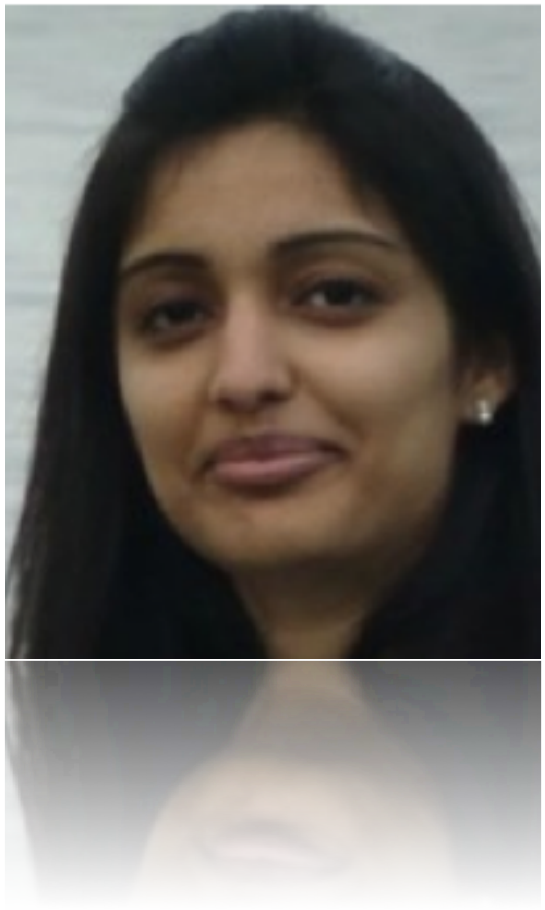
Use of Embedded FPGA Resources in Implementations of 14 Round 2 SHA-3 Candidates

Kris Gaj,
Rabia Shahid,
Malik Umar Sharif, and
Marcin Rogawski
George Mason University
U.S.A.

Co-Authors

Rabia Shahid



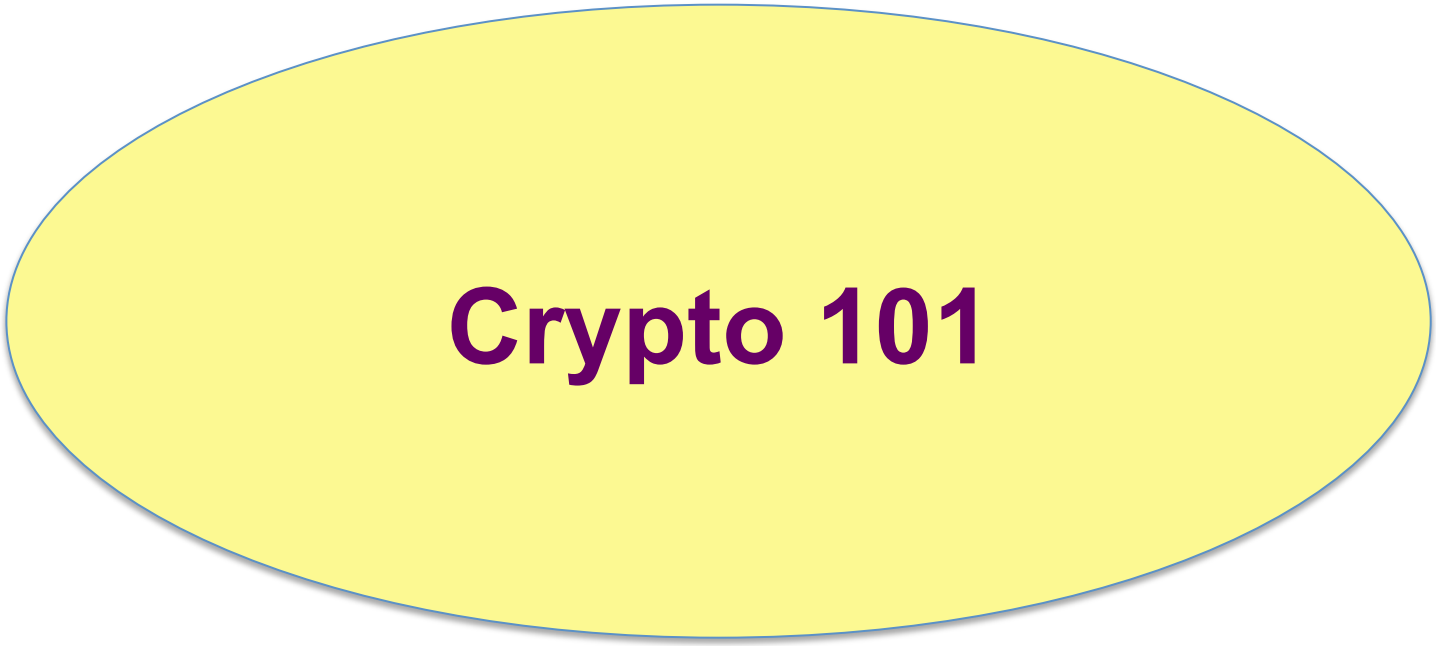
Malik Umar Sharif

Marcin Rogawski

PhD Students
in the Cryptographic Engineering Research Group at GMU

Outline

- Introduction: Crypto 101
- Background & Previous Work
- Initial Analysis & Methodology
- Results
- Conclusions


