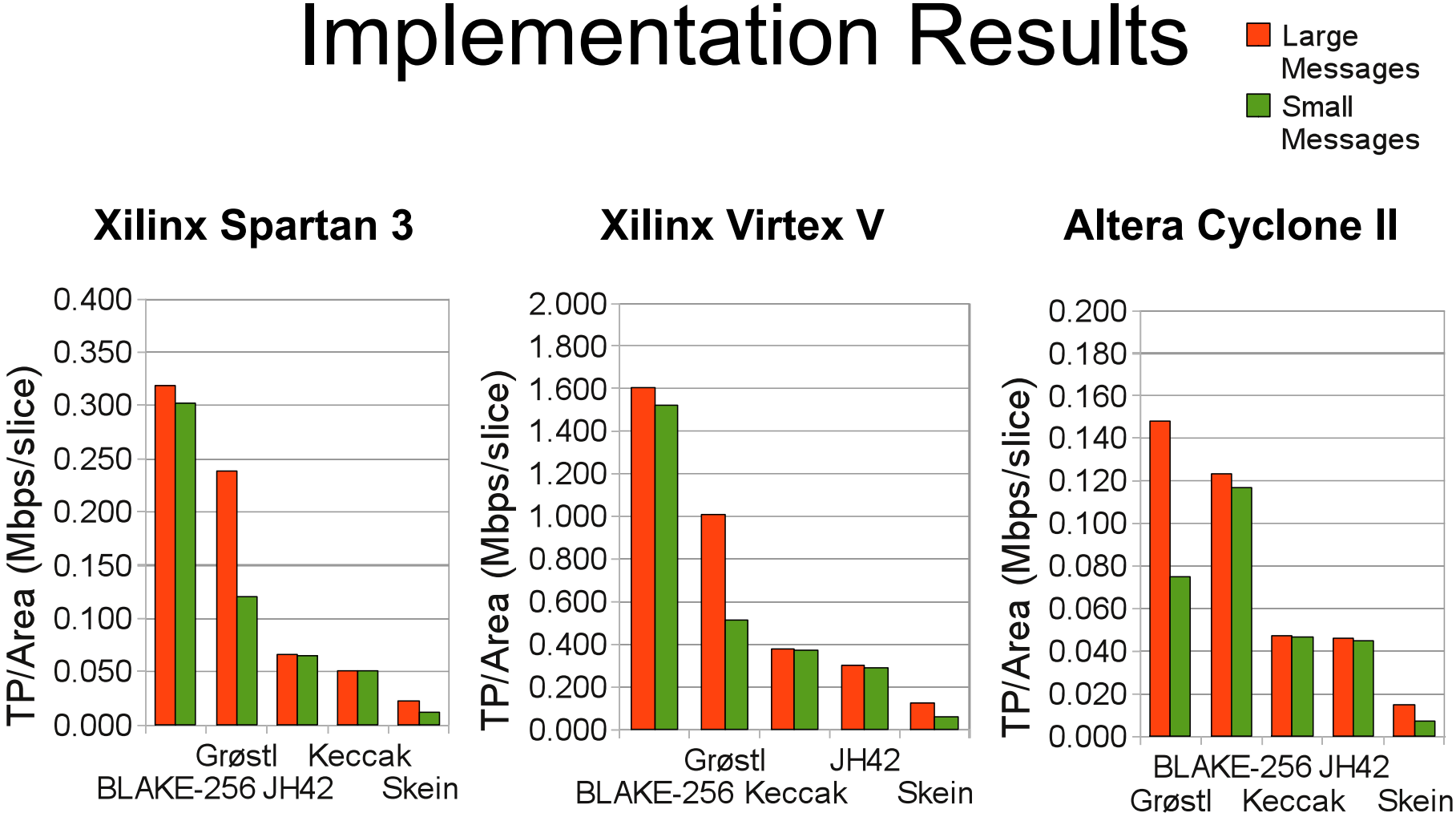
Assumptions

- Implementing for minimum area alone can lead to unrealistic run-times.
- $\Rightarrow$ Goal: Achieve the maximum Throughput/Area ratio for a given area budget.
- Realistic scenario:
  - System on Chip: Certain area only available.
  - Standalone: Smaller Chip, lower cost, but limit to smallest chip available, e.g. 768 slices on smallest Spartan 3 FPGA.

