Tim Grembowski, Roar Lien, Kris Gaj, Nghi Nguyen
George Mason University

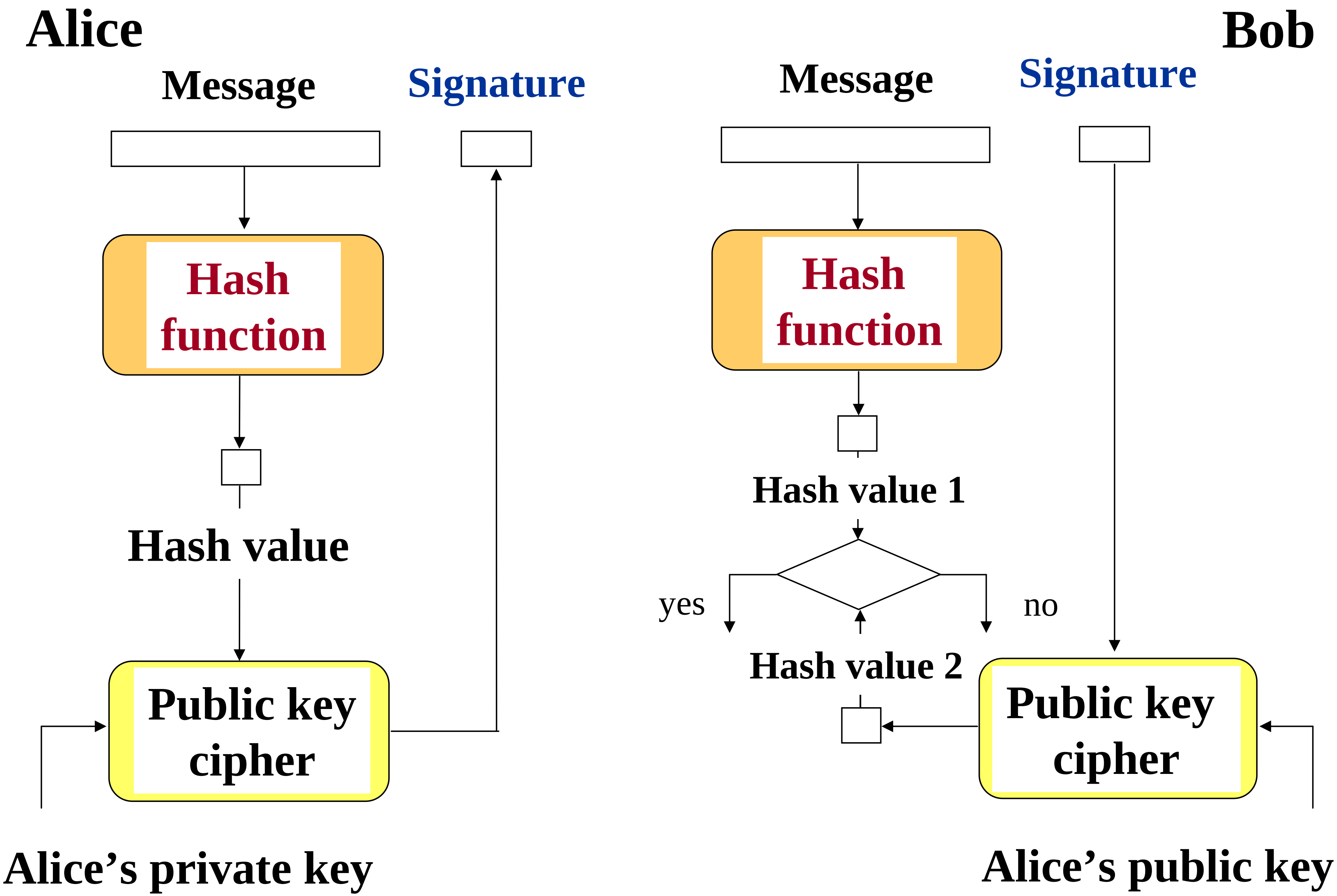
Peter Bellows, Jaroslav Flidr, Tom Lehman, and Brian Schott
USC - Information Sciences Institute

Comparative Analysis of the Hardware Implementations of Hash Functions SHA-1 and SHA-512

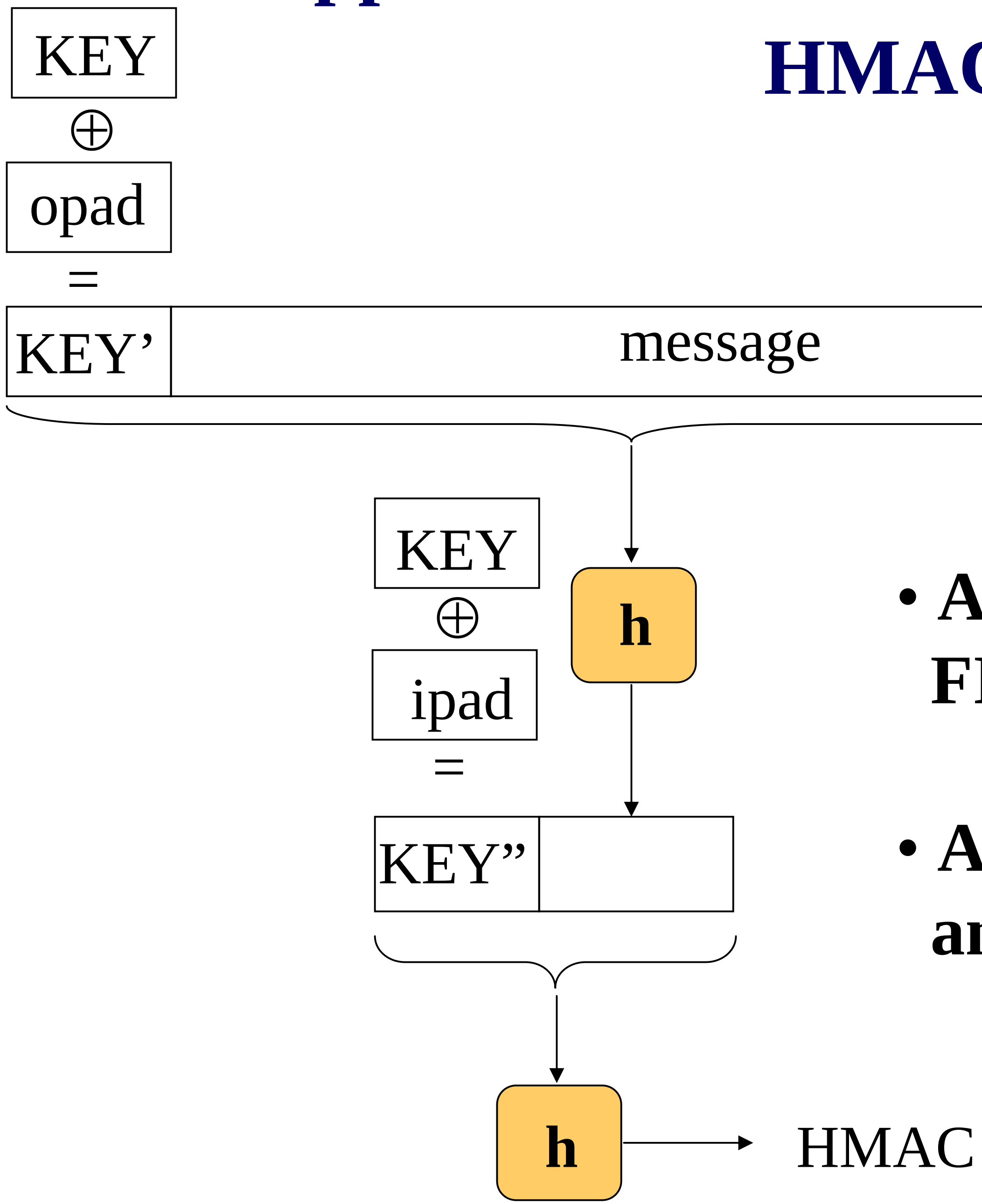
http://ece.gmu.edu/crypto-text.htm

Motivation
& Problem Statement

Applications of hash functions: Digital signatures



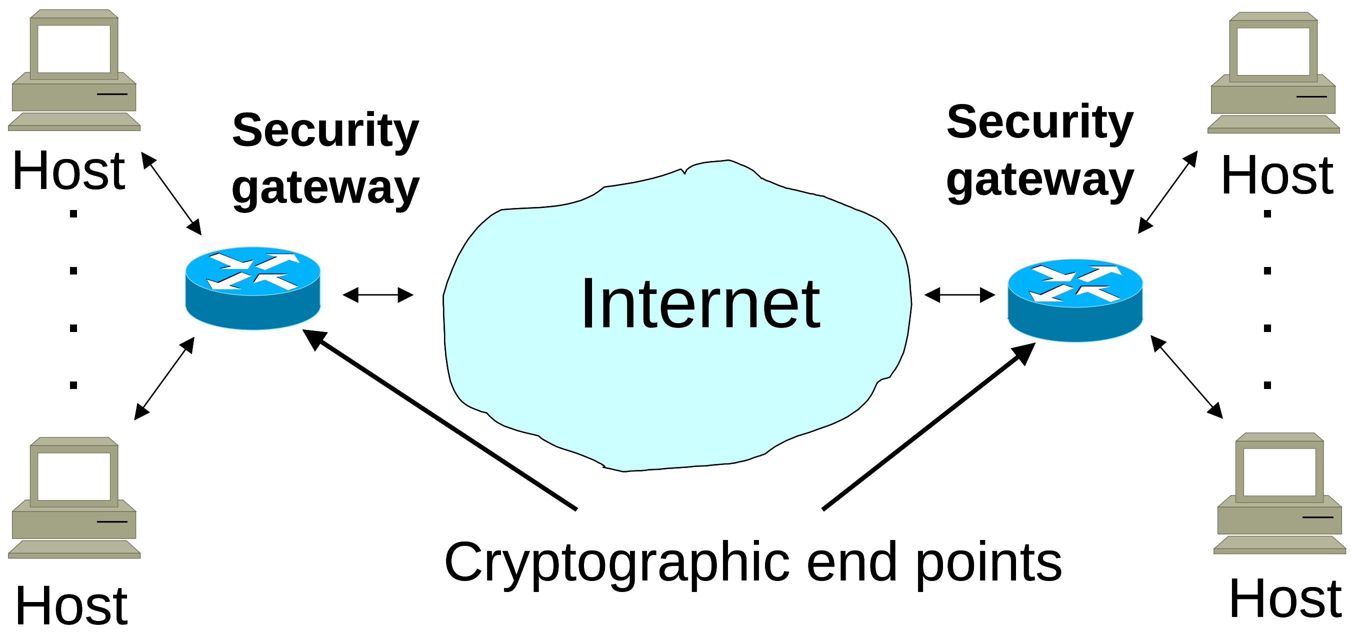
Applications of hash functions: MACs
HMAC



