

Cryptographic Contests: Toward Fair and Comprehensive Benchmarking of Cryptographic Algorithms in Hardware



**Kris Gaj
George Mason University**

Collaborators

Joint 3-year project (2010-2012) on benchmarking cryptographic algorithms in software and hardware sponsored by



software



**Daniel J. Bernstein,
University of Illinois
at Chicago**

FPGAs



**Jens-Peter Kaps
George Mason
University**

FPGAs/ASICs



**Patrick
Schaumont
Virginia Tech**

ASICs



**Leyla
Nazhand-Ali
Virginia Tech**



CERG @ GMU

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



**10 PhD students
8 MS students
co-advised by Kris Gaj & Jens-Peter Kaps**



Outline

- **Crypto 101**
- **Cryptographic standard contests**
- **Progress in evaluation methods**
 - **AES**
 - **eSTREAM**
 - **SHA-3**
- **Benchmarking tools for software and FPGAs**
- **Open problems**



Crypto 101

Cryptography is Everywhere



Buying a book on-line



Withdrawing cash from ATM



**Teleconferencing
over Intranets**



**Backing up files
on remote server**

Cryptographic Transformations Most Often Implemented in Practice

Secret-Key Ciphers

Hash Functions

Block Ciphers

Stream Ciphers

encryption

message & user
authentication

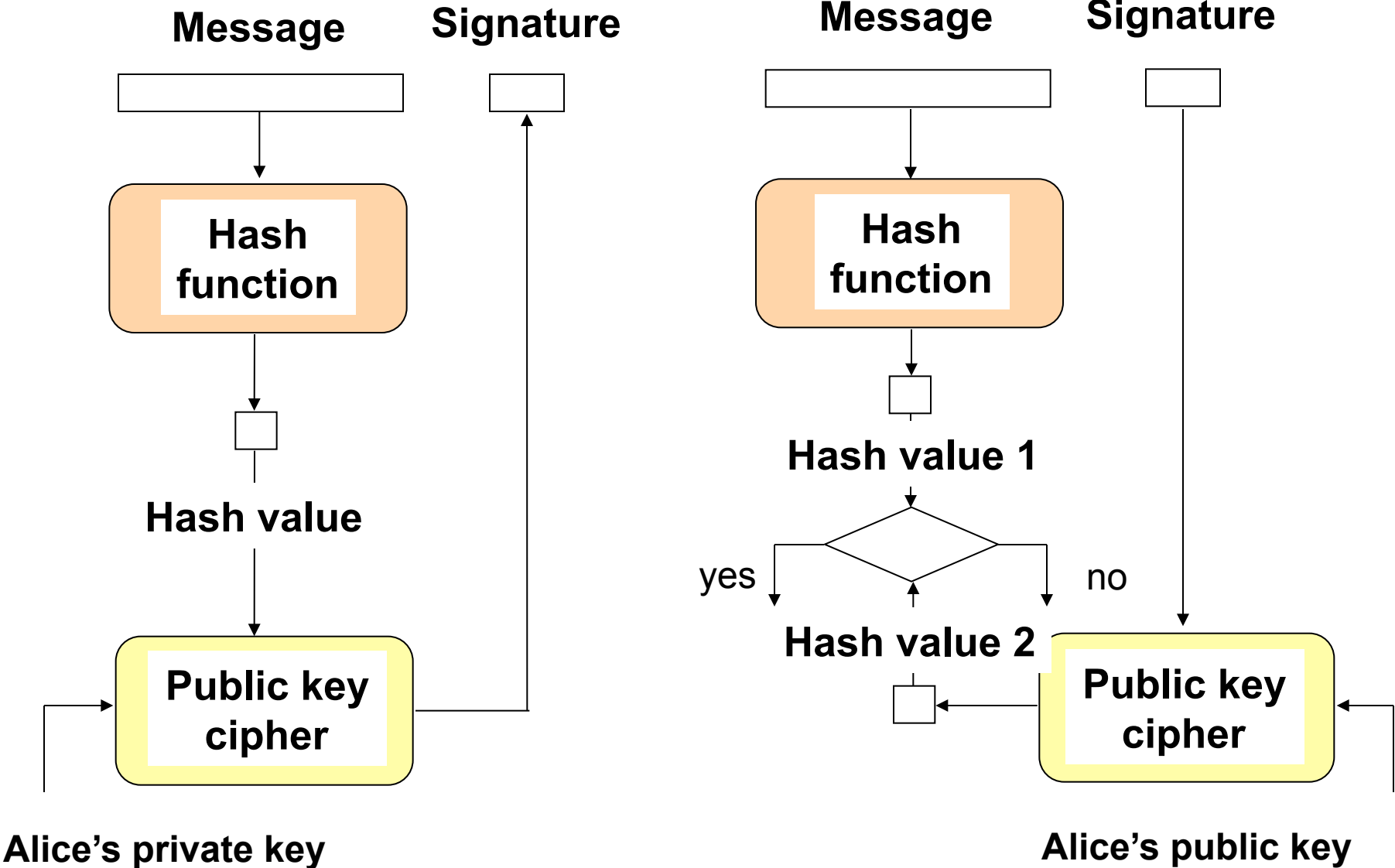
Public-Key Cryptosystems

digital signatures
key agreement
key exchange

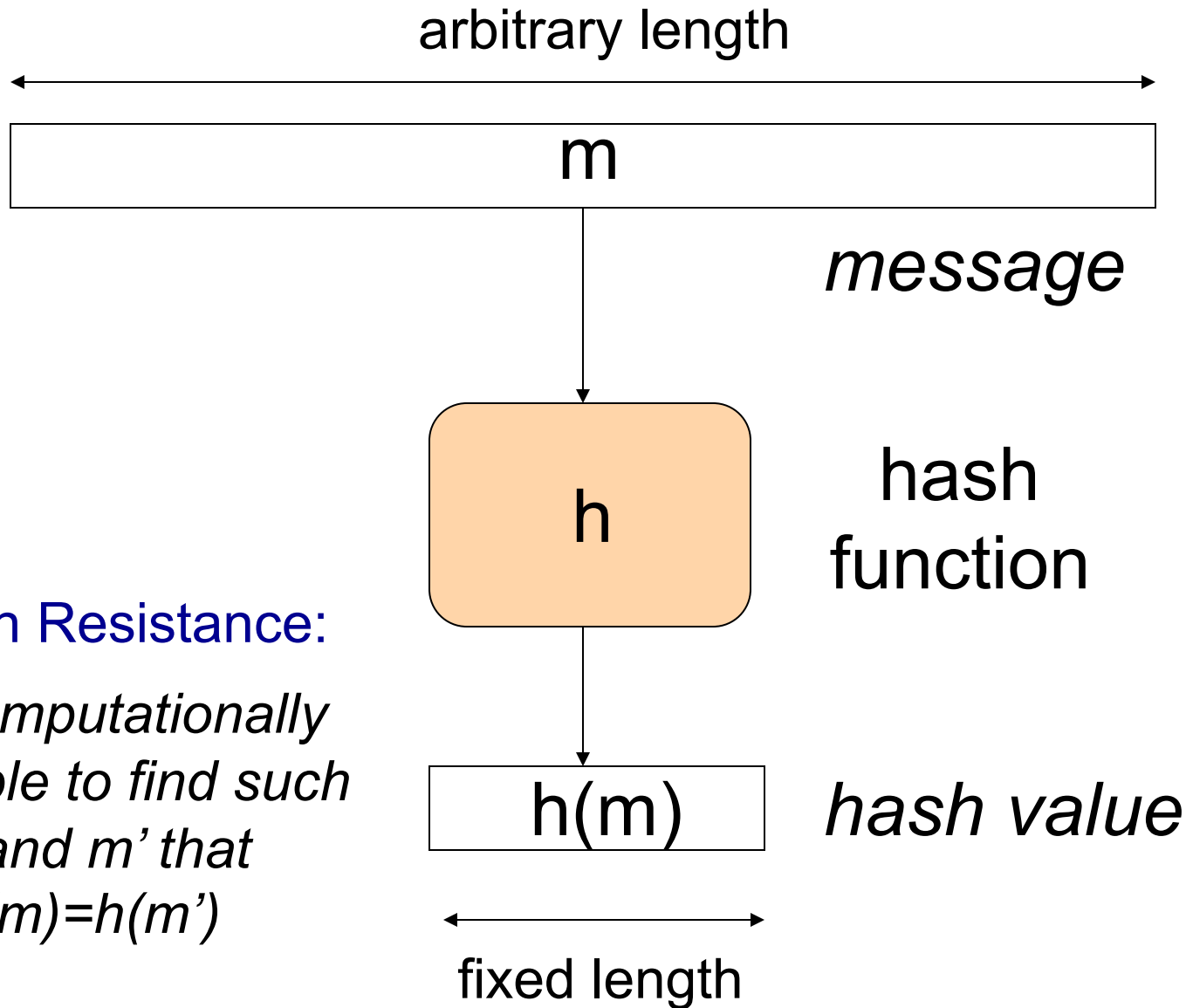
Hash Functions in Digital Signature Schemes

Alice

Bob



Hash Function



Collision Resistance:

It is computationally infeasible to find such m and m' that $h(m)=h(m')$



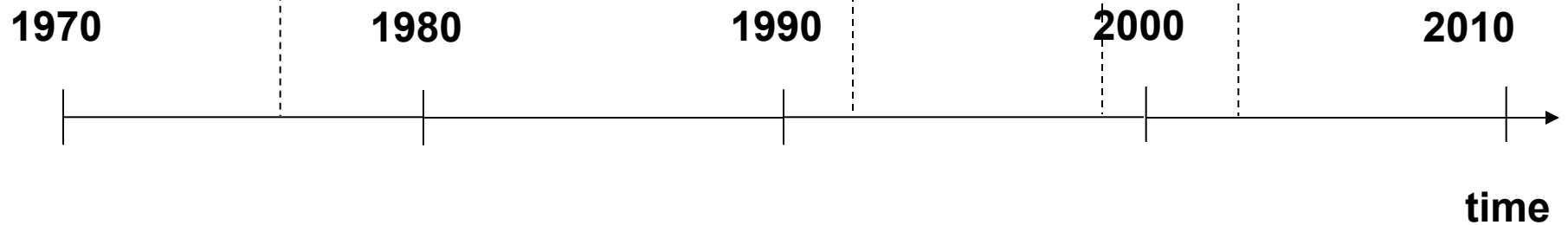
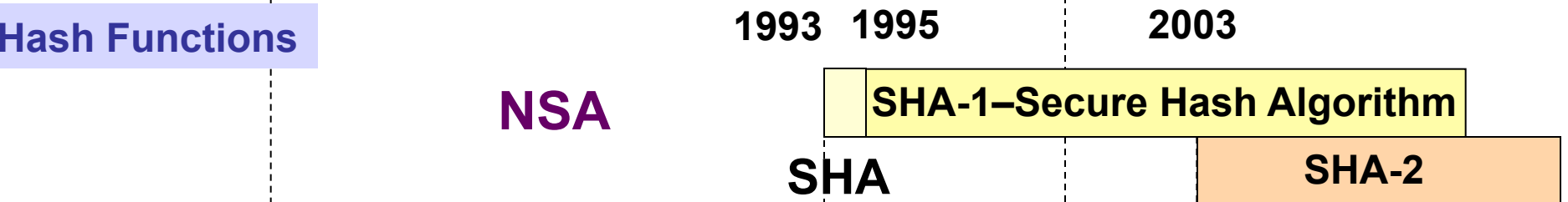
**Cryptographic
Standard
Contests**

Cryptographic Standards Before 1997

Secret-Key Block Ciphers



Hash Functions

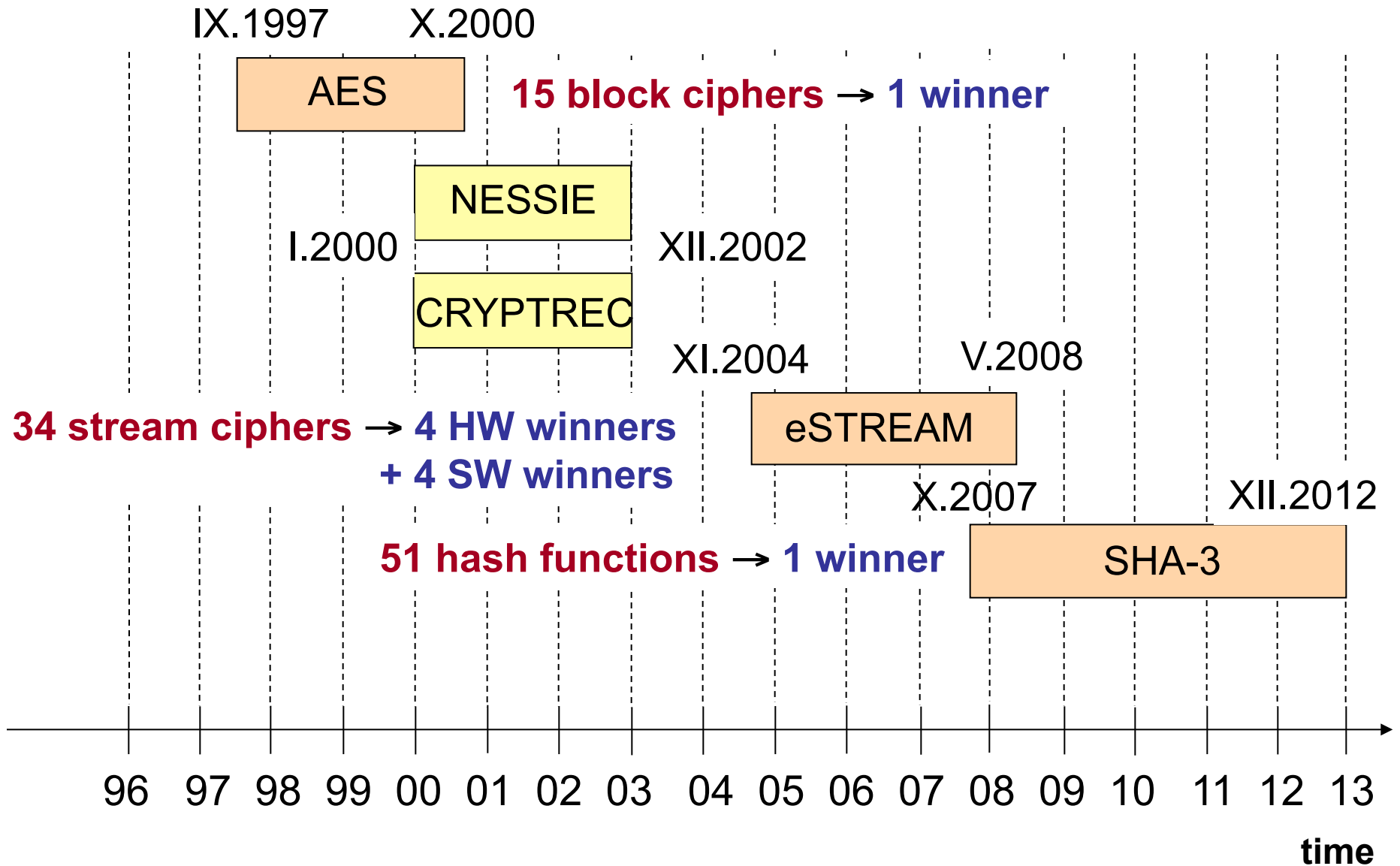


Why a Contest for a Cryptographic Standard?

- Avoid **back-door** theories
- Speed-up the **acceptance** of the standard
- **Stimulate** non-classified research on methods of designing a specific cryptographic transformation
- **Focus** the effort of a relatively small cryptographic community



Cryptographic Standard Contests



Cryptographic Contests - Evaluation Criteria

Security

Software Efficiency

μProcessors μControllers

Hardware Efficiency

FPGAs ASICs

Flexibility

Simplicity

Licensing

Specific Challenges of Evaluations in Cryptographic Contests

- Very wide range of possible applications, and as a result performance and cost targets

throughput: single Mbits/s to hundreds Gbits/s

cost: single cents to thousands of dollars

- Winner in use for the next 20-30 years, implemented using technologies not in existence today
- Large number of candidates
- Limited time for evaluation
- Only one winner and the results are final



Mitigating Circumstances

- Security is a primary criterion
- Performance of competing algorithms tend to vary significantly (sometimes as much as 500 times)
- Only relatively large differences in performance matter (typically at least 20%)
- Multiple groups independently implement the same algorithms (catching mistakes, comparing best results, etc.)
- Second best may be good enough



**AES
Contest
1997-2000**

Rules of the Contest

Each team submits

Detailed
cipher
specification

Justification
of design
decisions

Tentative
results
of cryptanalysis

Source
code
in C

Source
code
in Java

Test
vectors

AES: Candidate Algorithms



Canada:

CAST-256
Deal

USA:

Mars
RC6
Twofish
Safer+
HPC

Costa Rica:

Frog



Germany:

Magenta

Belgium:

Rijndael

France:

DFC

Israel, UK,

Norway:

Serpent



Korea:

Crypton

Japan:

E2



Australia:

LOKI97

AES Contest Timeline

June 1998

15 Candidates

CAST-256, Crypton, Deal, DFC, E2,
Frog, HPC, LOKI97, Magenta, Mars,
RC6, Rijndael, Safer+, Serpent, Twofish,

Round 1

Security
Software efficiency

August 1999

5 final candidates

Mars, RC6, Twofish (USA)
Rijndael, Serpent (Europe)

