F5-2: (cryptographic) hash function

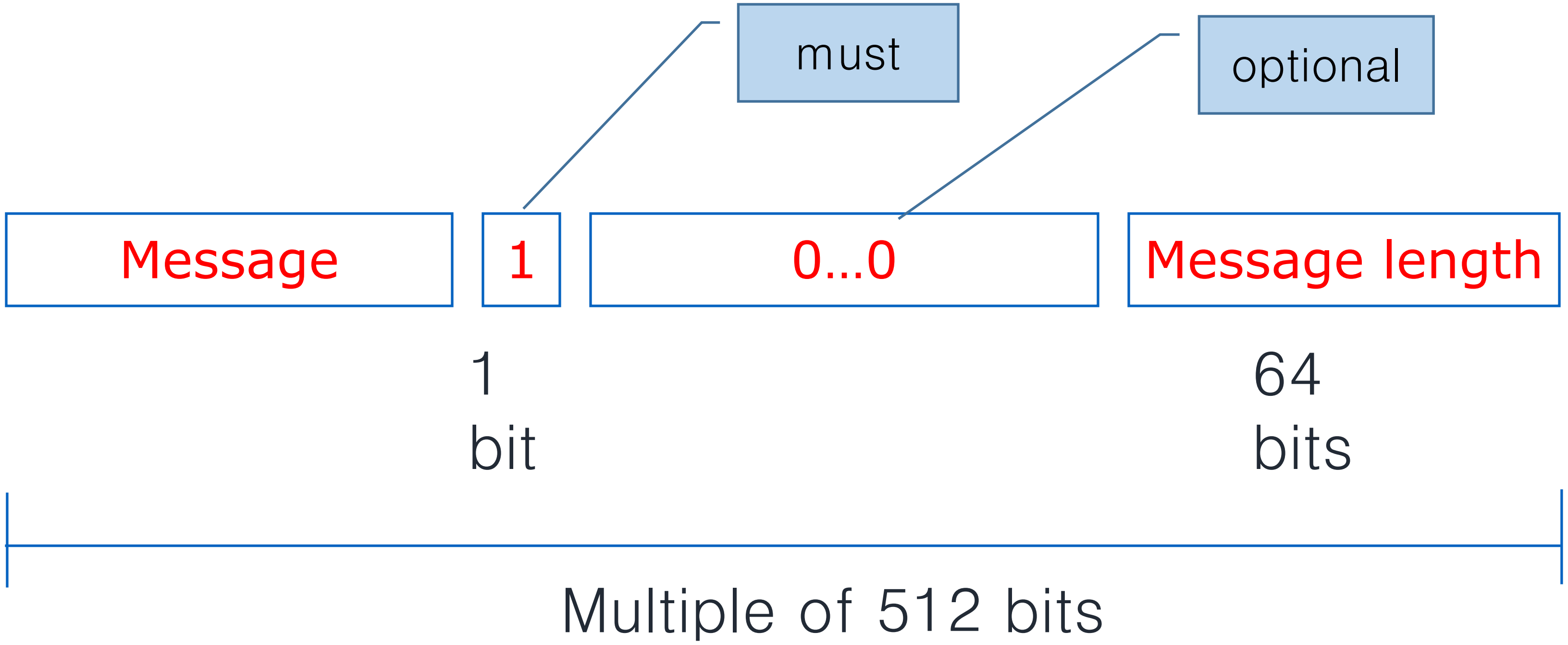
tkkwon@snu.ac.kr

some slides from Dan Boneh@Stanford

SHA-1

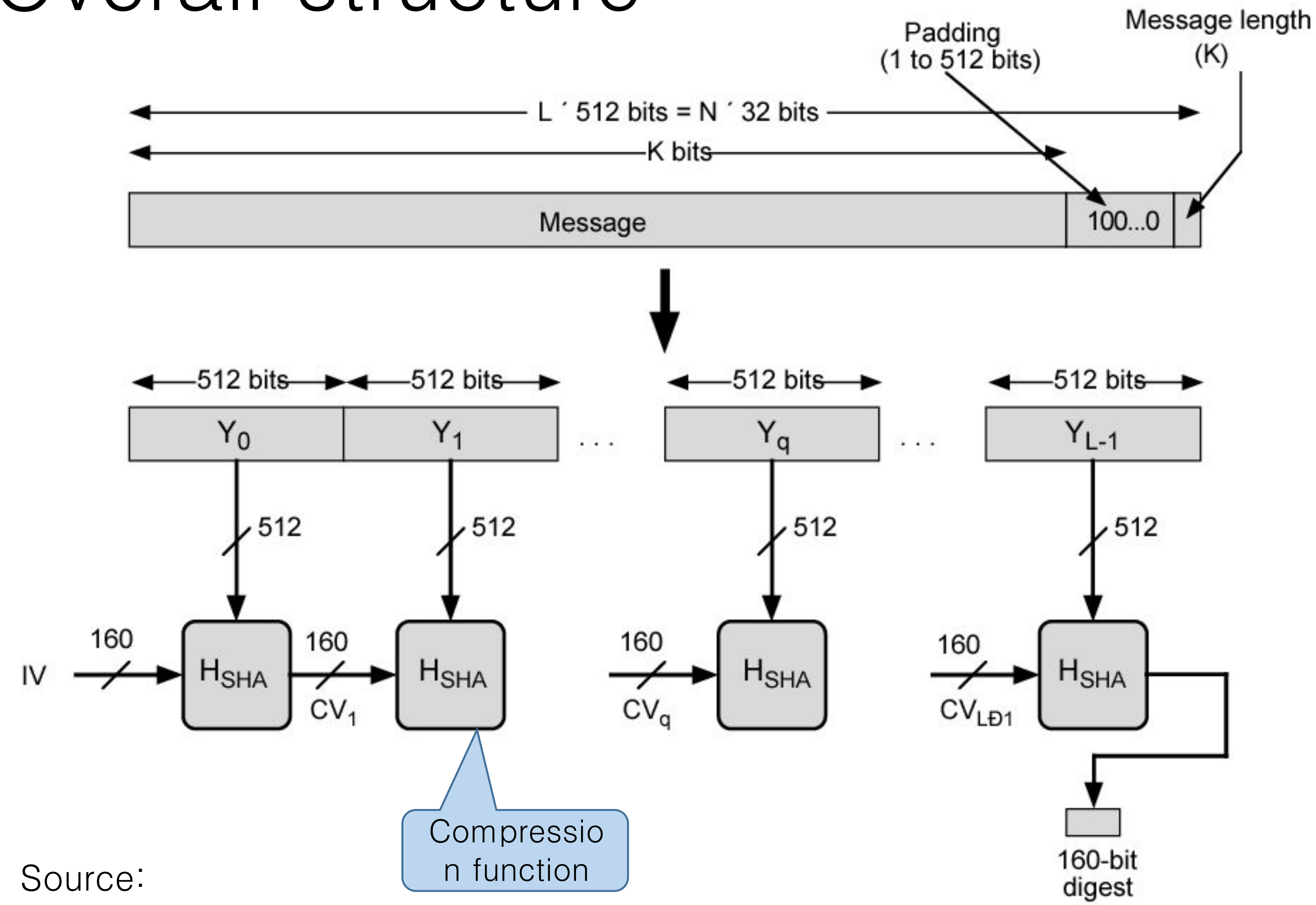
- Secure hash algorithm 1
- It was published as FIPS PUB 180-1 by NSA in 1995
- produces 160-bit hash values
- Merkle-Damgård + Davies-Meyer structure
- It is not recommended for use since 2005
- Microsoft, Google, Apple and Mozilla have all announced that their respective browsers stop accepting SHA-1 SSL certificates starting from 2017

Padding first



If a message is 512k+447 bit long: k+1 blocks
If 512k+448 bit long: k+2 blocks

Overall structure



Compression function

Source: Stallings

Some notation and terminology

- Digest Length = 160 bit
- Message Block = 512 bit
- Sub-block (or word) size = 32 bit
- 512/32 = 16 total Sub-blocks (or words)
- No. of Rounds = 4
- 80 iterations (t:0~79): (# of Rounds = 4) X (Iterations per round = 20)
- Chaining Value (CV) = 5*32=160 bits =[A,B,C,D,E]
- K[t] = constants per round (32 bits each where t=0 to 79)
- Output: five 32-bit sub-blocks

*CV: chain variable

SHA-1 Overview (1/2)

1. Padding: Length of the message is 64 bits short of multiple of 512 after padding (bit sequence 100…0).
2. Append: a 64-bit length value of original message is taken.
3. Divide the input into 512-bit blocks
4. Initialize CV (i,.e. CV_0): 5-word (160-bit) buffer (A,B,C,D,E) to

(A=01 23 45 67,
B=89 AB CD EF,
C=FE DC BA 98,
D=76 54 32 10,
E=C3 D2 E1 F0)

Nothing Up My Sleeve numbers

4A. Constants in a compression fn.