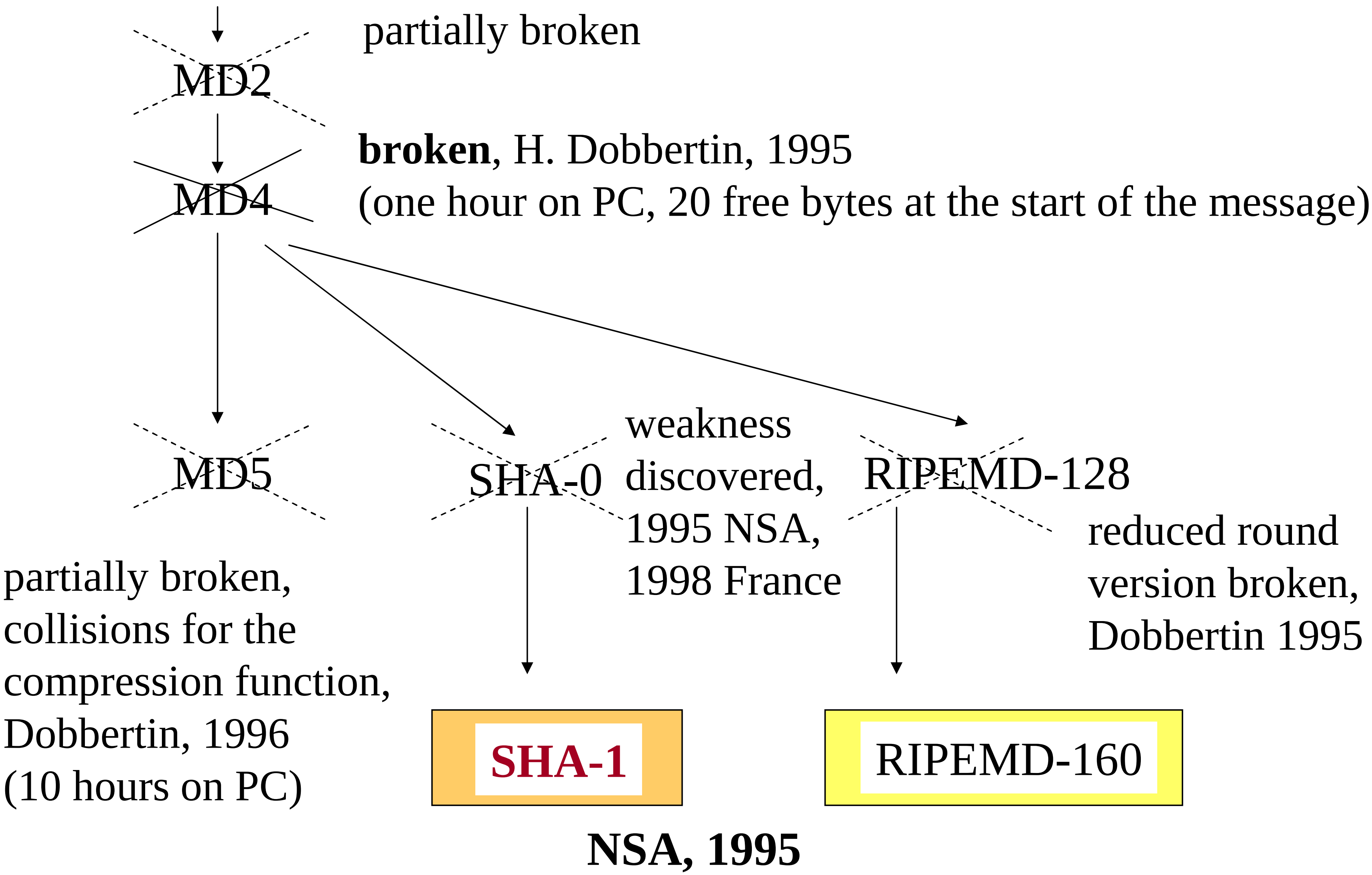
- **American standard FIPS 198**
- **Arbitrary hash function and key size**

Use of hash functions in communication protocols

- SSL
- IPSec



Security of hash functions



NSA/NIST family of cryptographic algorithms

	Algorithm	Complexity of the best known attack
Hash function	SHA-1	2^{80}
Digital signature	DSA	2^{80}
Encryption	Skipjack	2^{80}

New encryption standard – AES

Encryption algorithm	Complexity of the exhaustive key search	New hash functions with equivalent security
AES-128	2^{128}	SHA-256
AES-192	2^{192}	SHA-384
AES-256	2^{256}	SHA-512

Questions asked

1. **Does the increased security of SHA-512 come at the cost of**
    - **decreased speed**
    - **increased area**
    - **decreased speed to area ratio**

    compared to SHA-1?

2. **How does the speed of SHA-512 compares to the speed of AES-256?**

3. **Can SHA-512 be implemented with the speed of 1 Gbit/s using the current generation of FPGA devices?**

Conceptual Comparison

Conceptual comparison
Features affecting security and functionality

	SHA-1	SHA-256	SHA-384	SHA-512
Size of hash value	160	256	384	512
Complexity of the best attack	2^{80}	2^{128}	2^{192}	2^{256}
Equivalently secure secret-key cipher	Skipjack	AES-128	AES-192	AES-256
Message size	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$

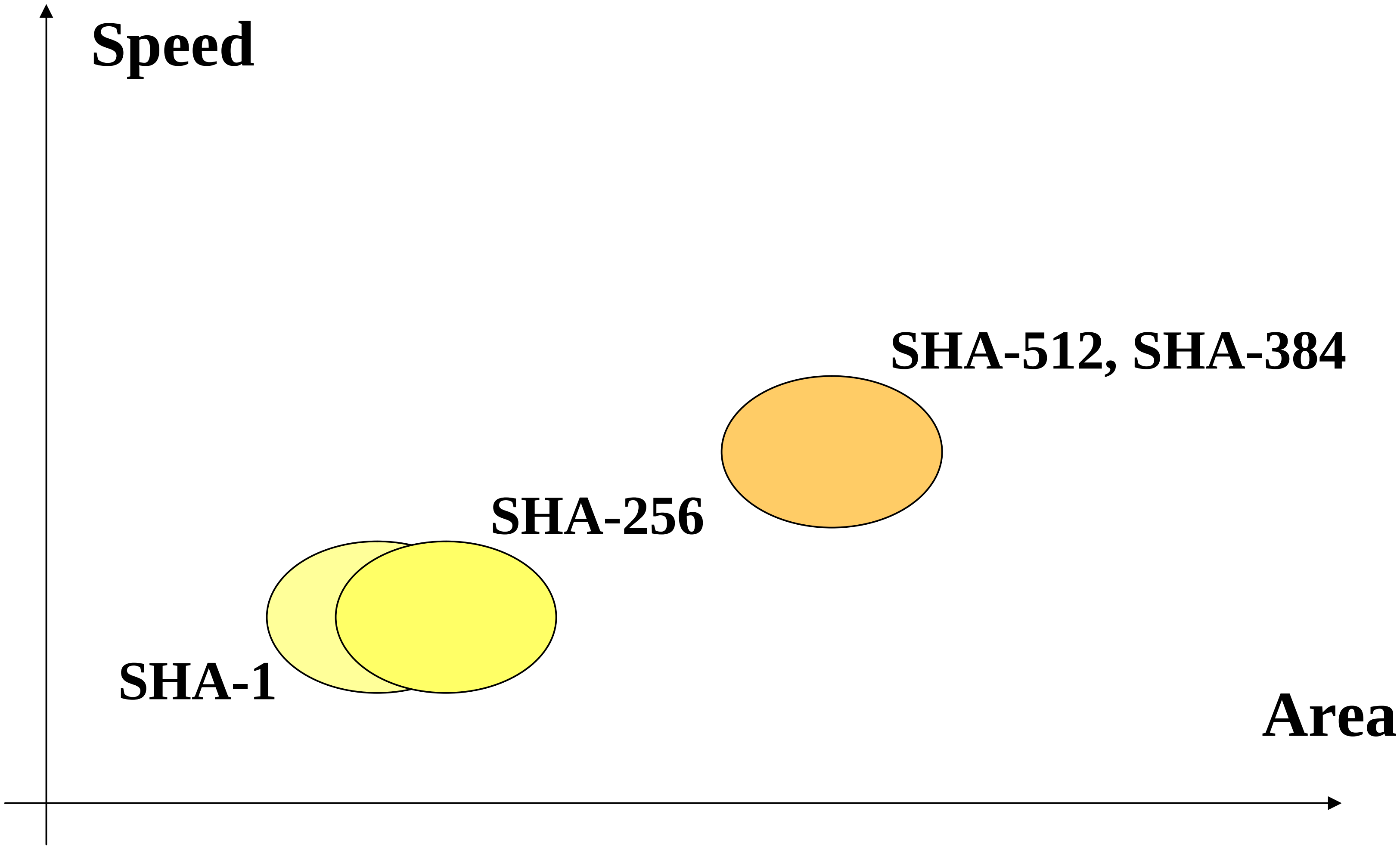
Conceptual comparison
Features affecting implementation speed

	SHA-1	SHA-256	SHA-384	SHA-512
Message block size	512	512	1024	1024
Number of digest rounds	80	64	80	80
Number of operands added in the critical path	5+1	7+1	7+1	7+1

Conceptual comparison
Features affecting implementation area

	SHA-1	SHA-256	SHA-384	SHA-512
Word size	32	32	64	64
Number of words	5	8	8	8
Round-dependent operations	f_t	None	None	None
Number of constants K_t	4	64	80	80