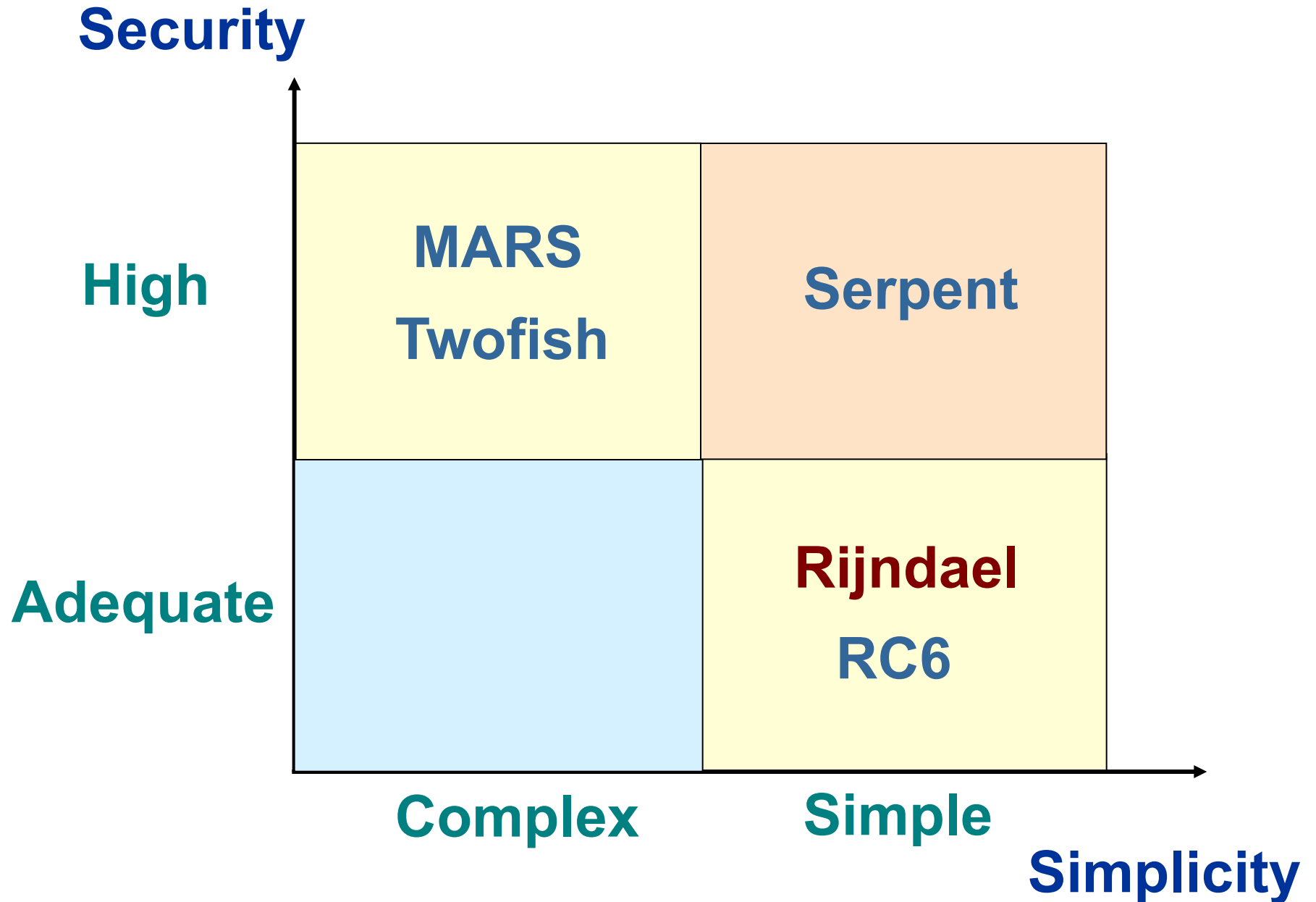
Round 2

Security
Software efficiency
Hardware efficiency

October 2000

1 winner: Rijndael
Belgium

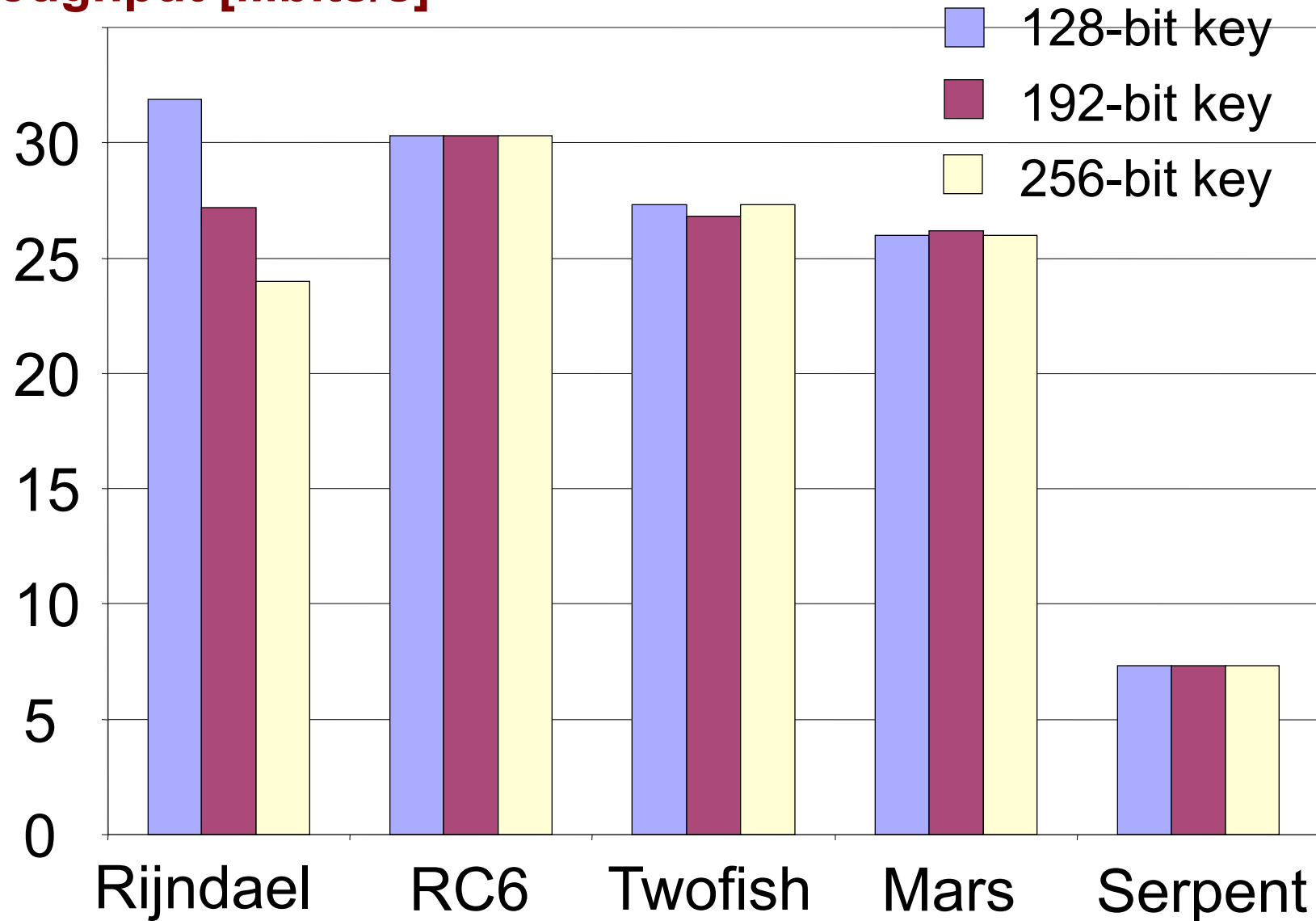
NIST Report: Security & Simplicity



Efficiency in software: NIST-specified platform

200 MHz Pentium Pro, Borland C++

Throughput [Mbits/s]



NIST Report: Software Efficiency

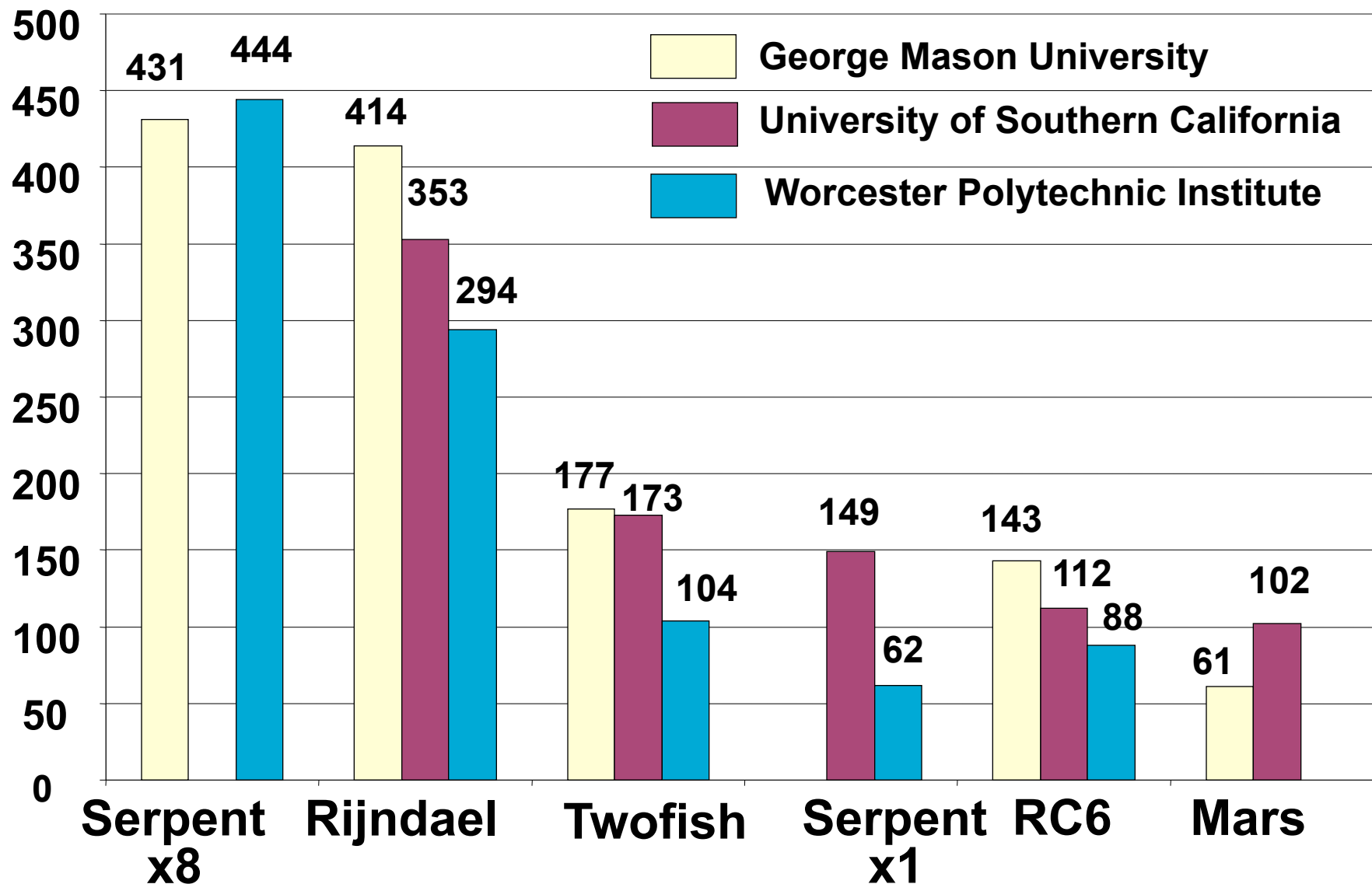
Encryption and Decryption Speed

	32-bit processors	64-bit processors	DSPs
high	RC6	Rijndael Twofish	Rijndael Twofish
medium	Rijndael Mars Twofish	Mars RC6	Mars RC6
low	Serpent	Serpent	Serpent

Efficiency in FPGAs: Speed

Xilinx Virtex XCV-1000

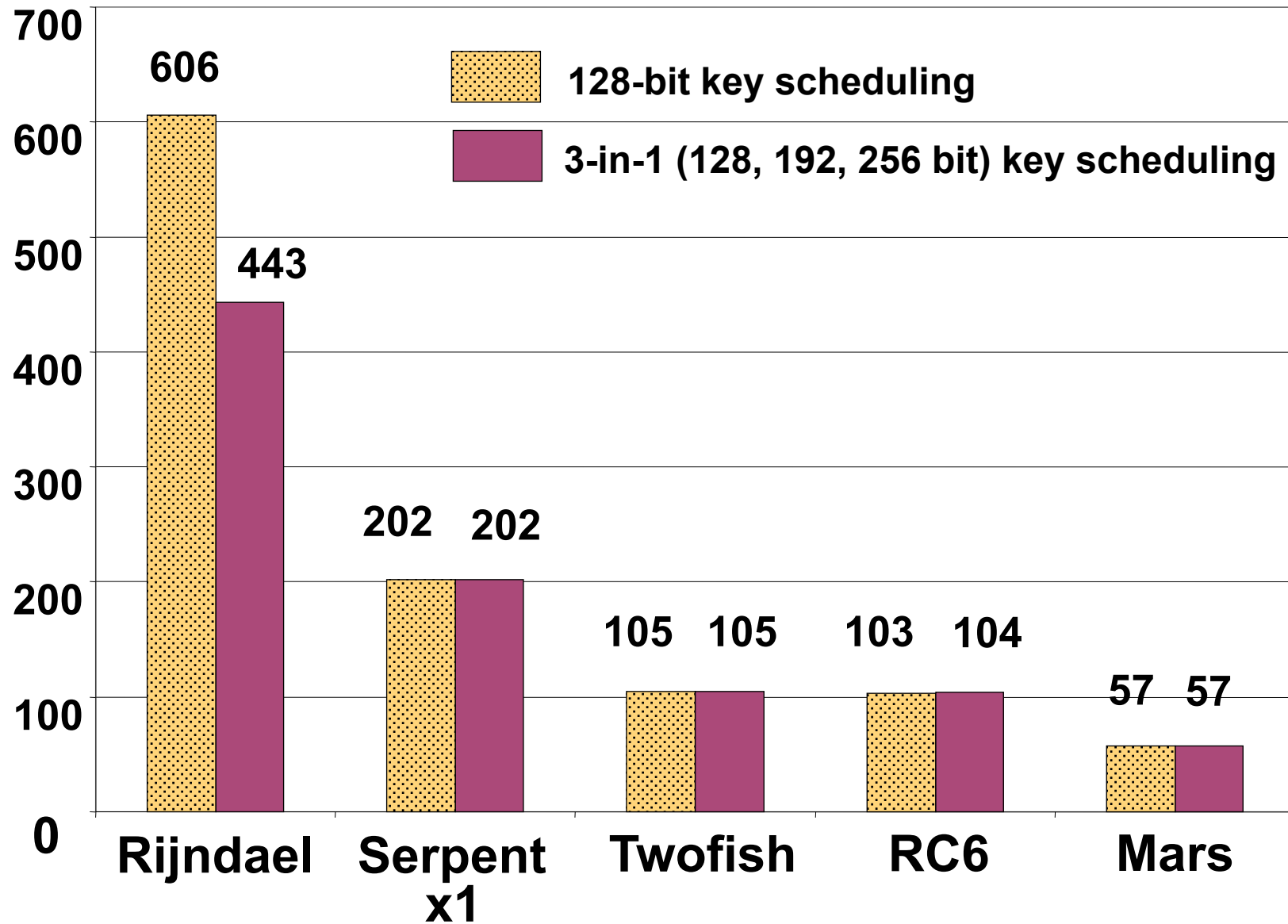
Throughput [Mbit/s]



Efficiency in ASICs: Speed

MOSIS 0.5 μ m, NSA Group

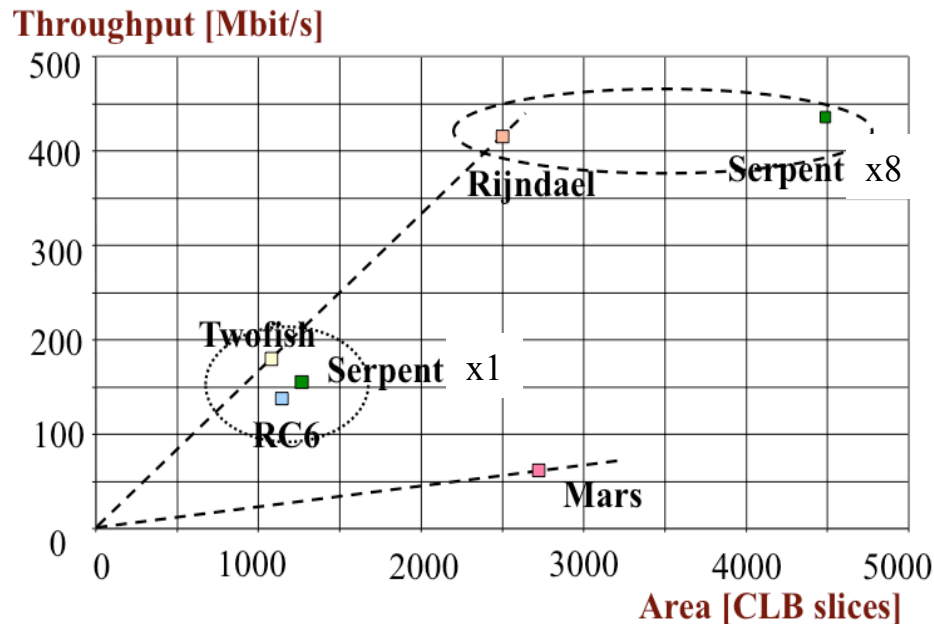
Throughput [Mbit/s]



Lessons Learned

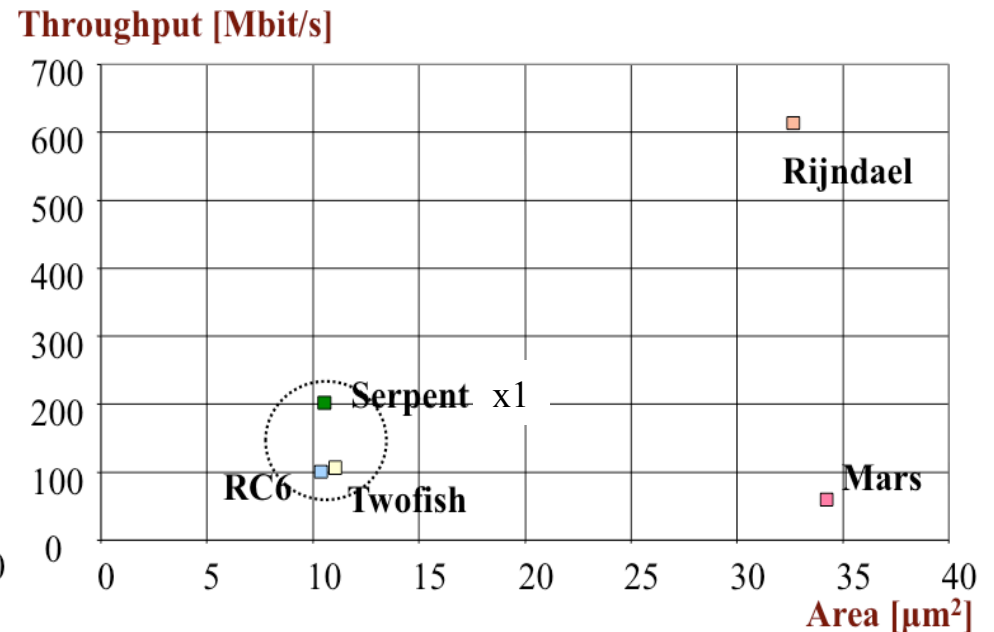
Results for ASICs matched very well results for FPGAs, and were both very different than software

FPGA



GMU+USC, Xilinx Virtex XCV-1000

ASIC



NSA Team, ASIC, 0.5 μm MOSIS

Serpent fastest in hardware, slowest in software

Lessons Learned

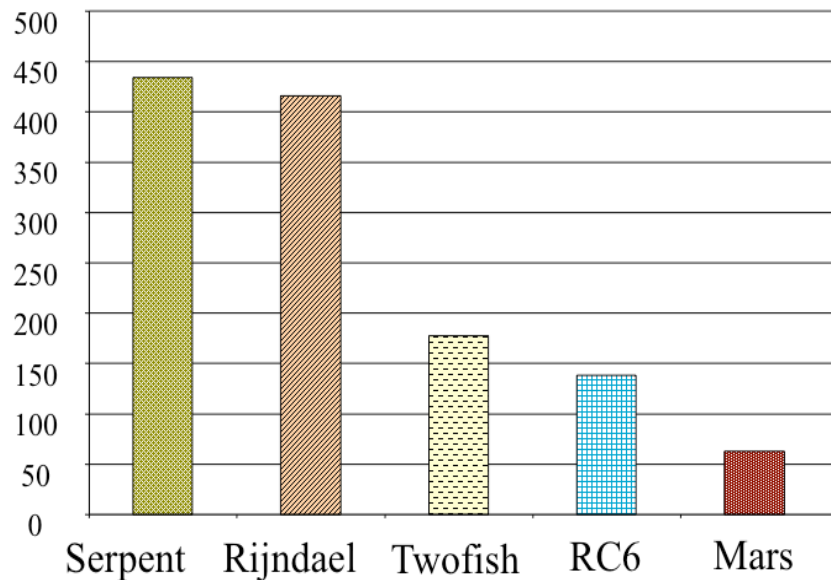
Hardware results matter!

