BLAKE

2012 update

Jean-Philippe Aumasson, Luca Henzen,
Willi Meier, Raphael C.-W. Phan

Secure

Secure

Like the 4 others

BLAKE-256, 14 rounds

Hash: **2.5**-round preimage (2^{241})

Li, Xu; 2009

Perm: **8**-round distinguisher (2^{242})

Biryukov, Nikolic, Roy; 2011

BLAKE-512, 16 rounds

Hash: **2.5**-round preimage (2^{481})

Li, Xu; 2009

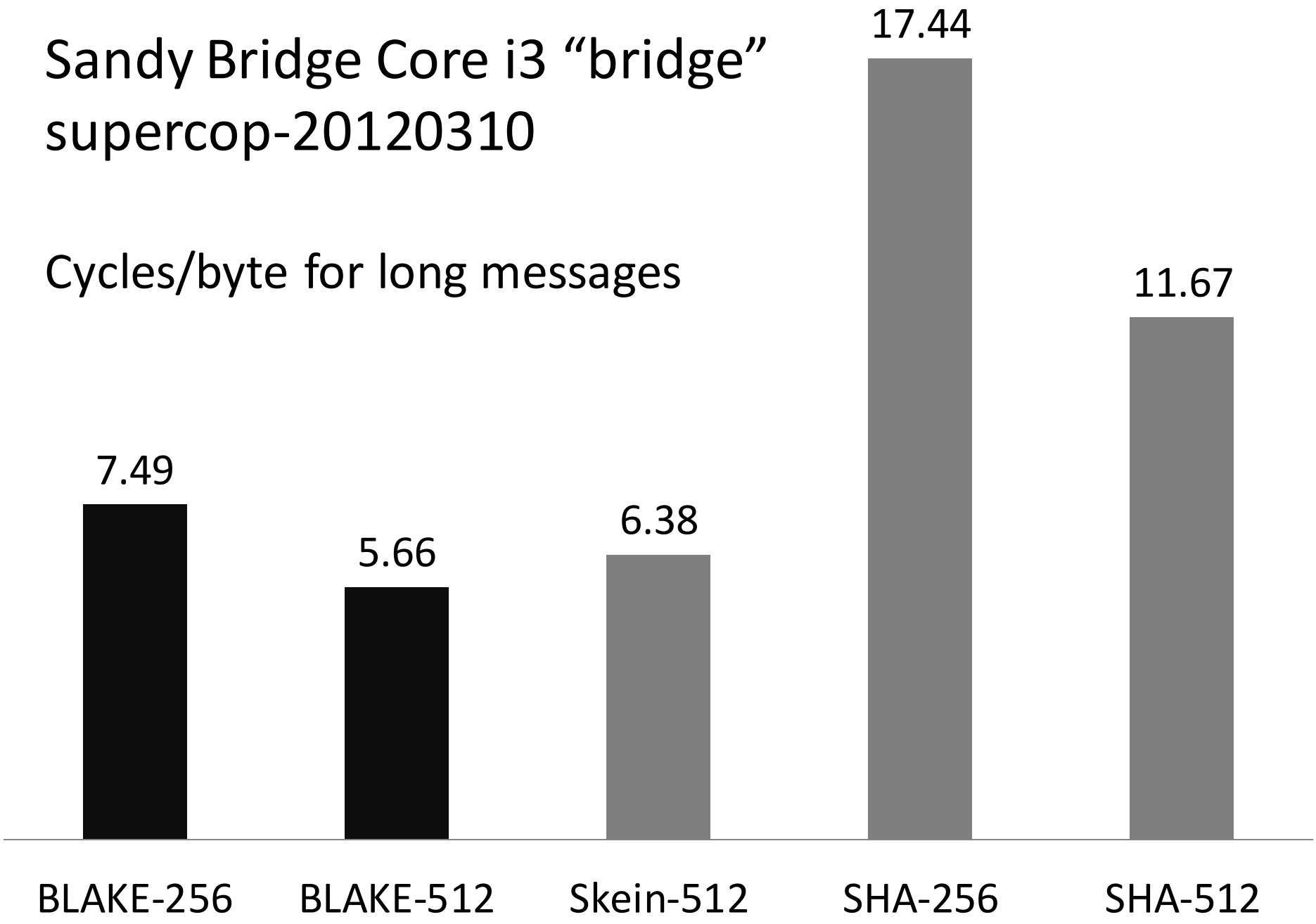
Comp: **5**-round near-collision (2^{216})