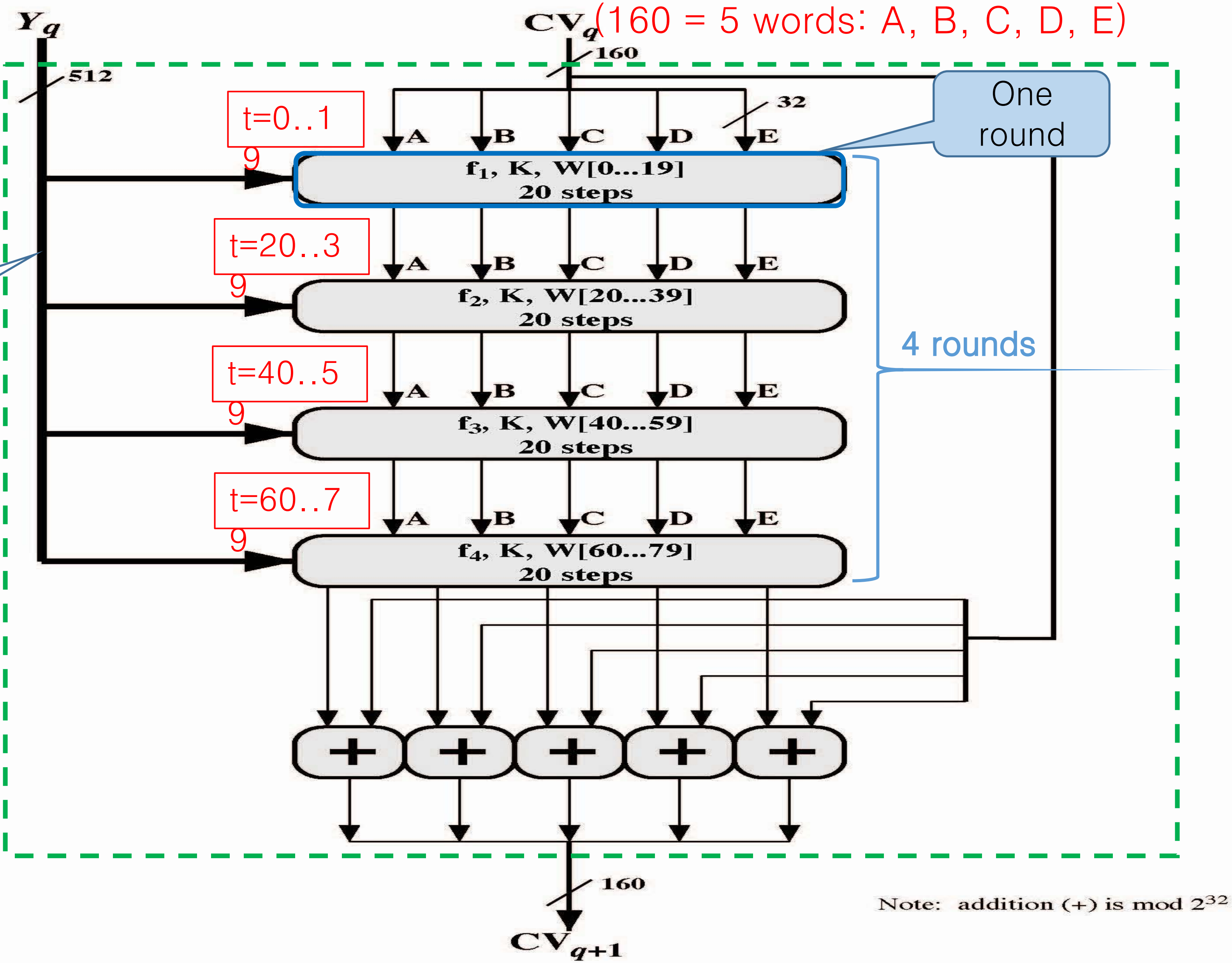
$K_0 - K_{19}$ = 5A827999
$K_{20} - K_{39}$ = 6ED9EBA1
$K_{40} - K_{49}$ = 8F1BBCDC
$K_{60} - K_{79}$ = CA62C1D6

SHA-1 overview (2/2)

5. Process Blocks: now the actual algorithm begins. message in 16-word (512-bit) chunks:
   - Copy CV into a single buffer for storing temporary intermediates as well as the final results.
   - Divide the current 512-bit blocks into 16 sub-blocks (W[0]..W[15]), each consisting of 32 bits.
   - Has # of rounds=4, each round consisting of 20 bit/step iteration operations on message block & buffer
   - expand 16 words into 80 words (W[0..79]) by mixing & shifting.
   - K[t] is one of 4 constants depending on iteration t ranging 0..79
   - Form a new buffer value by adding output to input.
6. output hash value is the final buffer value

q–th msg block
(512 = 16
words)

q–th chaining value
(160 = 5 words: A, B, C, D, E)