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

e.g. DES, AES, RC4

encryption

Hash Functions

e.g. SHA-1, SHA-2

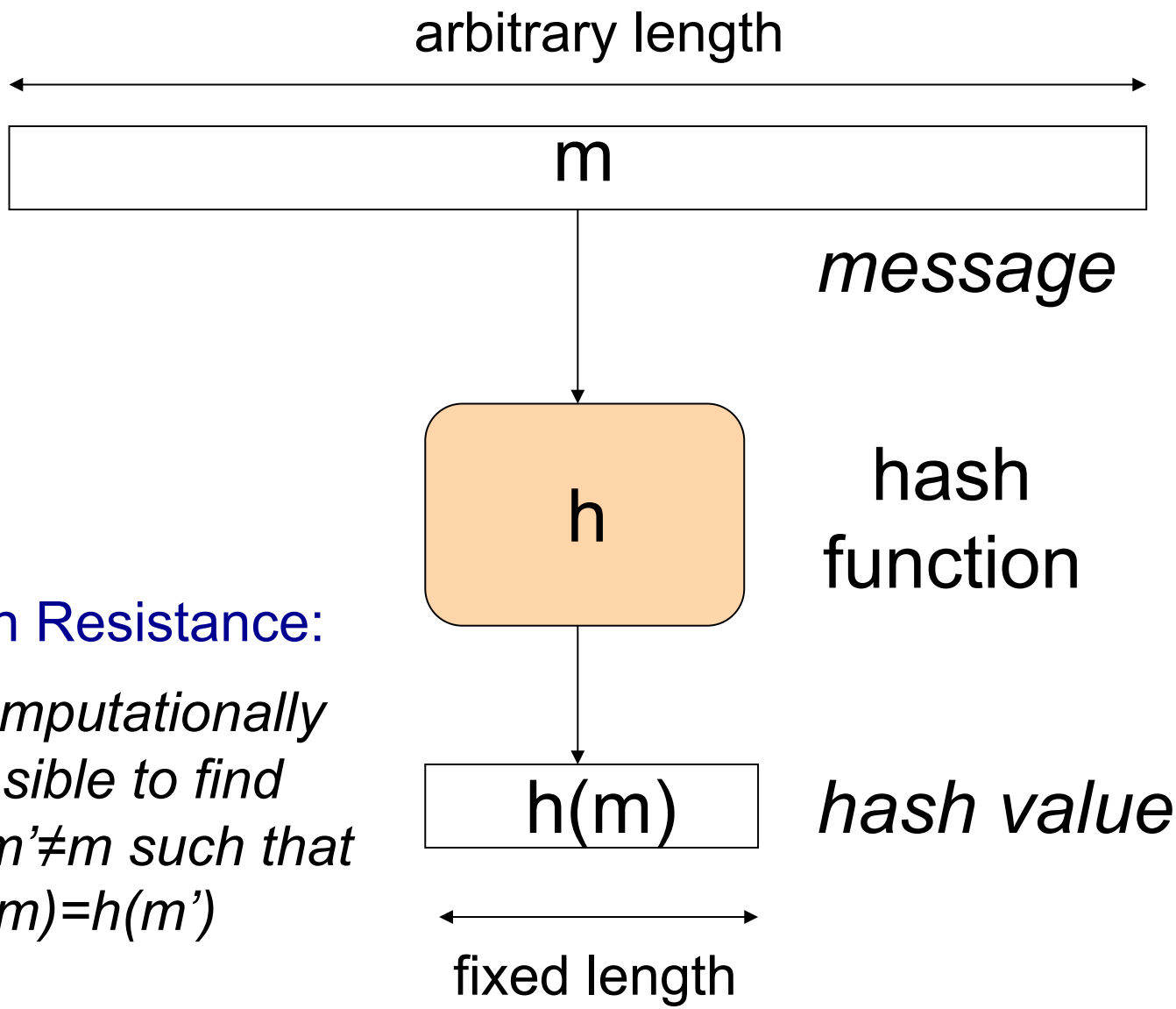
message & user authentication

Public-Key Cryptosystems

e.g. RSA, ECC

digital signatures
key agreement
