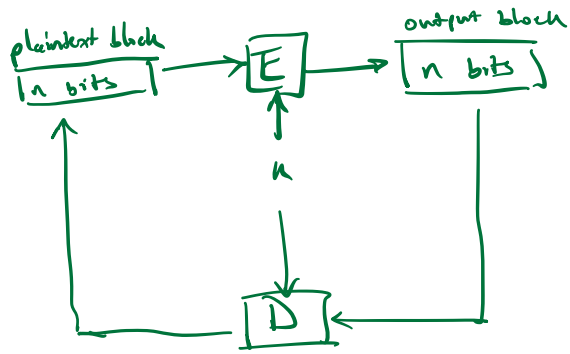
Block Ciphers

pair of efficient algos (E, D)



ex. 3DES : $n = 64$ bit blocks, key size: 128 bits
AES : $n = 128$ bits, key size = 128, 192, 256 bits

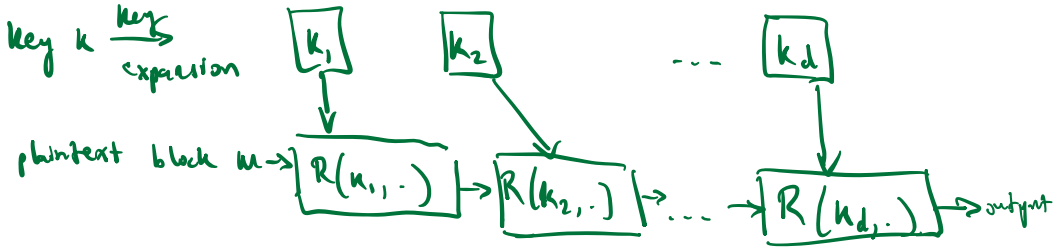
Definition: a block cipher over (K, X) is a pair of "eff" algos (E, D) s.t. $\forall k \in K, \forall x \in X$,

$$D(k, E(k, x)) = x$$

Secure block cipher: For $k \xleftarrow{R} K$, $\pi(x) := E(k, x) : X \to X$ should look like random one-to-one function on X.
(X finite set)

Advanced Encryption Standard (AES) (2000)

Most block ciphers are built by iteration



$R(k, m)$: Round function, d rounds

$y = R(k, x) \Rightarrow x = R^{-1}(k, y)$, R^{-1} easy to calculate

block cipher inversion: alg D, applies keys in reverse order using R^{-1}

3DES: $d = 48$ rounds

AES 128: $d = 10$

AES 192: $d = 12$

AES 256: $d = 14$

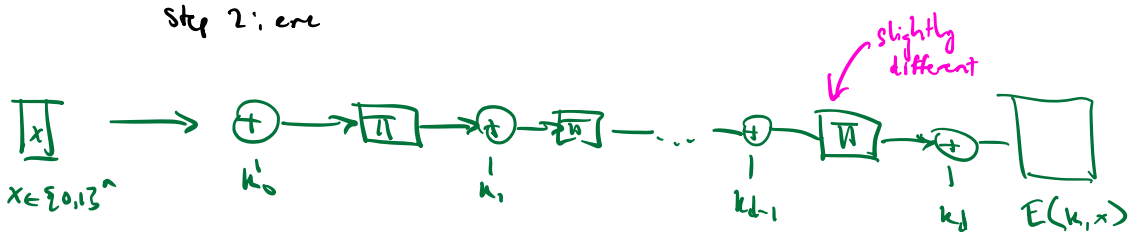
Abstract AES Iterated Even-Mansour Cipher (IEM)

IEM: $\pi : \{0,1\}^n \to \{0,1\}^n$ fixed one-to-one fn on $\{0,1\}^n$, $n = 128$

Block cipher: alg $E(k, x)$ $x \in \{0,1\}^n$

step 1: key expansion $k \to k_1 \ldots k_d \in \{0,1\}^n$

step 2: enc



Thm: if π is a random one-to-one fn

then $\{ k \xleftarrow{R} K : \pi(x) := E(k, x) \}$

"looks like" a random one-to-one fn on X,

provided n, d are sufficiently large

Performance

Cipher	block/key size	speed (MB/s)
chacha20	1 256	643
3DES	64/168	30 ← too slow!
AES128	128/128	163
AES256	128/256	115

} no hw accel

$\boxed{\text{14x w/ hw accel!}}$

AES in hardware

AES-NI (Intel, AMD, ARM)

→ aesenc, $\underbrace{\text{aesenclast}}_{\text{last round}}$ – one round of AES

aesenc $\underbrace{\text{xmm1}}_{\text{state}}$ $\underbrace{\text{xmm2}}_{\text{round key}}$

output xmm1

AES128: load round keys into $\text{xmm}_2 \ldots \text{xmm}_n$

load input x into xmm1

exec [aesenc xmm1, xmm2
aesenc xmm1, xmm3
⋮
aesenclast xmm1, xmm_n]

→ aesdec, aesdeclast : one round of AES dec.