t=0..19

t=20..39

t=40..59

t=60..79

One round

4 rounds

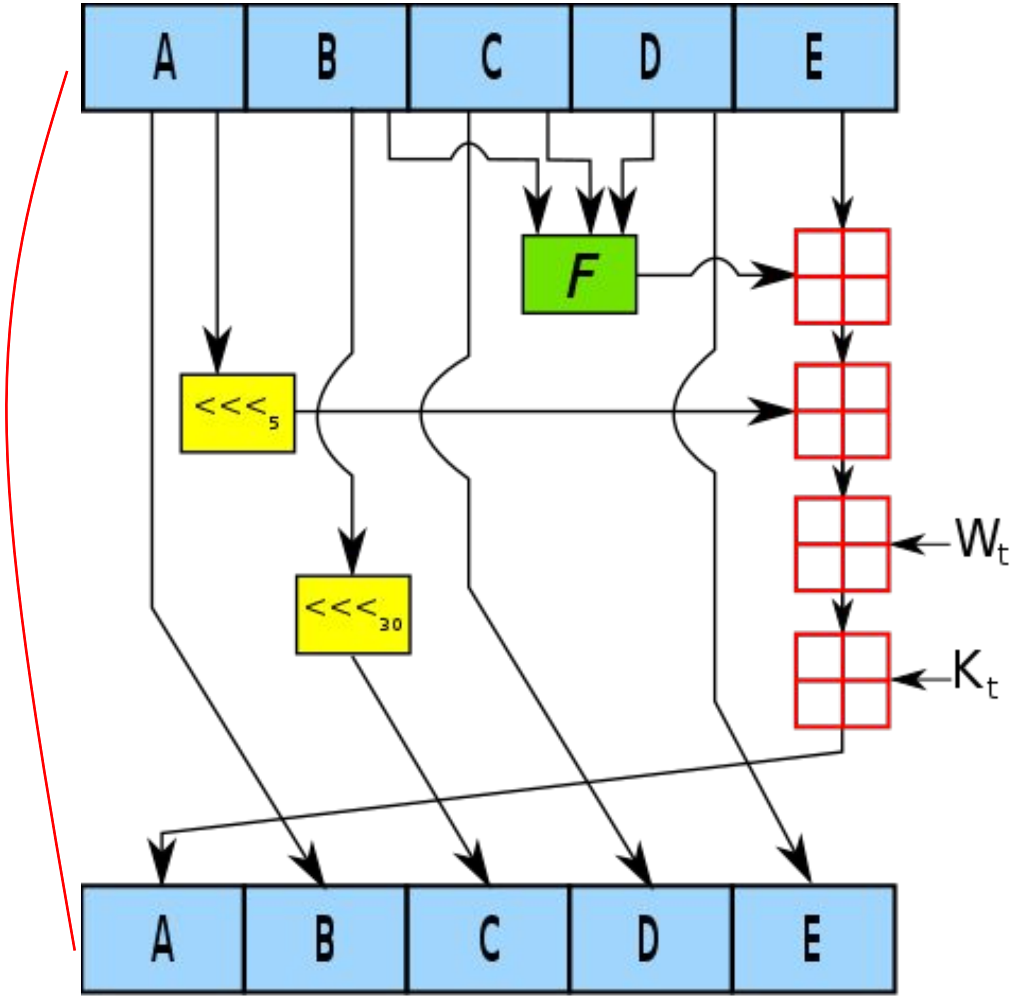
W[0..15]
→W[0..79]

Compression function

Figure 12.5 SHA-1 Processing of a Single 512-bit Block (SHA-1 Compression Function)

f_r is some function of some inputs

one round in a Compression Function (SHA−1)



Source: wikipedia

A, B, C, D, E (of CV): each 32-bit word of the state;
F (= *f*$_t$) is a nonlinear fn. that varies at each round;
<<<$_n$ denotes a left bit rotation by *n* bits;
W$_t$ is the expanded message word of round/step t;
W$_t$ is the incoming msg block when t<16;
K$_t$ is the constant that varies at each round;
⊞ denotes addition modulo 2^{32}.

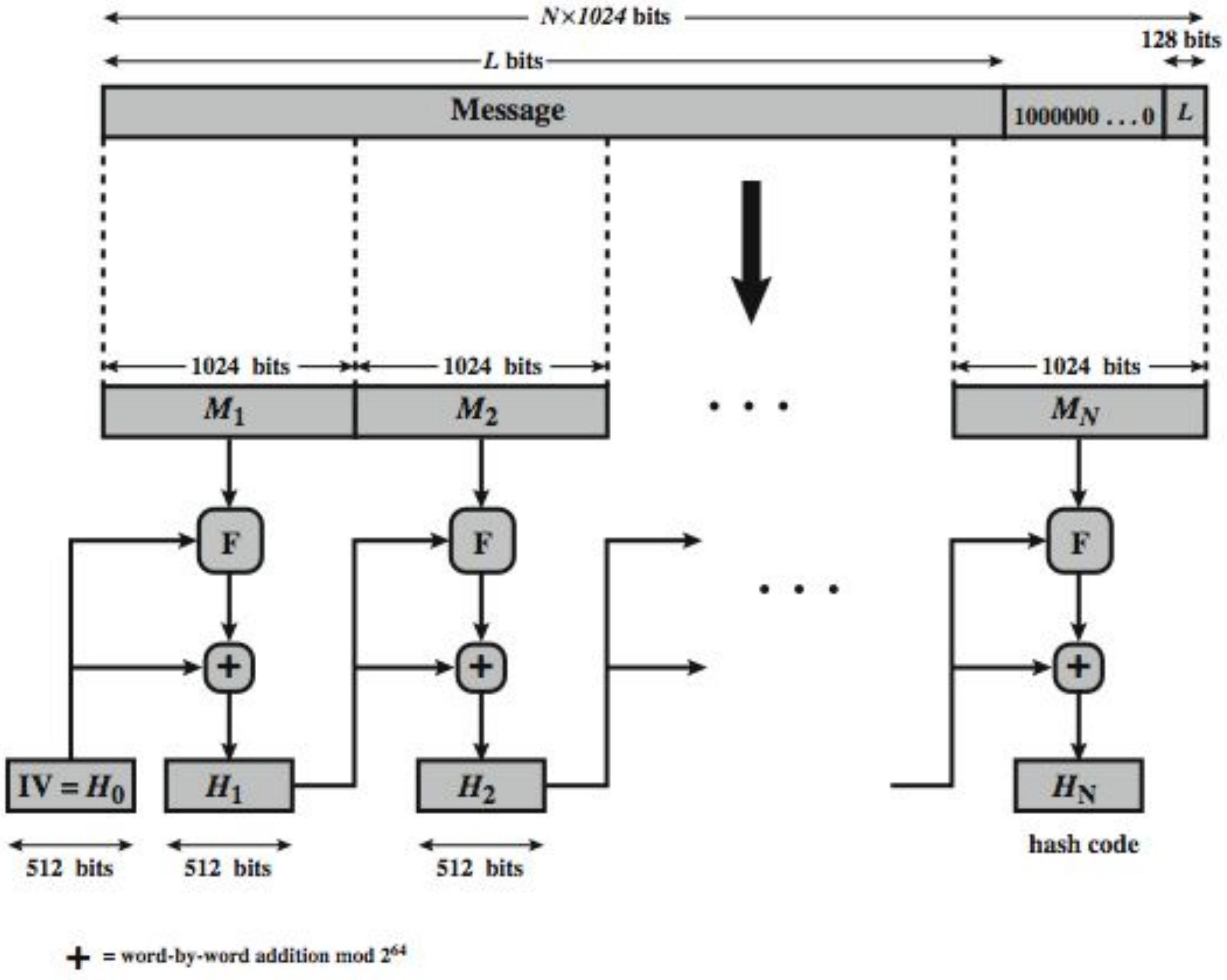
- $W_t = (W_{t-3} \oplus W_{t-8} \oplus W_{t-14} \oplus W_{t-16}) \ll 1$ 16≤t≤7

