

ONE TIME PAD

XOR

XOR	
00	0
01	1
10	1
11	0

$$\begin{aligned} \text{MESSAGGIO} &= M_0 M_1 M_2 \dots \dots M_{12} \\ &= 0110101110101 \end{aligned}$$

$$\begin{aligned} \text{PAD} &= K_0 K_1 K_2 \dots \dots K_{12} \\ &= 101101100011 \end{aligned}$$

il PAD è un insieme di bit casuali, ed è lungo quanto il messaggio da cifrare.

$$\text{CIFRATO} = C_i \leftarrow M_i \oplus K_i$$

$$\begin{array}{cccc} M_0 & M_1 & M_2 & M_{12} \\ \oplus & \oplus & \oplus & \dots \oplus \\ K_0 & K_1 & K_2 & K_{12} \end{array}$$

$$\begin{array}{ccccc} = & 0 & 1 & 1 & 1 \\ & \oplus & \oplus & \oplus & \oplus \\ & 1 & 0 & 1 & 1 \end{array}$$

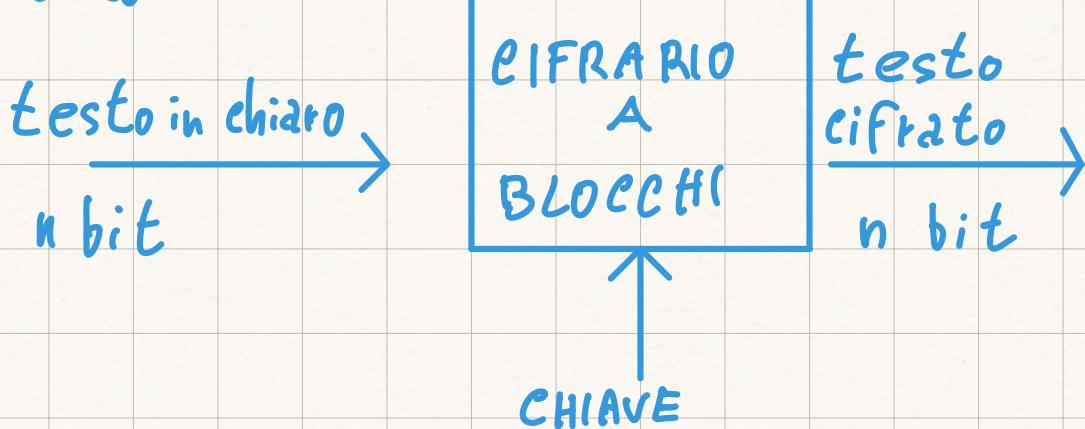
$$= 110110001010 \quad \text{risultato}$$

questo schema di cifratura è **PERFETTAMENTE SEGRETO** nel caso inviamo 1 solo messaggio.

Se invieremo più messaggi doveremo ripetere il PAD e questo permette l'attuazione di diverse tecniche per indovinarlo.

CIFRARI A BLOCCHI

la sicurezza risiede nella segretezza della chiave, la quale deve essere scelta uniformemente a caso.