key exchange

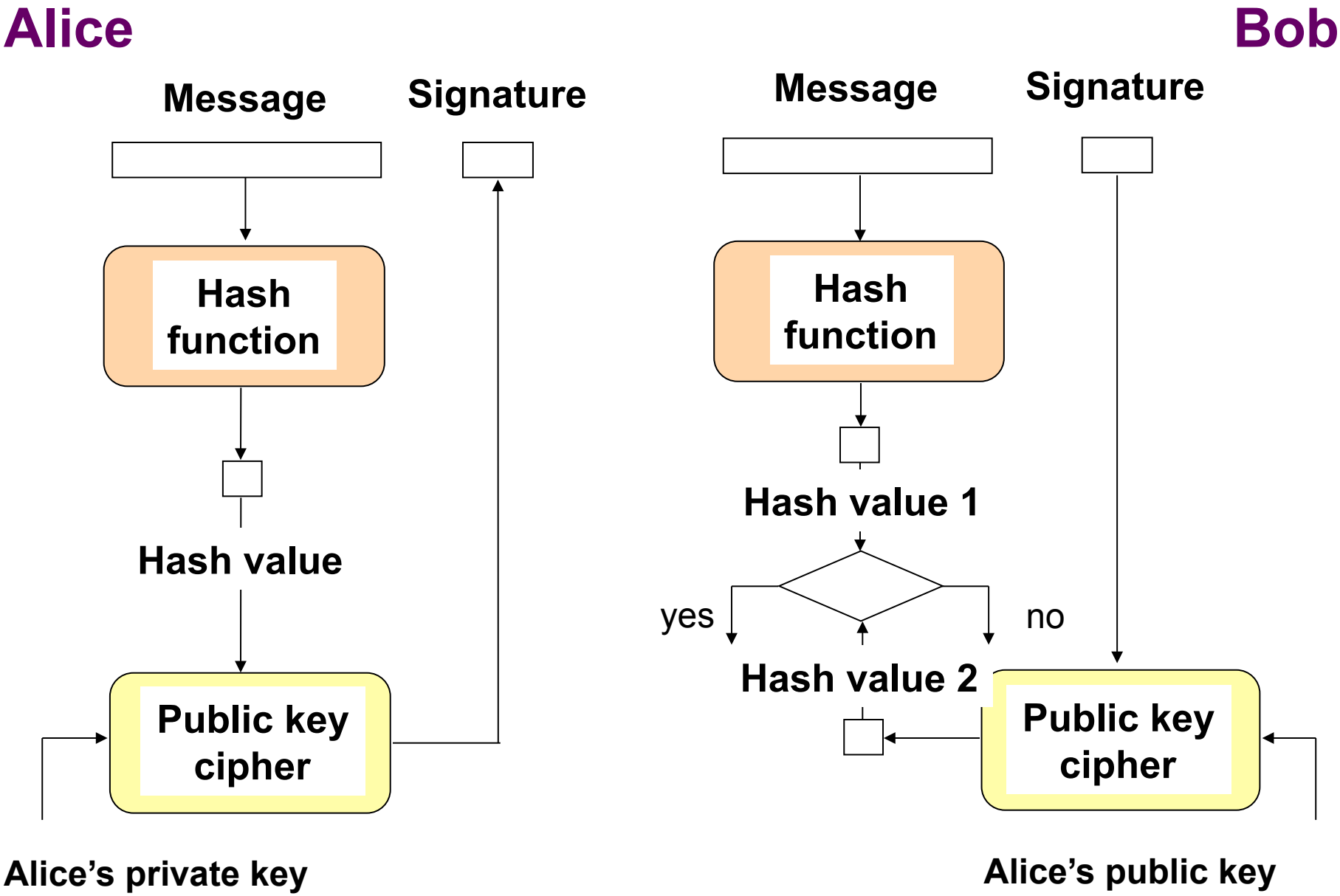
Hash Function



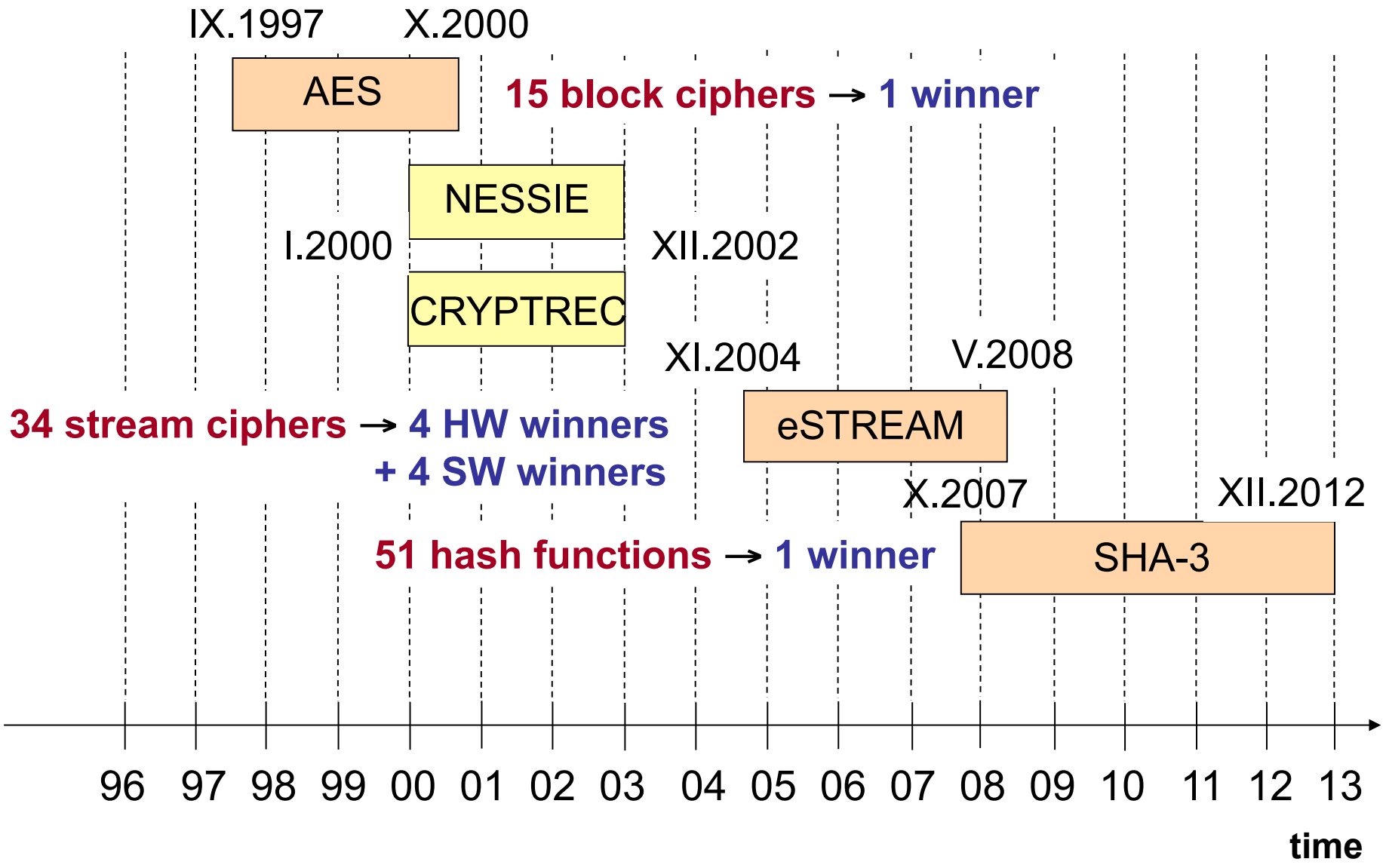
Collision Resistance:

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

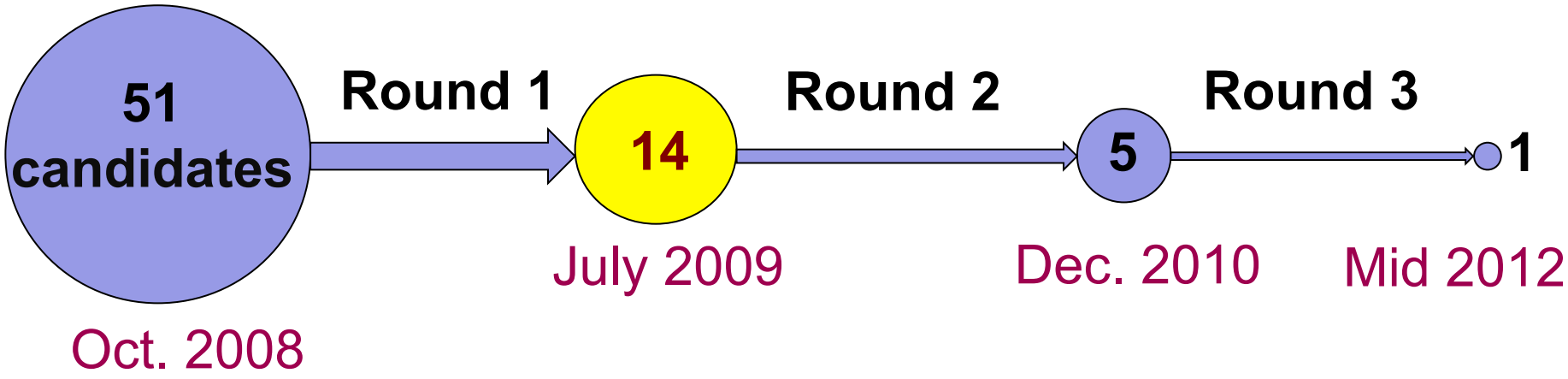
Hash Functions in Digital Signature Schemes