Target

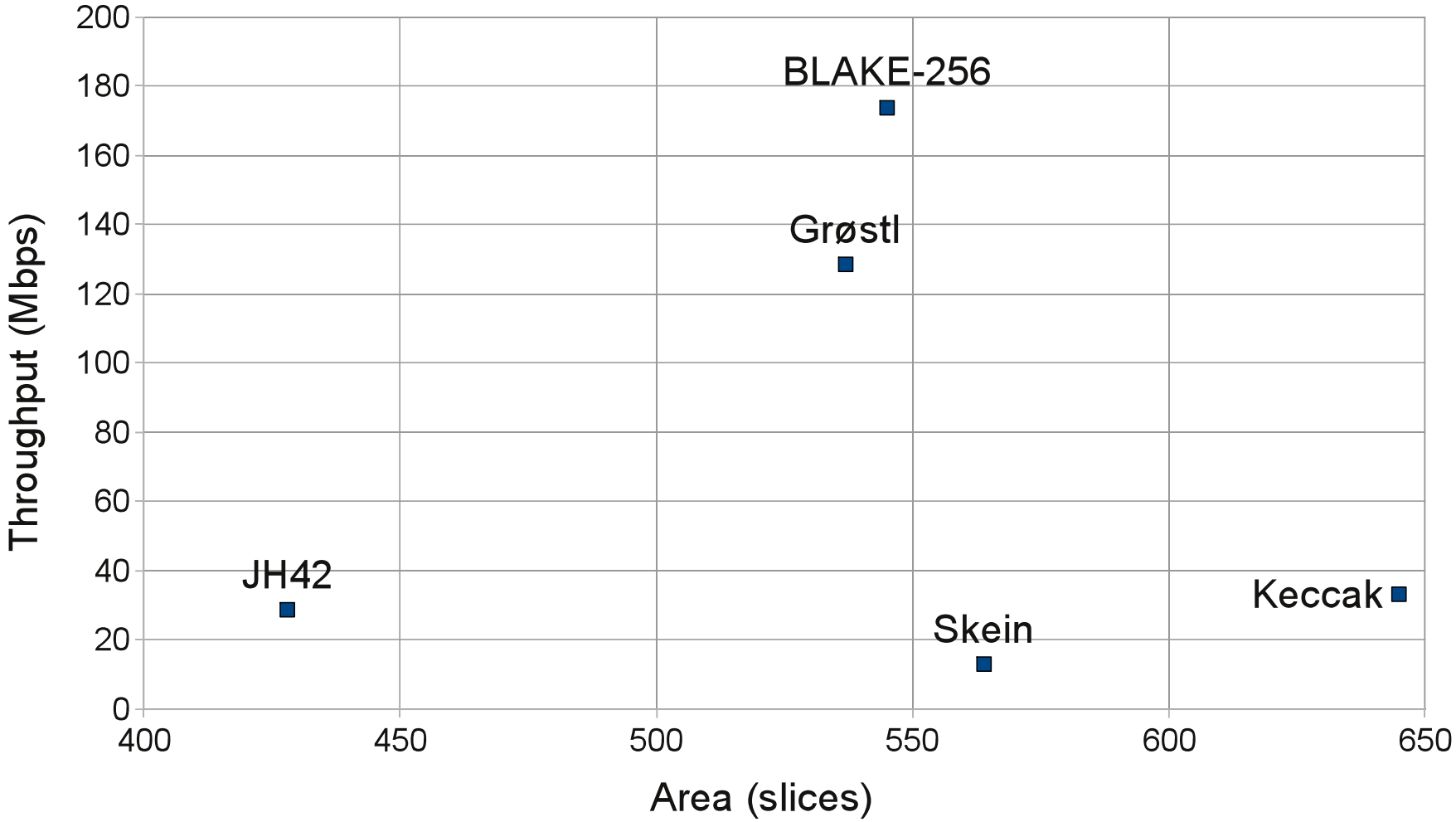
- Xilinx Spartan 3e, low cost FPGA family
- Budget: 500 slices, 1 Block RAM (BRAM)

Implementation Results



- Xilinx Spartan 3, ISE 12.3, after P&R, Optimized through ATHENa

Implementation Results



- Xilinx Spartan 3, ISE 12.3, after P&R, Optimized through ATHENa

Detailed Results

Algorithm	Area (slices)	Block RAMs	Maximum Delay (ns) T	Large Messages		Short Messages	
				TP (Mbps)	TP/Area (Mbps/slice)	TP (Mbps)	TP/Area (Mbps/slice)
BLAKE-256	545	1	8.42	173.8	0.32	164.8	0.302
Grøstl	537	1	6.95	128.3	0.24	64.9	0.121
JH42	428	1	9.74	28.5	0.07	27.7	0.065
Keccak	645	1	11.41	33.2	0.05	32.9	0.051
Skein	564	1	14.85	12.0	0.02	6.4	0.011

- Xilinx Spartan 3, ISE 12.3, after P&R, Optimized through ATHENa

Reproducibility of Results

GMU Source Codes

- First batch of GMU Source Codes made available at the ATHENa website at:

  http://cryprography.gmu.edu/athena

- Included in this release:
  - best non-pipelined architectures for each of the 14 Round 2 candidates and SHA-2
  - best non-pipelined architectures for each of the 5 Round 3 candidates
  - Each code supports two variants: with 256-bit and 512-bit output.

GMU Database of Results

- Available in the ATHENa database at

  http://cryptography.gmu.edu/athenadb

  20 functions (14 Round 2 SHA-3 + 5 Round 3 SHA-3 + SHA-2)

  x 2 variants

  x 11 FPGA families = 440 combinations

(440-not_fitting) = 379 optimized results

Support for easy replication of all results.

We invite other groups to submit results to our database

Invitation to Use ATHENa & ATHENa Database

http://cryptography.gmu.edu/athena

User

FPGA Synthesis and Implementation

6

5

Database query

Ranking of designs

2

3

HDL + scripts + configuration files

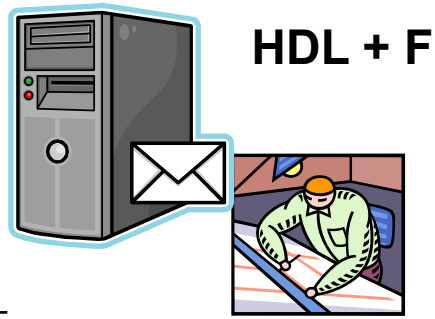
Result Summary + Database Entries

ATHENa Server

1

ATHENa scripts and configuration files

HDL + FPGA Tools



4

Database Entries

Designer

Interfaces + Testbenches

0

Thank you!

Questions?

Questions?

CERG: **http:/cryptography.gmu.edu**

ATHENa: **http:/cryptography.gmu.edu/athena**