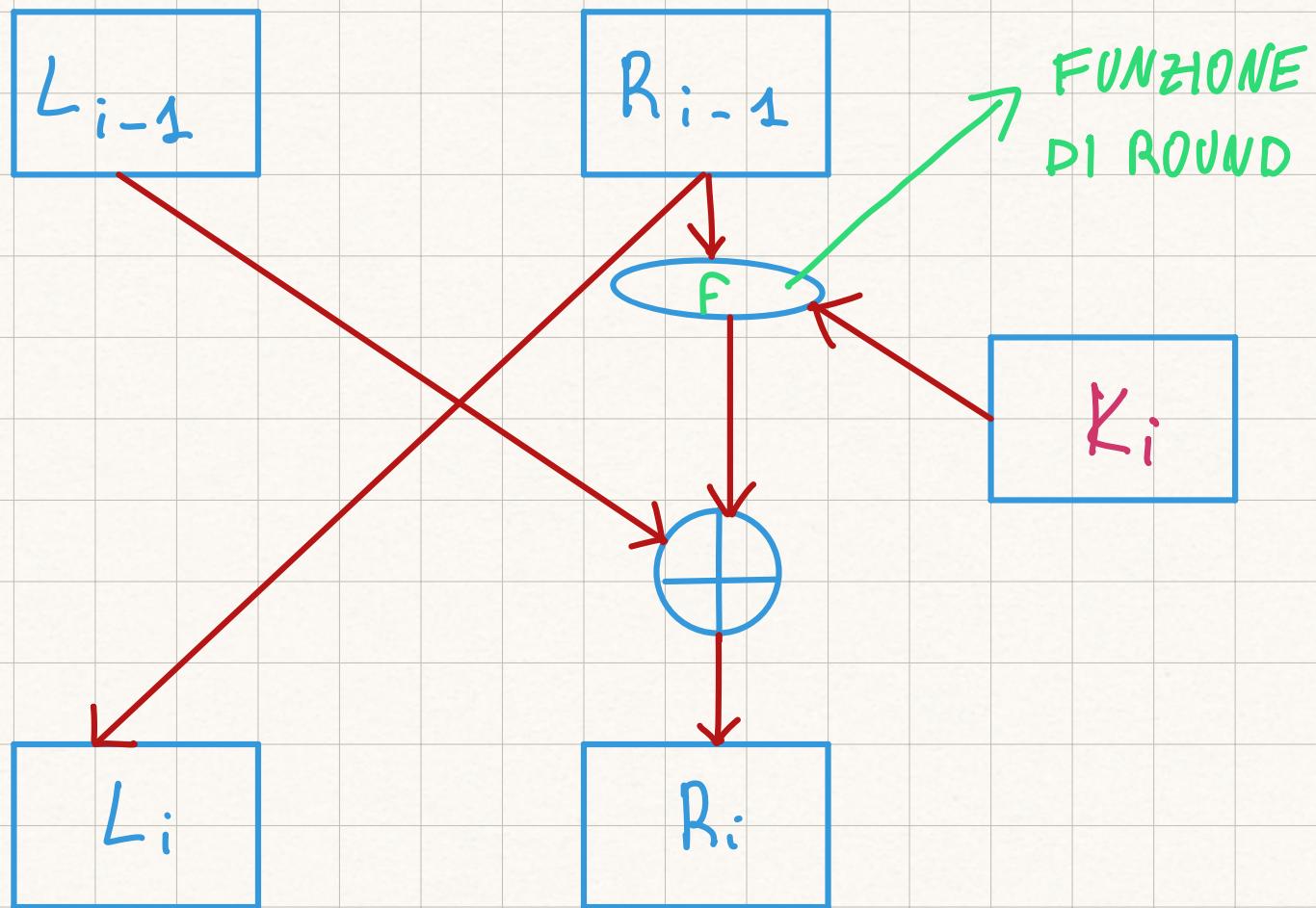


CIFRARI DI FEISTEL

Basato sul principio:

- 1) DIFFUSIONE: indipendenza del testo cifrato da quello in chiaro.
- 2) CONFUSIONE: relazione fra chiave segreta e cifrato difficile da indovinare.