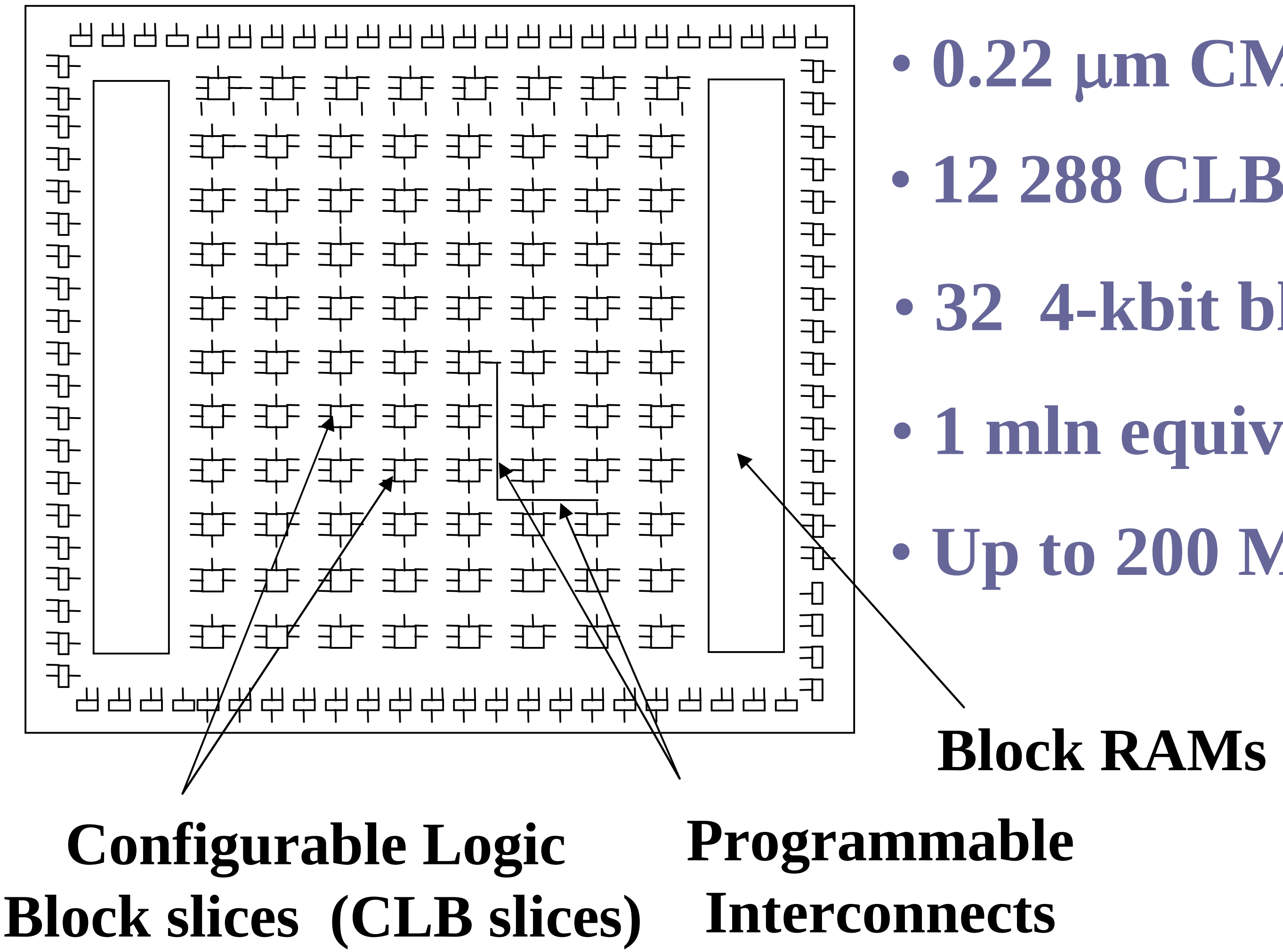
Results of conceptual comparison



Design Methodology

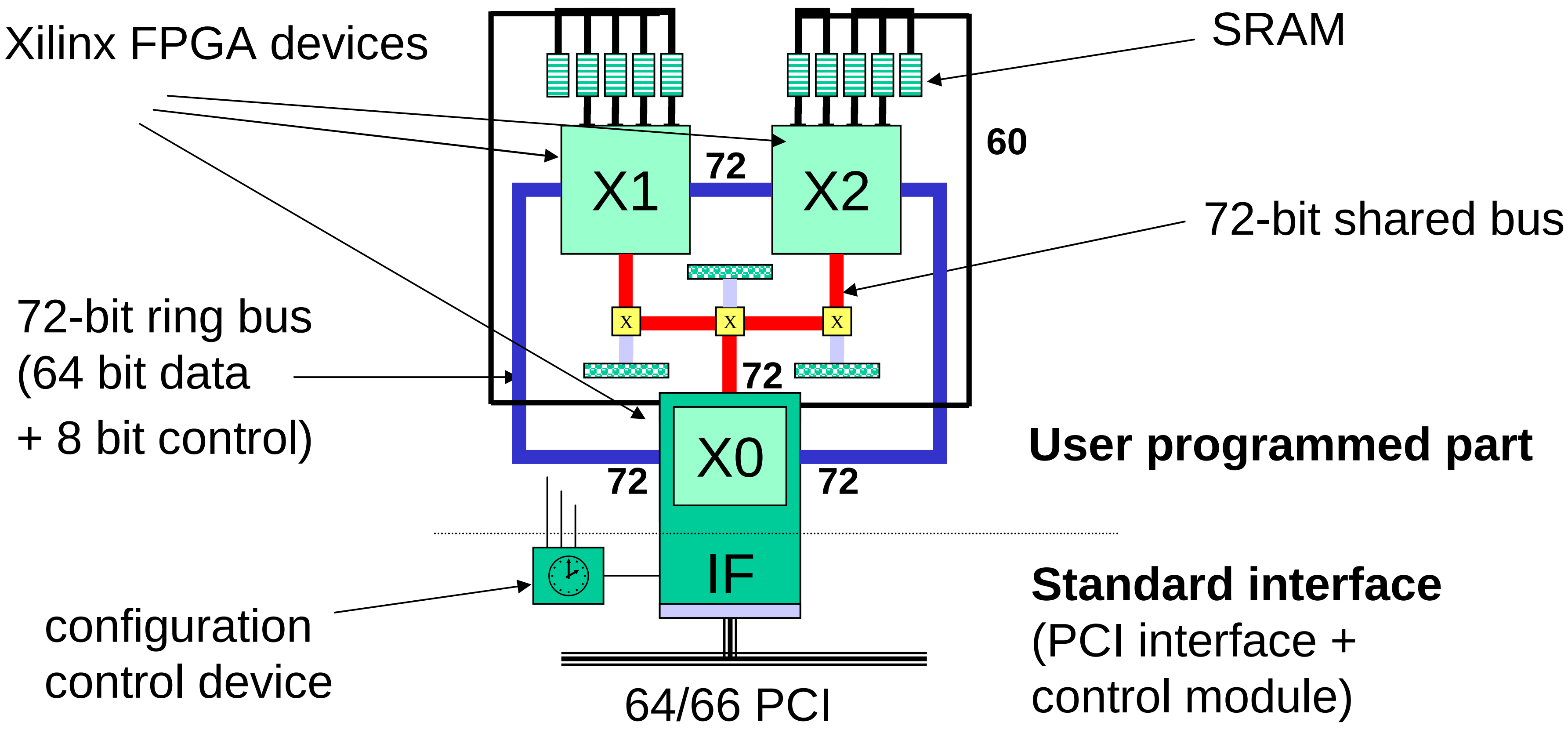
Target FPGA devices

Xilinx Virtex - XCV 1000



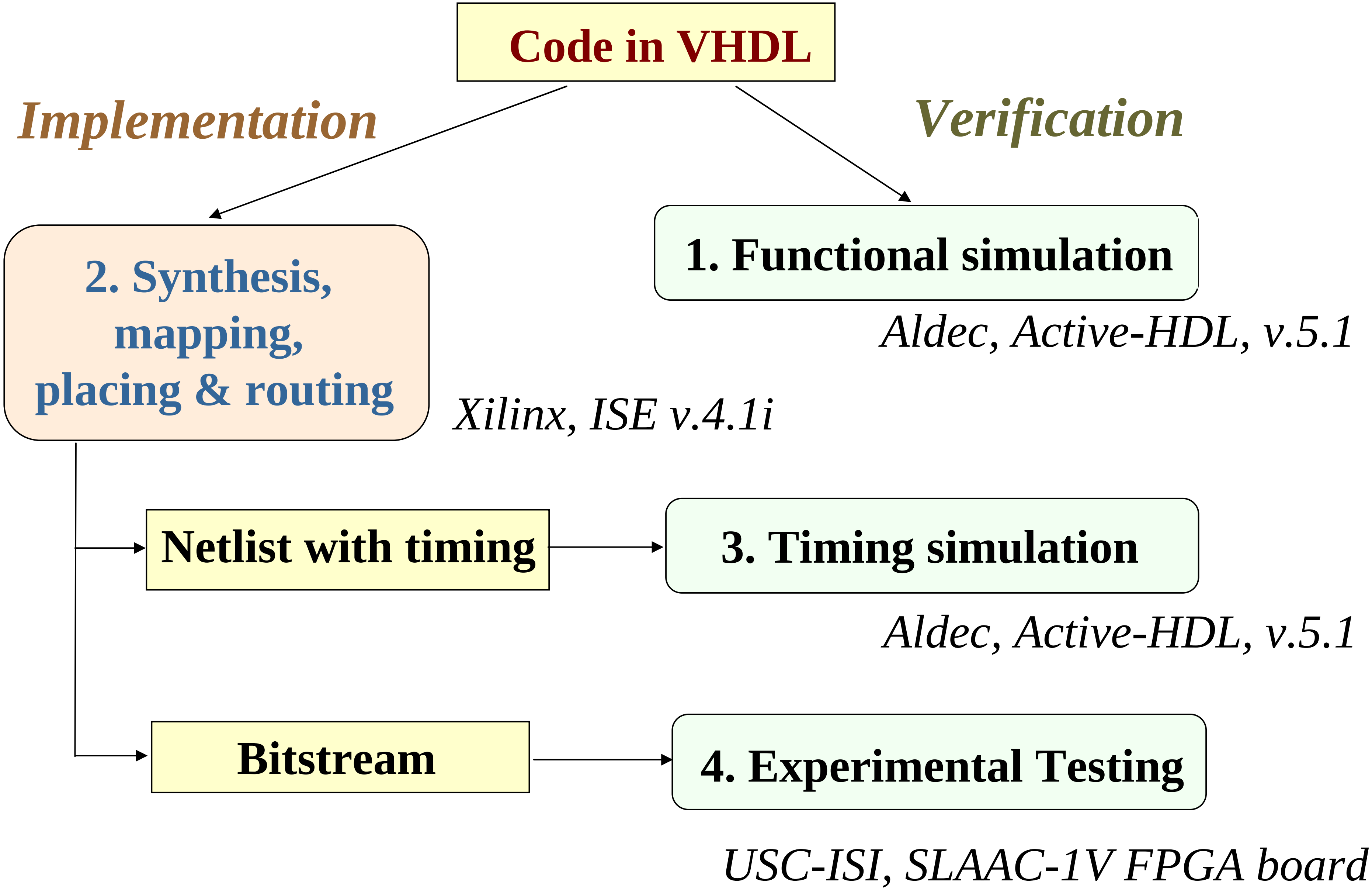
- 0.22 µm CMOS process
- 12 288 CLB slices
- 32 4-kbit block RAMs
- 1 mln equivalent logic gates
- Up to 200 MHz clock

SLAAC-1V



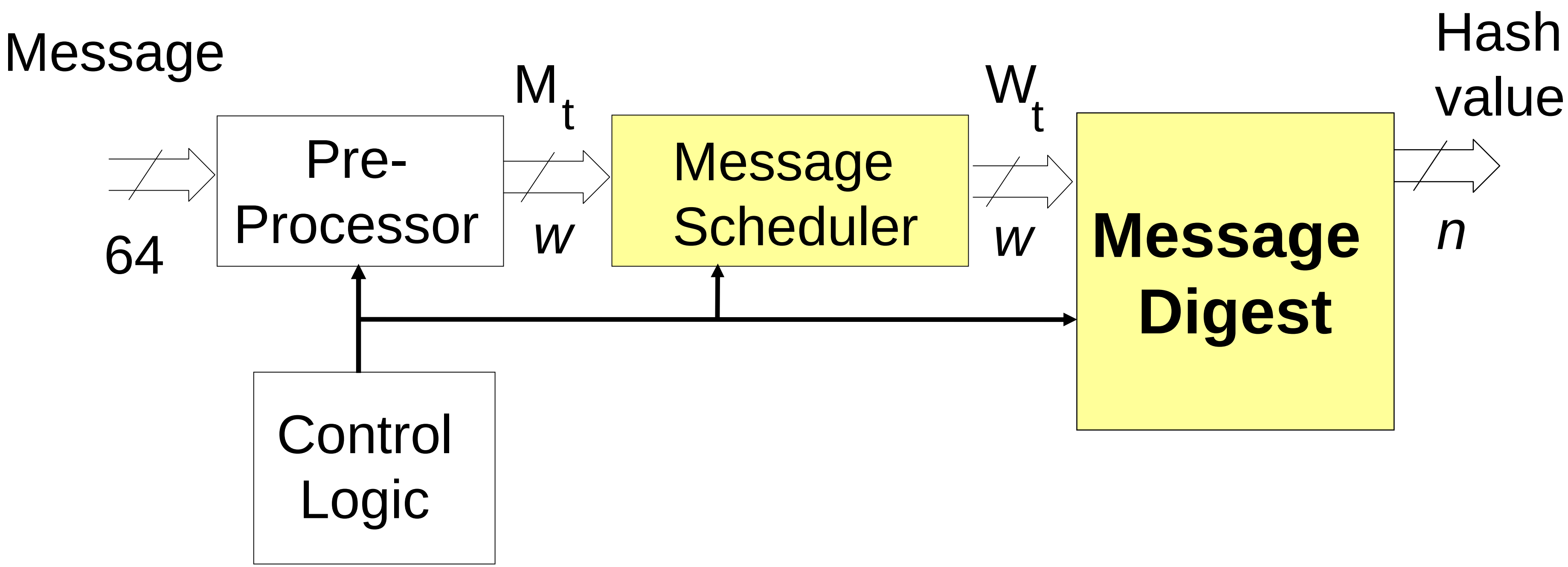
Xilinx FPGA devices
SRAM
60
72
X1
X2
72-bit shared bus
72-bit ring bus
(64 bit data
+ 8 bit control)
X
X
X
72
X0
72
72
User programmed part
IF
Standard interface
(PCI interface +
control module)
configuration
control device
64/66 PCI