Final round of the AES Contest, 2000

Speed in FPGAs

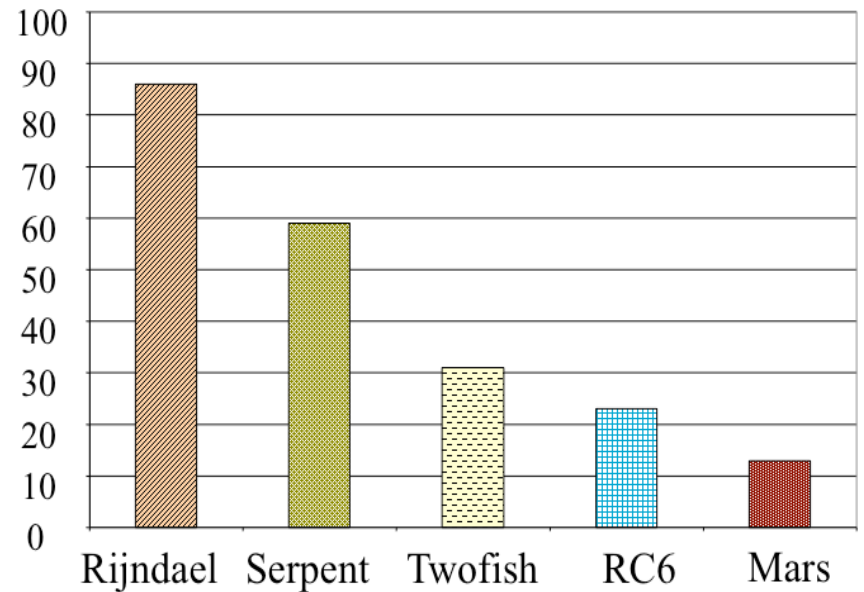
GMU results

Speed [Mbit/s]



Votes at the AES 3 conference

votes



Limitations of the AES Evaluation

- Optimization for **maximum throughput**
- **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



**eSTREAM
Contest
2004-2008**



eSTREAM - Contest for a new stream cipher standard

PROFILE 1 (SW)

- Stream cipher suitable for **software implementations** optimized for **high speed**
- Key size - 128 bits
- Initialization vector – 64 bits or 128 bits

PROFILE 2 (HW)

- Stream cipher suitable for **hardware implementations** with **limited memory, number of gates, or power supply**
- Key size - 80 bits
- Initialization vector – 32 bits or 64 bits

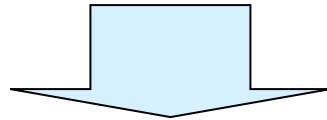
eSTREAM Contest Timeline

April 2005

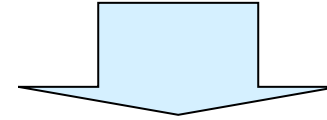
PROFILE 1 (SW)

PROFILE 2 (HW)

23 Phase 1 Candidates

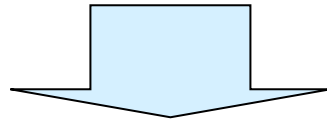


25 Phase 1 Candidates

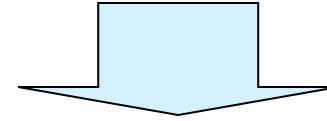


July 2006

13 Phase 2 Candidates

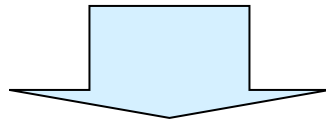


20 Phase 2 Candidates

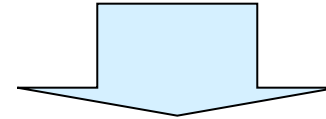


April 2007

8 Phase 3 Candidates



8 Phase 3 Candidates



May 2008

4 winners:

**HC-128, Rabbit,
Salsa20, SOSEMANUK**

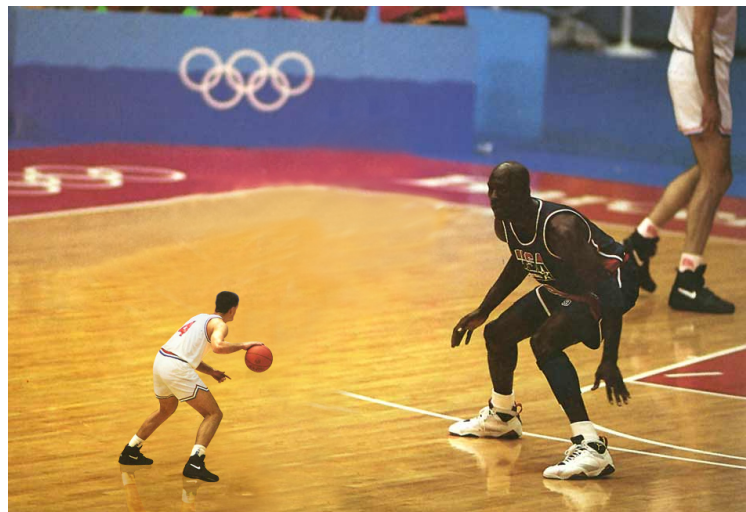
4 winners:

**Grain v1, Mickey v2,
Trivium, ~~F-FCSR-H v2~~**

Lessons Learned

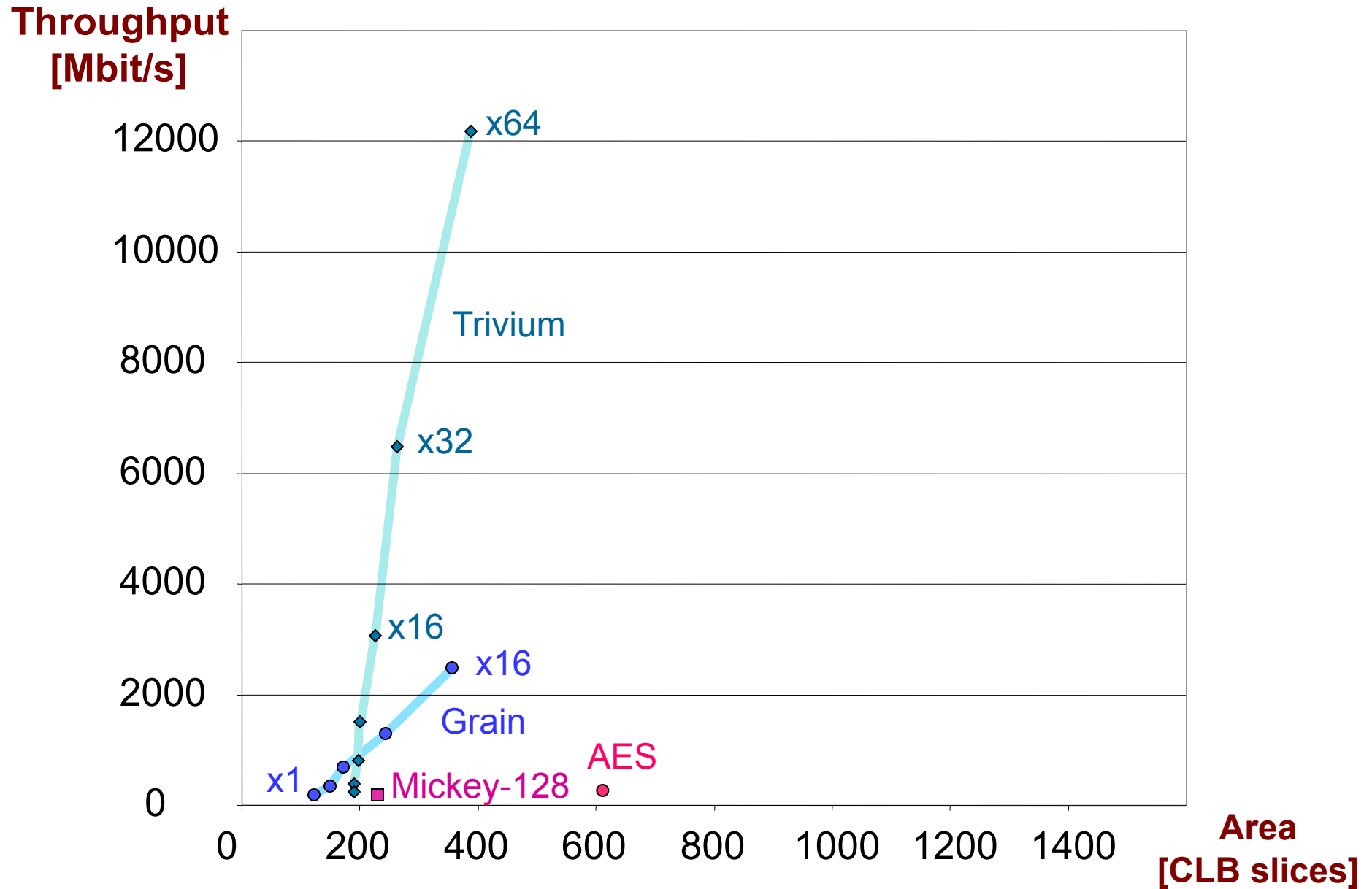
**Very large differences among
8 leading candidates**

- ~30 x** in terms of area (Grain v1 vs. Edon80)
- ~500 x** in terms of the throughput to area ratio (Trivium (x64) vs. Pomaranch)



Hardware Efficiency in FPGAs

Xilinx Spartan 3, GMU SASC 2007



ASIC Evaluations

- **Two major projects**

- T. Good, M. Benaissa, University of Sheffield, UK
(Phases 1-3) – 0.13 μ m CMOS

eSCARGO 

- F.K. Gürkaynak, et al., ETH Zurich, Switzerland
(Phase 1) - 0.25 μ m CMOS

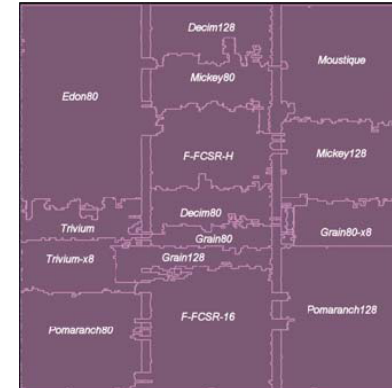
- **Two representative applications**

- **WLAN @ 10 Mbits/s**
- **RFID / WSN @ 100 kHz clock**

eSTREAM ASIC Evaluations

New compared to AES:

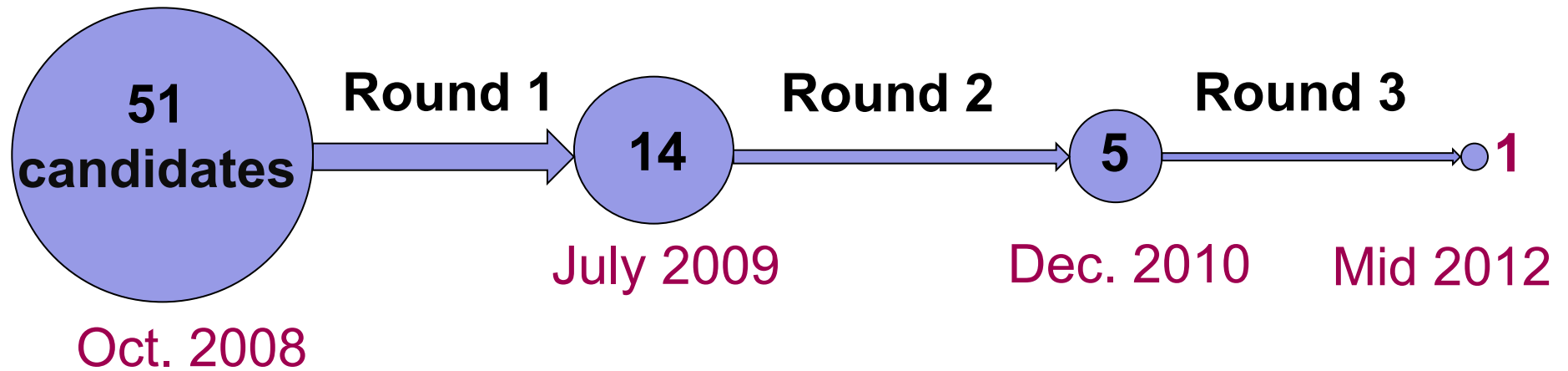
- **Post-layout** results, followed by
- Actually **fabricated ASIC chips** (0.18 μ m CMOS)
- More complex performance measures
 - **Power x Area x Time**
- New types of analyses
 - Power x Latency vs. Area
 - Throughput/Area vs. Energy per bit





**SHA-3
Contest
2007-2012**

NIST SHA-3 Contest - Timeline



SHA-3 Round 2

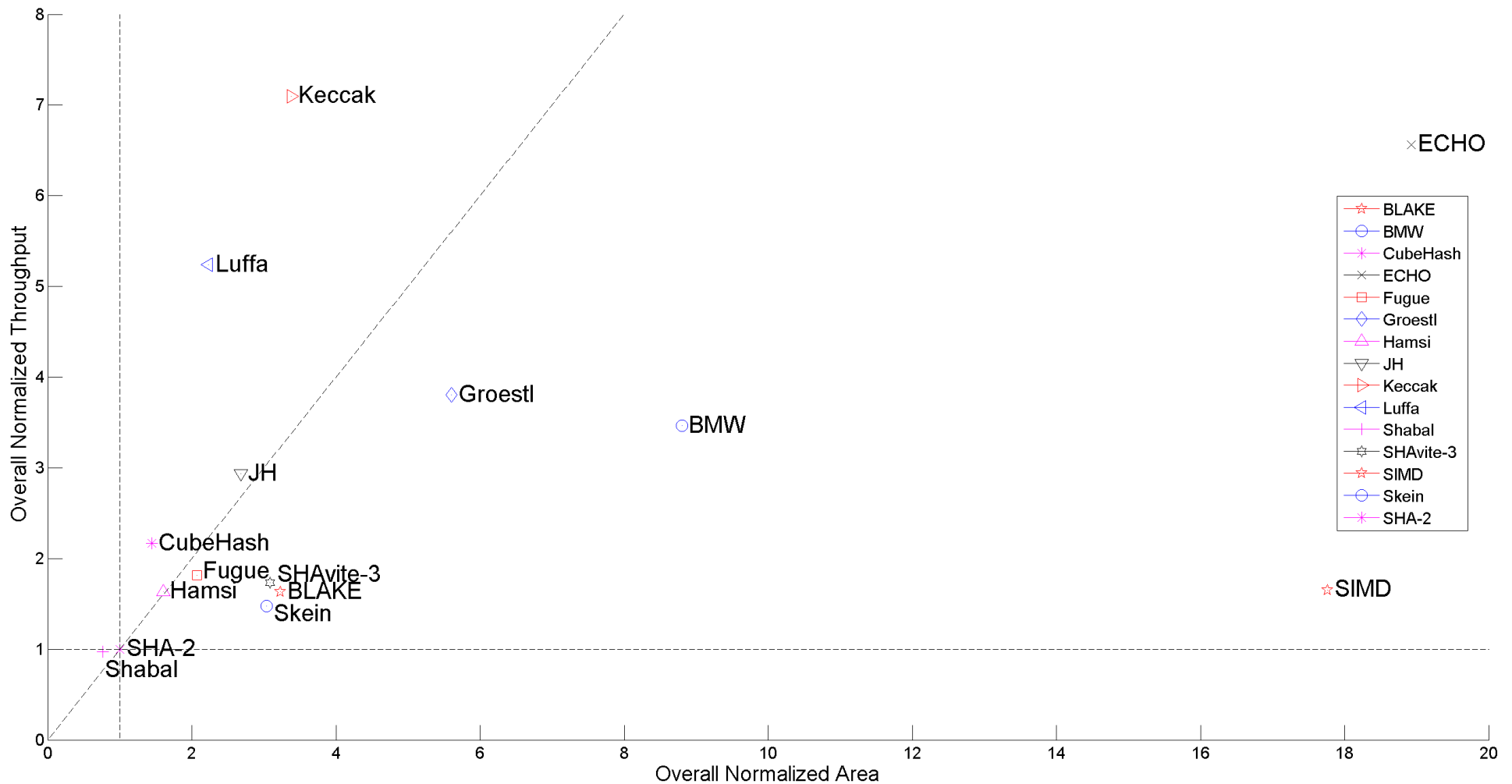
Features of the SHA-3 Round 2 Evaluation

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

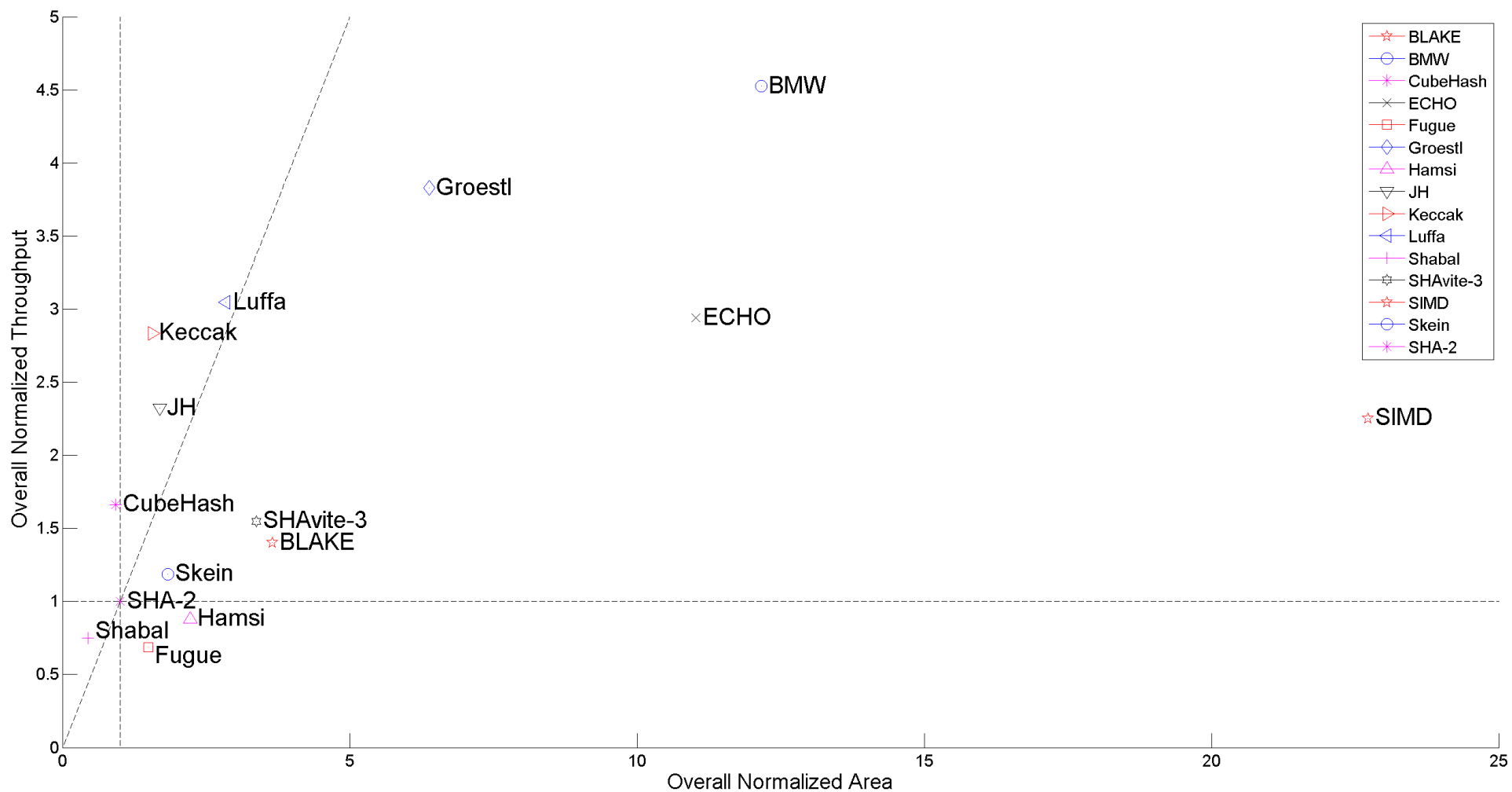
But still...