STRUTTURA DI UN ROUND



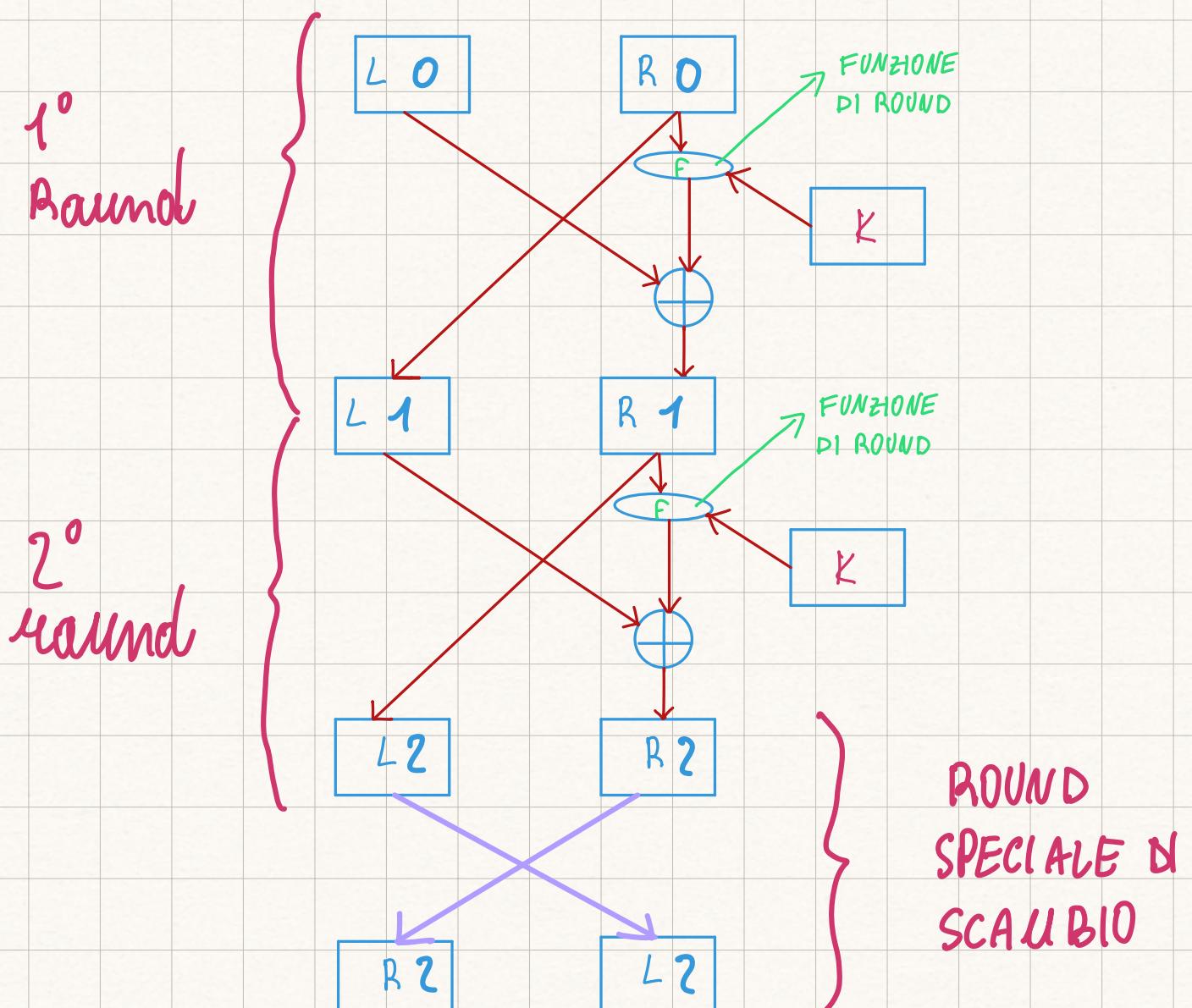
quindi per ogni round, $L_i = R_{i-1}$ mentre