Methodology and Tools



Hardware Architectures

General block diagram of SHA-1 and SHA-512



For SHA-1:

w=32

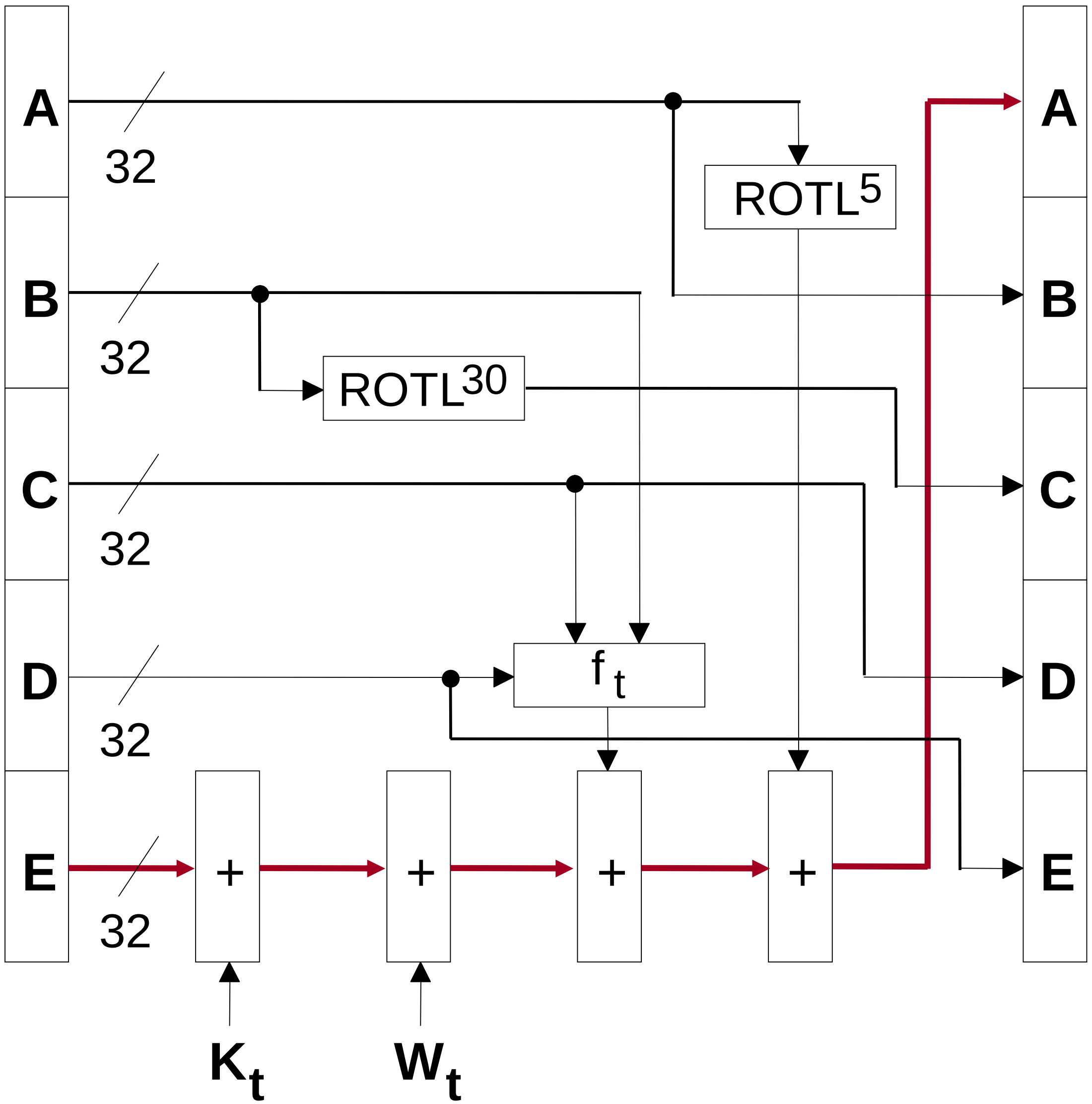
n=160

For SHA-512:

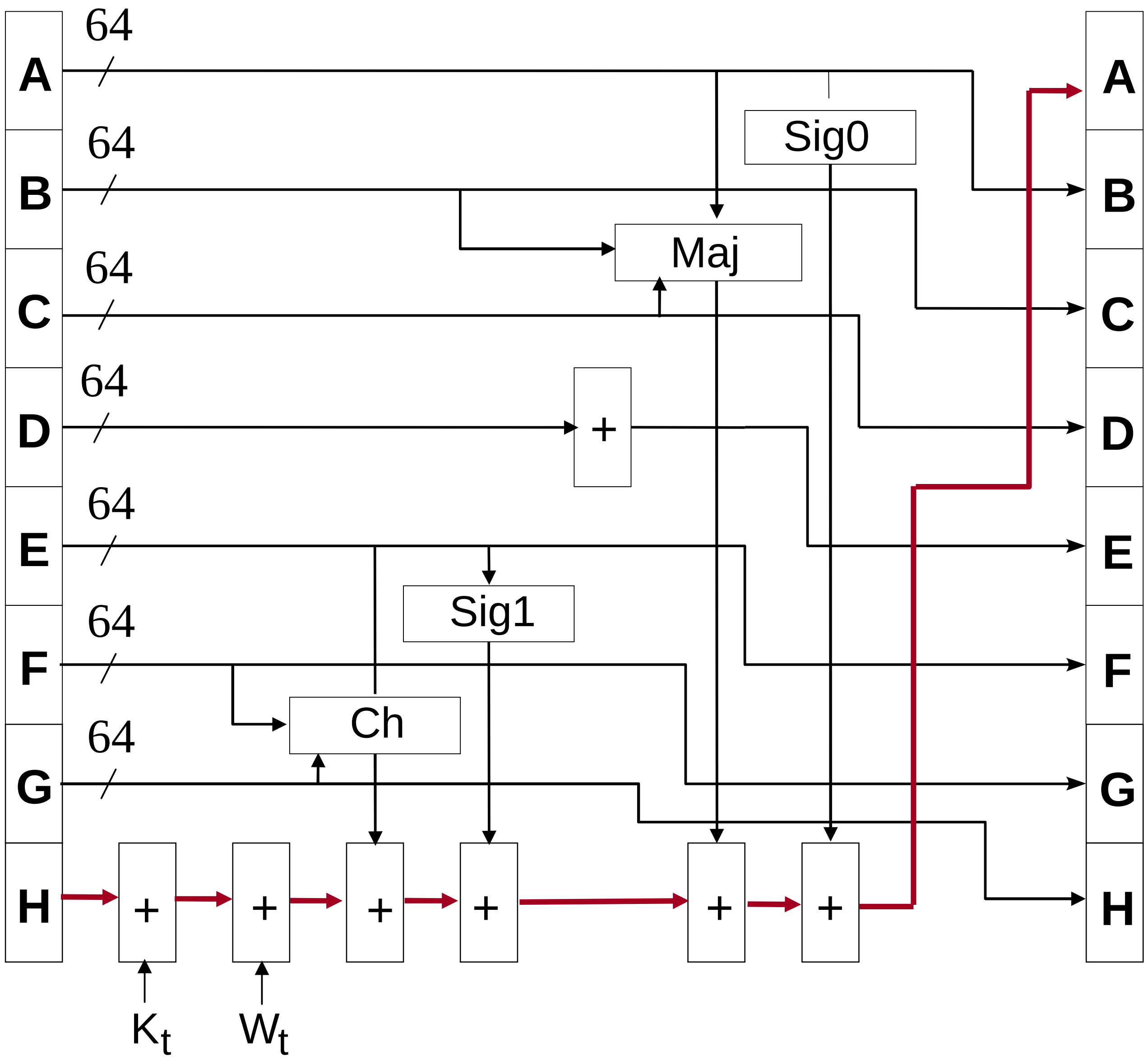
w=64

n=512

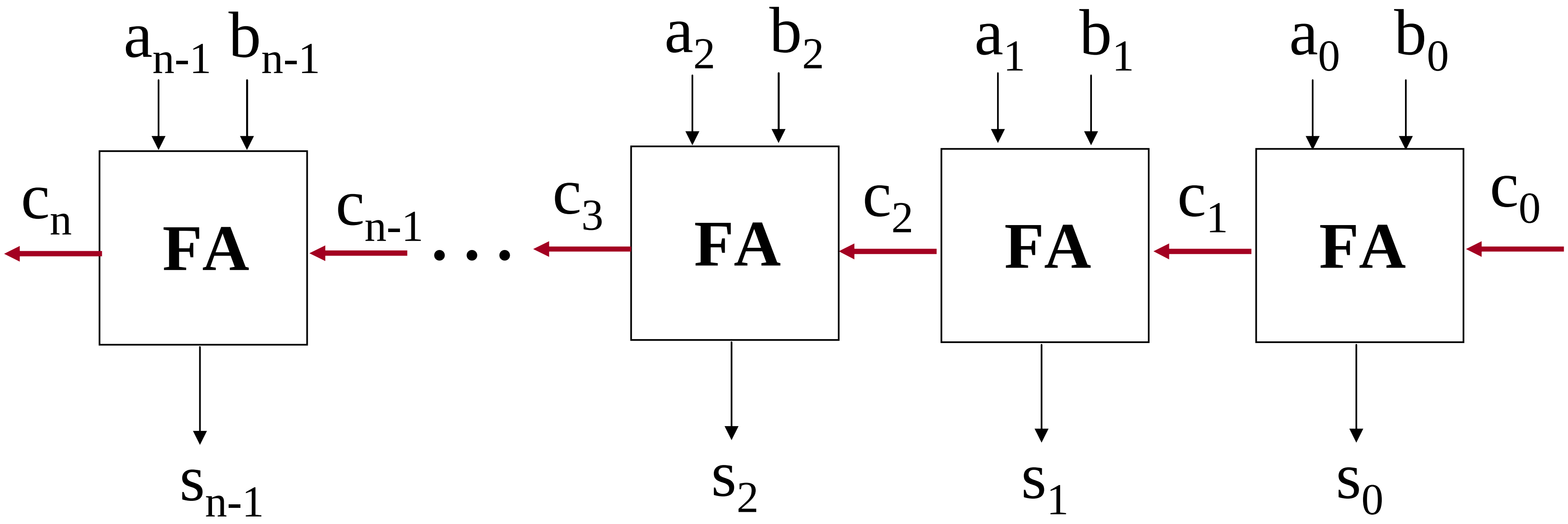
Message digest unit of SHA-1 Functional block diagram