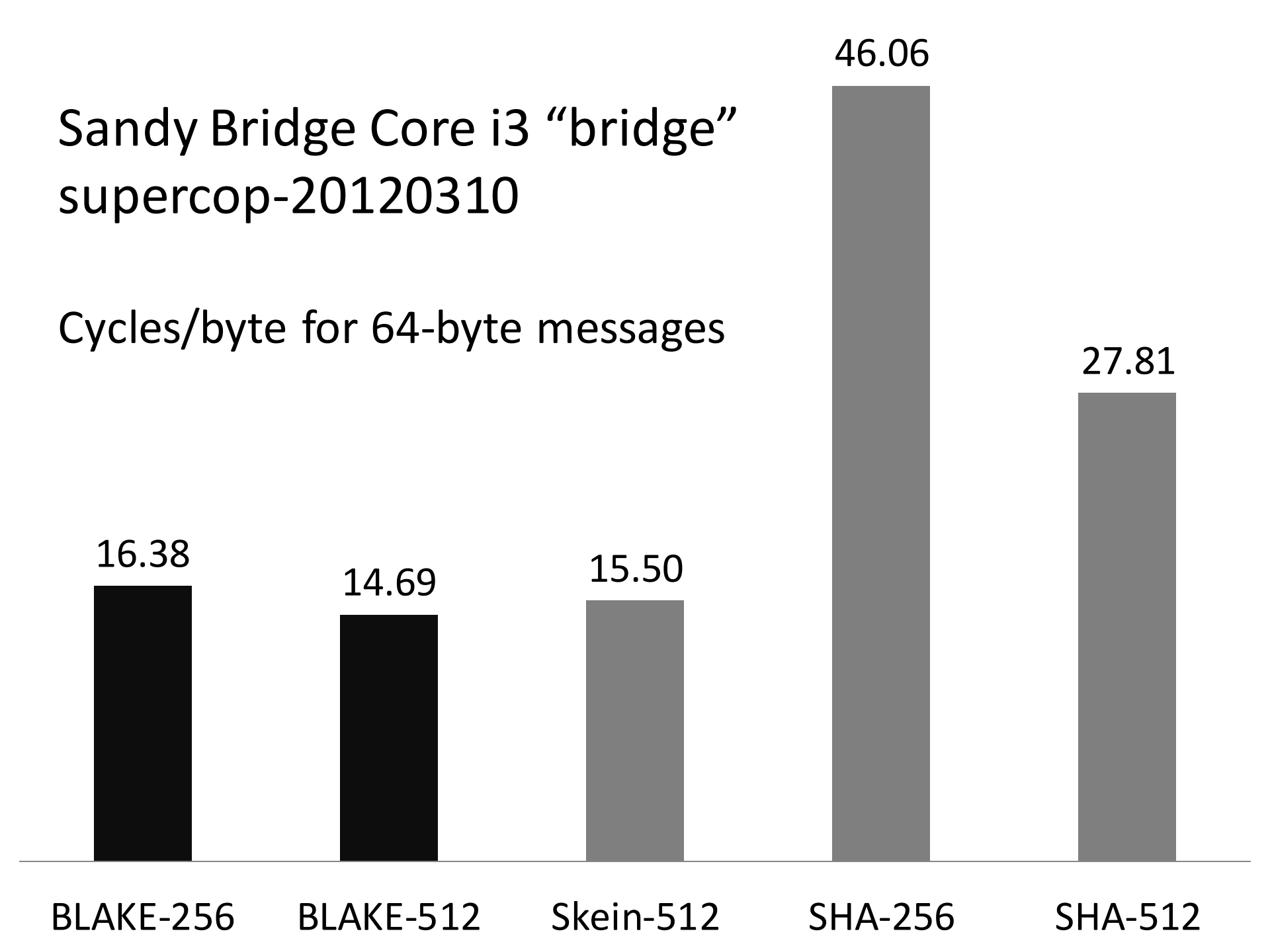
Su, Wu, Wu, Dong; 2010

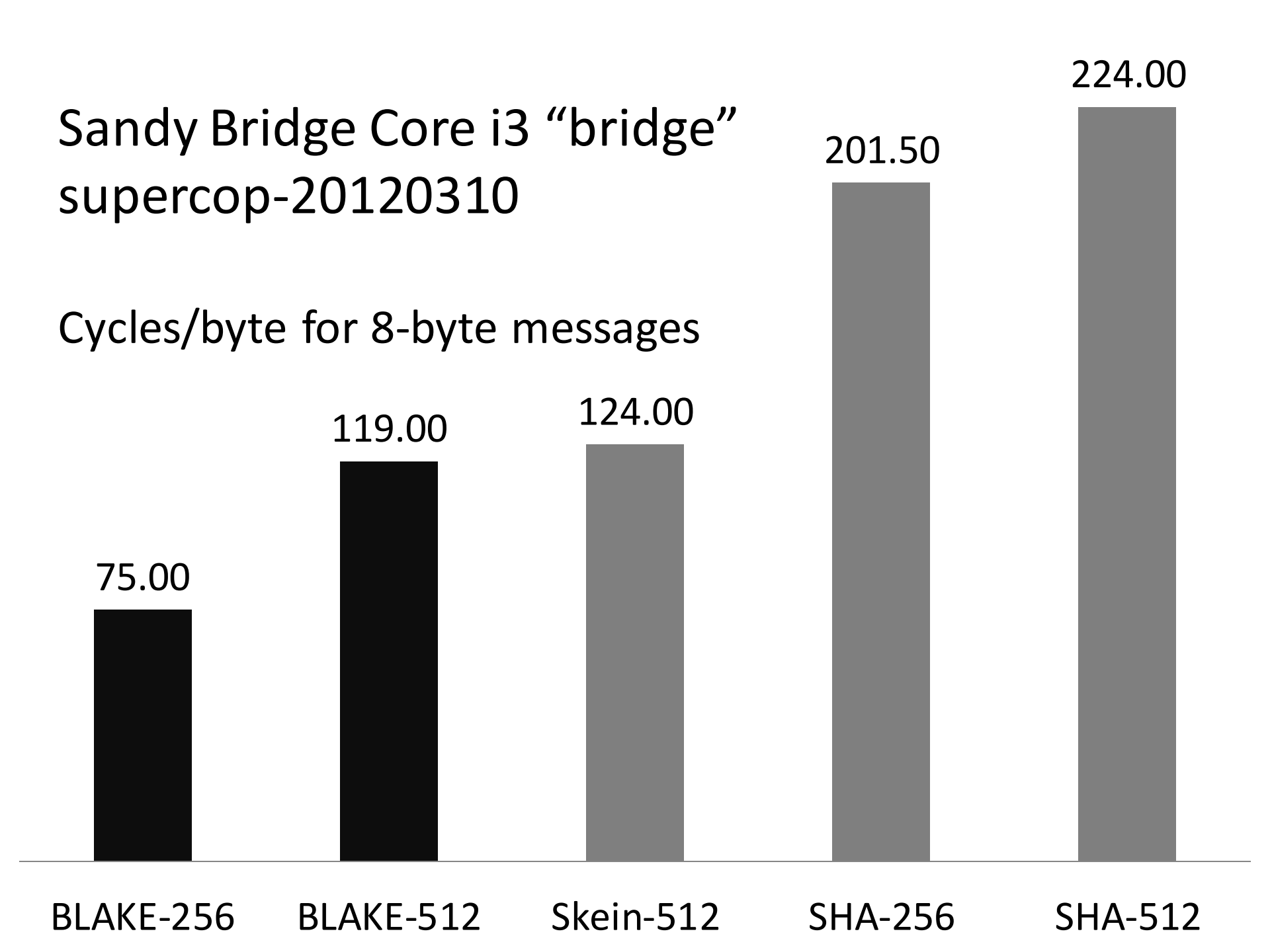
Fast

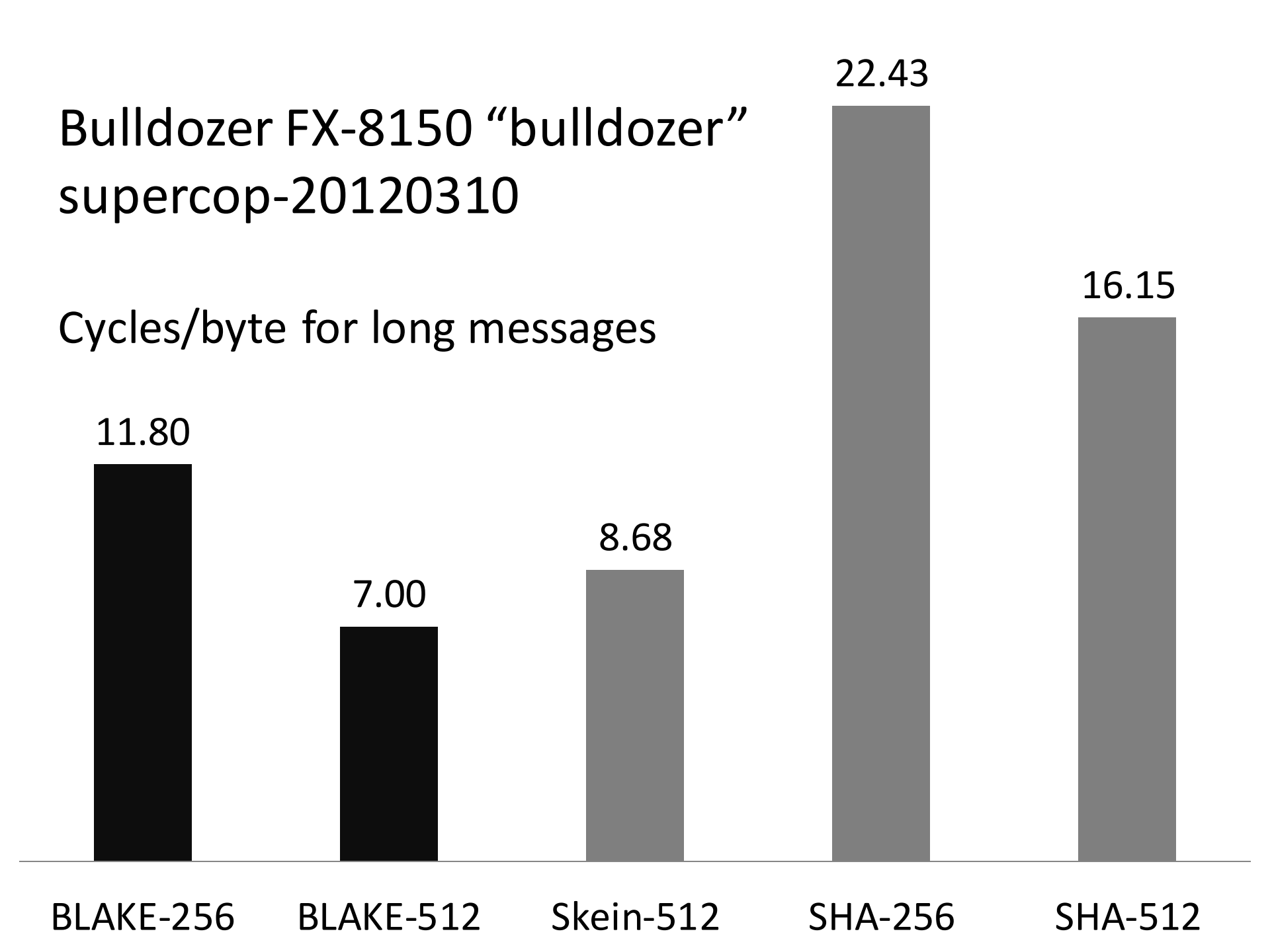
Sandy Bridge Core i3 “bridge” supercop-20120310