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

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



Performance Metrics

Primary

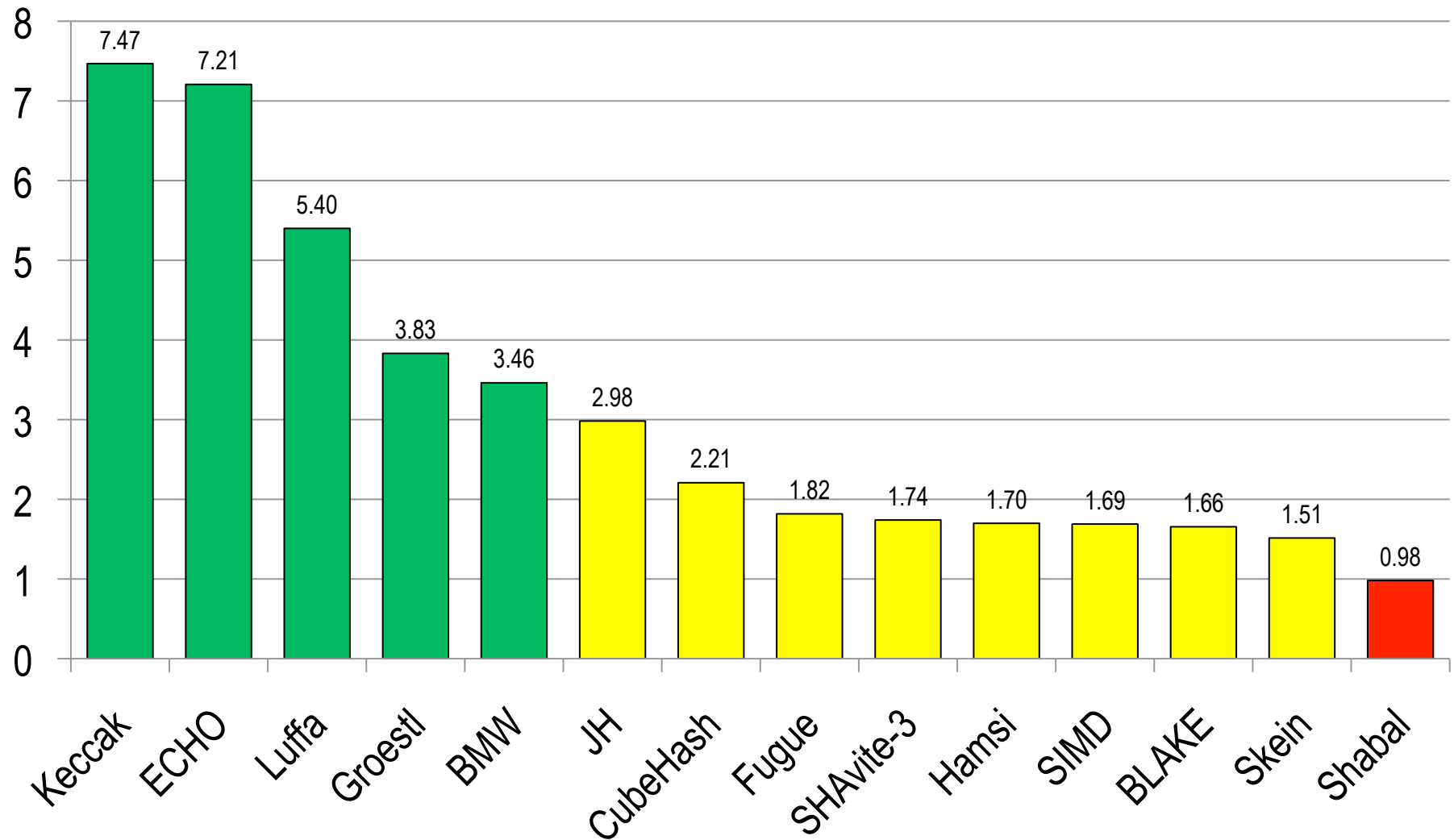
1. Throughput
(single message)
3. Throughput / Area

Secondary

2. Area
3. Hash Time for
Short Messages
(up to 1000 bits)

Overall Normalized Throughput: 256-bit variants of algorithms

Normalized to SHA-256, Averaged over 10 FPGA families



256-bit variants

512-bit variants

	Thr/Area	Thr	Area	Short msg.	Thr/Area	Thr	Area	Short msg.
BLAKE	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Red
➔ BMW	Yellow	Green	Yellow	Yellow	Yellow	Green	Red	Yellow
CubeHash	Green	Yellow	Green	Red	Green	Yellow	Green	Red
➔ ECHO	Yellow	Green	Red	Green	Yellow	Green	Red	Green
Fugue	Yellow	Yellow	Yellow	Yellow	Yellow	Red	Green	Yellow
➔ Groestl	Yellow	Green	Yellow	Yellow	Yellow	Green	Yellow	Green
Hamsi	Green	Yellow	Green	Yellow	Yellow	Red	Yellow	Yellow
➔ JH	Green	Yellow	Yellow	Yellow	Green	Yellow	Green	Green
➔ Keccak	Green	Green	Yellow	Green	Green	Green	Green	Green
➔ Luffa	Green	Green	Yellow	Green	Green	Green	Yellow	Green
Shabal	Green	Red	Green	Red	Green	Red	Green	Red
SHAvite-3	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow
➔ SIMD	Red	Yellow	Red	Red	Red	Yellow	Red	Red
Skein	Yellow	Yellow	Yellow	Yellow	Yellow	Yellow	Green	Yellow


SHA-3 Round 3

SHA-3 Contest Finalists



New in Round 3

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

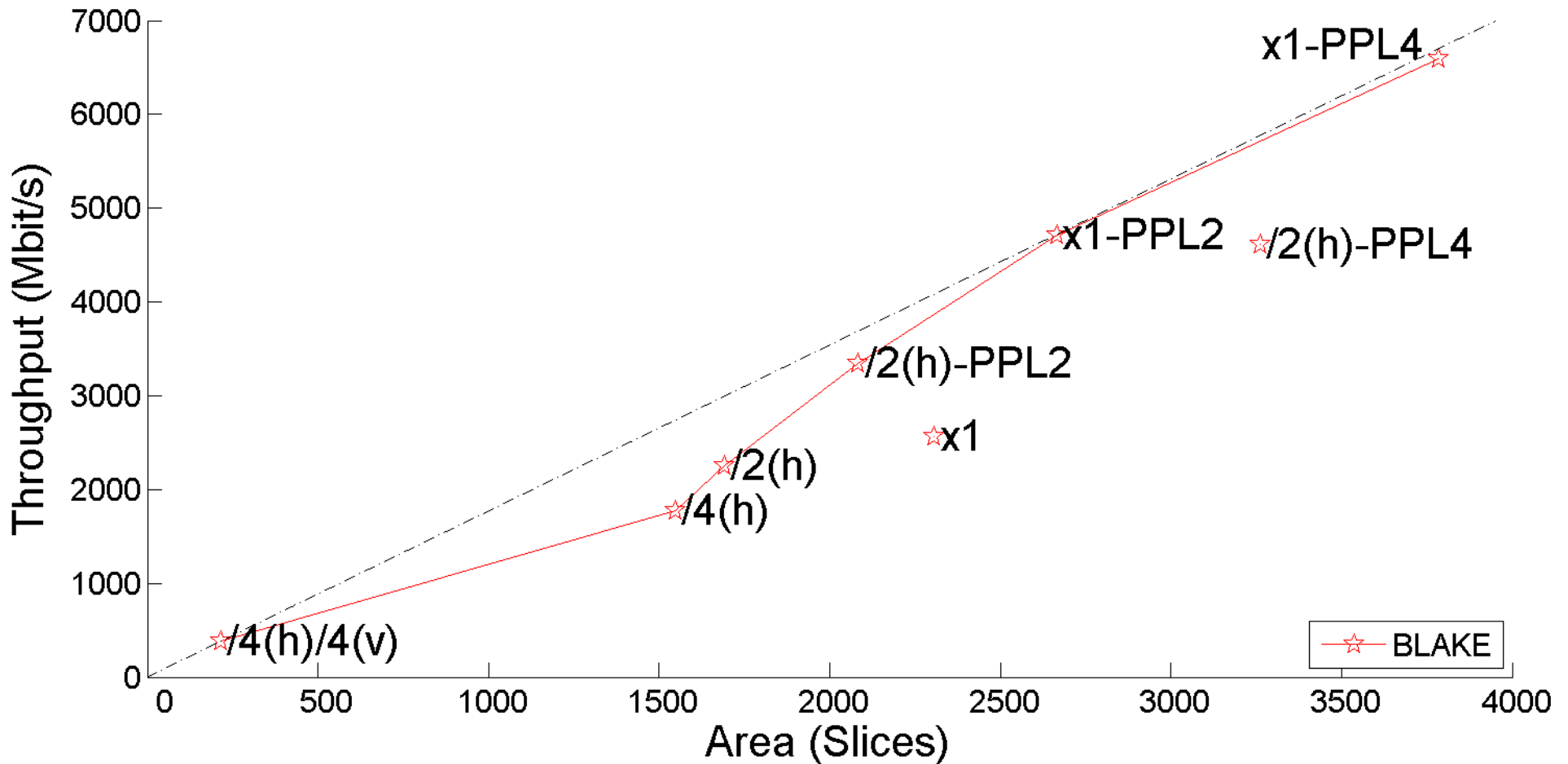


**SHA-3
Multiple
High-Speed
Architectures**

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

BLAKE-256 in Virtex 5



$x1$ – basic iterative architecture

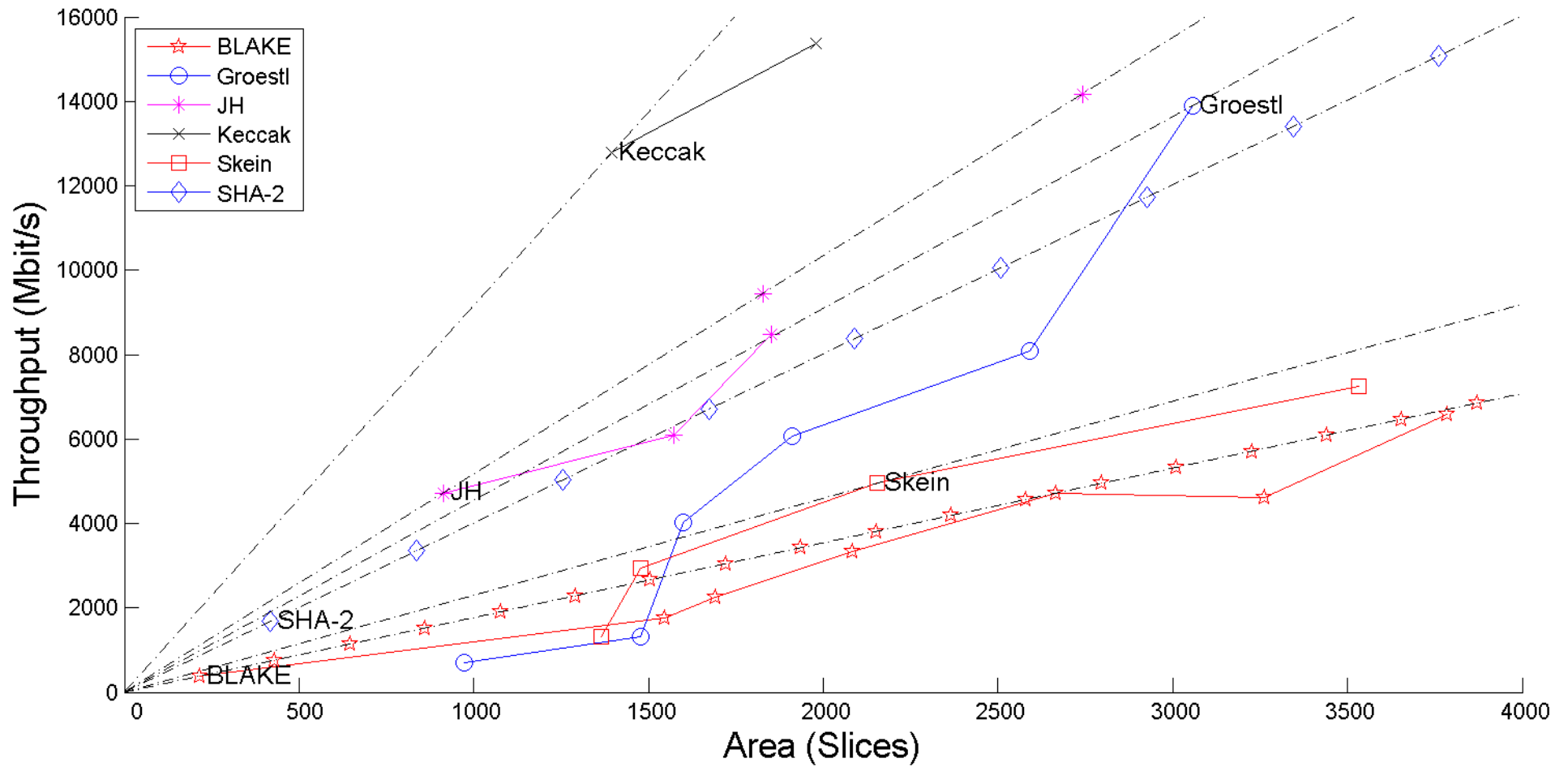
$/k(h)$ – horizontal folding by a factor of k

xk – unrolling by a factor of k

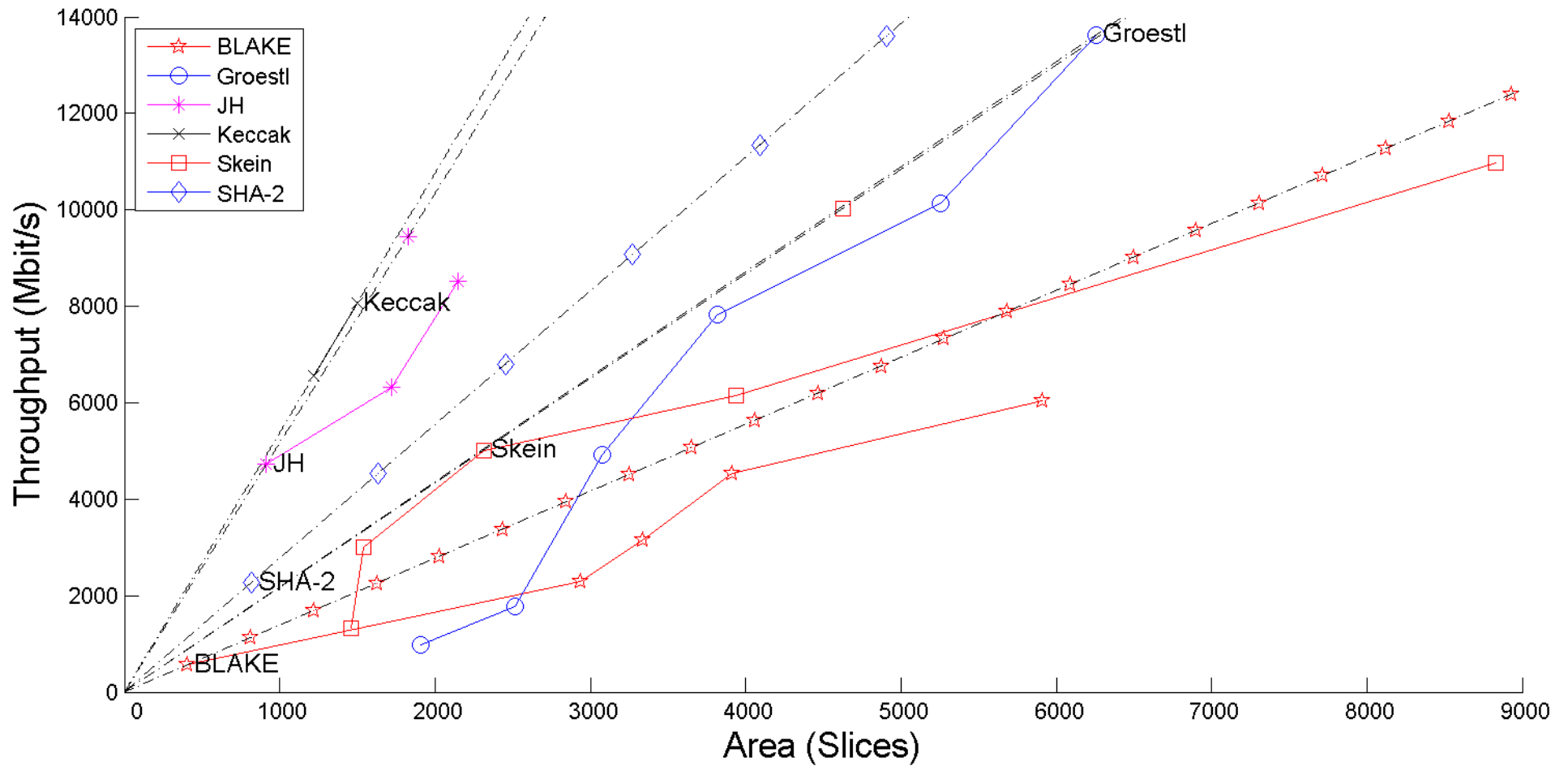
$/k(v)$ – vertical folding by a factor of k