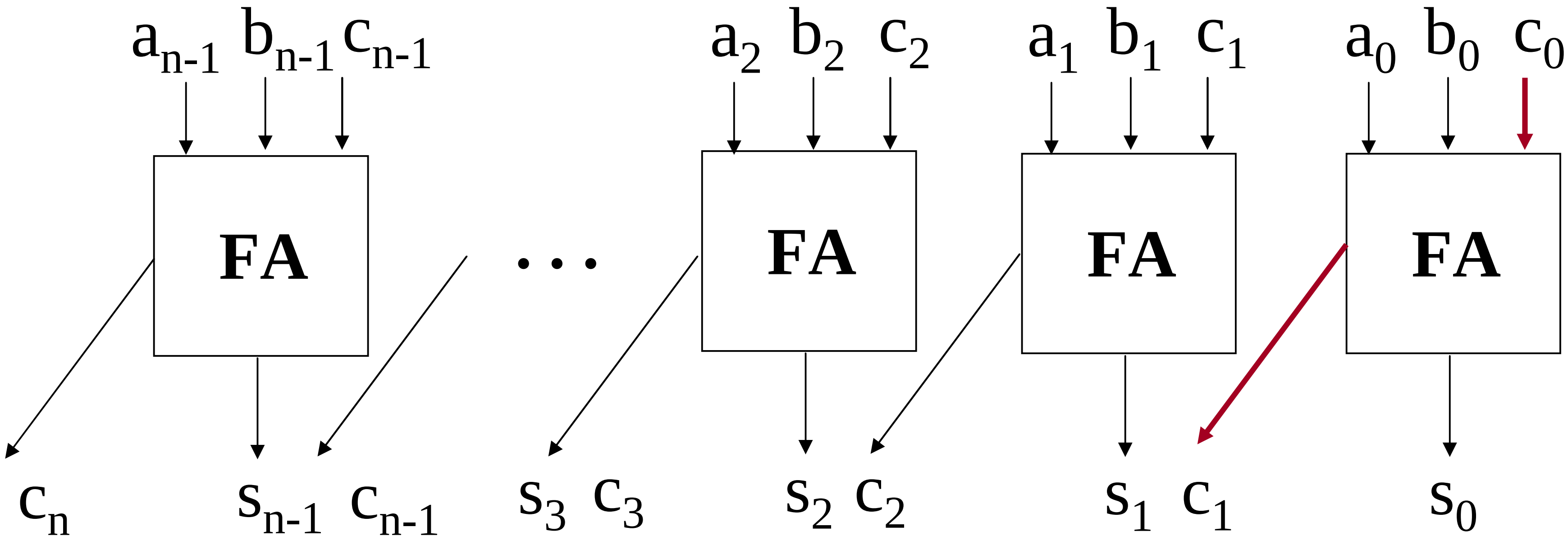
Message digest unit of SHA-512
Functional block diagram



Ripple-Carry Carry Propagate Adder (CPA)



Carry Save Adder (CSA)



Operation of a Carry Save Adder (CSA)

Example

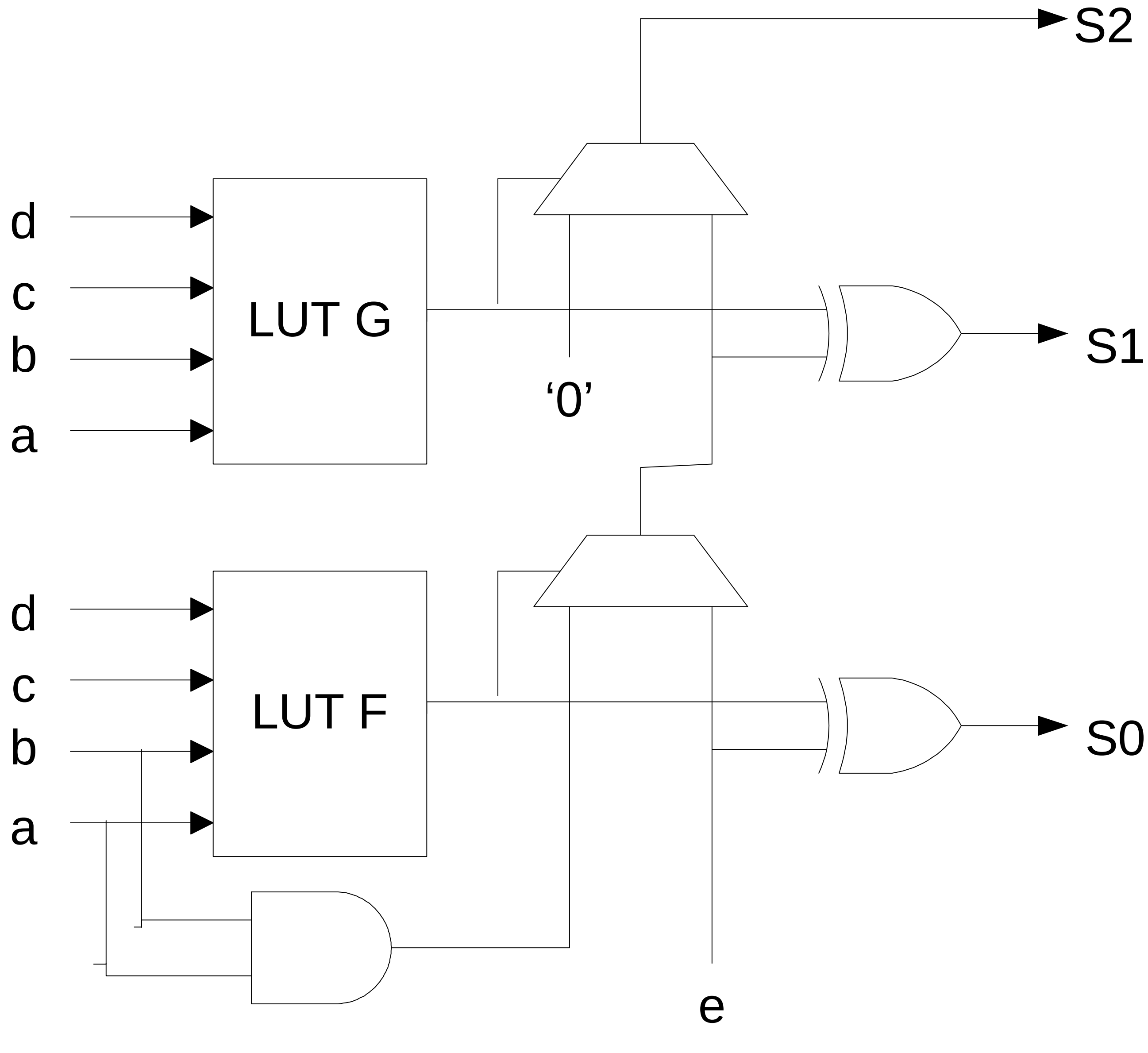
	2^{i+2}	2^{i+1}	2^{i}	2^{i-1}	2^{i-2}
a	0	1	0	1	0
b	1	1	0	1	1
c	1	0	1	1	1
s	0	0	1	1	0
c	1	0	1	1	0

Operation of a 5-to-3 Parallel Counter

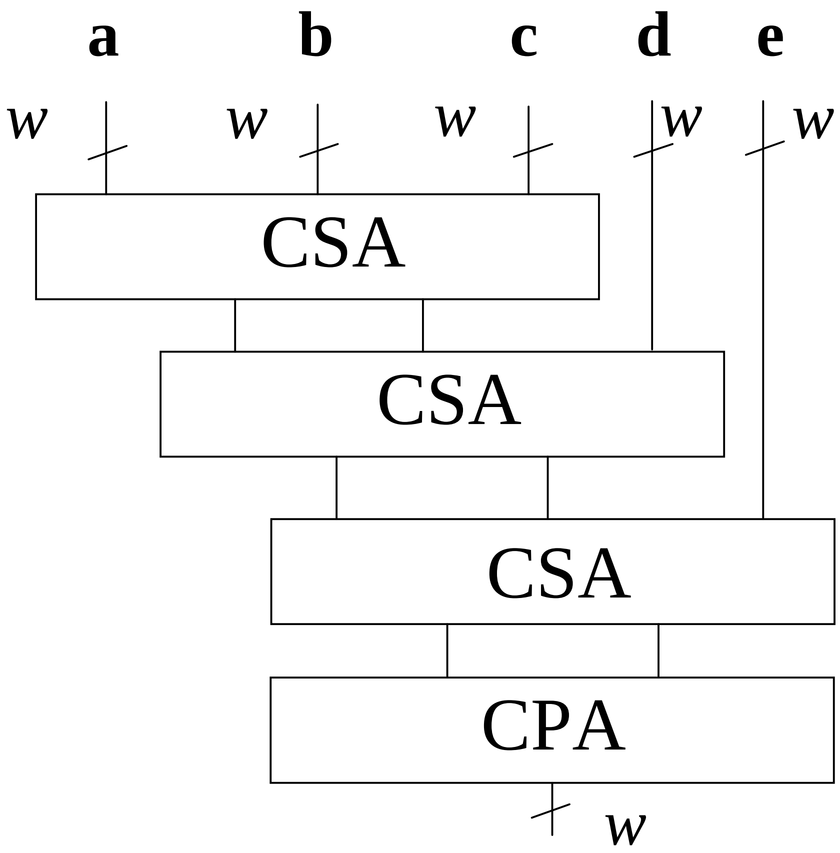
Example

	2^{i+2}	2^{i+1}	2^{i}	2^{i-1}	2^{i-2}
a	0	1	0	1	0
b	1	1	0	1	1
c	1	0	1	1	1
d	1	0	1	1	1
e	1	1	1	1	1
s0	0	1	1	1	0
s1	1	1	0	0	0
s2	0	1	1	0	1

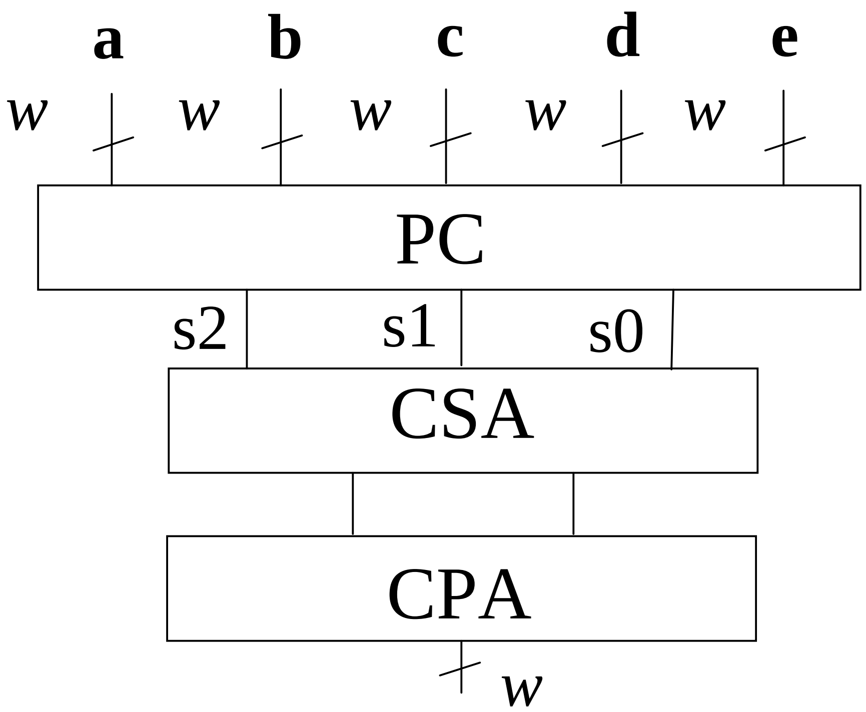
Implementation of 1-bit of 5-to-3 parallel counter using single CLB slice of a Virtex 1000 FPGA