$$R_i = (L_{i-1} \oplus F(R_{i-1}))$$

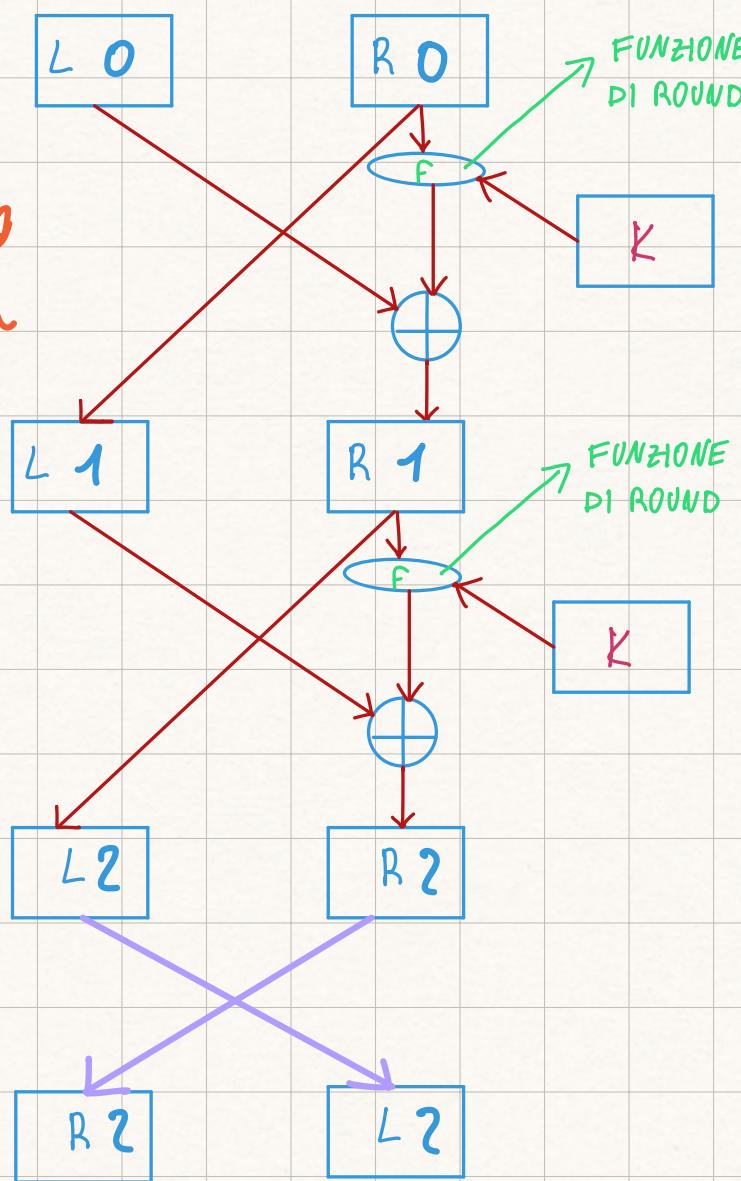
K_i

un cifrario di Feistel è composto da più
ROUND

ESEMPIO A DUE ROUND



Troppo pochi
round rendono
la decifrazione
immediata
e lo scatenamento
non sicuro.