Cycles/byte for long messages



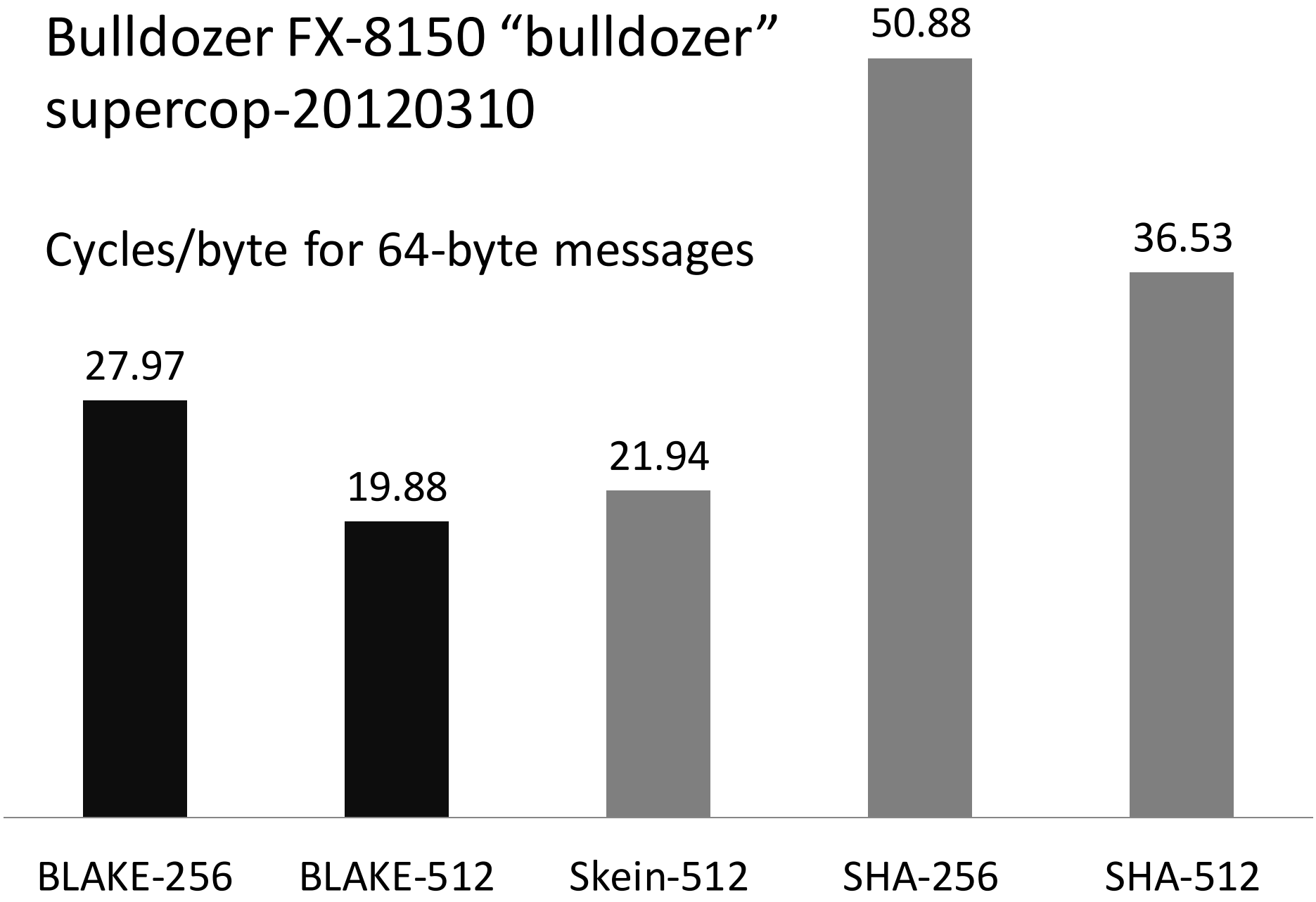

Sandy Bridge Core i3 “bridge”
supercop-20120310
Cycles/byte for 64-byte messages
16.38
14.69
15.50
46.06
27.81
BLAKE-256
BLAKE-512
Skein-512
SHA-256
SHA-512