Carry Save Adder vs. 5-to-3 Parallel Counter

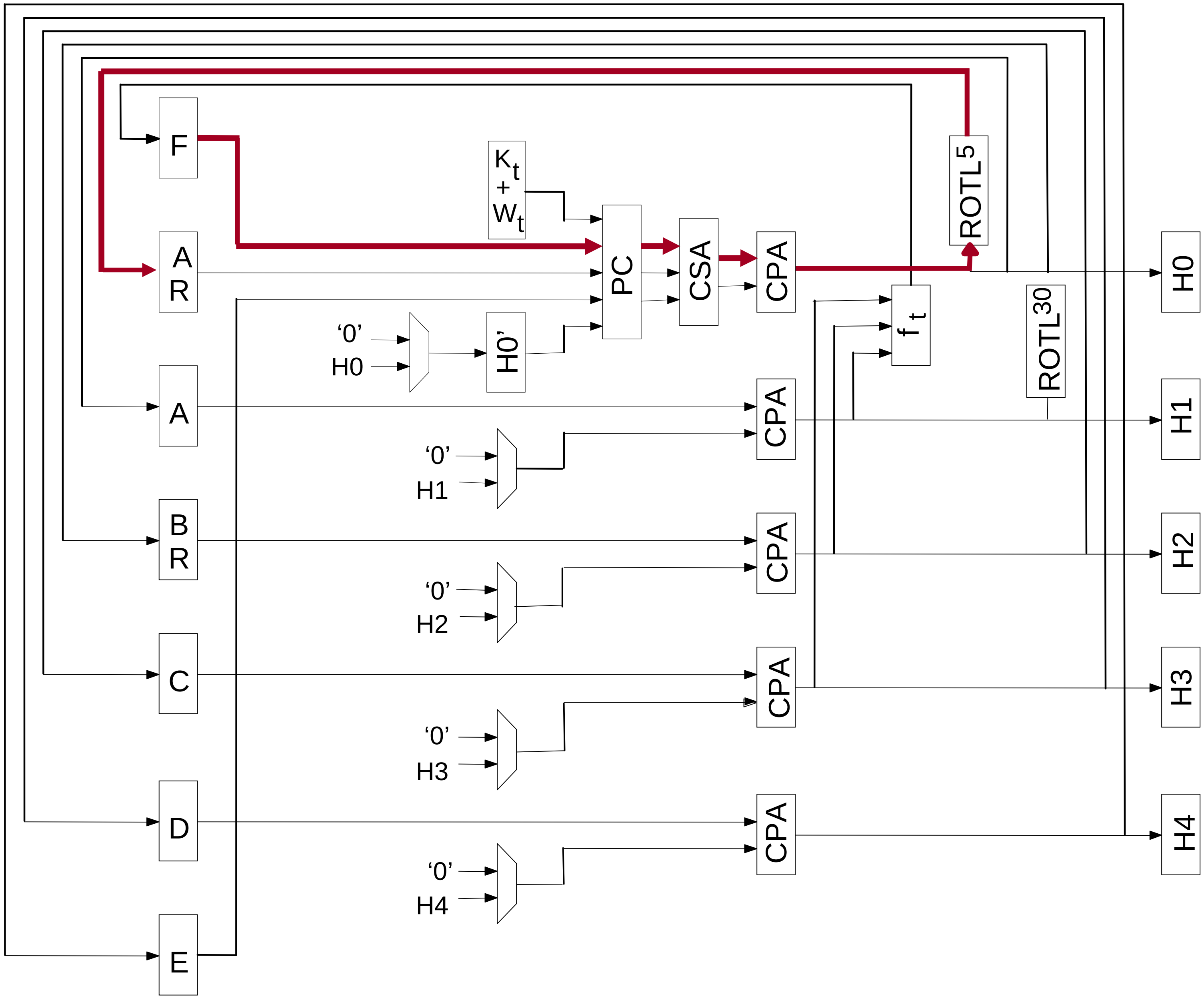


y=a+b+c+d+e mod 2^w

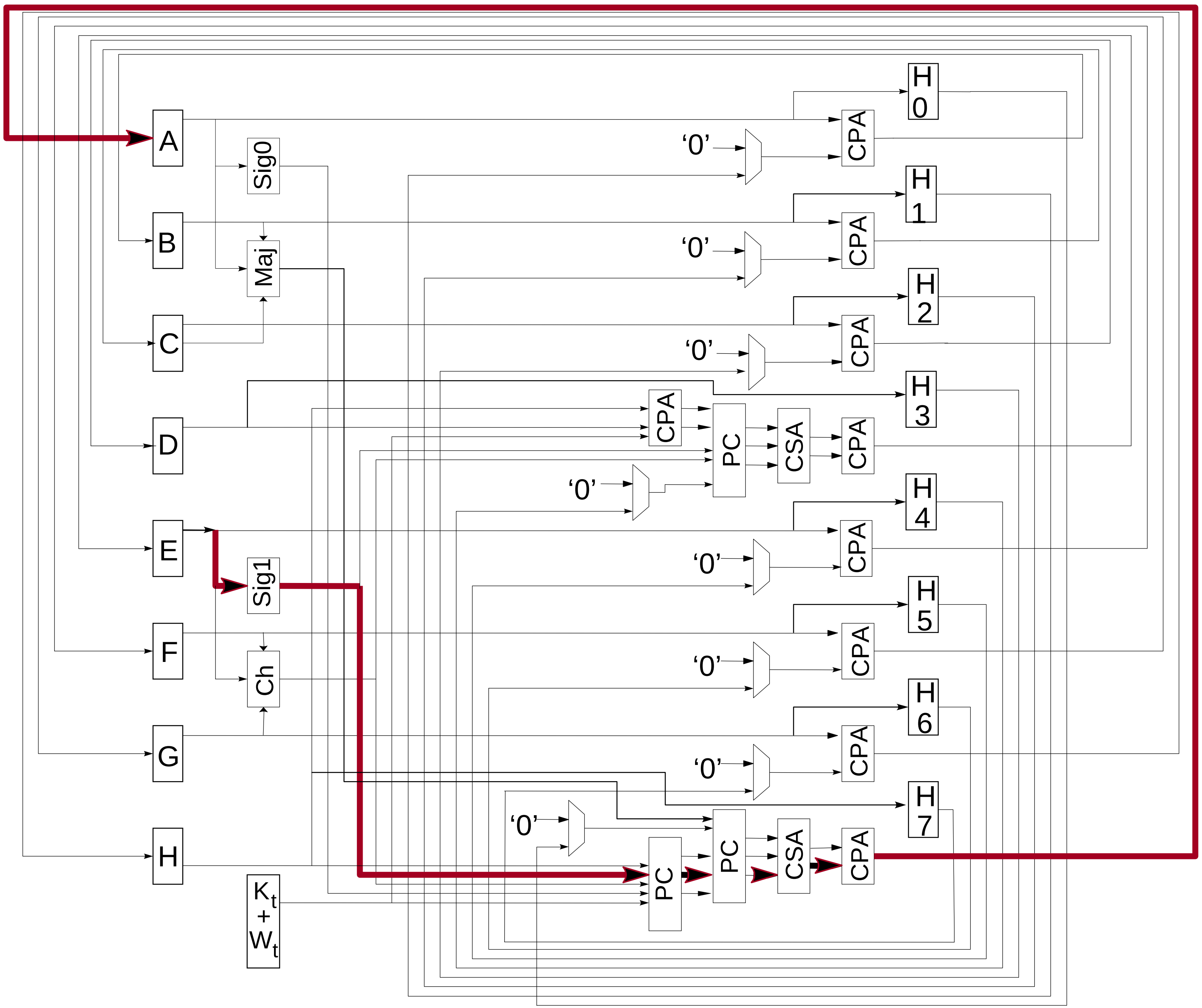


y=a+b+c+d+e mod 2^w

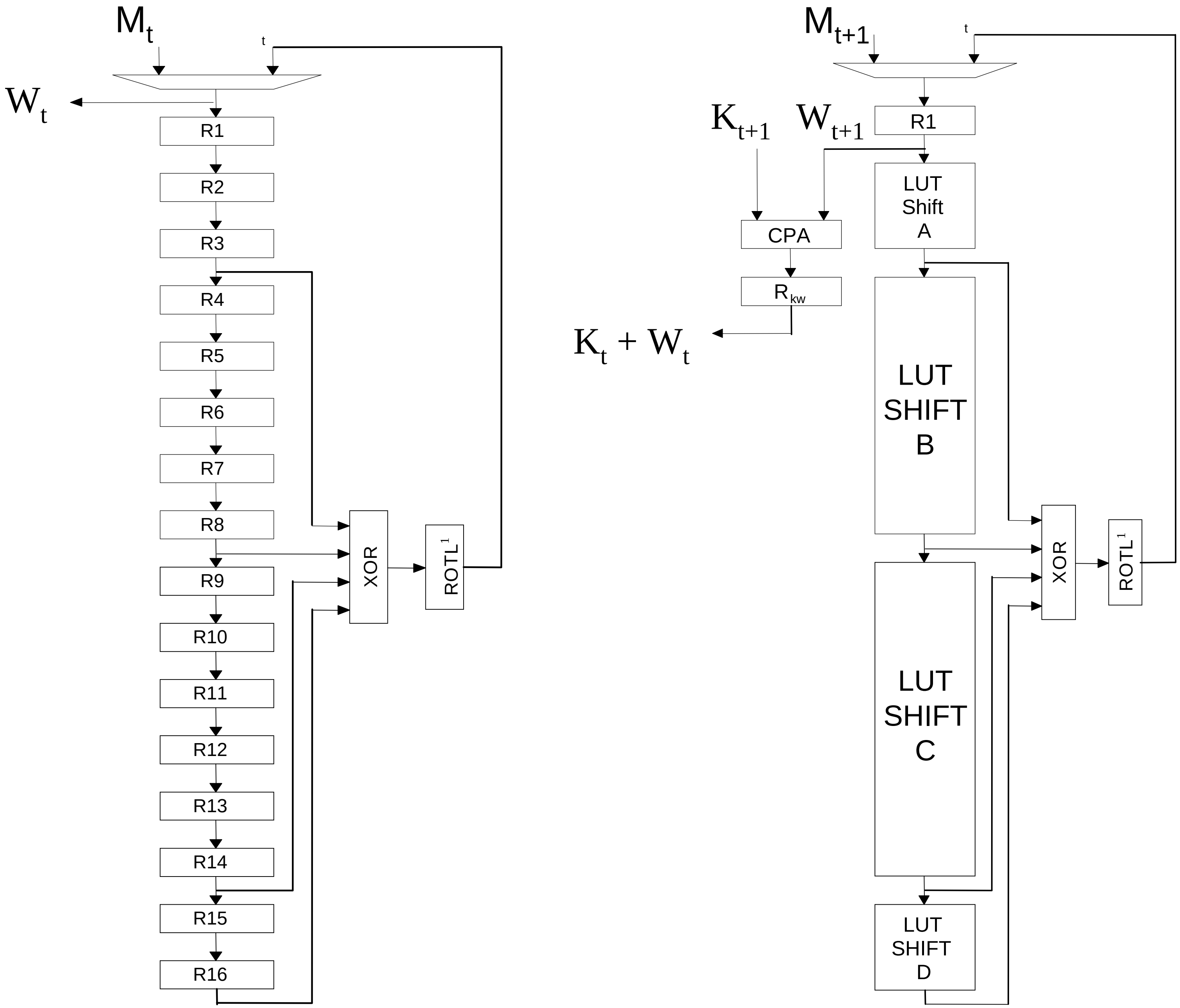
Message digest unit of SHA-1: Our implementation