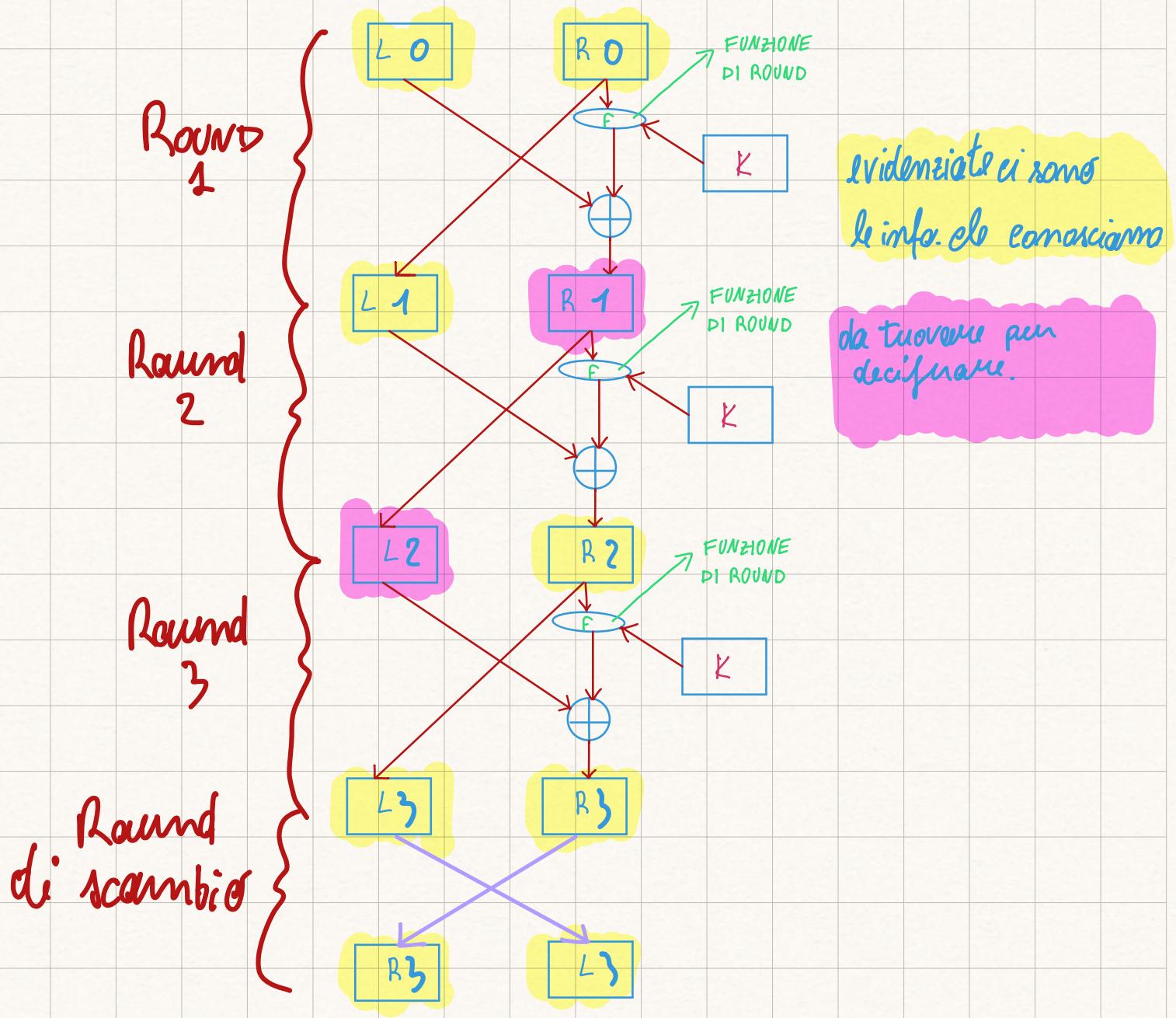


in questo caso tutti gli elem.
sono già stati percorso
questa rete è molto semplice
(ha 2 round)

Se avessimo avuto più round darevo calcolare L_{i-1}, L_{i-2}, \dots in questo modo:

$$L_{i-1} = R_i \oplus f_{K_{i-1}}(R_{i-1})$$

Decifrazione in una rete a 3 Round



RISOLUZIONE:

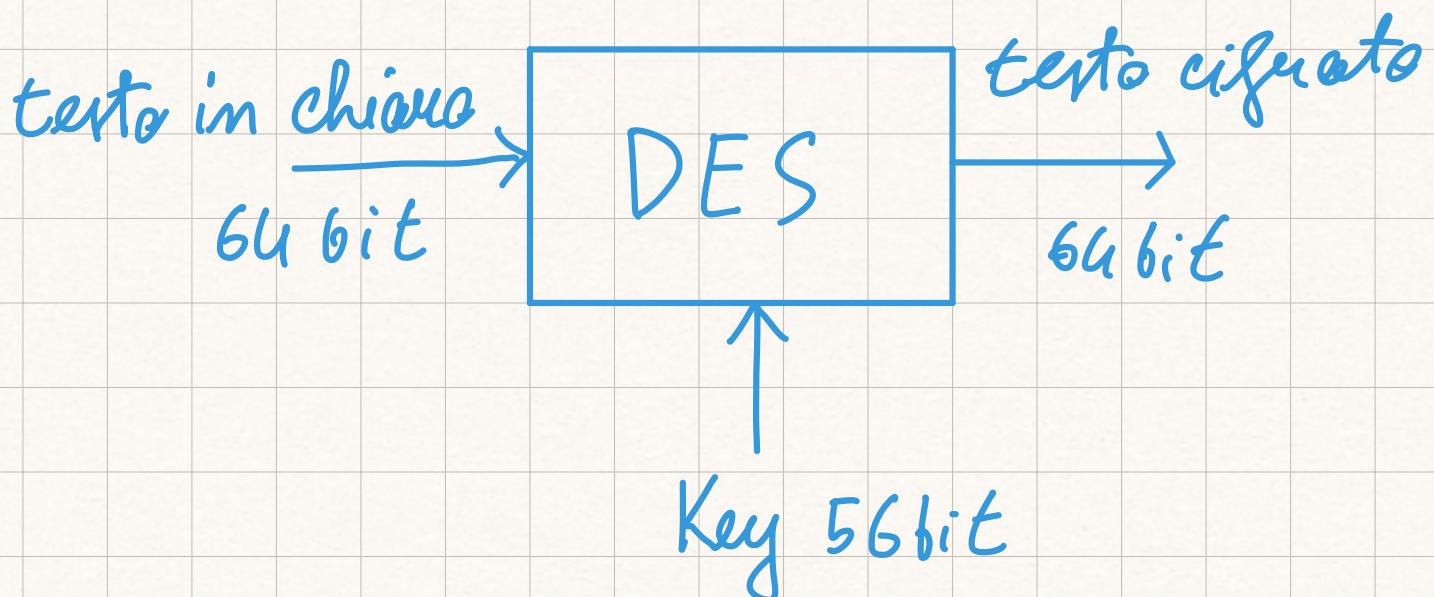
Ricavo L_2 : $R_3 = L_2 \oplus f_K(R_2) \rightarrow L_2 = R_3 \oplus f_K(R_1)$
 R_3 ed R_1 sono noti. Quindi: si cerca a ricavare L_2 .