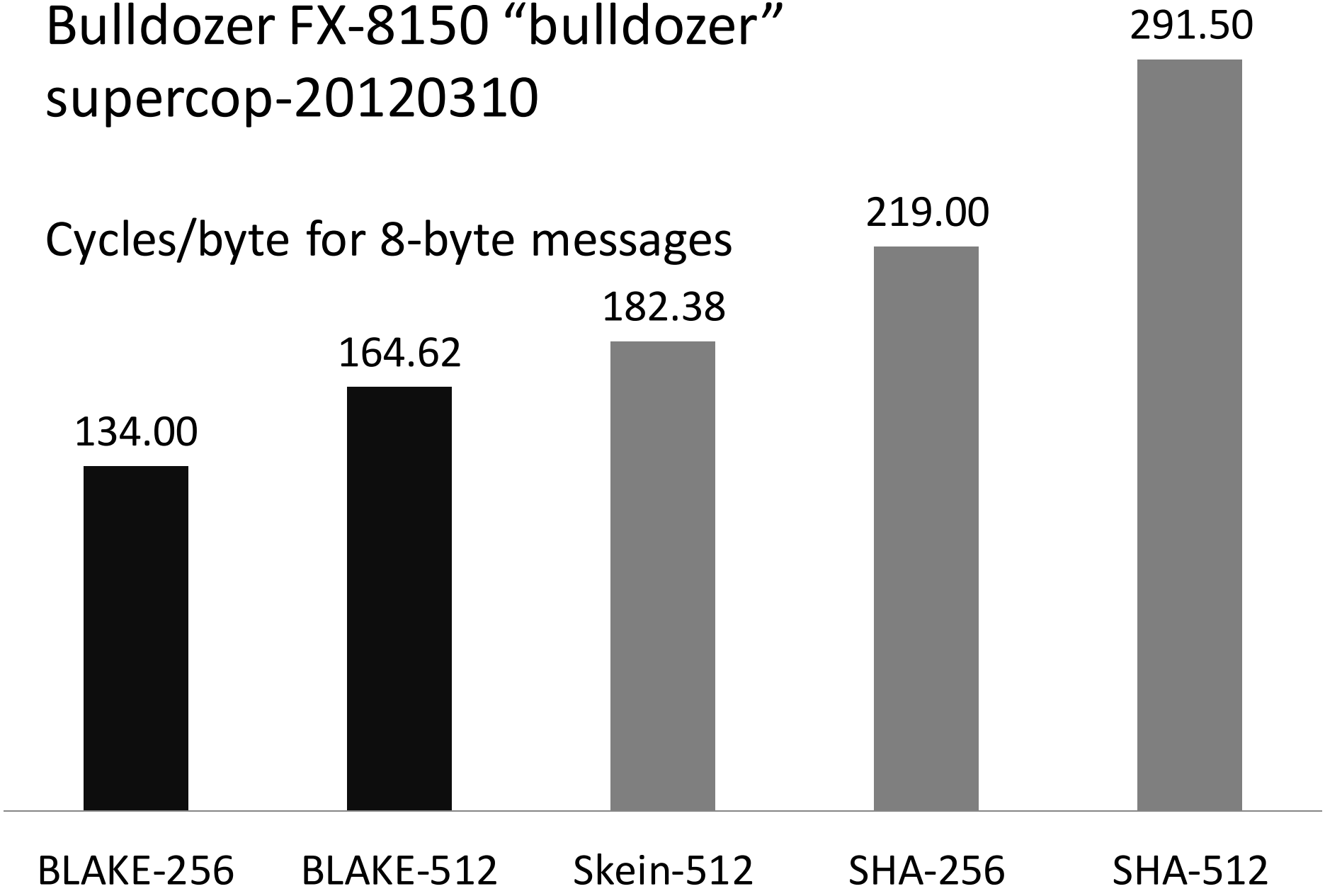

Sandy Bridge Core i3 “bridge”
supercop-20120310
Cycles/byte for 8-byte messages
75.00
119.00
124.00
201.50
224.00
BLAKE-256
BLAKE-512
Skein-512
SHA-256
SHA-512

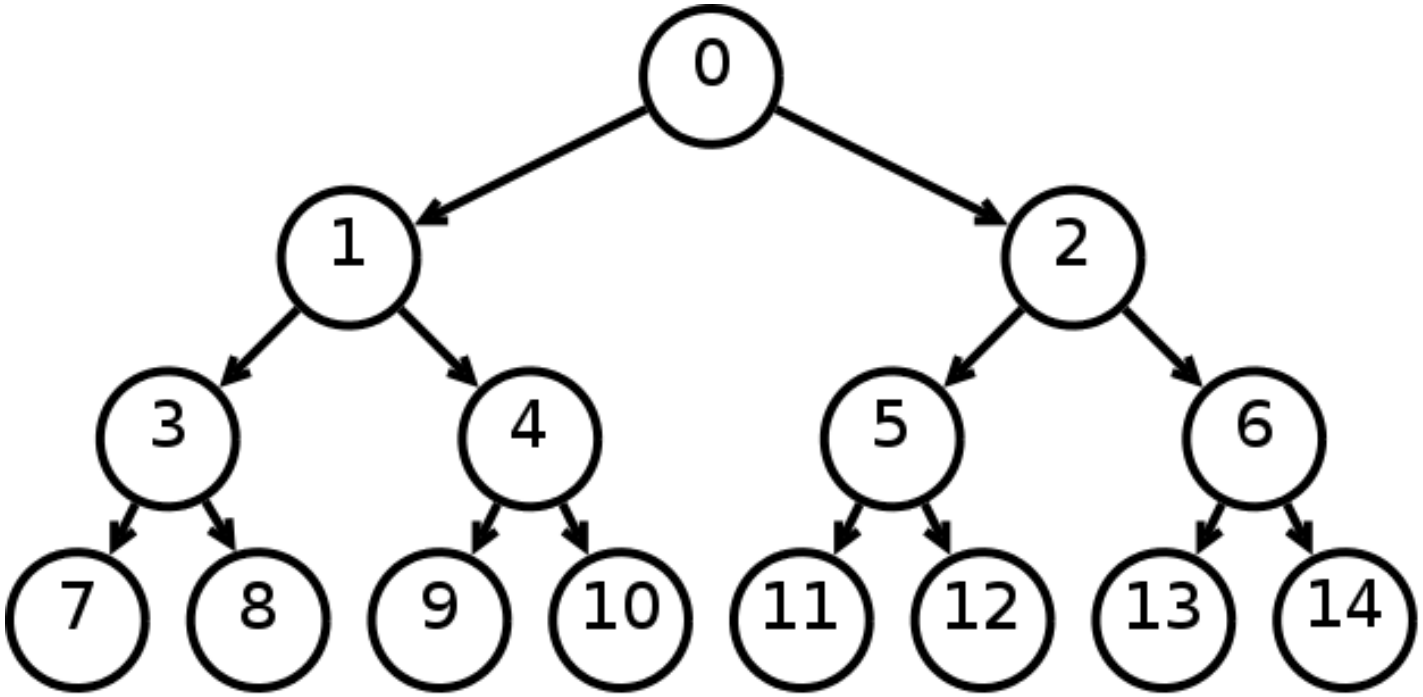

Bulldozer FX-8150 “bulldozer”
supercop-20120310
Cycles/byte for long messages
11.80
7.00
8.68
22.43
16.15
BLAKE-256
BLAKE-512
Skein-512
SHA-256
SHA-512