$xk-PPLn$ – unrolling by a factor of k with n pipeline stages

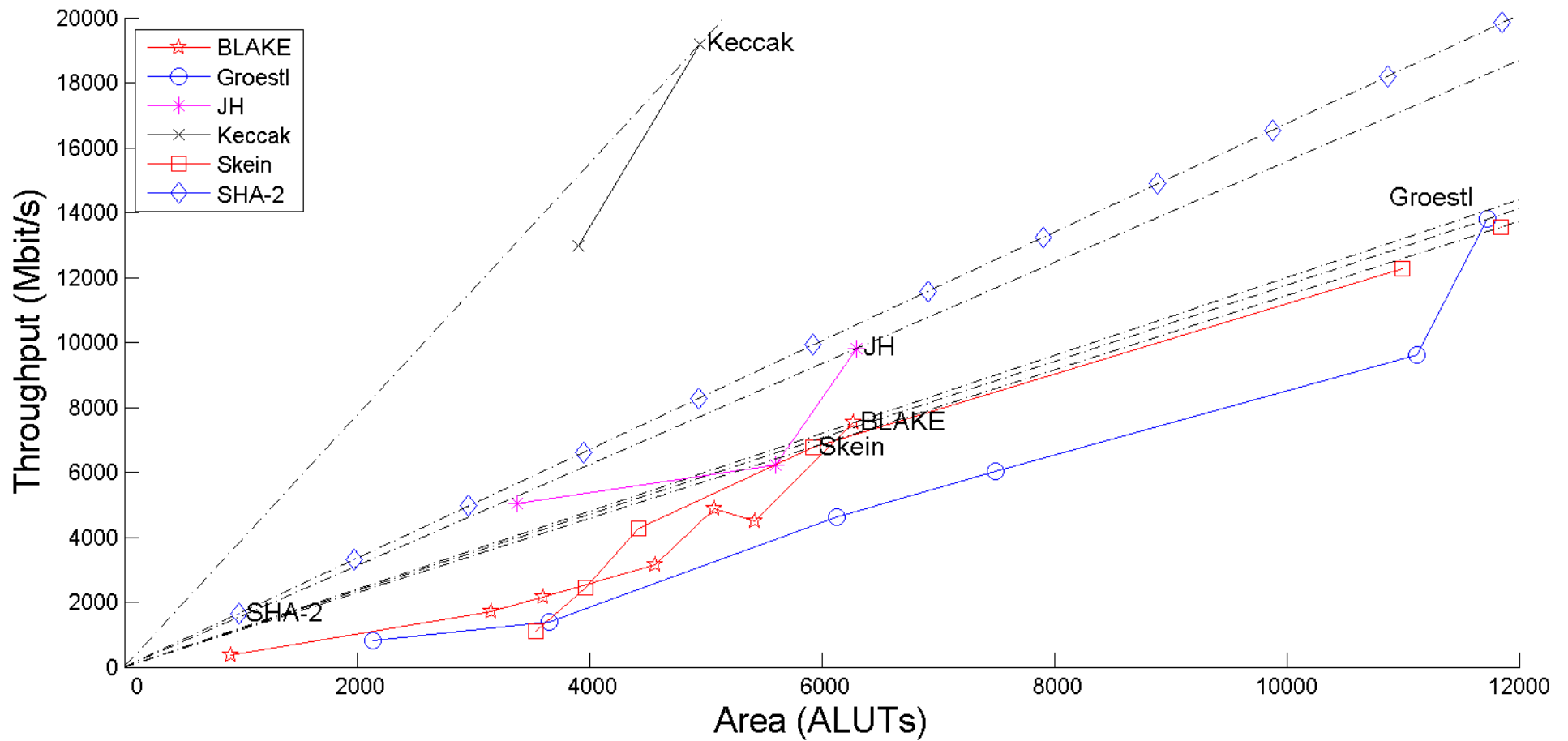
256-bit variants in Virtex 5



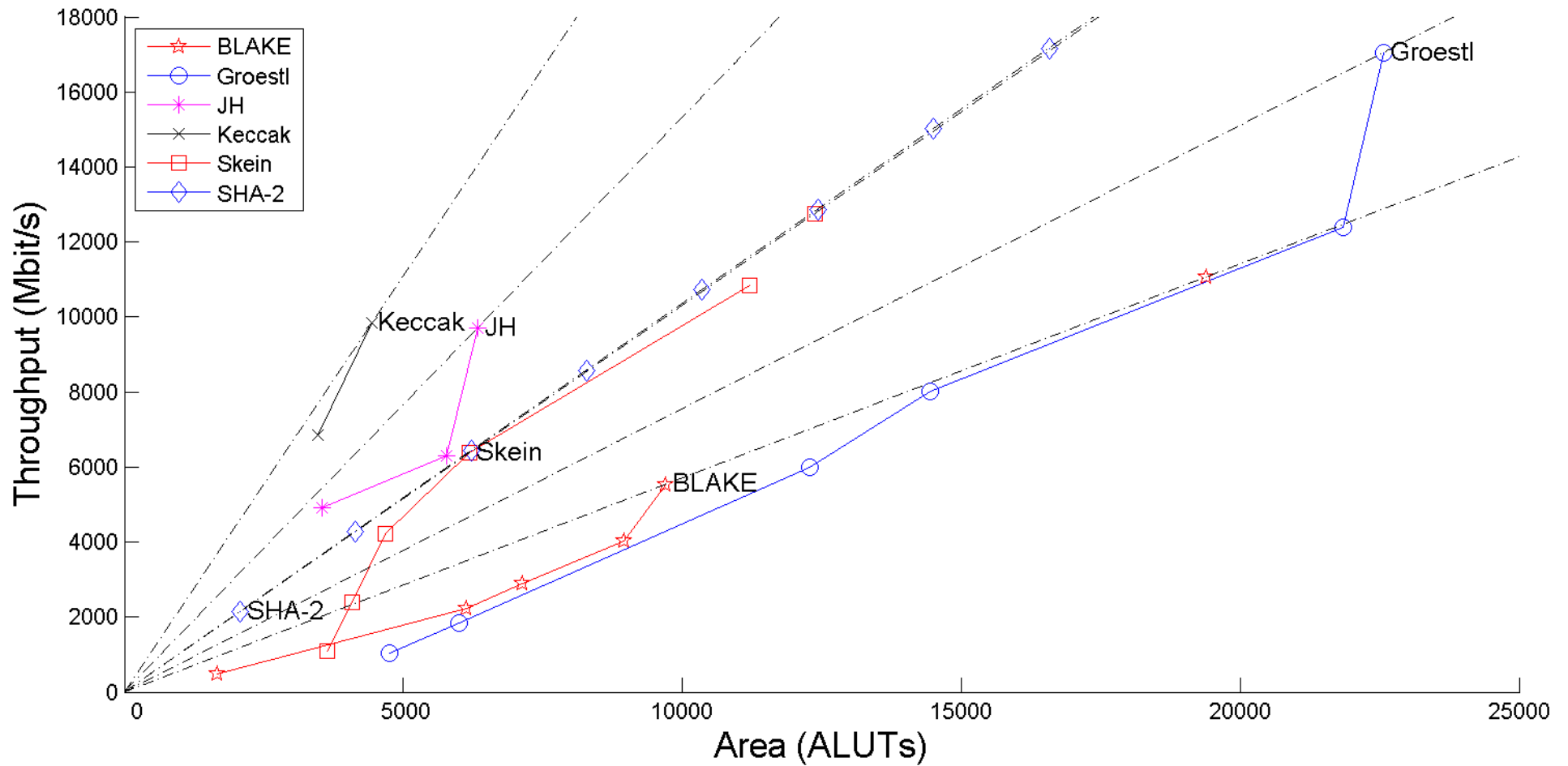
512-bit variants in Virtex 5




256-bit variants in Stratix III



512-bit variants in Stratix III





**SHA-3
Lightweight
Implementations**

Study of Lightweight Implementations in FPGAs

- **Two major projects**
 - J.-P. Kaps, et al., George Mason University, USA
 - F.-X. Standaert, UCL Crypto Group, Belgium
- **Target:**
 - Low-cost FPGAs (Spartan 3, Spartan 6, etc.)
for stand-alone implementations
 - High-performance FPGAs (e.g., Virtex 6)
for system-on-chip implementations

Typical Assumptions – GMU Group

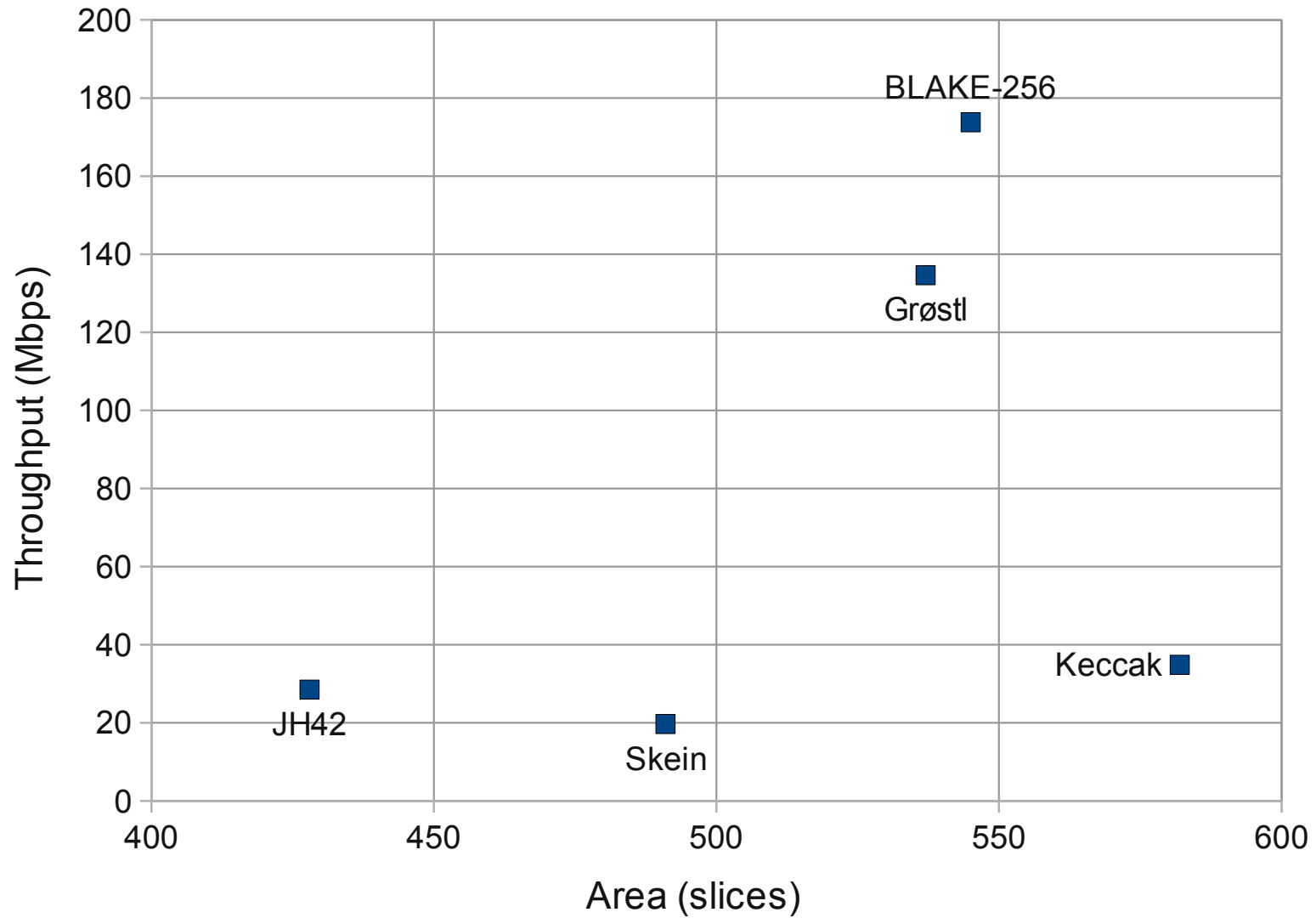
Assumptions

- Implementing for minimum area alone can lead to unrealistic run-times.
- ⇒ 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.

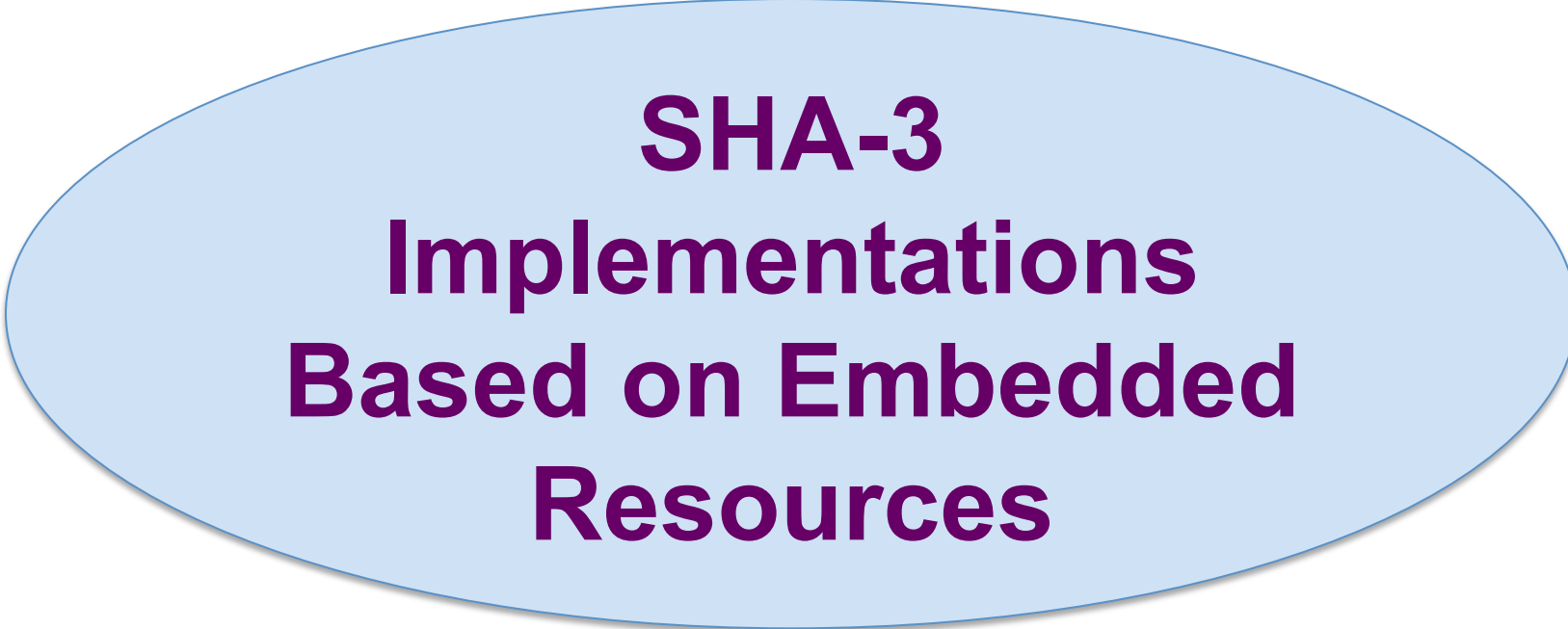
Target

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

Implementation Results

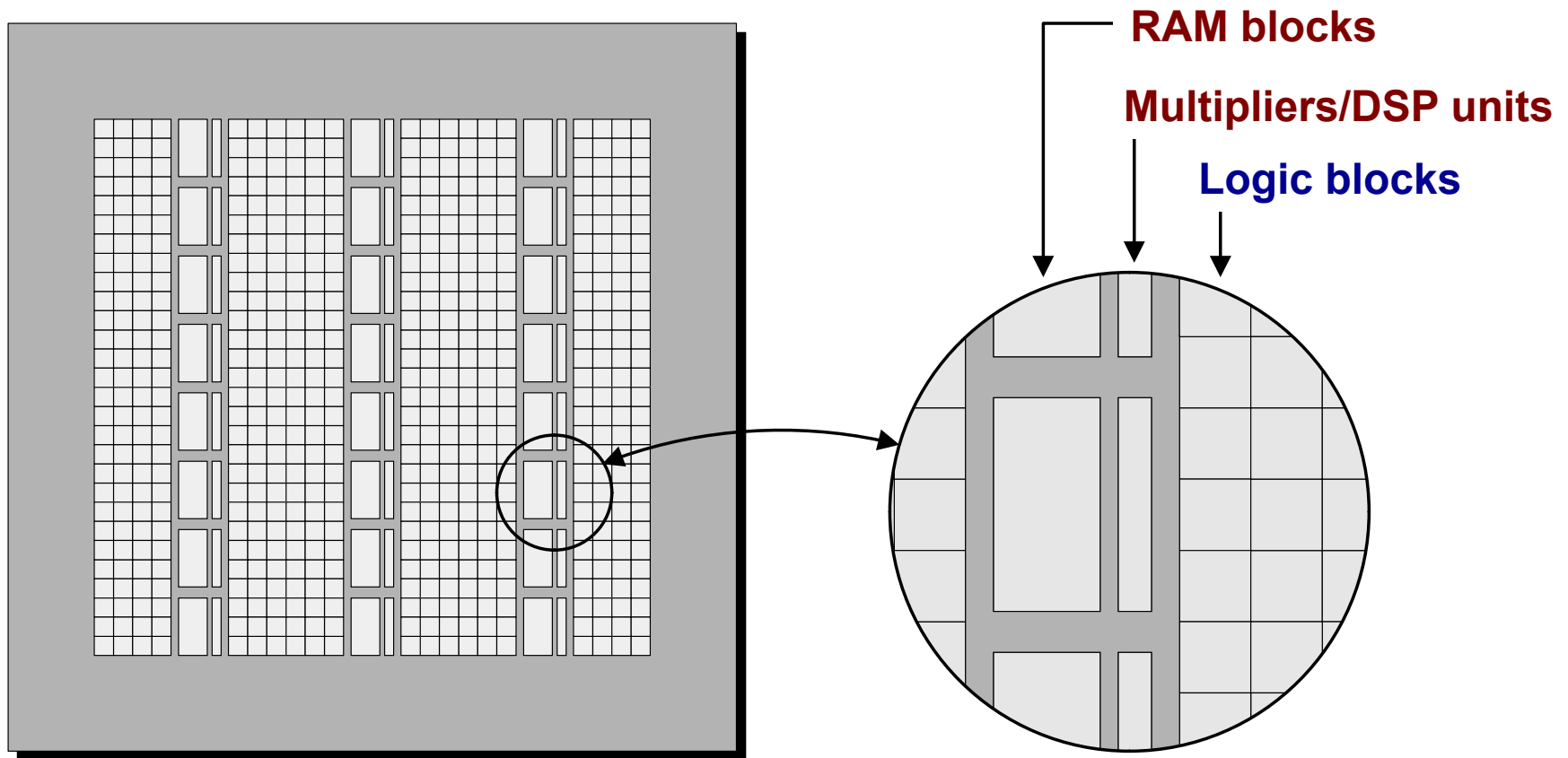


- Xilinx Spartan 3, ISE 12.3, after P&R, Optimized using ATHENa



**SHA-3
Implementations
Based on Embedded
Resources**

Implementations Based on the Use of Embedded Resources in FPGAs



(#Logic blocks, #Multipliers/DSP units, #RAM_blocks)

Resource Utilization Vector

(#Logic blocks, #Multipliers/DSP units, #RAM blocks)

Xilinx

Spartan 3: (#CLB_slices, #multipliers, #Block_RAMs)

Virtex 5: (#CLB_slices, #DSP units, #Block_RAMs)

Altera

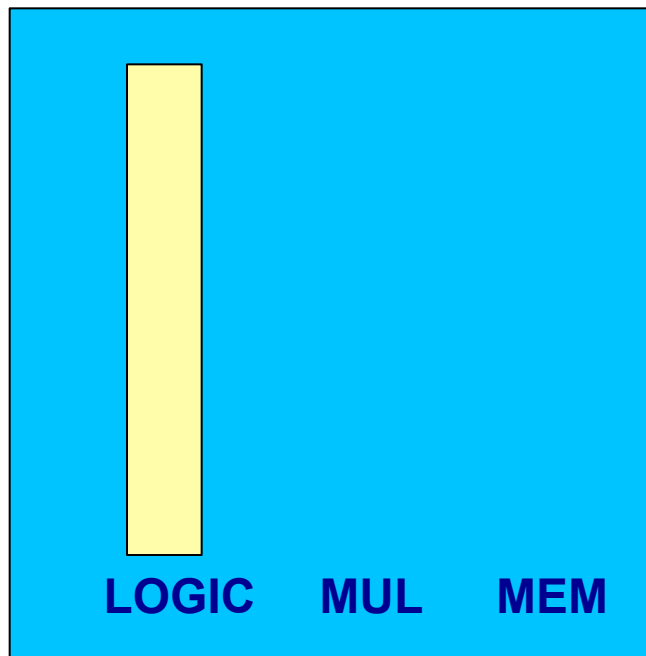
Cyclone III: (#LEs, #multipliers, #RAM_bits)