Cryptographic Standard Contests



NIST SHA-3 Contest - Timeline

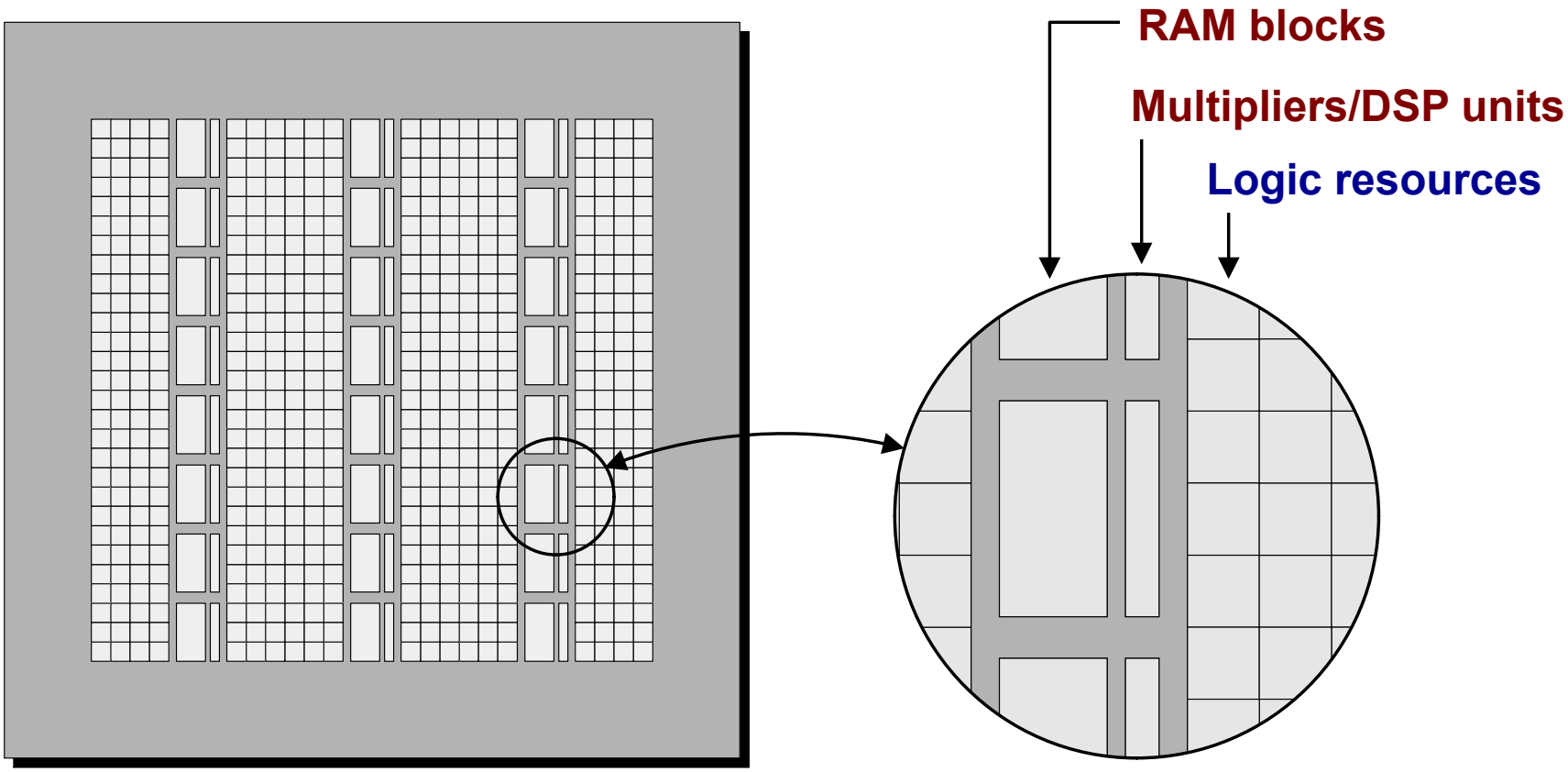


14 Round 2 Candidates & the Current Standard SHA-2 selected for our study:

- large variety of cryptographic design approaches
- respectable security: none broken so far
- likely to remain in limited use even if not standardized

Background
& Previous Work

Implementations Based on the Use of Embedded Resources in FPGAs



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

Resource Utilization Vector

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

Xilinx

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

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

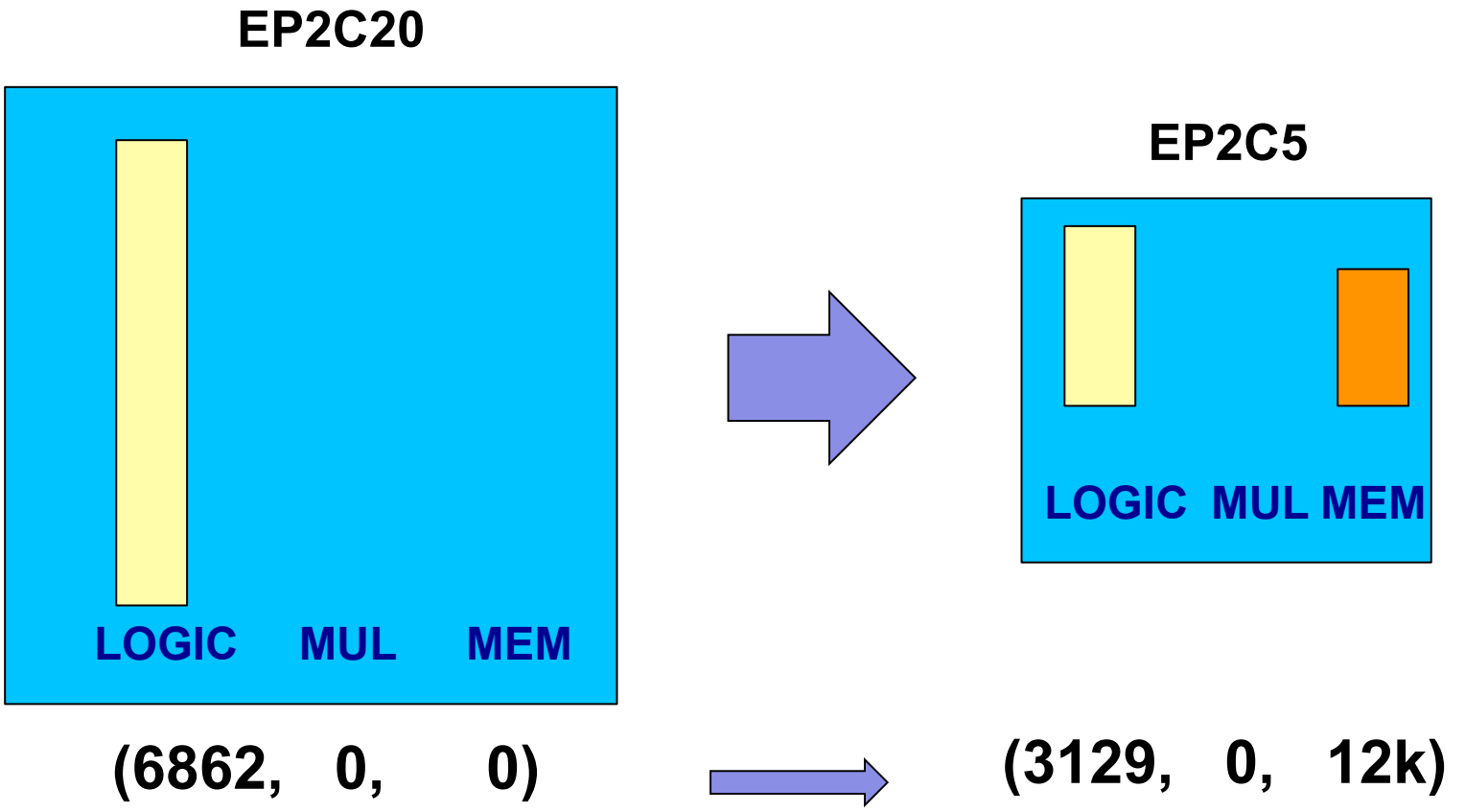
Altera

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

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

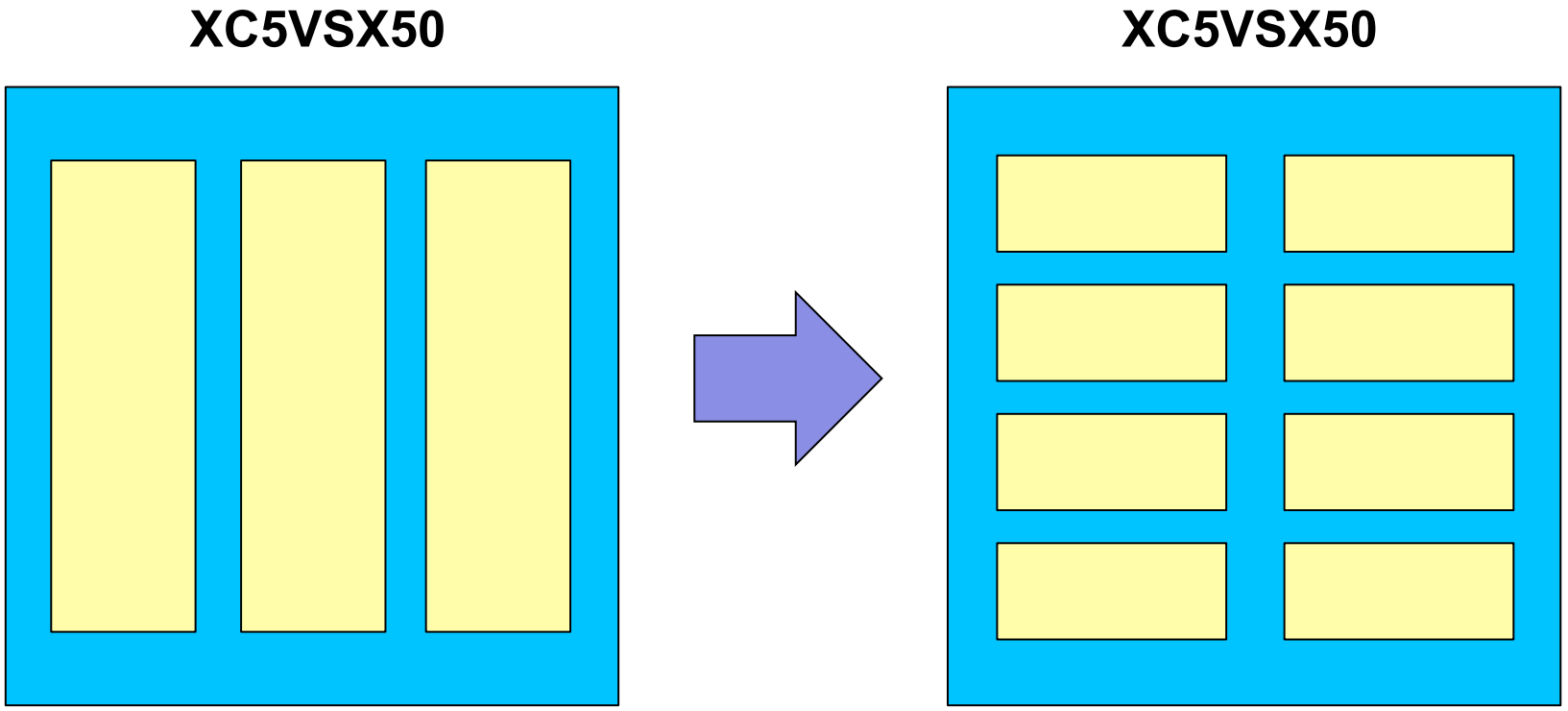
Fitting a Single Core in a Smaller FPGA Device