$$f_t(B,C,D) = \begin{cases} (B \wedge C) \vee ((\neg B) \wedge D) & \text{if } \ 0 \leq t \leq 19 \\ B \oplus C \oplus D & \text{if } 20 \leq t \leq 39 \\ (B \wedge C) \vee (B \wedge D) \vee (C \wedge D) & \text{if } 40 \leq t \leq 59 \\ B \oplus C \oplus D & \text{if } 60 \leq t \leq 79 \end{cases}$$

SHA-2

- Published by NSA in 2001
- SHA-2 family consists of six hash functions with digests (hash values) that are 224, 256, 384 or 512 bits: SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, SHA-512/256
- Great increase in mixing between bits of the words compared to SHA-1
- Still Merkle-Damgård structure, subject to length-extension attacks
- F is the compression function

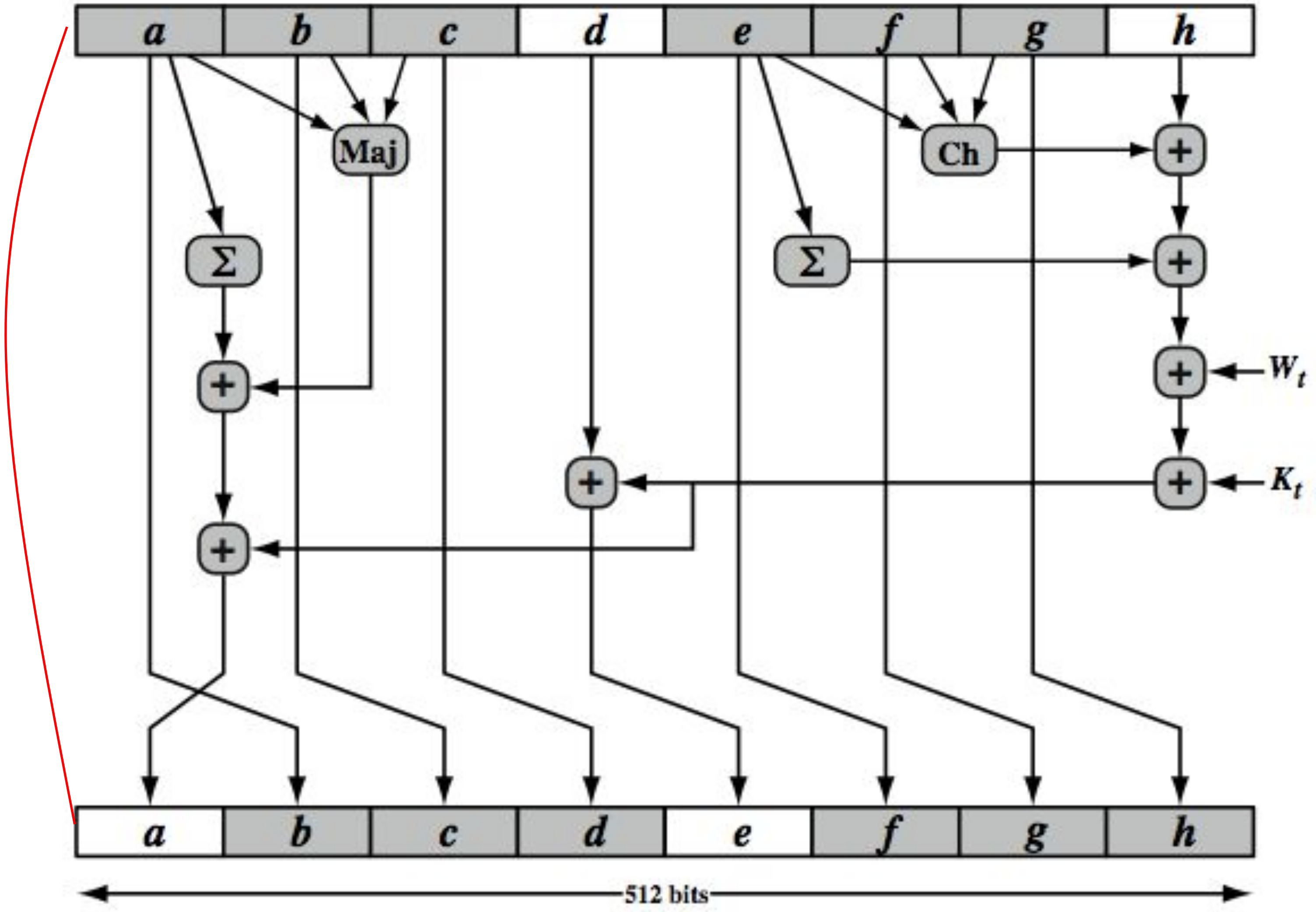
SHA-512



SHA-512 (F: compression fn.)

- heart of the algorithm