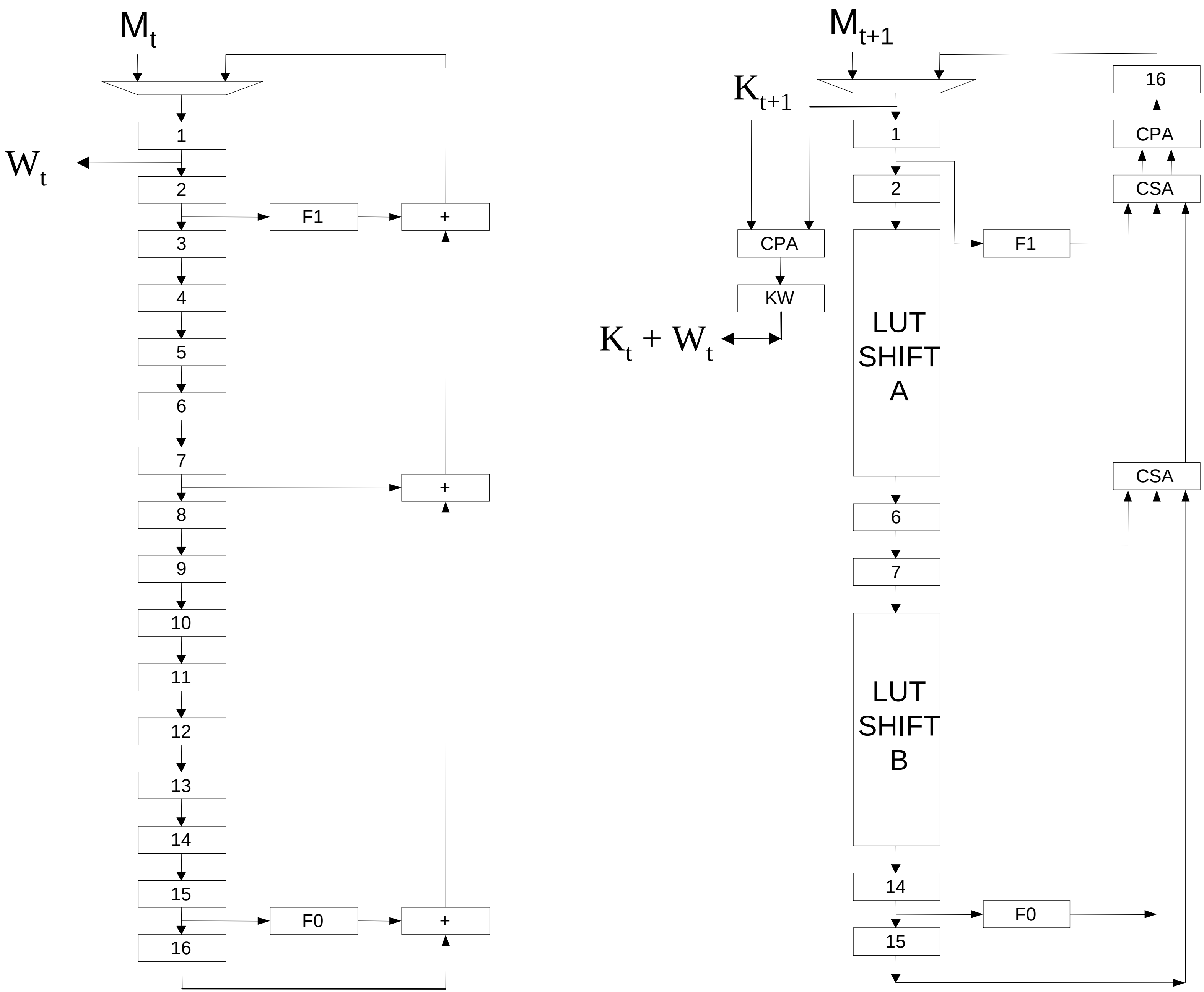
Message digest unit of SHA-512: Our implementation



SHA-1: Message Scheduler Unit



SHA-512: Message Scheduler Unit



Testing Procedure

Testing Procedure

1. Functional testing

Digital Signature Standard Validation System (DSSVS) User's Guide

- Known Answer Tests
- Monte Carlo Test

2. Maximum clock frequency test

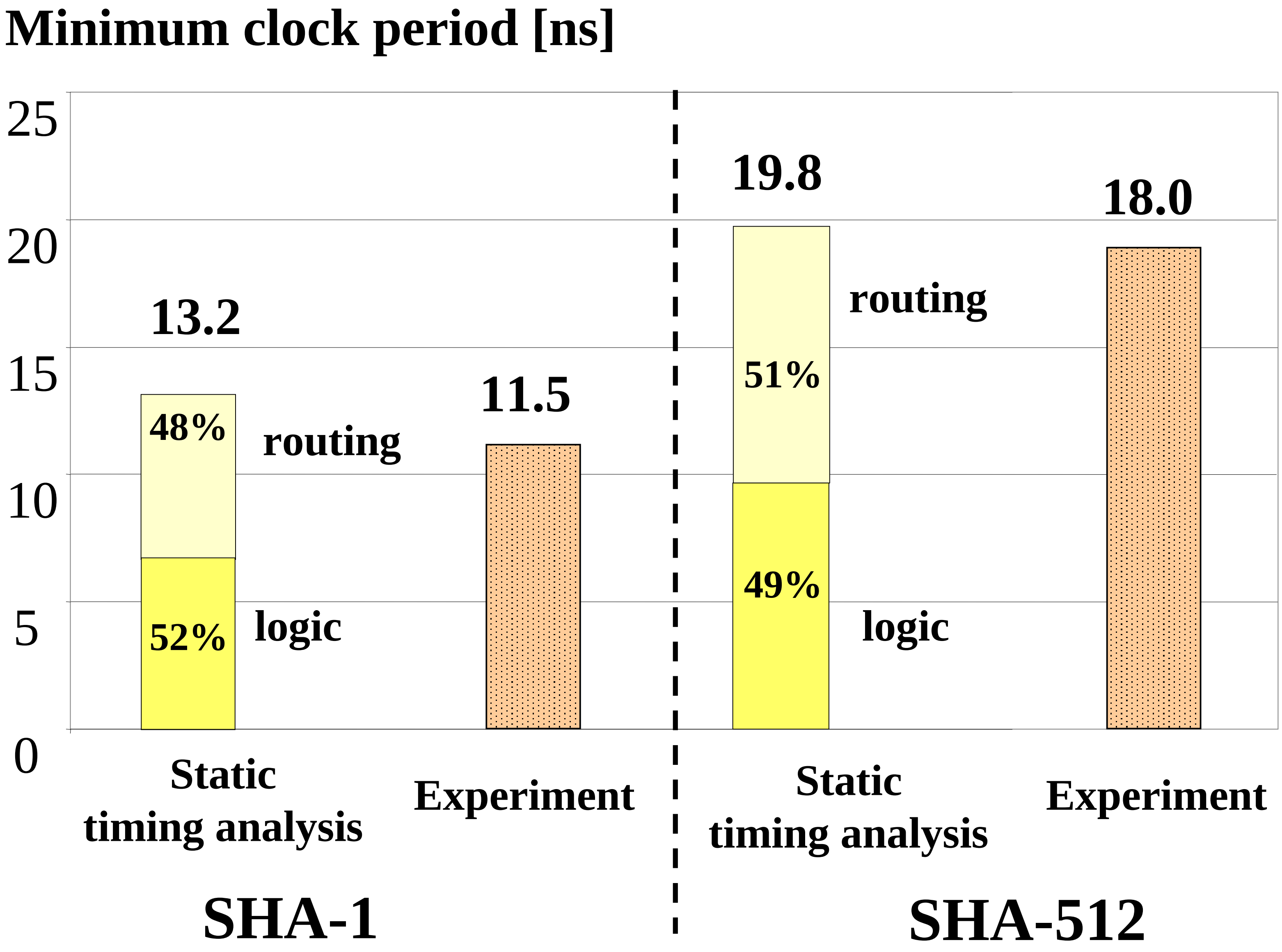
- clock frequency varied using binary search
- 30 x 3 MB of pseudorandom data hashed
- results compared with results from software implementation

3. Maximum data throughput test

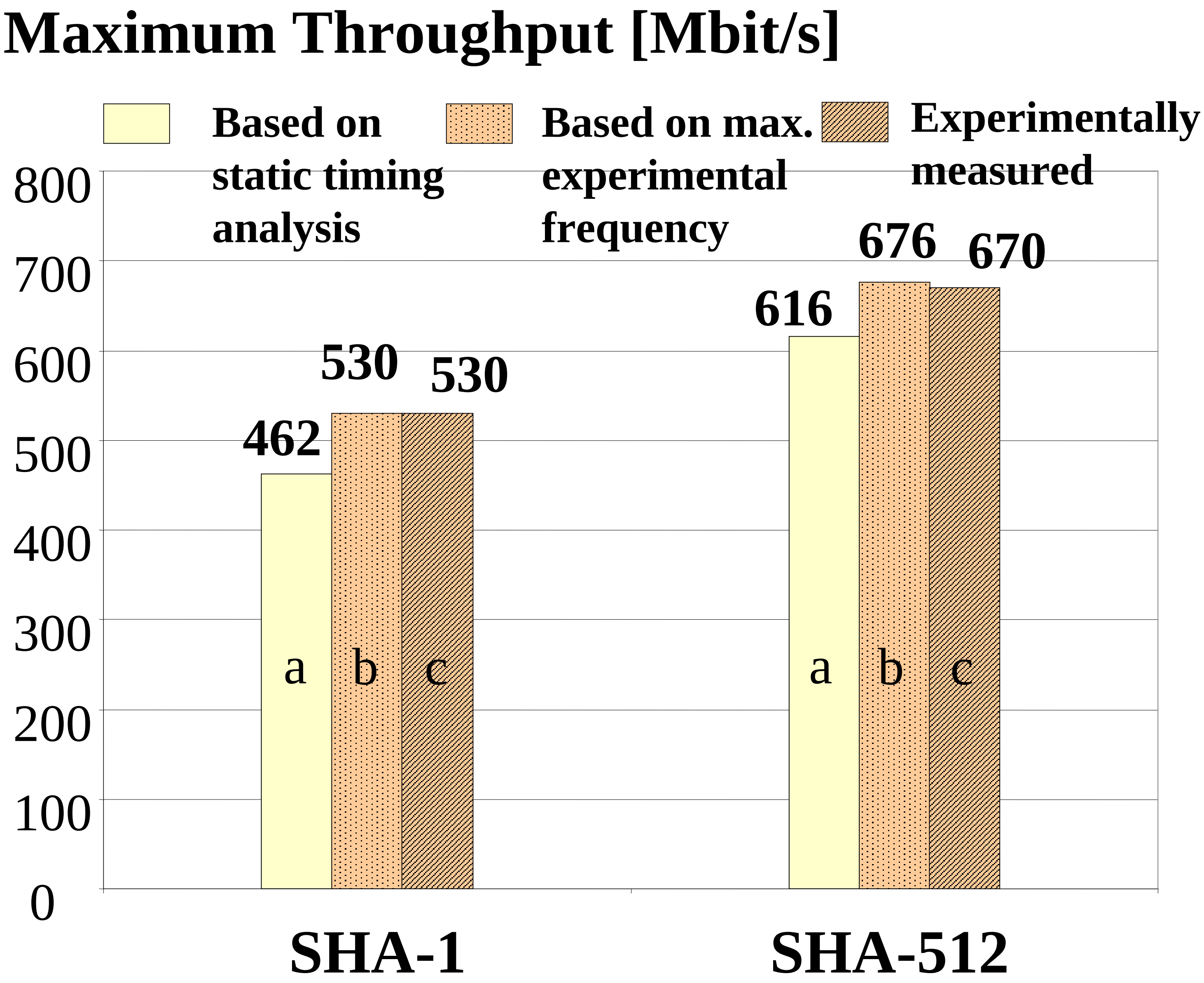
- maximum clock frequency
- 3 MB of pseudorandom data hashed
- time necessary to complete all operations determined