BLAKE in Altera Cyclone II



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

BLAKE in Virtex 5



Cumulative Throughput 6.8 Gbit/s → 20.6 Gbit/s

Most Related Previous Work

- **Secret Key Ciphers - Advanced Encryption Standard**
  - Drimer et al., PhD Thesis 2009, ACM TRETS 2010
  - Güneysu et al., JCEN 2011

  FPGA implementations using **DSP units** and **Block RAMs** of Xilinx Virtex 5

- **Public Key Ciphers – RSA, Elliptic Curve Cryptosystems**
  - Suzuki et al., CHES 2007, IEICE TF 2011
  - Güneysu et al., JCEN 2011

  FPGA implementations using **DSP units** of Xilinx Virtex 4 & Virtex 5

Initial Analysis & Methodology

Basic Operations of 13 SHA-3 Candidates & SHA-2

Function	MUL	mADD	ADD	Tables of Constants
BLAKE		mADD3	ADD	Message Expansion Table
BMW		mADD17	ADD, SUB	
CubeHash			ADD	
ECHO				AES S-box or T-box
Fugue				AES S-box or T-box
Groestl				AES S-box or T-box
Hamsi				Message Expansion Table
JH				Round Constants
Keccak				Round Constants
SHAvite-3				AES S-box or T-box
SIMD	x185, x233	mADD3	ADD	Twiddle Factors
Shabal	x3, x5		ADD, SUB	
Skein			ADD-64	
SHA-2		mADD6	ADD, SUB	Round Constants

7 Functions Implemented Using DSP Units

Function	MUL	mADD	ADD	Tables of Constants
BLAKE		mADD3	ADD	Message Expansion Table
BMW		mADD17	ADD, SUB	
CubeHash			ADD	
ECHO				AES S-box or T-box
Fugue				AES S-box or T-box
Groestl				AES S-box or T-box
Hamsi				Message Expansion Table
JH				Round Constants
Keccak				Round Constants
SHAvite-3				AES S-box or T-box
SIMD	x185, x233	mADD3	ADD	Twiddle Factors
Shabal	x3, x5		ADD, SUB	
Skein			ADD-64	
SHA-2		mADD6	ADD, SUB	Round Constants

10 Functions Implemented Using Block Memories

Function	MUL	mADD	ADD	Tables of Constants
BLAKE		mADD3	ADD	Message Expansion Table
BMW		mADD17	ADD, SUB	
CubeHash			ADD	
ECHO				AES S-box or T-box
Fugue				AES S-box or T-box
Groestl				AES S-box or T-box
Hamsi				Message Expansion Table
JH				Round Constants
Keccak				Round Constants
SHAvite-3				AES S-box or T-box
SIMD	x185, x233	mADD3	ADD	Twiddle Factors
Shabal	x3, x5		ADD, SUB	
Skein			ADD-64	
SHA-2		mADD6	ADD, SUB	Round Constants

3 Functions Implemented Using DSP Units & BRAMs

Function	MUL	mADD	ADD	Tables of Constants
BLAKE		mADD3	ADD	Message Expansion Table
BMW		mADD17	ADD, SUB	
CubeHash			ADD	
ECHO				AES S-box or T-box
Fugue				AES S-box or T-box
Groestl				AES S-box or T-box
Hamsi				Message Expansion Table
JH				Round Constants
Keccak				Round Constants
SHAvite-3				AES S-box or T-box
SIMD	x185, x233	mADD3	ADD	Twiddle Factors
Shabal	x3, x5		ADD, SUB	
Skein			ADD-64	
SHA-2		mADD6	ADD, SUB	Round Constants

Functions Benefiting Most From the Respective Embedded Resources

DSP Units (5)	DSP Units & Block Memories (1)	Block Memories (8)
• BMW • CubeHash • Shabal • SIMD • Skein	• SHA-2	• BLAKE • ECHO • Fugue • Groestl • Hamsi • JH • Keccak • SHAvite-3

Research Questions

Can the Use of Embedded Resources

1. Increase
   Throughput?

2. Significantly Reduce
   Use of Logic Resources (#CLB slices, #ALUTs)?

3. Increase
   Throughput/#Logic Resources Ratio?

FPGAs Used

- High Speed FPGA Families
  - Xilinx Virtex 5
  - Altera Stratix III
- Low Cost FPGA Families
  - Xilinx Spartan 3
  - Altera Cyclone II

Results

Can the Use of Embedded Resources

1. **Increase Throughput**?