Stratix III: (#ALUTs, #DSP units, #RAM_bits)

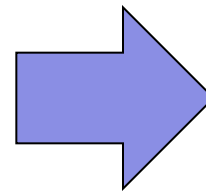
Fitting a Single Core in a Smaller FPGA Device

BLAKE in Altera Cyclone II

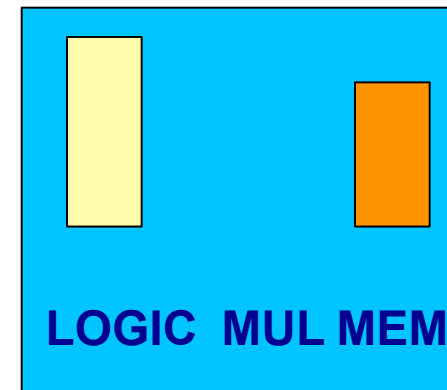
EP2C20



(6862, 0, 0)
LEs, MULs, bits



EP2C5

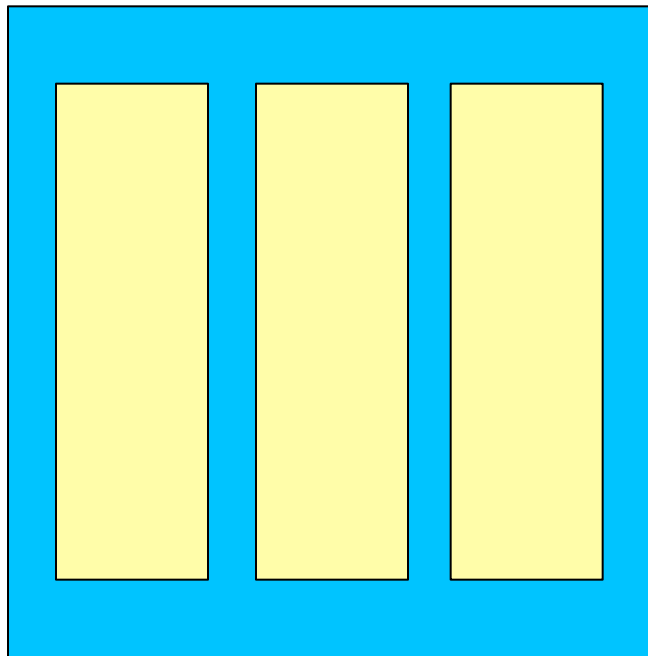


(3129, 0, 12k)
LEs, MULs, bits

Fitting a Larger Number of Identical Cores in the same FPGA Device

BLAKE in Virtex 5

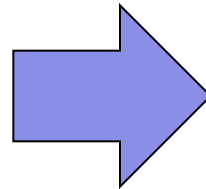
XC5VSX50



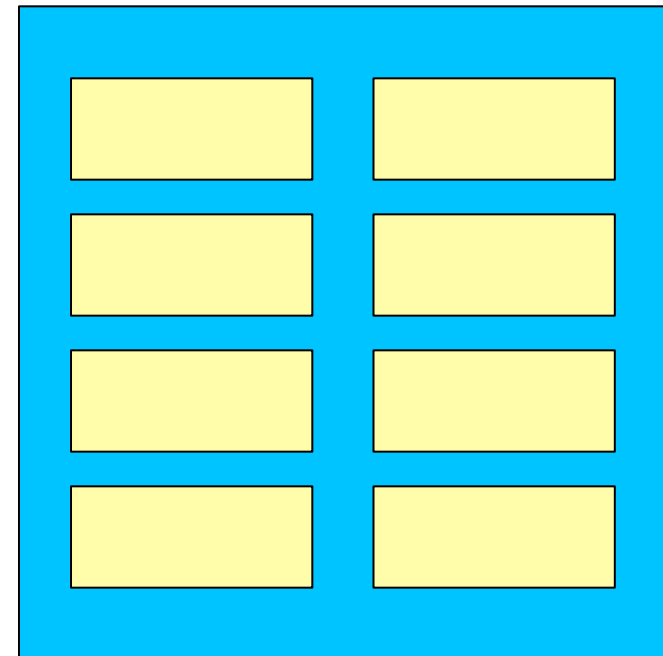
3 BLAKE cores

Cumulative
Throughput

6.8 Gbit/s



XC5VSX50

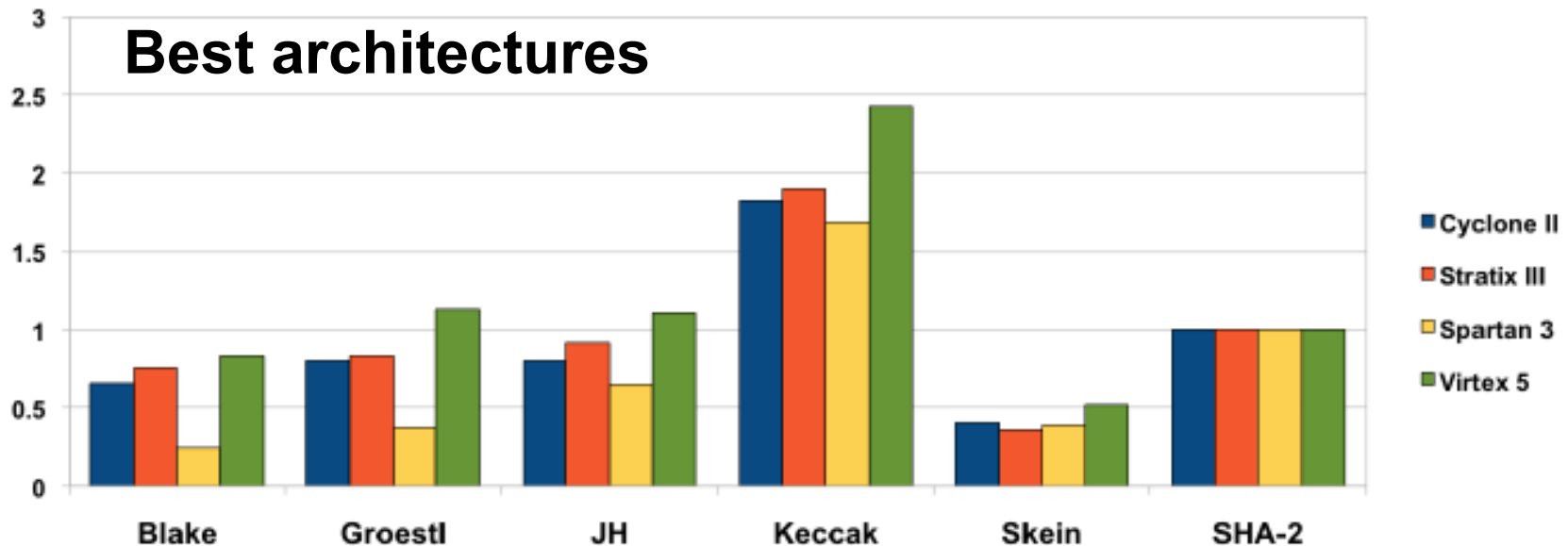
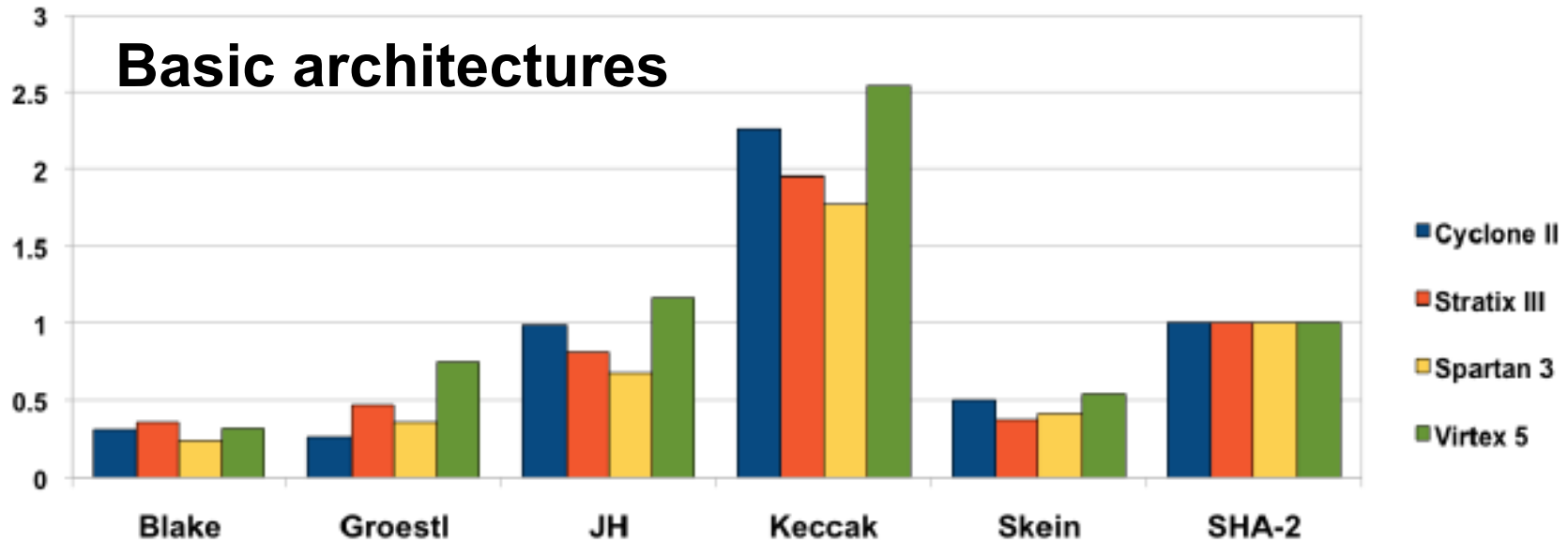


8 BLAKE cores

20.6 Gbit/s



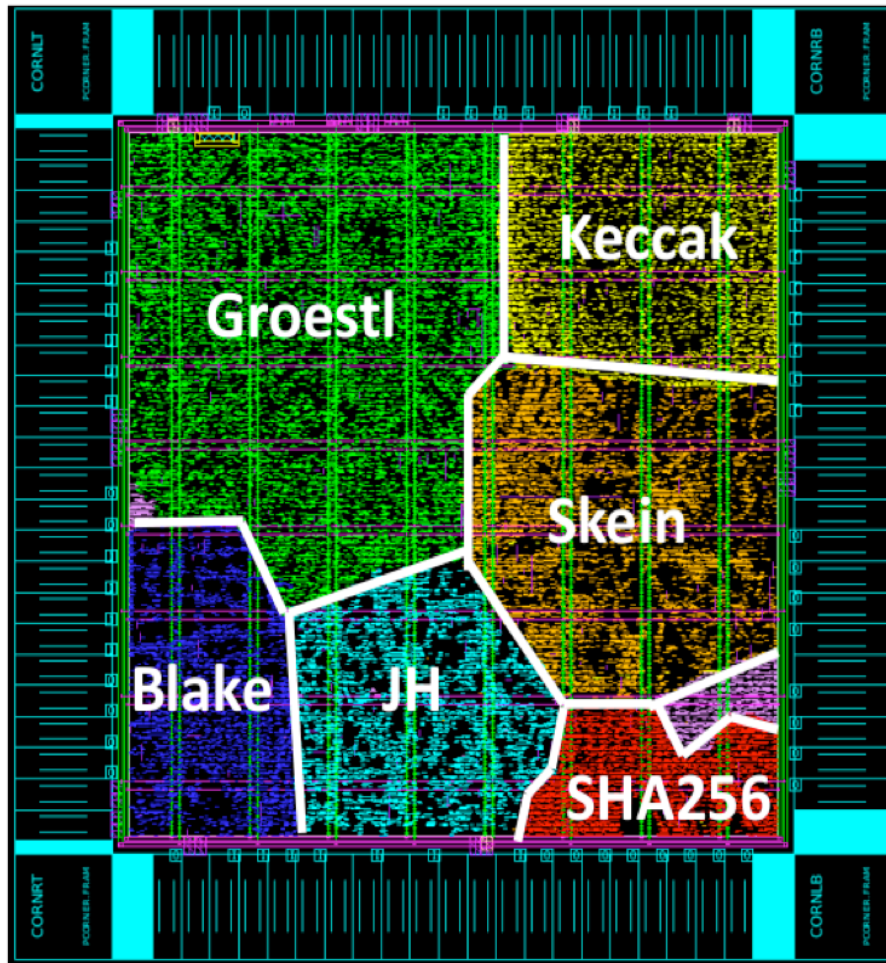
Cumulative Throughput for the Largest Device of a Given Family





**SHA-3
in ASICs**

Virginia Tech ASIC



- IBM MOSIS 130nm process
- The first ASIC implementing 5 final SHA-3 candidates
- Taped-out in Feb. 2011, successfully tested this Summer
- Multiple chips made available to other research labs

**Presentation at DSD in session
AHSA-1: Architectures and Hardware for Security Applications (1)
today, Thursday @ 10:30am**

FPGA Evaluations - Summary

	AES	eSTREAM	SHA-3
Multiple FPGA families	No	No	Yes
Multiple architectures	No	Yes	Yes
Use of embedded resources	No	No	Yes
Primary optimization target	Throughput	Area Throughput/ Area	Throughput/ Area
Experimental results	No	No	Yes
Availability of source codes	No	No	Yes
Specialized tools	No	No	Yes

ASIC Evaluations - Summary

	AES	eSTREAM	SHA-3
Multiple processes/ libraries	No	No	Yes
Multiple architectures	No	Yes	Yes
Primary optimization target	Throughput	Power x Area x Time	Throughput /Area
Post-layout results	No	Yes	Yes
Experimental results	No	Yes	Yes
Availability of source codes	No	No	Yes
Specialized tools	No	No	No



**Benchmarking
Tools**

Tools for Benchmarking Implementations of Cryptography

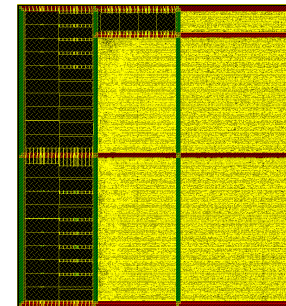
Software



FPGAs



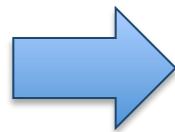
ASICs



eBACS

D. Bernstein (UIC)
T. Lange (TUE)

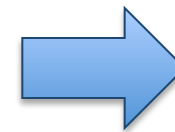
2006-present



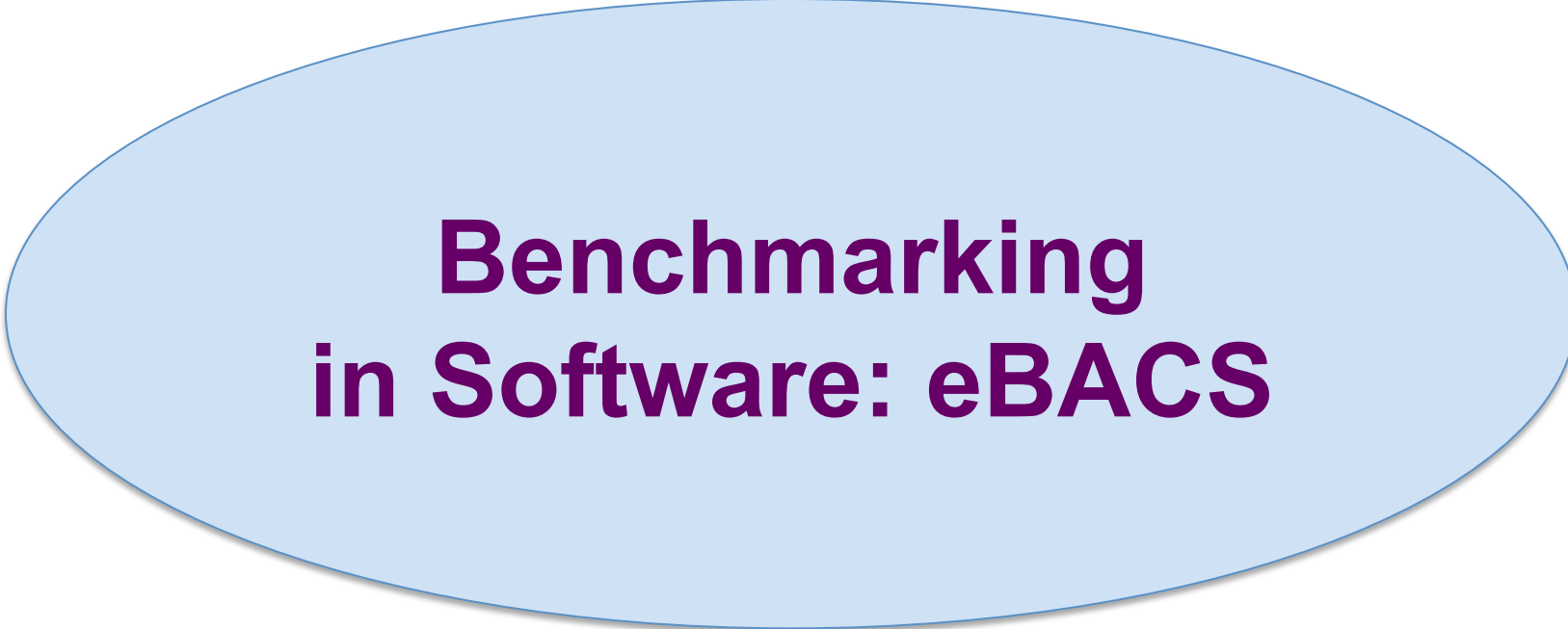
ATHENa

K. Gaj,
J. Kaps, et al.
(GMU)

2009-present



?