Bulldozer FX-8150 “bulldozer” supercop-20120310

Cycles/byte for 64-byte messages



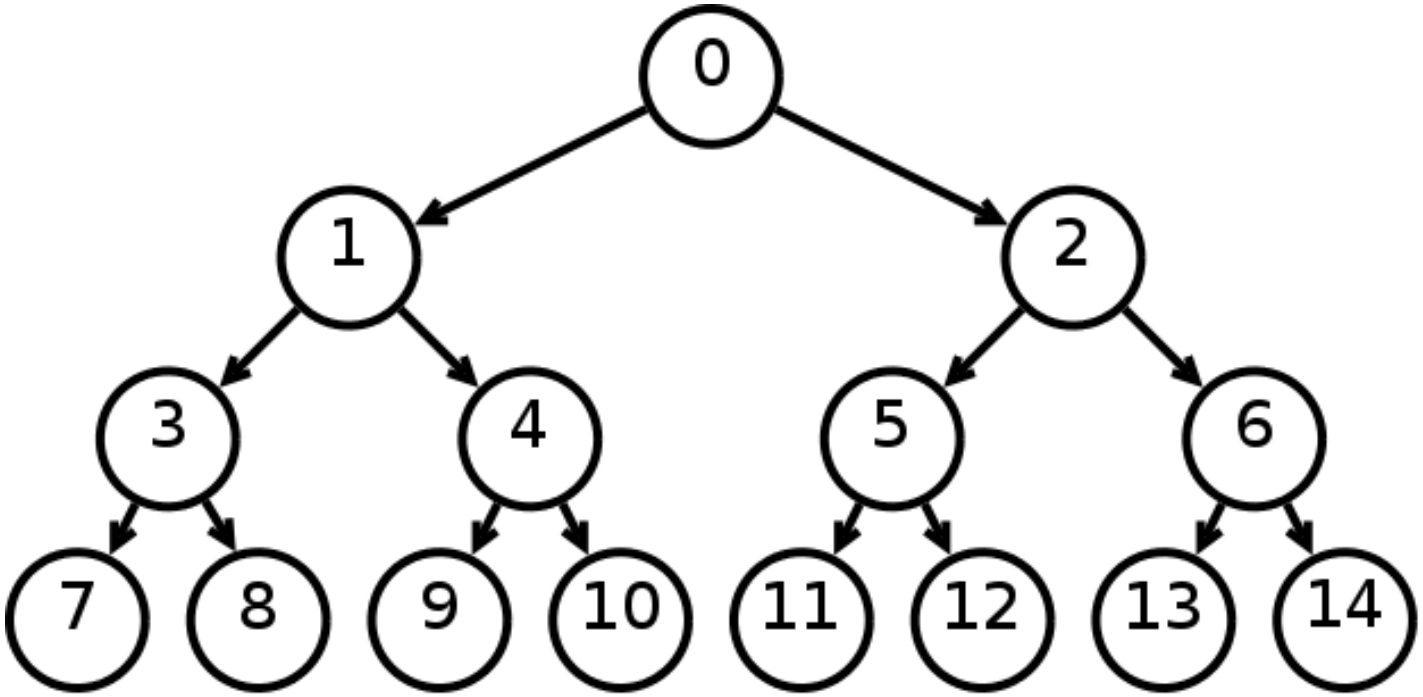

Bulldozer FX-8150 “bulldozer”
supercop-20120310
Cycles/byte for 8-byte messages
134.00
164.62
182.38
219.00
291.50
BLAKE-256
BLAKE-512
Skein-512
SHA-256
SHA-512



Single-thread treed BLAKE-256 faster than serial

(≈ 25% on “bulldozer”)

Obviously much faster when multi-threaded

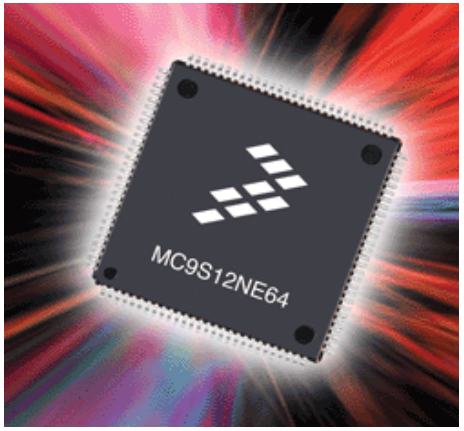


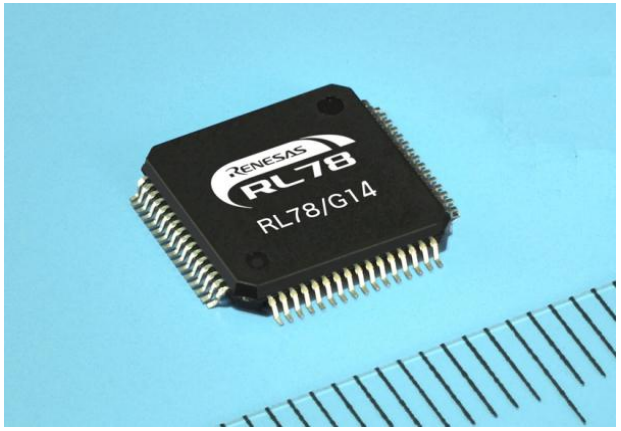
Single-thread treed BLAKE-512:

< 5.40 cycles/byte on “bulldozer”

(7.00 in serial mode)

“the low-end does not go away”