No **- 7 functions**

Marginally (less than 10%) **- 5 functions**

Yes (more than 10%) - 2 functions

Fugue: Spartan 3, Cyclone II
SHAvite-3: Spartan 3, Cyclone II, Stratix III

Reasons for No Increase in Throughput

- **interconnect delays between reconfigurable logic and embedded resources** greater than those within reconfigurable logic;
  the only exception for low-cost families based on 4-input LUTs

- **addition/subtraction faster using logic resources** (fast carry chains) rather than DSP units;
  drop in frequency higher for Altera DSP units

Can the Use of Embedded Resources

2. Significantly Reduce
Use of Logic Resources (#CLB slices, #ALUTs)?

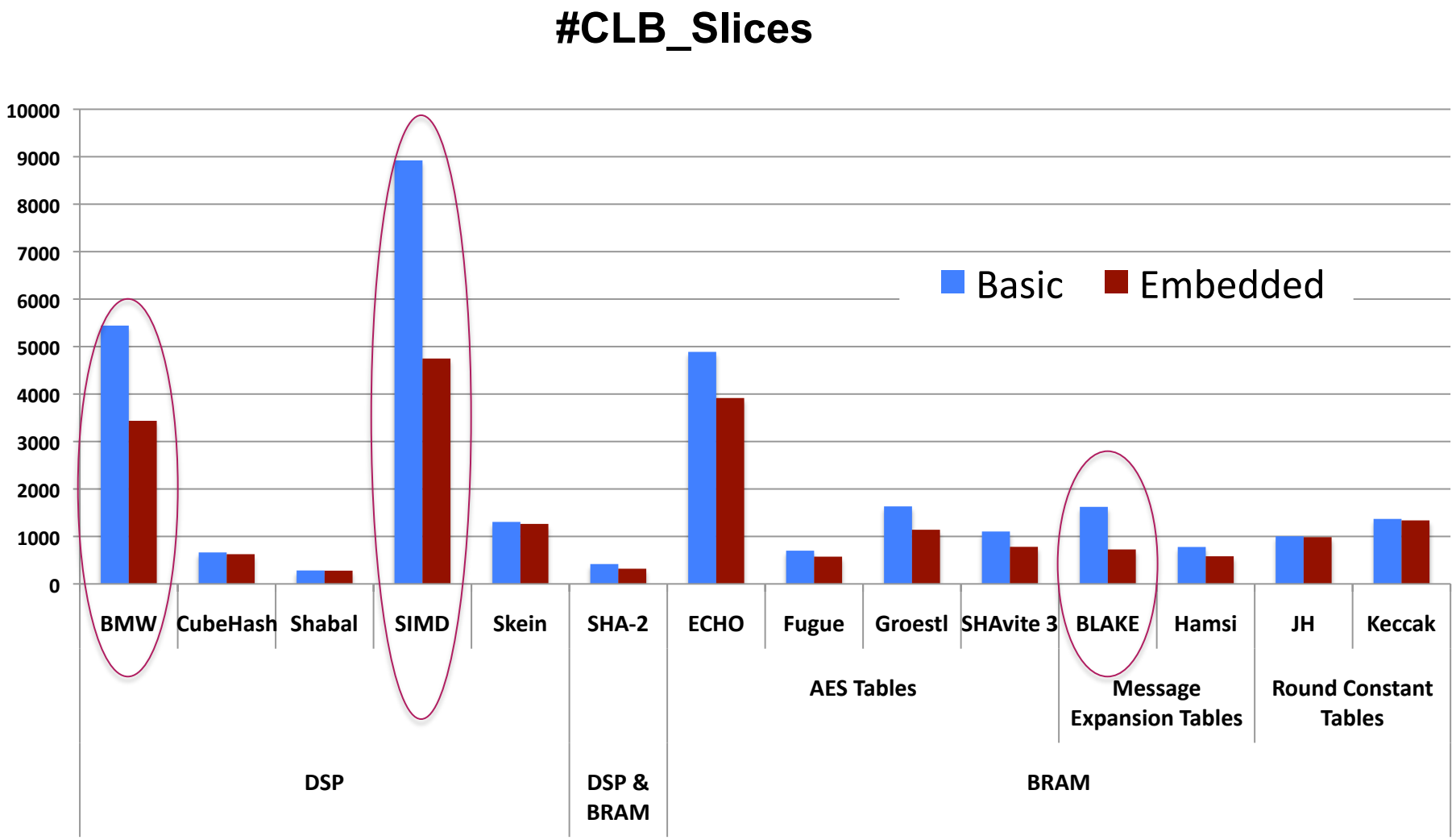
No (Less than 20%) - 5 functions

Partially (20-40%) **- 5 functions**

Yes (>40%) **- 4 functions**

BLAKE	- **all families**
Groestl, SHAvite-3	- all except Virtex 5
SIMD	- Virtex 5