Benchmarking in Software: eBACS

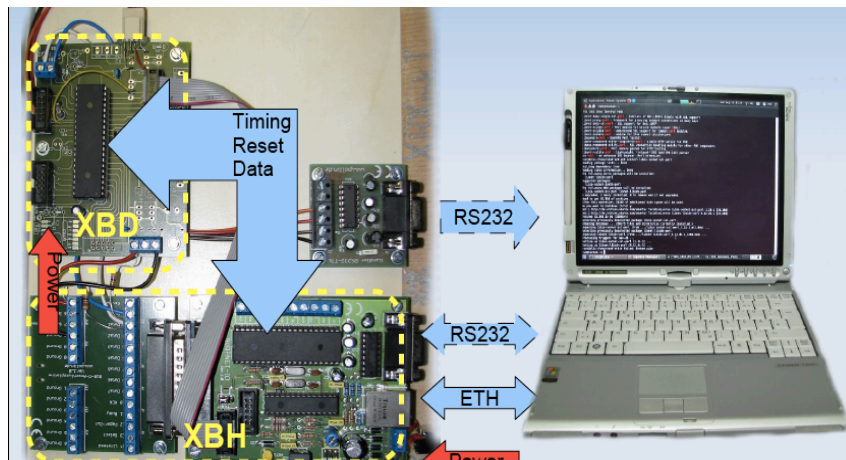
eBACS: ECRYPT Benchmarking of Cryptographic Systems:

<http://bench.cr.yp.to/>

SUPERCOP - toolkit developed by D. Bernstein and T. Lange for measuring performance of cryptographic software

- measurements on multiple machines (currently over 90)
- each implementation is recompiled multiple times (currently over 1600 times) with various compiler options
- time measured in clock cycles/byte for multiple input/output sizes
- median, lower quartile (25th percentile), and upper quartile (75th percentile) reported
- standardized function arguments (common API)

SUPERCOP Extension for Microcontrollers – XBX: 2009-present



Allows on-board timing measurements

Supports at least the following microcontrollers:

8-bit:

Atmel ATmega1284P (AVR)

32-bit:

TI AR7 (MIPS)

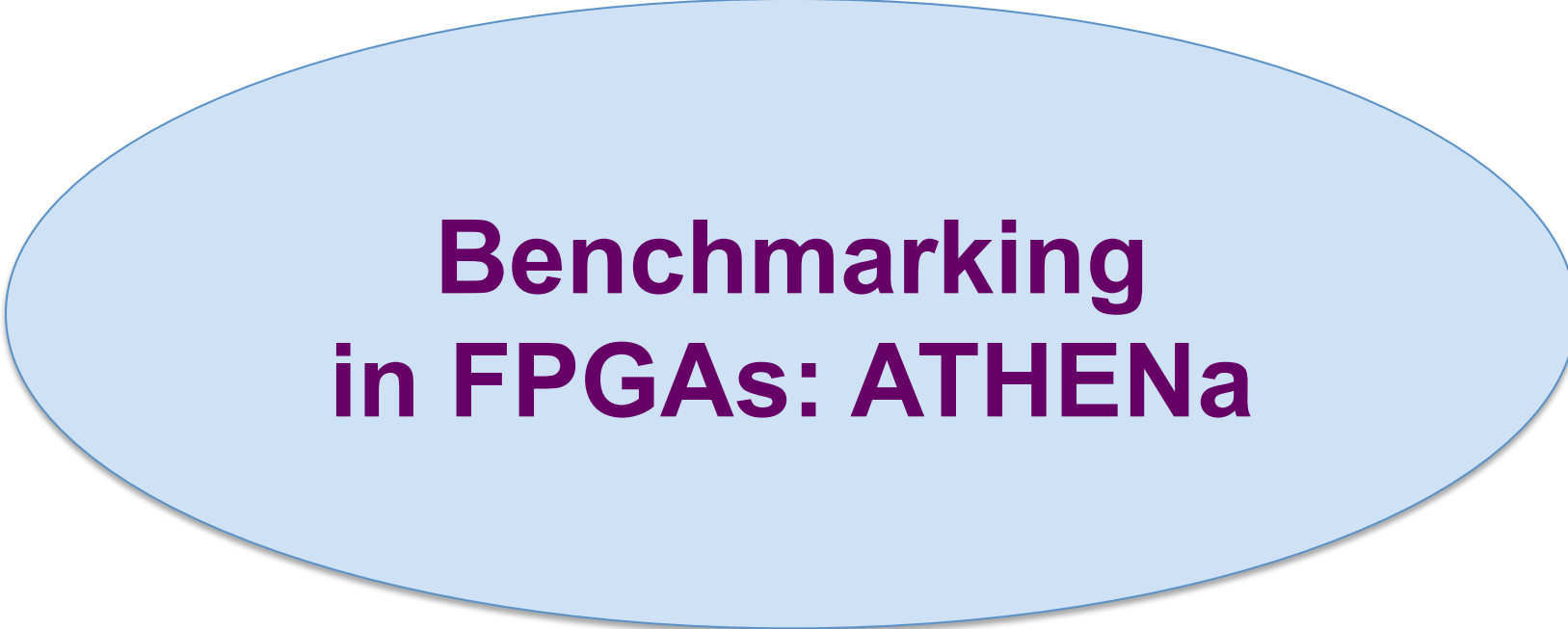
Atmel AT91RM9200 (ARM 920T)

Intel XScale IXP420 (ARM v5TE)

Cortex-M3 (ARM)

Developers:

- Christian Wenzel-Benner, ITK Engineering AG, Germany
- Jens Gräf, LiNetCo GmbH, Heiger, Germany



Benchmarking in FPGAs: ATHENa

ATHENa – Automated Tool for Hardware Evaluation

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



Open-source benchmarking environment, written in Perl, aimed at AUTOMATED generation of OPTIMIZED results for MULTIPLE hardware platforms.

The most recent version 0.6.2 released in June 2011. Full features in ATHENa 1.0 to be released in 2012.

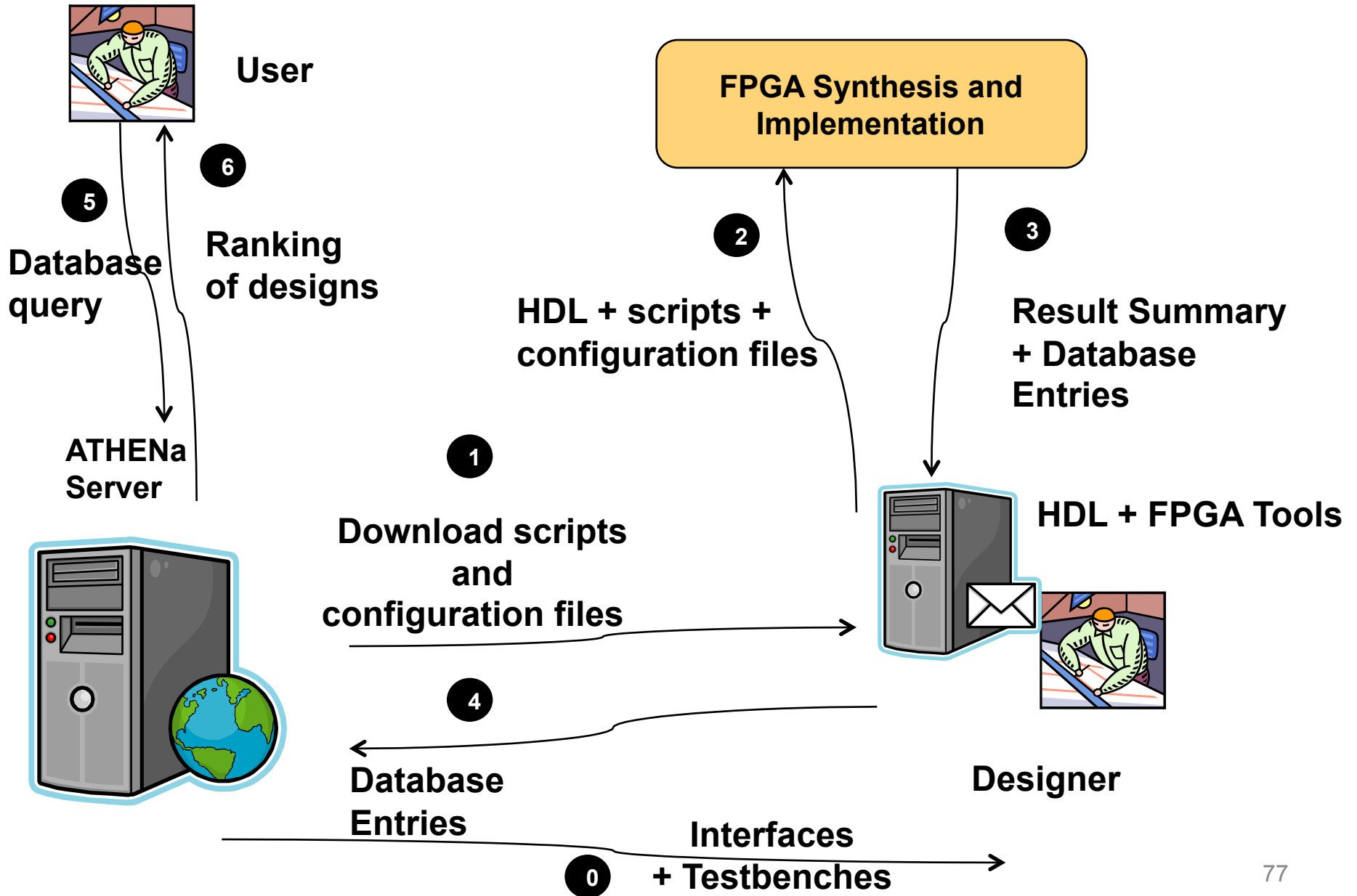
Why Athena?



"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"

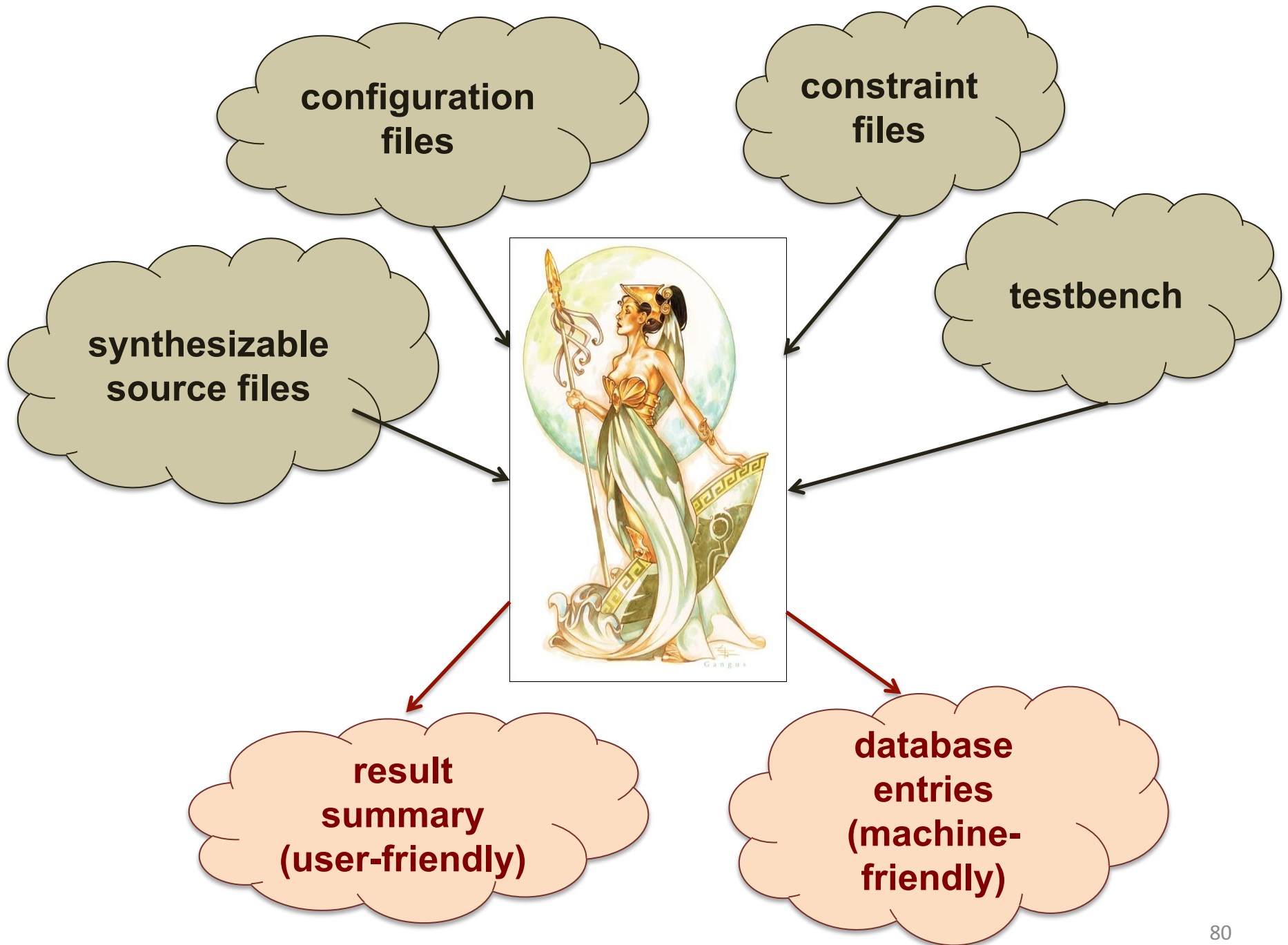
Basic Dataflow of ATHENa



Three Components of the ATHENa Environment

- **ATHENa Tool**
- **ATHENa Database of Results**
- **ATHENa Website**

ATHENa - Tool



ATHENa Major Features (1)

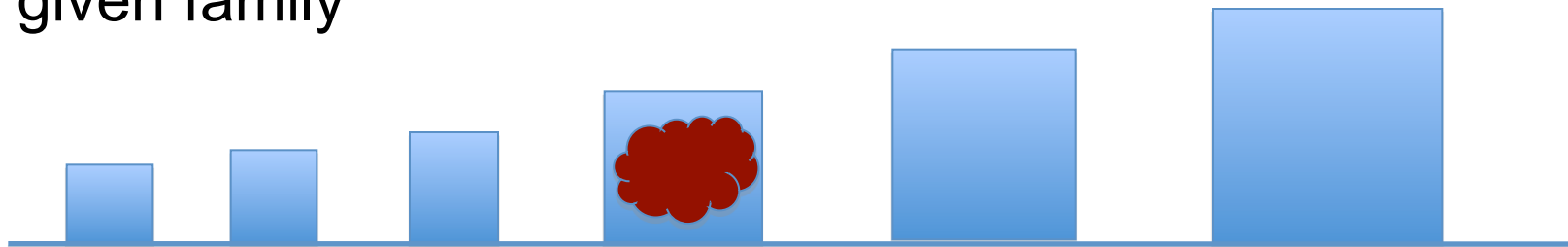
- synthesis, implementation, and timing analysis in **batch mode**
- support for devices and tools of **multiple FPGA vendors:**



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

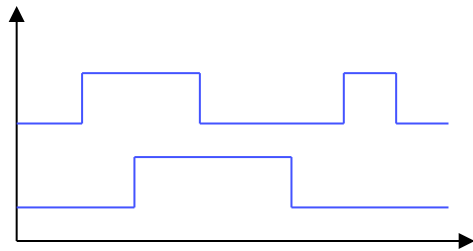


- automated choice of a **best-matching device** within a given family



ATHENa Major Features (2)

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

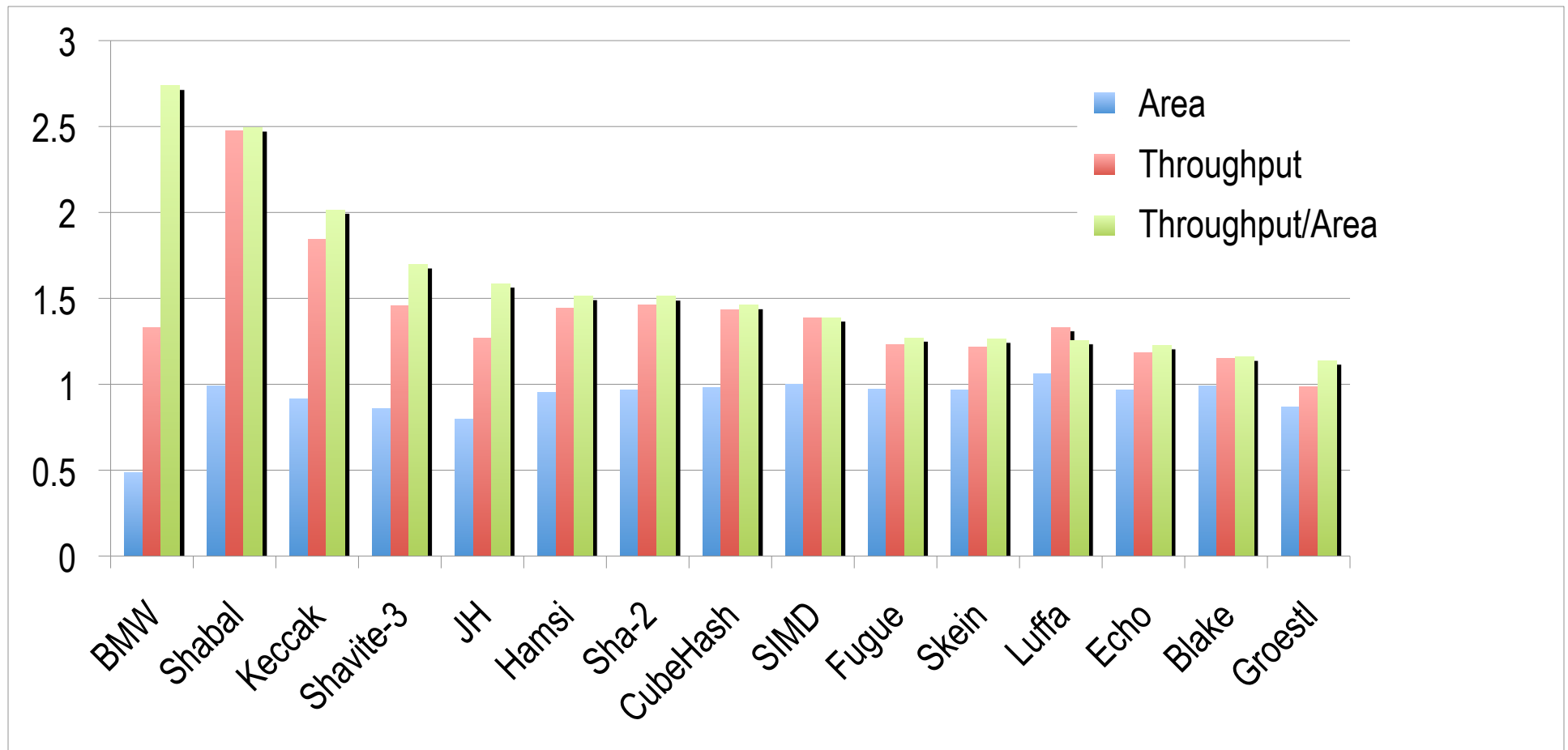


OR



- 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

Relative Improvement of Results from Using ATHENa Virtex 5, 512-bit Variants of Hash Functions



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

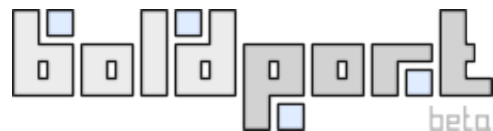
Other (Somewhat) Similar Tools



ExploreAhead (part of PlanAhead)



Design Space Explorer (DSE)



Boldport Flow



EDAx10 Cloud Platform

Distinguishing Features of ATHENa

- Support for **multiple tools** from **multiple vendors**
- Optimization strategies aimed at the **best possible performance** rather than design closure
- Extraction and **presentation of results**
- Seamless **integration with** the ATHENa **database of results**



**ATHENa – Database
of Results**

ATHENa Database – Result View

- Algorithm parameters
- Design parameters
 - Optimization target
 - Architecture type
 - Datapath width
 - I/O bus widths
 - Availability of source code
- Platform
 - Vendor, Family, Device
- Timing
 - Maximum clock frequency
 - Maximum throughput
- Resource utilization
 - Logic blocks (Slices/LEs/ALUTs)
 - Multipliers/DSP units
- Tools
 - Names & versions
 - Detailed options
- Credits
 - Designers & contact information