8- and 16-bit MCUs still widely used in automotive and industrial systems

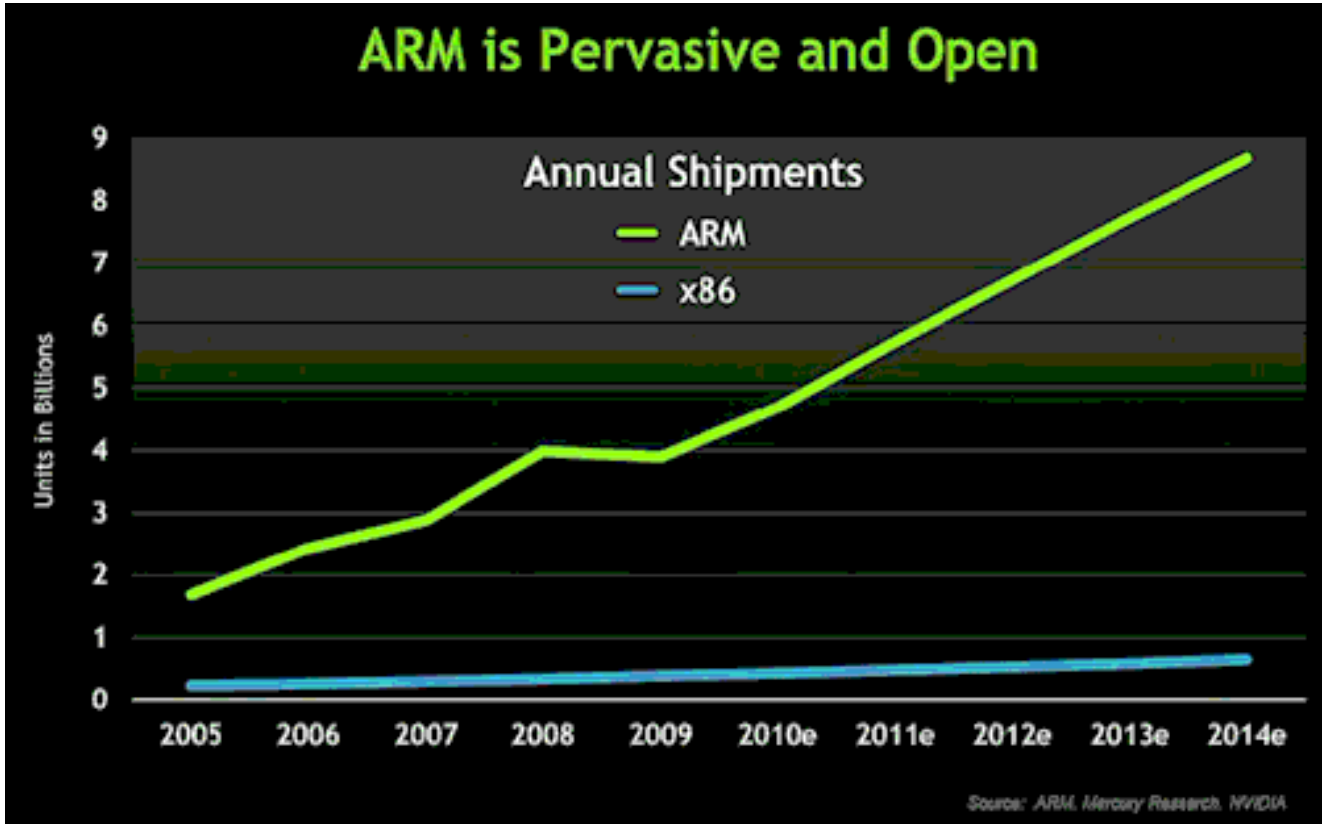
Example: RFID readers, smartmeters

BLAKE: fastest and lowest-memory on ARM cores and MCUs

Faster than SHA-2 on 8- and 16-bit (rotations by 8, 16, 32 help a lot)