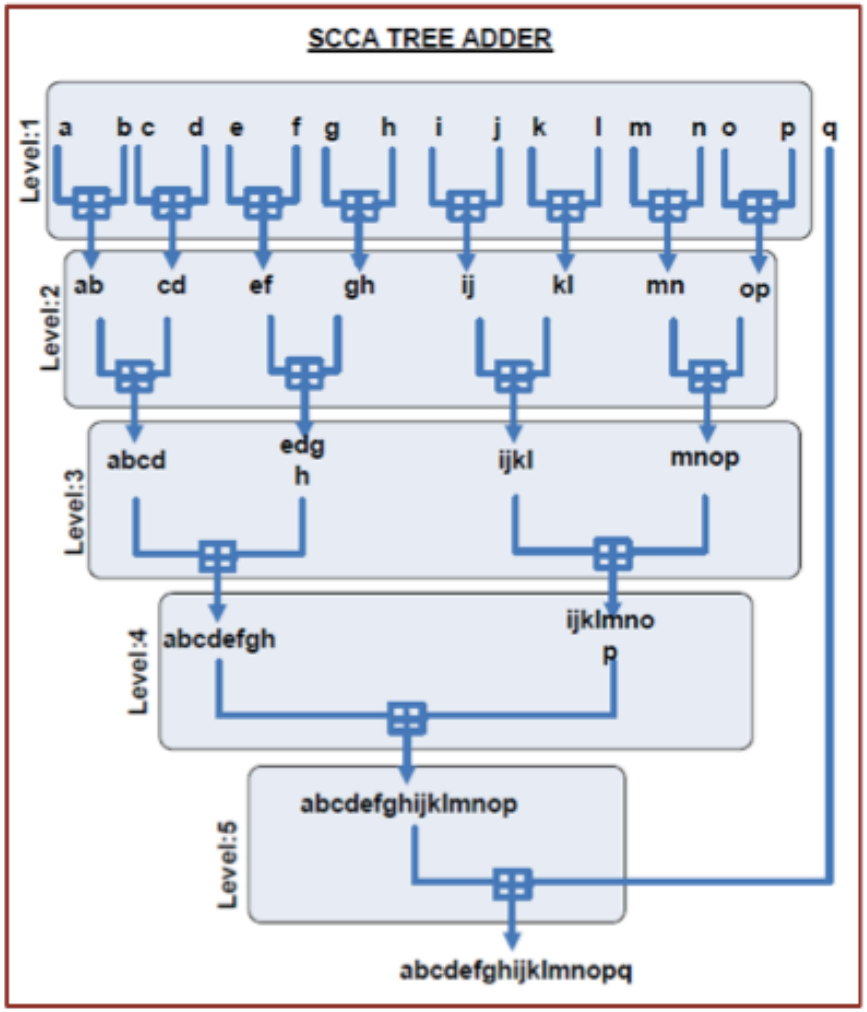
Xilinx Virtex 5 Results

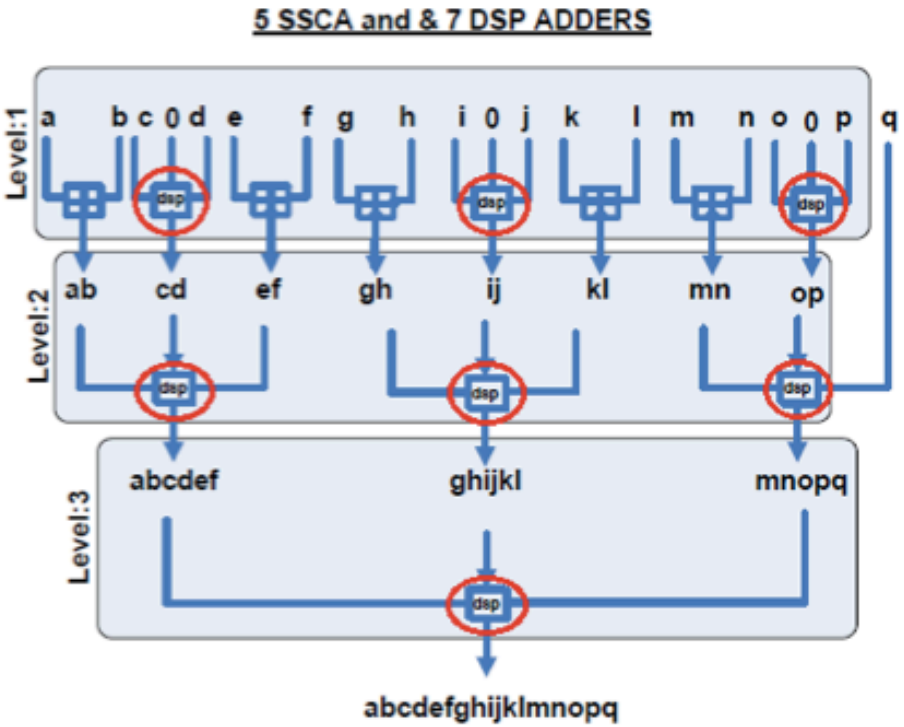


Reasons for Particularly Large Reduction

- BLAKE: complex initial transformation based on **look-up tables matching the size of BRAM**
- SIMD: use of **DSP units for multiplication**
- BMW: use of **DSP units in cascaded mode for** multi-operand **addition**

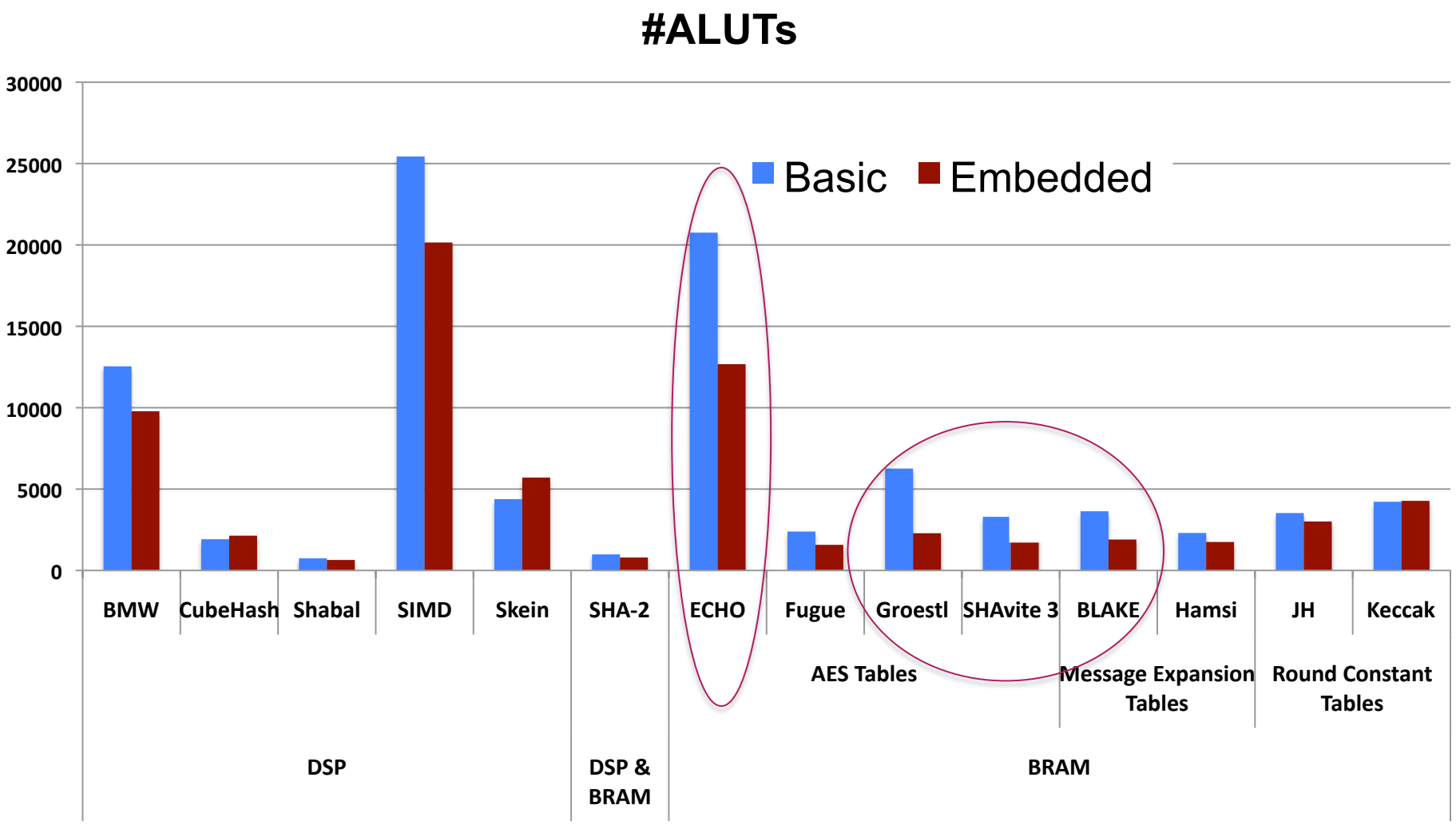
Multi-operand Addition in BMW





Dsp – DSP based adder
+ - Simple Carry Chain Adder
(inferred by "+" in VHDL)