- processing message blocks (each 1024 bits long)
- Each CV (and digest) is 512 bits: 8 words each 64 bits long
- consists of 80 rounds (t)
  - updating a 512-bit buffer
  - using a 64-bit value W_t derived from the current message block
  - and a round constant based on cube root of first 80 prime numbers
- compared with SHA-1
  - block and word sizes are doubled
    - 512→1024, 32→64
  - CV/IV is more than tripled
    - 160→512

SHA-512: compression fn. F



Ch(e,f,g) = (e ∧ f) ⊕ (¬ e ∧ g)

Maj(a,b,c) = (a ∧ b) ⊕ (a ∧ c) ⊕ (b ∧ c)

$\sum(a) = <<<_{28}(a) \oplus <<<_{34} (a) \oplus <<<_{39}(a)$

$\sum(e) = <<<_{14}(e) \oplus <<<_{18} (a) \oplus <<<_{41}(a)$

⊞ = addition modulo 2^64

K_t = a 64-bit additive constant

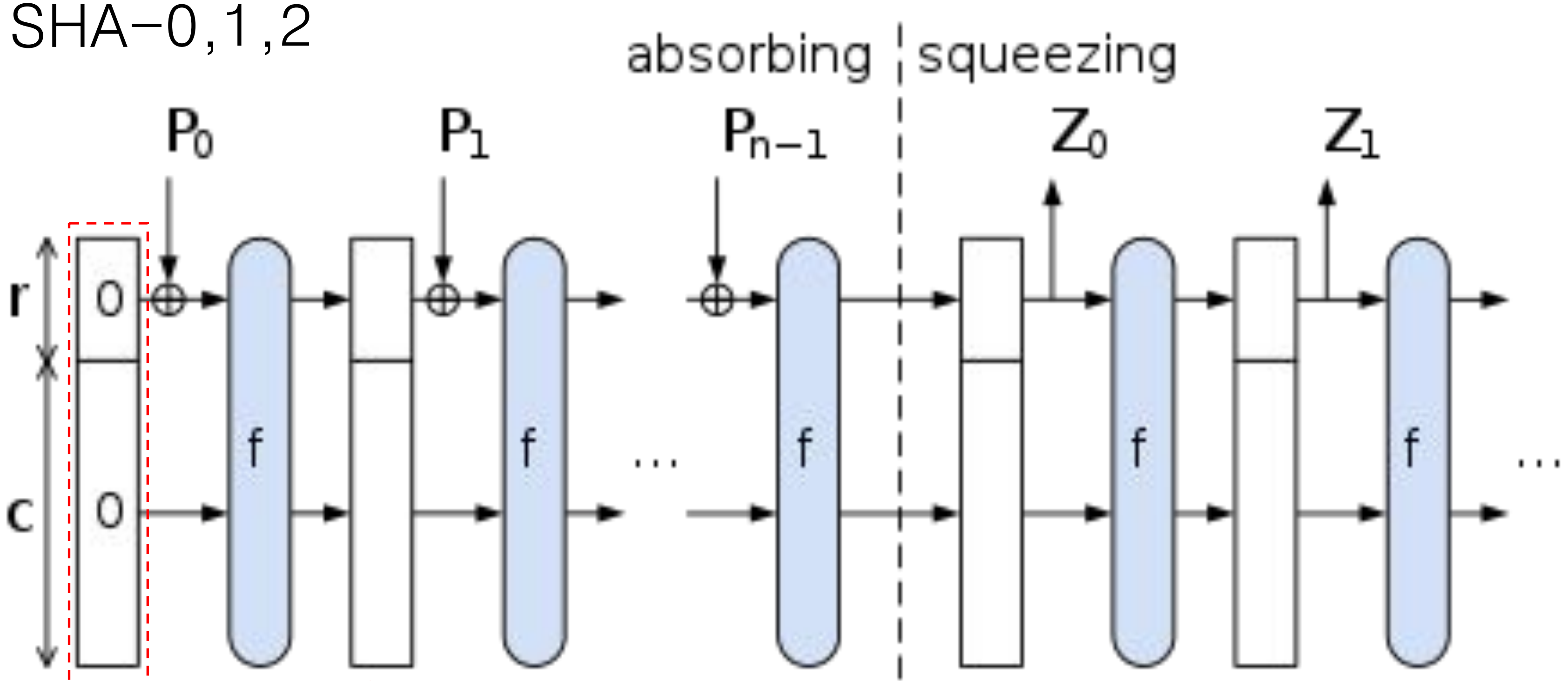
W_t = a 64-bit word derived from the current 1024-bit input block.