Results

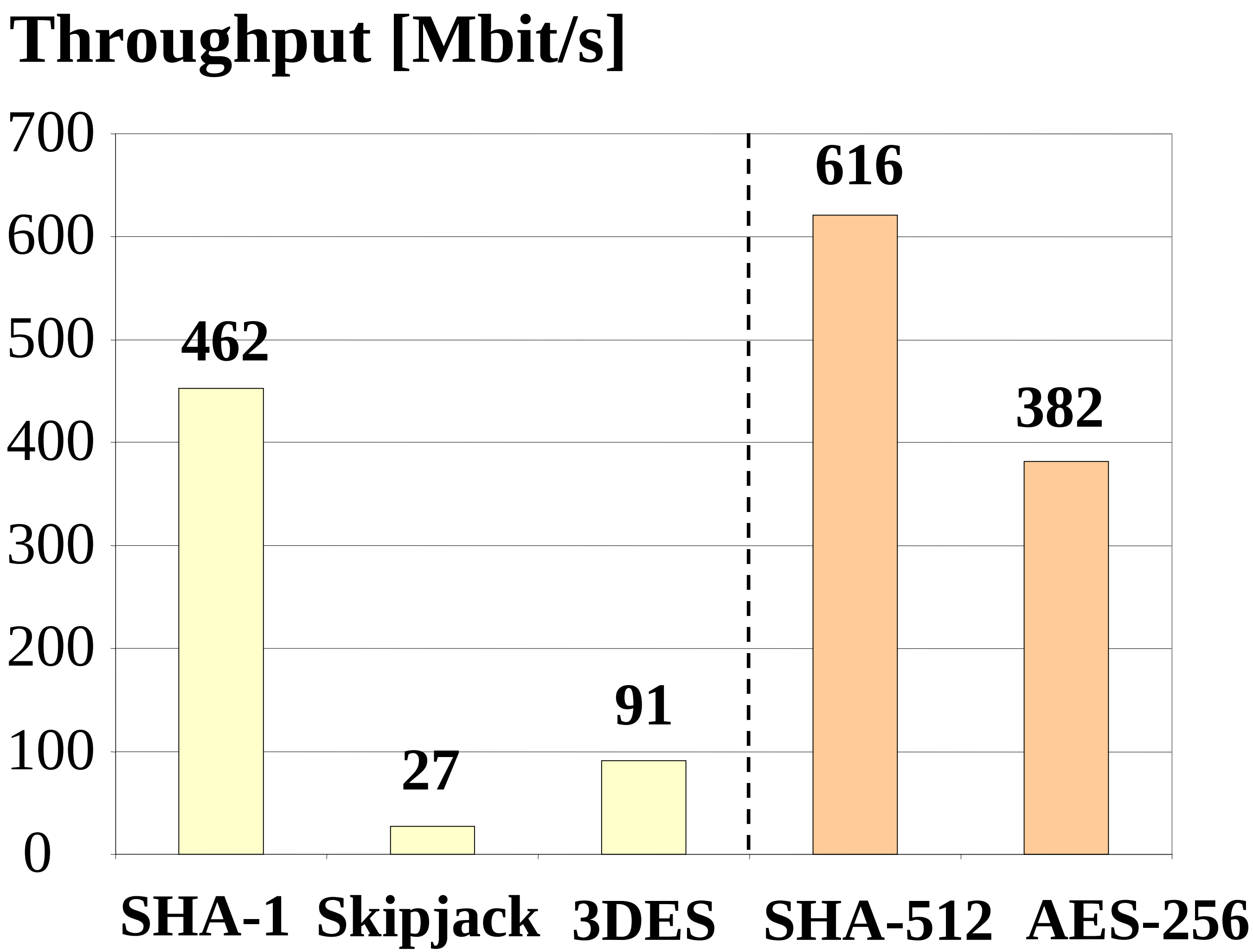
SHA-1, SHA-512: Minimum clock period



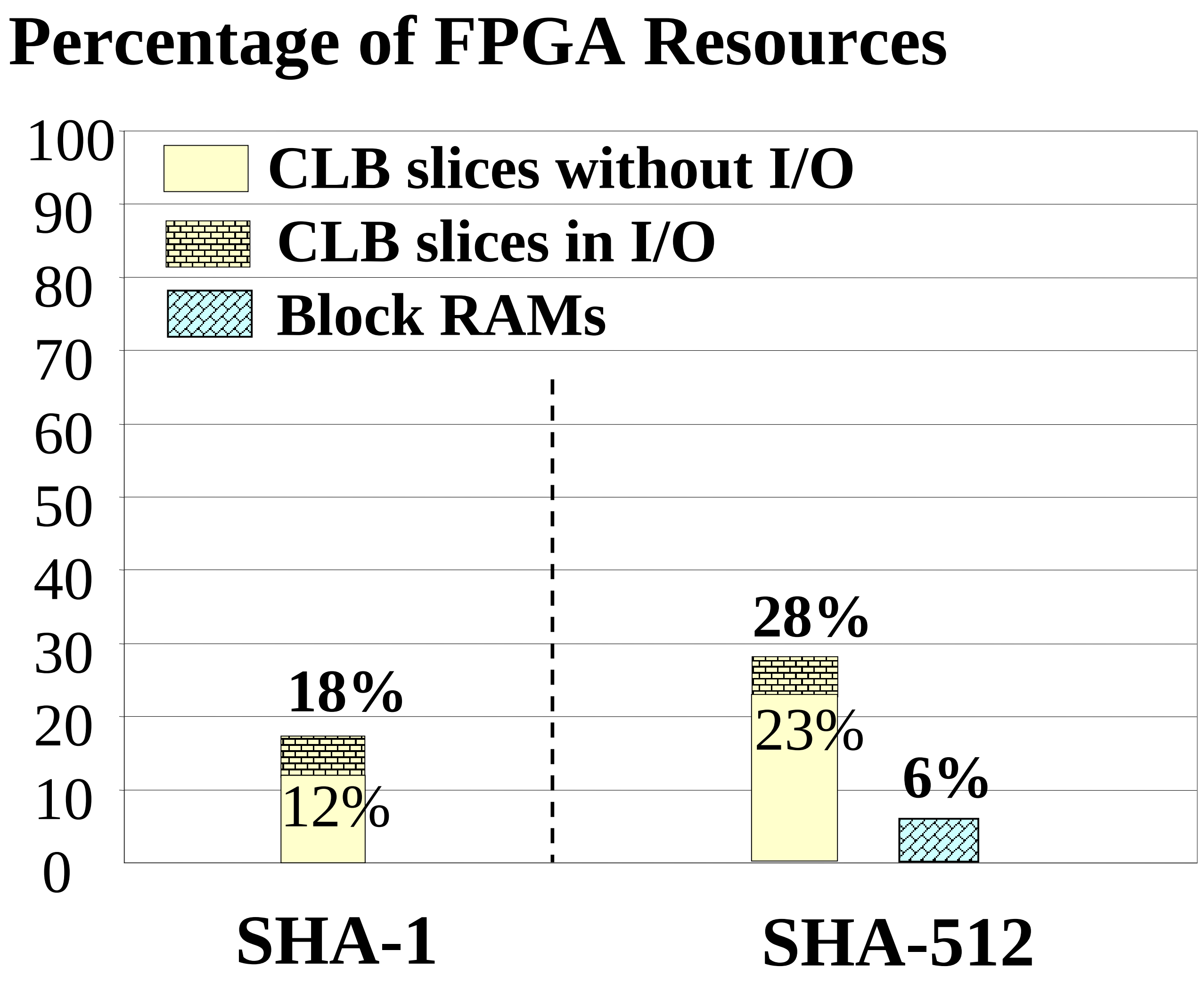
SHA-1, SHA-512: Maximum data throughput



Comparison with encryption algorithms



SHA-1, SHA-512: Area



Possible improvements and extensions (1)

- **manual floorplanning and routing**

Problem: not portable among FPGA families

- **parallel processing** using
    - multiple independent execution units
    - pipelining

Problem: require multiple independent streams of data (messages, packets)

Possible improvements and extensions (2)

• **loop unrolling** of the message digest

several (2, 4, 5, or 8) message digest rounds implemented as combinational logic and executed in a single clock cycle

Problem: substantial increase in the circuit area

Conclusions

Answers to our questions

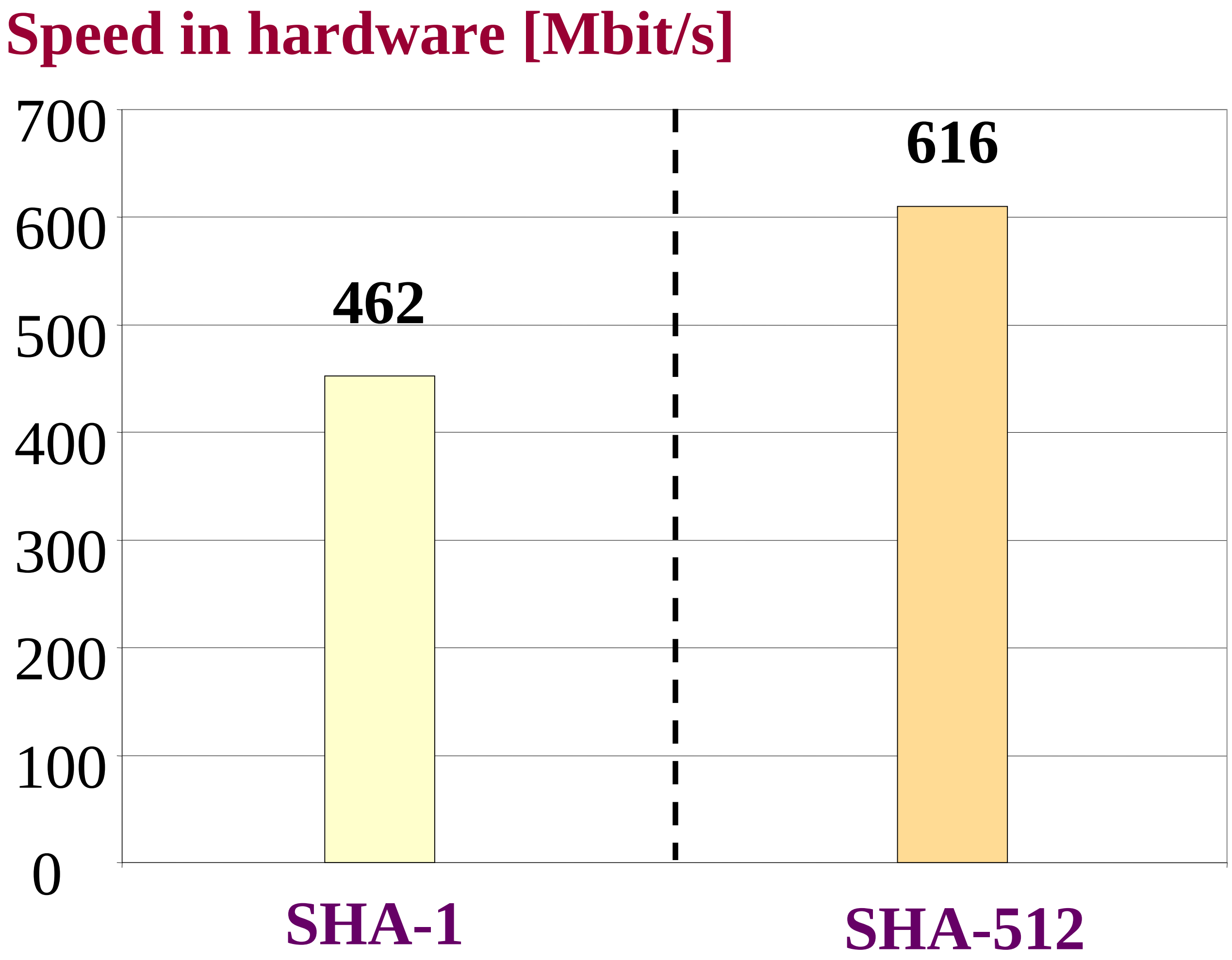
1. Does the increased security of SHA-512 come at the cost of
   - decreased speed — no, SHA-512 33% faster
   - increased area — yes, ~ 2 times
   - decreased speed to area ratio — yes, ~30%

   compared to SHA-1?

2. How does the speed of SHA-512 compares to the speed AES-256? — 60% faster

3. Can SHA-512 be implemented with the speed of 1 Gbit/s using the current generation of FPGA devices?
   - using two streams of data - yes
   - using one stream of data - to be determined

Security and hardware speed for hash functions



Complexity of the best attack	2^{80}	2^{256}
the same as	*Skipjack*	*AES-256*

Conclusions

- **Design of cryptographic hash functions does not involve a trade-off between**
  hardware speed and cryptographic security

- **More secure hash functions may require substantially more hardware resources (area, memory)**