→ aeskeygen assist : AES key expansion

Runs in constant time; resistant to timing attacks

On Skylake, (old), aesenc took 4 cycles

Fully pipelined: can issue aesenc instr every cycle

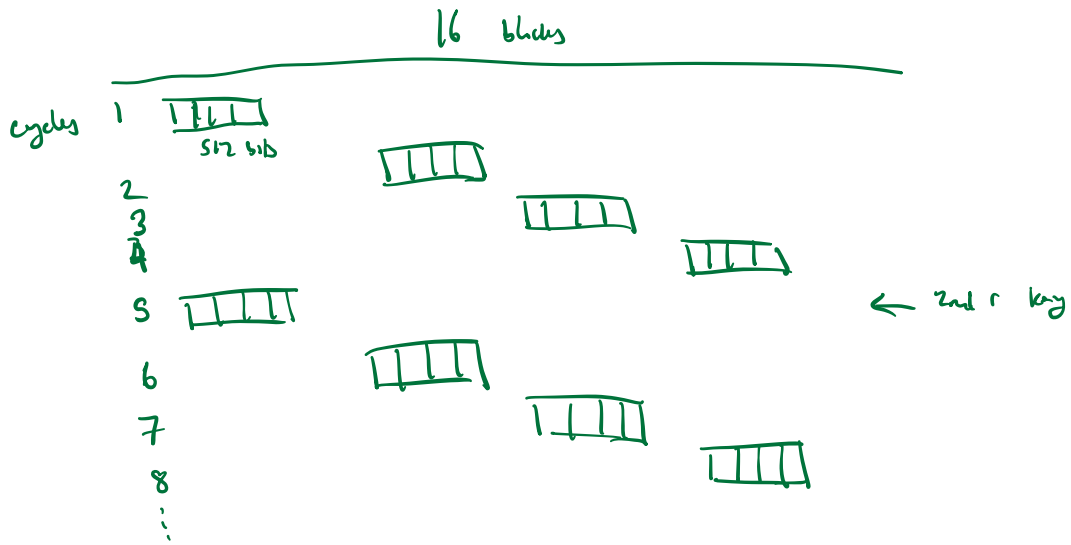
On Ice Lake (2019): vectorized aesenc (AVX2)

apply aesenc to 4 blocks in parallel

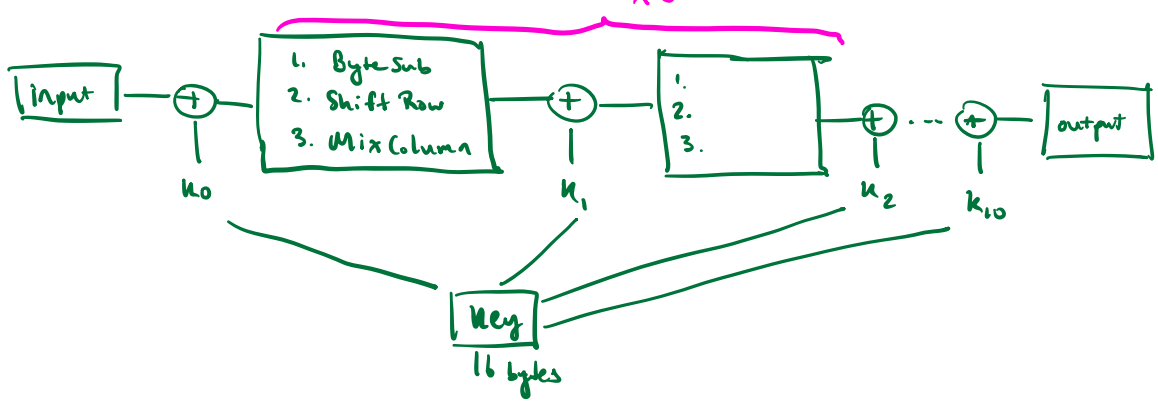
fully pipelined

⇒ encrypting 16 blocks in parallel

takes only 43 cycles

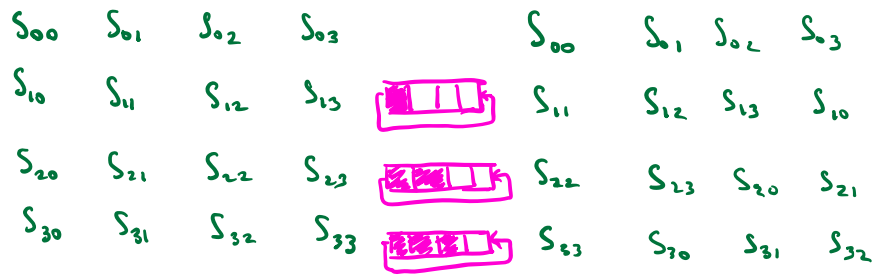


AES128: 10 rounds of Even-Mansour ×10



1. ByteSub: replaces 1-byte by another, 256 byte lookup table (invertible)

2. Shift Rows (invertible)



3. MixColumn: multiplies every column by invertible matrix