Dramatic growth of embedded market

Example: ARM-based SoCs



BLAKE on 

Takes advantage of NEON extensions

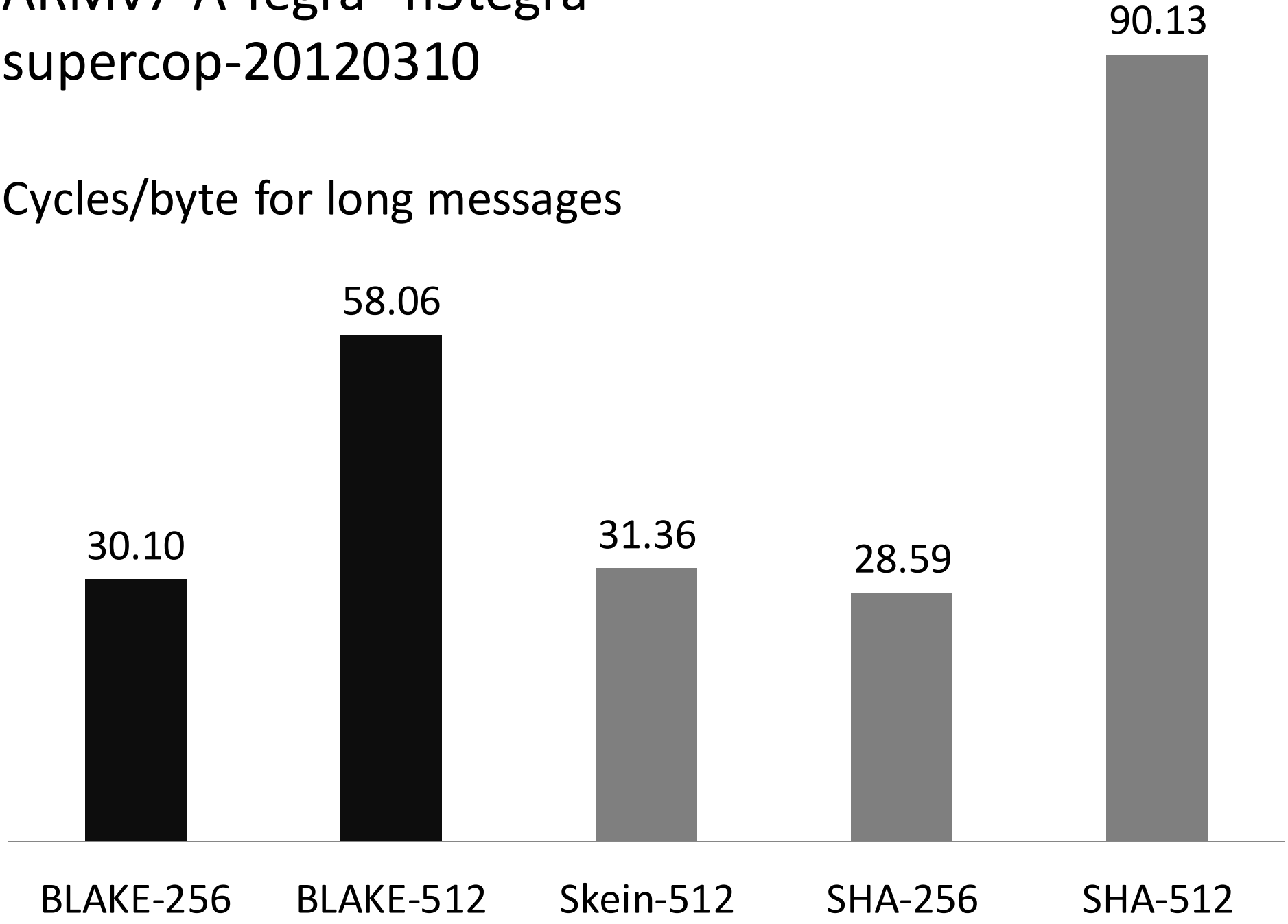
x4 speed-up of BLAKE-512 on Cortex A8

Optimizations for non-NEON cores

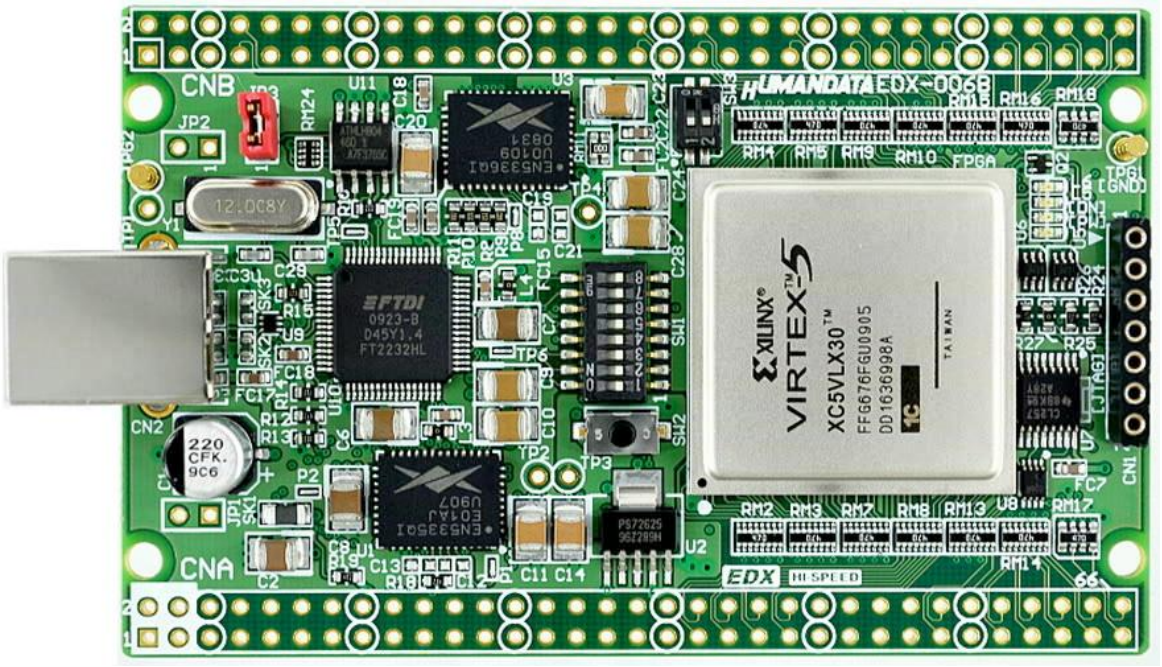
Rescheduling to exploit op-and-rotate

ARMv7-A Tegra “h5tegra” supercop-20120310

Cycles/byte for long messages



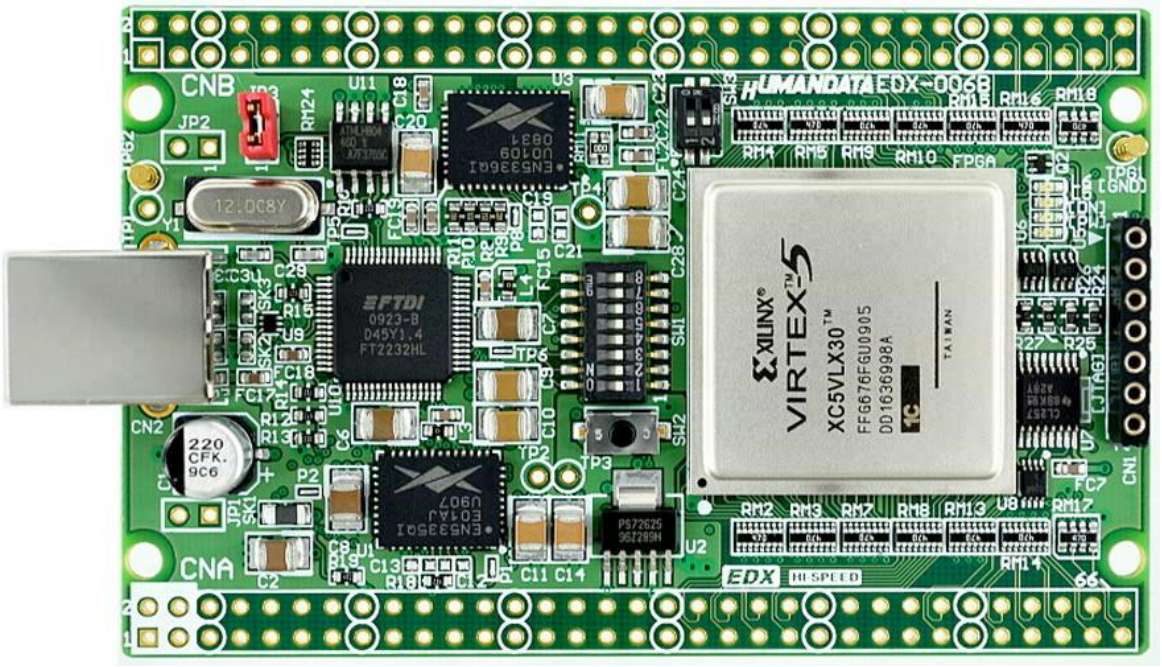
FPGA



"BLAKE is the algorithm with the **highest flexibility**, and the largest number of potential architectures. (…) It is also the only algorithm that has a relatively efficient architecture that is **smaller than the basic iterative architecture of SHA-2**."