* Ch(): choose depending on e
* Maj(): majority of inputs

SHA-3

f: permutation fn. that uses xor, and & not operations
r: size of the part of the state that is combined with message blocks
c: size of the part of the state that is not combined with message blocks
P_i: input (r bits each)
Z_i: output (r bits each)
S (state): a block of b bits, b=r+c, initial state is all 0 bits.

- NIST, 2015.
- SHA-3 is a subset of the broader cryptographic primitive family: Keccak
- No Merkle–Damgård structure
- Different structure from SHA-0,1,2
- sponge construction



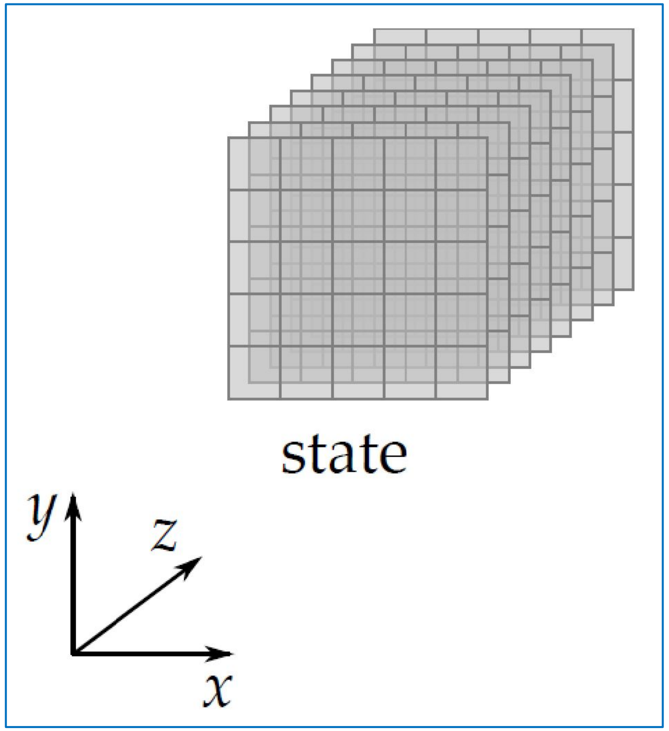
state

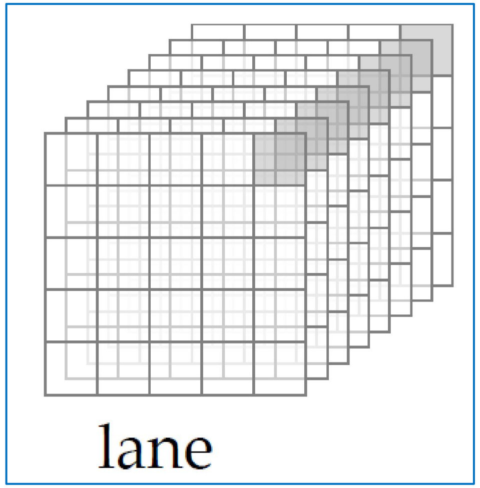
Source:

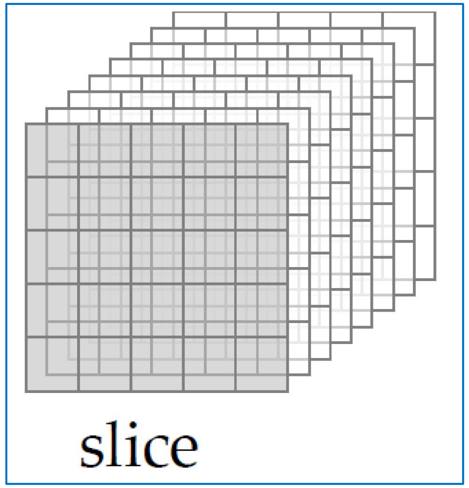
SHA-3

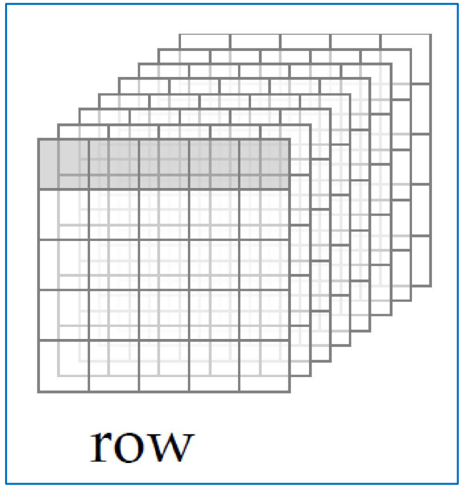
Keccak: l is 0..6
SHA-3: l = 6

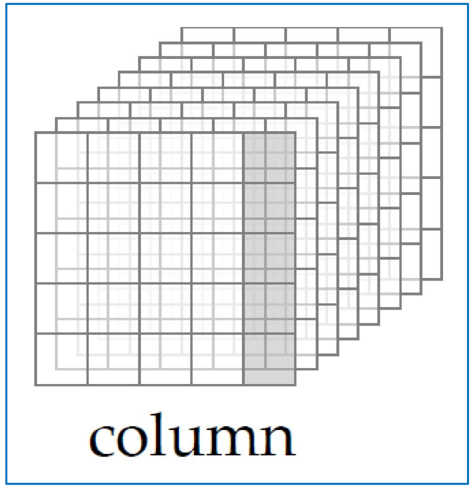
- State: $5*5*2^l$, 2^l=1,2,4,⋯,32,64
- lane, slice,⋯



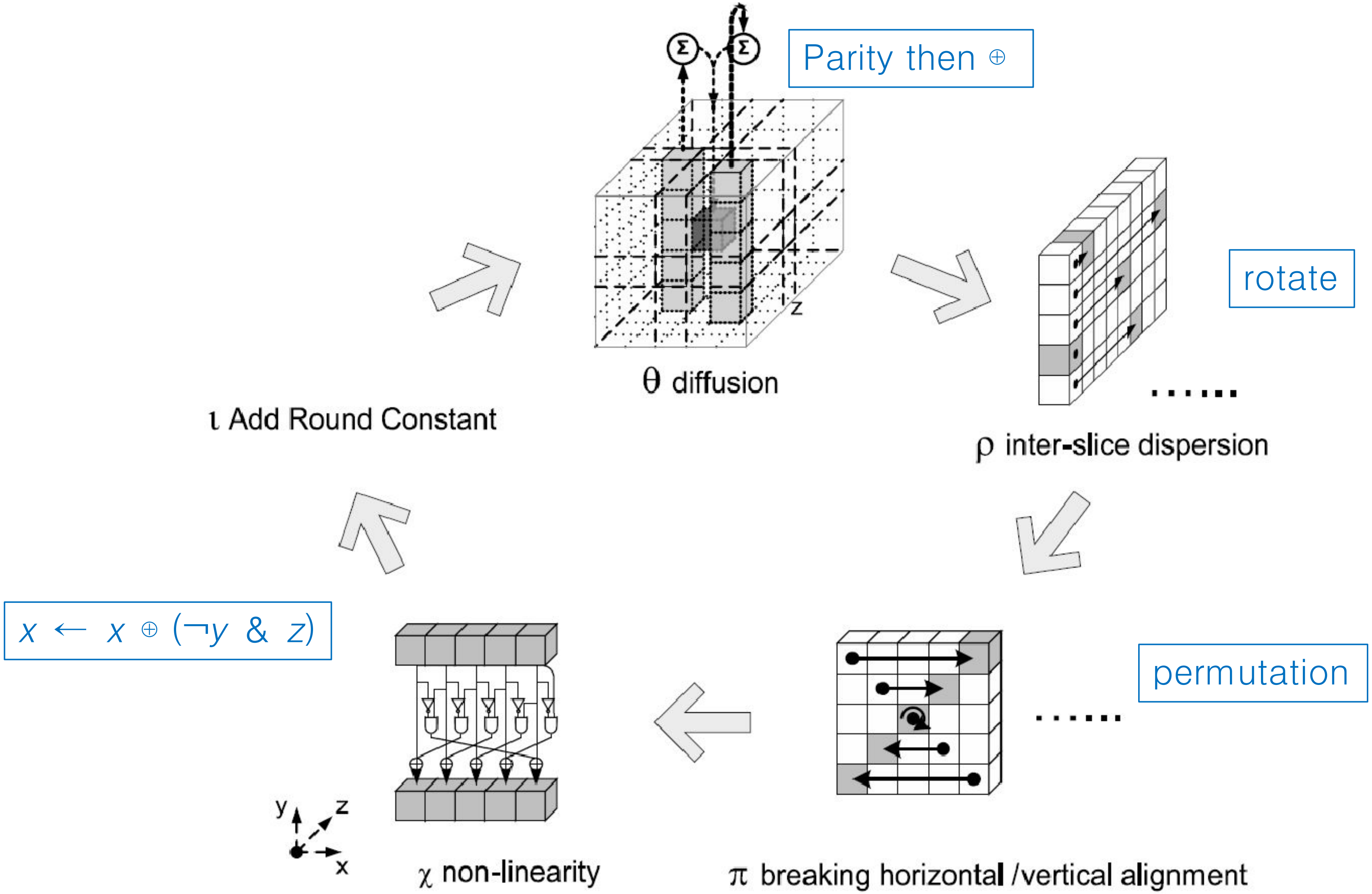








A high-level view of the permutation fn. f



Parity then ⊕

rotate

permutation

$x \leftarrow x \oplus (\neg y \ \& \ z)$