Altera Stratix III Results



Can the Use of Embedded Resources

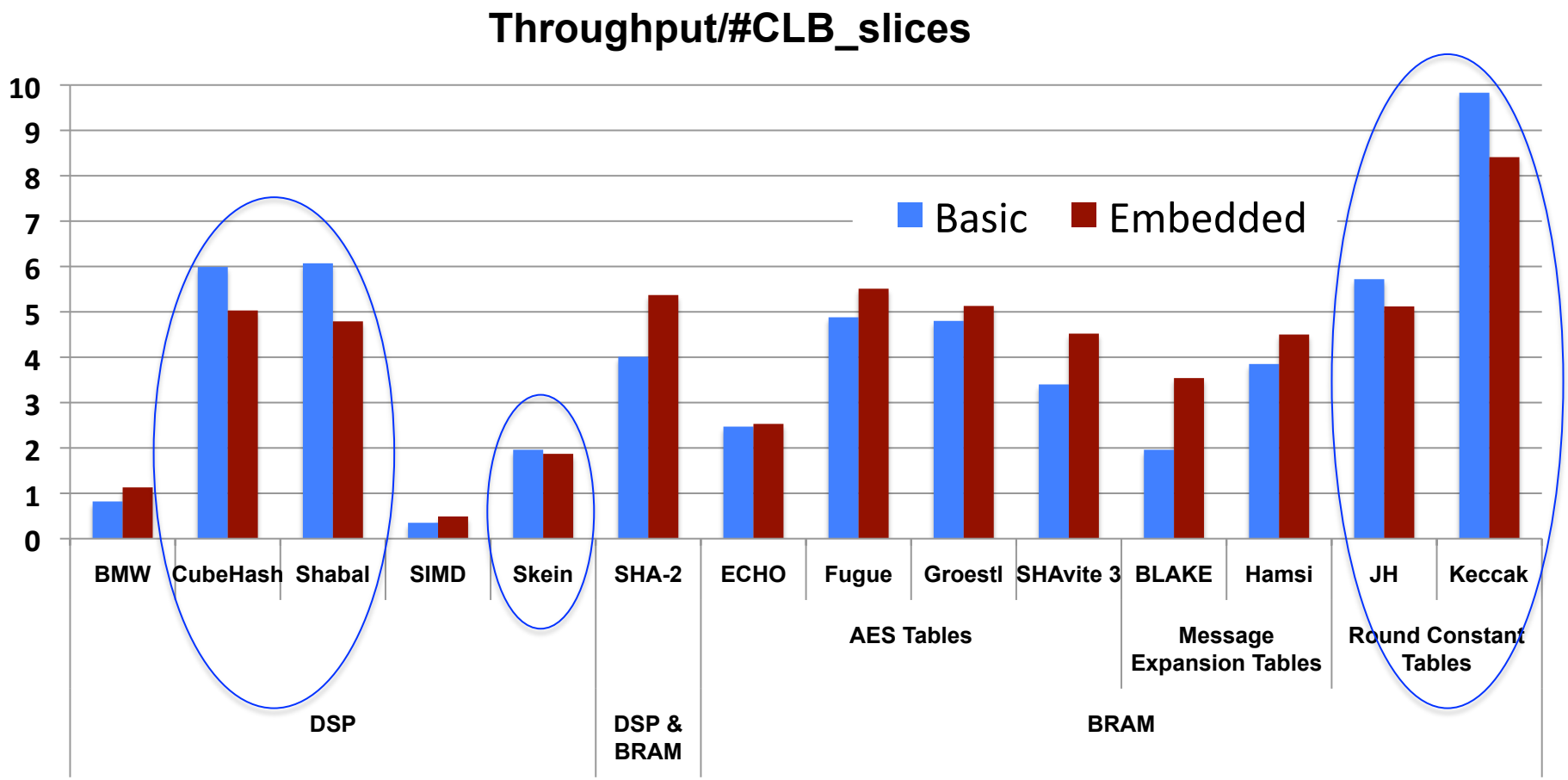
3. Increase Throughput/#Logic Resources Ratio?

No - 4 functions
(CubeHash, Shabal, Skein, Keccak)

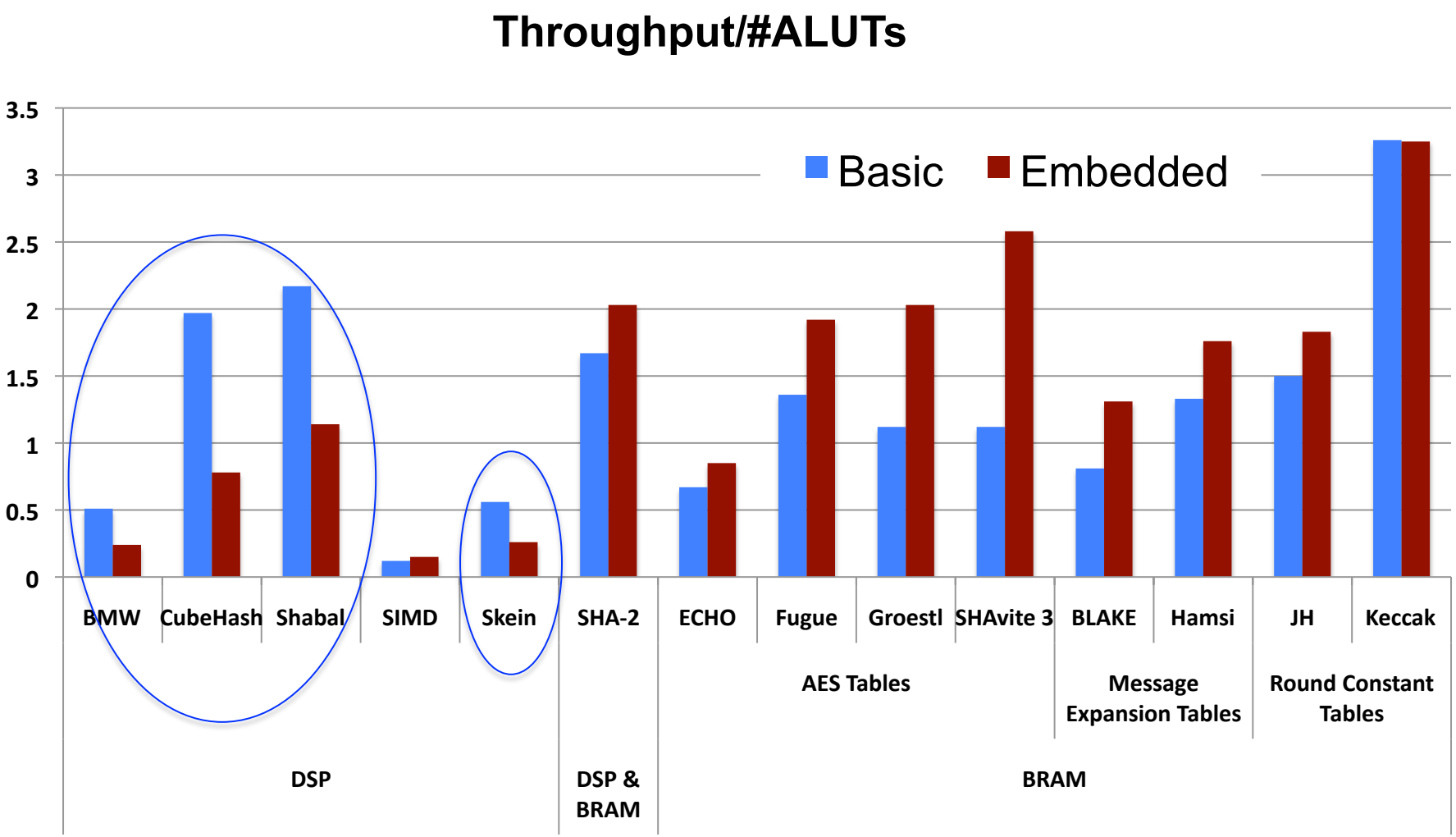
Yes - **10 functions**

more than 100% - Groestl: Spartan 3, Cyclone II
SHAvite-3: Spartan 3, Stratix III
Fugue: Cyclone II
ECHO: Spartan 3, Cyclone II

Xilinx Virtex 5 Results



Altera Stratix III Results



Conclusions

Conclusions

- **14 modern hash functions implemented using 4 FPGA families with and without using embedded resources**

- **No or marginal improvement in Throughput**

- **Greater than 20% savings in the amount of logic resources** possible for functions based on large look-up tables (such as AES-based hash functions, BLAKE, and Hamsi)

- **Limited advantage of using DSP units**
  - only for functions using multipliers and multi-operand addition
  - improvement larger for Virtex 5 than Stratix III

- Significant **difference in performance across FPGA families**

Thank you!

Questions?

Questions?

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

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