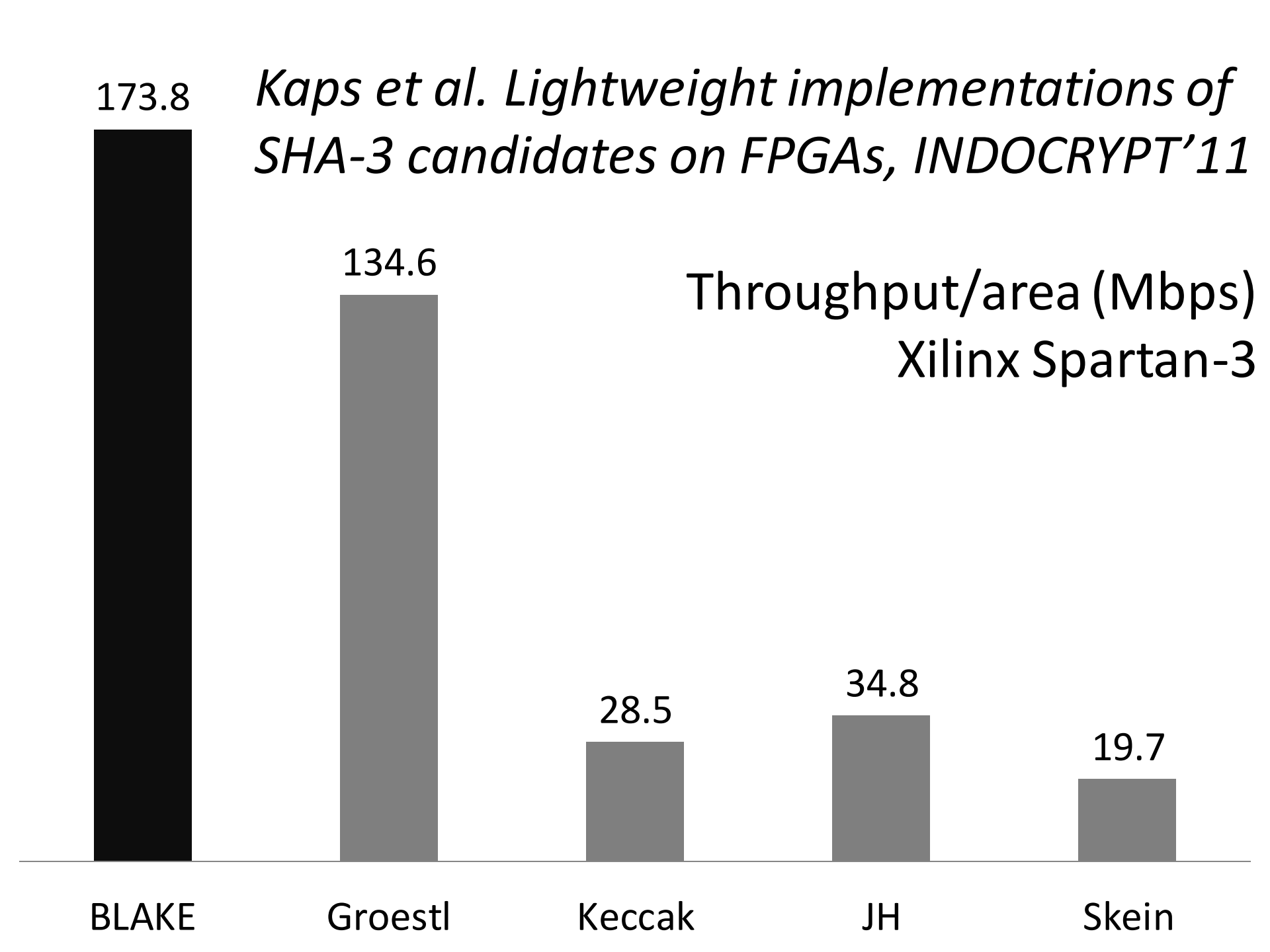
Homisirikamol, Rogawski, Gaj; CHES 2011

FPGA



"the BLAKE family offers one of the **best area-time trade-offs** and leads to the **smallest coprocessors** on reconfigurable devices."

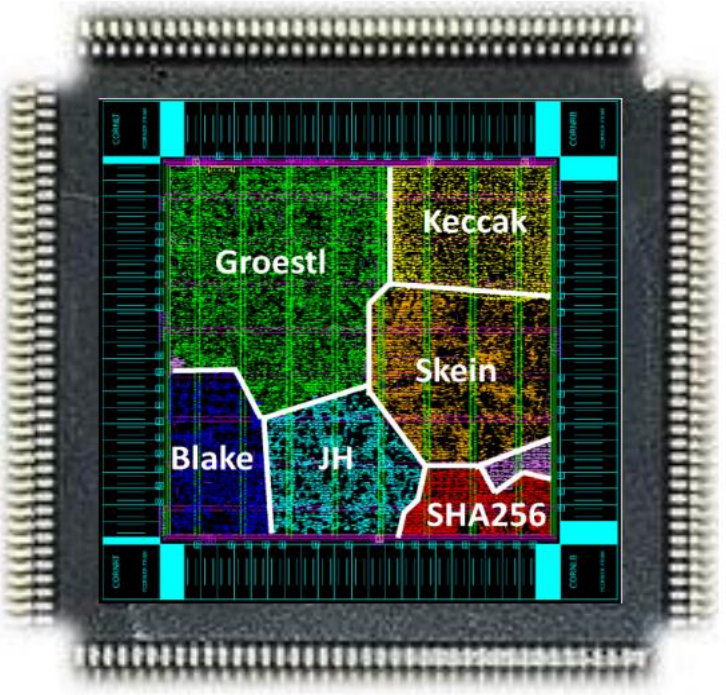
Beuchat, Okamoto, Yamazaki; FPT 2010


173.8
134.6
28.5
34.8
19.7
BLAKE
Groestl
Keccak
JH
Skein
Kaps et al. Lightweight implementations of SHA-3 candidates on FPGAs, INDOCRYPT'11
Throughput/area (Mbps)
Xilinx Spartan-3

ASIC

For optimal efficiency (throughput/area):

- Most scalable
- Most compact



130 nm: ≈2 Gbps, 34 kGE

Guo et al.; DATE 2012

Knežević et al.; IEEE T VLSI, 2011 (Credit: Guo et al.)

Simple

Example: most complex lines in Cantu's Haskell implementation

```
a' = a + b + (messageword (i2) `xor` constant (i2 + 1))
d' = (d `xor` a') `rotateR` rot0
c' = c + d'
b' = (b `xor` c') `rotateR` rot1
a'' = a' + b' + (messageword (i2 + 1) `xor` constant (i2))
d'' = (d' `xor` a'') `rotateR` rot2
c'' = c' + d''
b'' = (b' `xor` c'') `rotateR` rot3
```

https://github.com/killerswan/Haskell-BLAKE

Already many third-party implementations:

ARM11 asm	Schwabe, Yang, Yang	Java	Pornin
ARM thumb2	Wenzel-Benner	Java	Greim
AVR asm	von Maurich	Javascript	Drost
C (for AVR)	Otte	Matlab	Burgess, Jelley, Smith, Weston
C (HMAC)	Lazar		
C#	Reichl	Matlab	Steer
CUDA	Bos, Stefan	Perl	Gray
Go	Chestnykh	PHP	Correa
Haskell	Cantu	Python	Bugbee

+ eBASH C implementations by Bernstein, Kirst, Leurent, Neves, Pornin, Schwabe

Thanks to all!

Conclusion: BLAKE is...

Secure

with a LARGE margin

Fastest or close-second on desktops & servers

Best performer in embedded systems

Most flexible & scalable in FPGA and ASIC

Implementers-friendly

Thank you!

Thanks D.C. to show support!

(1025 Connecticut Av NW)