Details of Result ID 1125

Algorithm

Transformation Category: Cryptographic
Transformation: Hash
Group: SHA-3 Round 3
Algorithm: BLAKE
Hash Size [bits]: 512
Message Block Size [bits]: 1,024
Other Parameters: -
Specification: **Blake_FinalRnd.zip**
Formula for Message Size After Padding: -

Design

Design ID: **53**
Primary Optimization Target: Throughput/Area
Description Language: VHDL
Secondary Optimization Target: Throughput
Architecture Type: Folded
Datapath Width [bits]: 1,024
Padding: No
Minimum Message Unit: -
Input Bus Width [bits]: 64
Output Bus Width [bits]: 64
Implementation URL: **445.pdf**
Shared I/O Bus: No
Throughput Formula: $1024/(33*T)$
Execution Time Formula: -
Source Available: Yes
Source Code Files: **link**
Design Entry Date: 2011-05-02
Design Modify Date: 2011-08-02
Design Name: Blake_r3 (512)
Comments: -

Platform

Device Vendor: Altera
Family: Stratix II
Device: ep2s30f672c3

Timing

Requested Synthesis Clock Frequency [MHz]: -
Synthesis Clock Frequency [MHz]: -
Requested Implementation Clock Frequency [MHz]: -
Implementation Clock Frequency [MHz]: 71.910
Throughput [Mbits/s]: 2,231
Throughput/ALUTs [(Mbits/s)/ALUTs]: 0.311

Resource Utilization

ALUTs: 7,172
Flip Flops: 4,759
DSPs: 0
Memory Bits: 0

Tool Information

Synthesis Tool: Quartus II
Synthesis Tool Version: 10.1
Command Line Synthesis Tool Options: -

QSF Script Synthesis Tool Options: AUTO_DSP_RECOGNITION OFF
set_parameter -name h "512"

Implementation Tool: Quartus II

Implementation Tool Version: 10.1

Implementation Tool Options: --SEED=8001 --ONE_FIT_ATTEMPT=ON --EFFORT=STANDARD

Credits

Primary Designer Name(s): Ekawat Homsirikamol
Primary Designer Email(s): ehomsiri@gmu.edu
Co-designer Name(s): Marcin Rogawski, Kris Gaj

ATHENa Database – Compare Feature

Comparison of Result #s 1067 and 1056

Category	Result #1067	Result #1056
Algorithm		
Transformation Category:	Cryptographic	Cryptographic
Transformation:	Hash	Hash
Group:	SHA-3 Round 3	SHA-3 Round 3
Algorithm:	Skein	JH
Hash Size [bits]:	512	512
Message Block Size [bits]:	512	512
Other Parameters:		
Specification:	Skein_FinalRnd.zip	JH_FinalRnd.zip
Formula for Message Size After Padding:		
Design		
Design ID:	57	55
Primary Optimization Target:	Throughput/Area	Throughput/Area
Description Language:	VHDL	VHDL
Secondary Optimization Target:	Throughput	Throughput
Architecture Type:	Unrolled	Basic Iterative
Datapath Width [bits]:	512	1024
Padding:	No	No
Minimum Message Unit:		
Input Bus Width [bits]:	64	64
Output Bus Width [bits]:	64	64
Implementation URL:	445.pdf	445.pdf
Shared I/O Bus:	No	No
Throughput Formula:	512/(19*T)	512/(43*T)
Execution Time Formula:		
Source Available:	Yes	Yes
Source Code Files:	link	link
Design Entry Date:	2011-05-02 @ 01:04 EST	2011-05-02 @ 00:59 EST
Design Modify Date:	2011-08-02 @ 22:25 EST	2011-08-19 @ 17:39 EST
Design Name:	Skein_r3 (512)	JH_r3 (512)
Comments:		
Platform		
Device Vendor:	Xilinx	Xilinx
Family:	Virtex 6	Virtex 6
Device:	xc6vtx75tff784-3	xc6vtx75tff784-3
Timing		
Requested Synthesis Clock Frequency [MHz]:	147.7	431.2
Synthesis Clock Frequency [MHz]:	165.003	417.606
Requested Implementation Clock Frequency [MHz]:	147.7	431.2
Implementation Clock Frequency [MHz]:	127.016	411.015
Throughput [Mbits/s]:	3423	4894
Throughput/CLBs [(Mbits/s)/CLBs]:	2.997	5.114
Resource Utilization		
CLB Slices:	1142	957
LUTs:	4475	3581
Flip Flops:	3121	2902
DSPs:	0	0
BRAMs:	0	0
Tool Information		
Synthesis Tool:	Xilinx XST	Xilinx XST
Synthesis Tool Version:	12.3	12.3
Synthesis Tool Options:	-generics { h=512 adder_type=0 } -dsp_utilization_ratio 0 -opt_level 1 -bram_utilization_ratio 0 -opt_mode area	-generics { h=512 } -dsp_utilization_ratio 0 -opt_level 1 -bram_utilization_ratio 0 -opt_mode area -ram_style distributed
Implementation Tool:	Xilinx ISE	Xilinx ISE
Implementation Tool Version:	12.3	12.3
Map Options:		
Implementation Tool Options:	-ol high	-ol high
Credits		
Primary Designer Name(s):	Ekawat Homsirikamol	Ekawat Homsirikamol
Primary Designer Email(s):	ehomsiri@gmu.edu	ehomsiri@gmu.edu
Co-designer Name(s):	Marcin Rogawski, Kris GaJ	Marcin Rogawski, Kris GaJ
Co-designer Email(s):	mrogawsk@gmu.edu, kgaj@gmu.edu	mrogawsk@gmu.edu, kgaj@gmu.edu
Primary Designer Affiliation:	CERG @ GMU	CERG @ GMU
Co-Designer Affiliation:	CERG @ GMU	CERG @ GMU
Other		
Result Replication Files:	link	link
Result Entry Date:	2011-05-02 @ 15:15 EST	2011-05-02 @ 15:14 EST

Matching fields in grey
Non-matching fields in red and blue

Currently in the Database

Hash Functions in FPGAs

GMU Results for

- **20 hash functions**
(14 Round 2 SHA-3 + 5 Round 3 SHA-3 + SHA-2)
x **2 variants** (256-bit output & 512-bit output)
x **11 FPGA families** = **440 combinations**
-

(440-not_fitting) = 423 optimized results

Coming soon!

- **GMU results for Hash Functions in FPGAs**
 - Folded & unrolled architectures
 - Pipelined architectures
 - Lightweight architectures
 - Architectures based on embedded resources
- **Other Groups' results for Hash Functions in FPGAs**
- **Other Groups' results for Hash Functions in ASICs**
- **Modular Arithmetic (basis of public key cryptography)
in FPGAs & ASICs**

Possible Future Customizations

The same basic database can be customized and adapted for other domains, such as

- Digital Signal Processing
- Bioinformatics
- Communications
- Scientific Computing, etc.

ATHENa - Website

ATHENa Website

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

- **Download** of ATHENa Tool
- Links to **related tools**

SHA-3 Competition in FPGAs & ASICs

- **Specifications** of candidates
- **Interface** proposals
- RTL **source codes**
- **Testbenches**
- ATHENa database of **results**
- Related **papers & presentations**

GMU Source Codes

- best non-pipelined high-speed architectures for **14 Round 2 SHA-3 candidates** and **SHA-2**
- best non-pipelined high-speed architectures for **5 Round 3 SHA-3 candidates**
- Each code supports **two variants:**
with **256-bit** and **512-bit** output

ATHENa Result Replication Files

- **Scripts and configuration files sufficient to easily reproduce all results (without repeating optimizations)**
 - **Automatically created by ATHENa for all results generated using ATHENa**
 - **Stored in the ATHENa Database**
-

In the same spirit of **Reproducible Research** as:

- J. Claerbout (Stanford University)
“Electronic documents give reproducible research a new meaning,”
in *Proc. 62nd Ann. Int. Meeting of the Soc. of Exploration Geophysics*, **1992**,
<http://sepwww.stanford.edu/doku.php?id=sep:research:reproducible:seg92>
.....
- Patrick Vandewalle¹, Jelena Kovacevic², and Martin Vetterli¹ (¹EPFL, ²CMU)
Reproducible research in signal processing - what, why, and how.
IEEE Signal Processing Magazine, May **2009**. <http://rr.epfl.ch/17/>

Benchmarking Goals Facilitated by ATHENa

Comparing multiple:

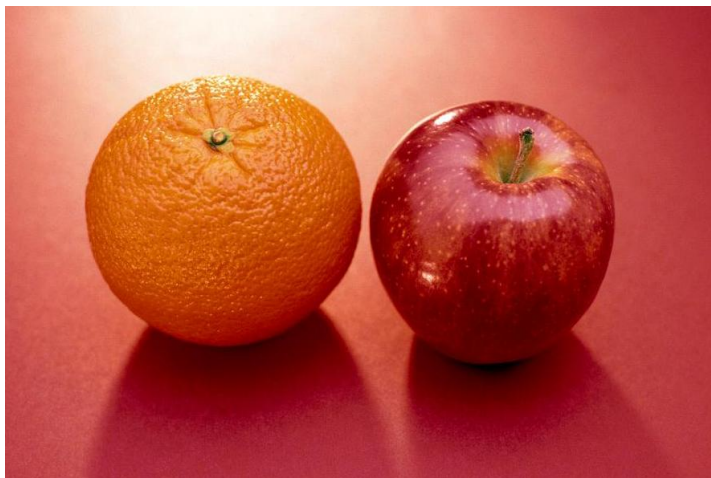
1. cryptographic **algorithms**
2. hardware **architectures or implementations** of the same cryptographic algorithm
3. hardware **platforms** from the point of view of their suitability for the implementation of a given algorithm, (e.g., choice of an FPGA device or FPGA board)
4. **tools and languages** in terms of quality of results they generate (e.g. Verilog vs. VHDL, Synplicity Synplify Premier vs. Xilinx XST, ISE v. 13.1 vs. ISE v. 12.3)



**Open
Problems**

Objective Benchmarking Difficulties

- lack of standard one-fits-all interfaces
- stand-alone performance vs. performance as a part of a bigger system
- heuristic optimization strategies
- time & effort spent on optimization



or



Objective Benchmarking Difficulties

- lack of convenient cost metric in FPGAs
- accuracy of power estimators in ASICs & FPGAs
- verifiability of results
- human factor (skills of designers, order of implementations, etc.)

How to measure hardware cost in FPGAs?

1. Stand-alone cryptographic core on an FPGA



Cost of the smallest FPGA that can fit the core?

Unit: USD [FPGA vendors would need to publish MSRP (manufacturer's suggested retail price) of their chips]

– not very likely, very volatile metric

or **size of the chip in mm²** - easy to obtain

2. Part of an FPGA System On-Chip

Resource utilization described by a vector:

(#CLB slices, #MULs/DSP units, #BRAMs) for Xilinx

(#LEs/ALUTs, #MULs/DSP units, #membits) for Altera

Difficulty of turning vector into a single number representing cost

Potential Problems with Publishing Source Codes

- **Export control** regulations for cryptography

Check: Bert-Jaap Koops, Crypto Law Survey

<http://rechten.uvt.nl/koops/cryptolaw/>

- **Commercial interests**

- **Competition with other groups** for

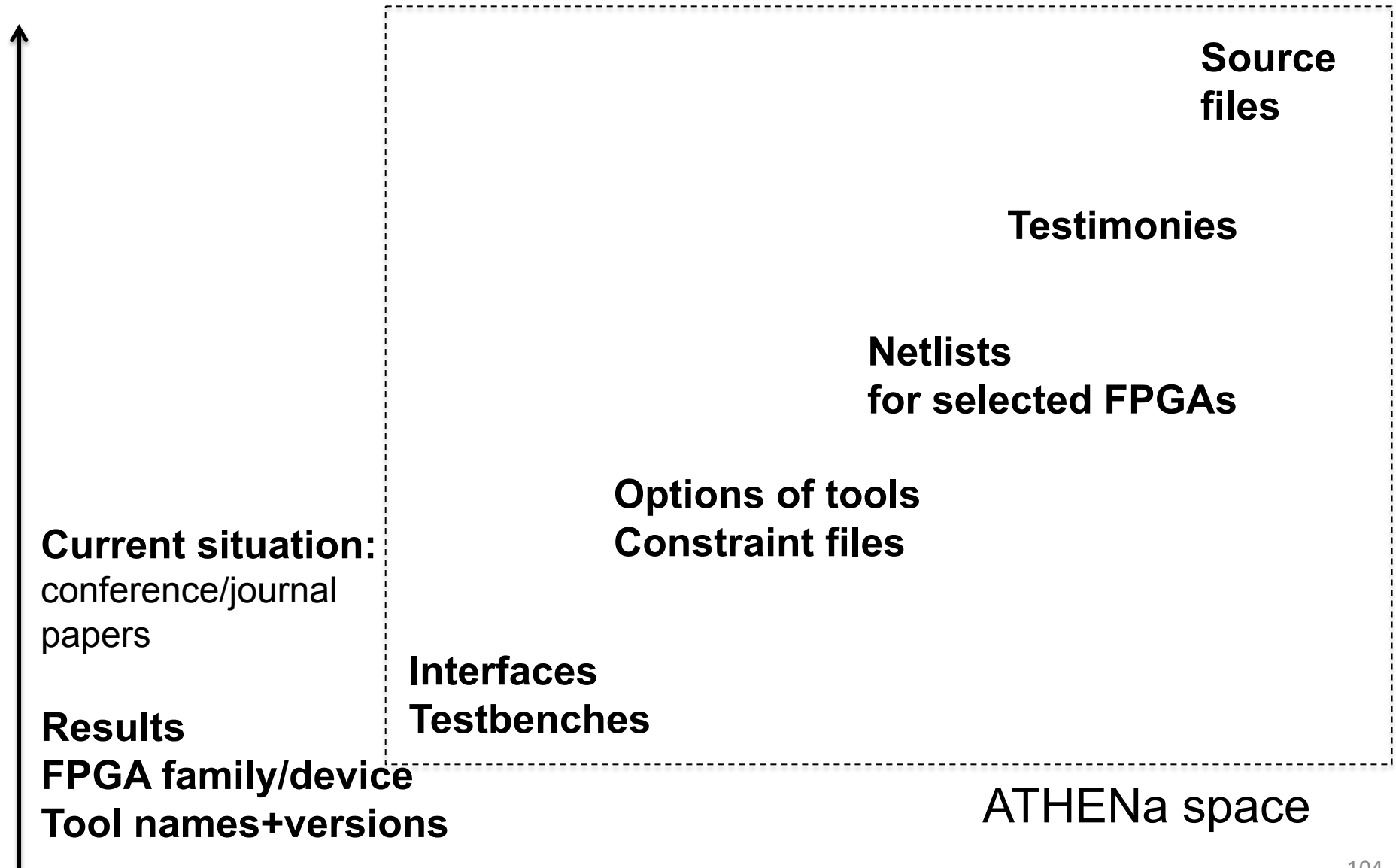
grants and publications in the most renowned journals and conference proceedings

Selected SHA-3 Source Codes Available in Public Domain

- **AIST-RCIS:** <http://www.rcis.aist.go.jp/special/SASEBO/SHA3-en.html>
- **University College Cork, Queens University Belfast, RMIT University, Melbourne, Australia:**
<http://www.ucc.ie/en/crypto/SHA-3Hardware>
- **Virginia Tech:** <http://rijndael.ece.vt.edu/sha3/soucecodes.html>
- **ETH Zurich:** <http://www.iis.ee.ethz.ch/~sha3/>
- **George Mason University:** <http://cryptography.gmu.edu/athena>
- **BLAKE Team:** <http://www.131002.net/blake/>
- **Keccak Team:** <http://keccak.noekeon.org/>

How to assure verifiability of results?

Level of openness



Initial Evaluation by High-Level Synthesis Tools?

	Initial number of candidates
AES	15
	↓
eSTREAM	34
	↓
SHA-3	51
	↓
Next Contest	???

- All hardware implementations so far developed using RTL HDL
- Growing number of candidates in subsequent contests
- Each submission includes reference implementation in C
- Results from High-Level Synthesis could have a large impact in early stages of the competitions
- Results and RTL codes from previous contests form interesting benchmarks for High-Level synthesis tools

Turning Thousands of Results into a Single Fair Ranking

- **Choosing** which **FPGA families / ASIC libraries** should be included in the comparison
 - wide range?
 - only most recent?
 - vendors with the largest market share?
 - wide spectrum of vendors?
- Methods for **combining multiple results** into single ranking

Thousands of results
on tens of platforms



- 1.
- 2.
- 3.
- 4.
- 5.

Turning Thousands of Results into Fair Ranking

- Deciding on most important **application scenarios**
 - Throughput – Cost – Power range
from RFIDs to High-speed security gateways
 - Assigning weights to different scenarios

**Help/recommendation from the system developers
highly appreciated**

Conclusions

- **Contests for cryptographic standards are important**
 - **Stimulate progress in design and analysis of cryptographic algorithms**
 - **Determine future of cryptography for the next decades**
 - **Promote cryptology: Are easy to understand by general audience**
 - **Provide immediate recognition and visibility worldwide.**
- **Digital System Designers and Software Engineers can play an important role in these contests**
 - **Co-designers of new cryptographic algorithms**
 - **Evaluators**
 - **Tool developers**
 - **Early adopters of new standards**
- **Get involved! It is fun!**

Conferences & Journals



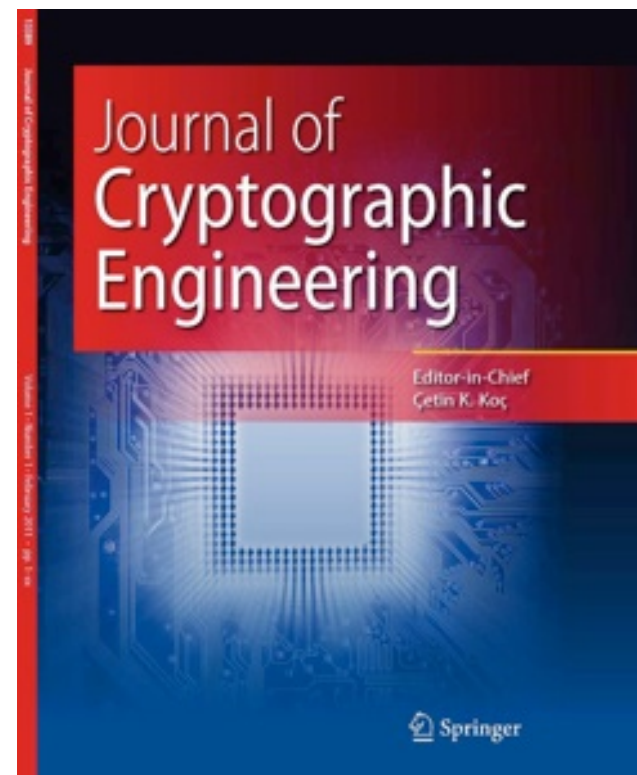
ECRYPT II
↓↑↔↻⊕⊖^ ↓

**Conferences
& workshops
devoted
to specific contests**

**Next: The 3rd SHA-3
Candidate Conference
Washington, D.C.,
March 22-23, 2012**



**Since 1999
USA-Europe-Asia
CHES 2011, Nara, Japan
Sep. 28-Oct. 1, 2011**



Since Jan. 2011

Thank you!

Questions?



Questions?

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