Ricavando L_2 ho automaticamente R_1 e quindi il gioco è fatto.

DES

È un esempio di cifrario di Feistel.

Data **E**nryption **S**tandard

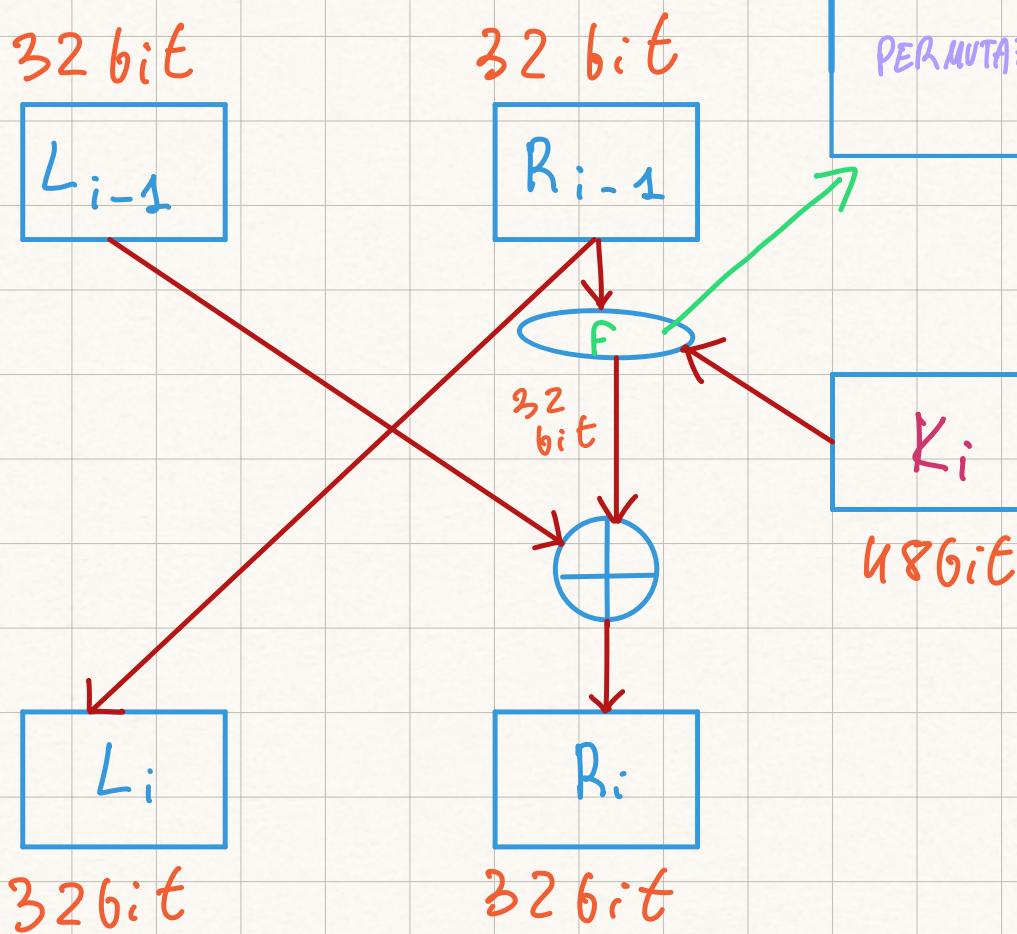
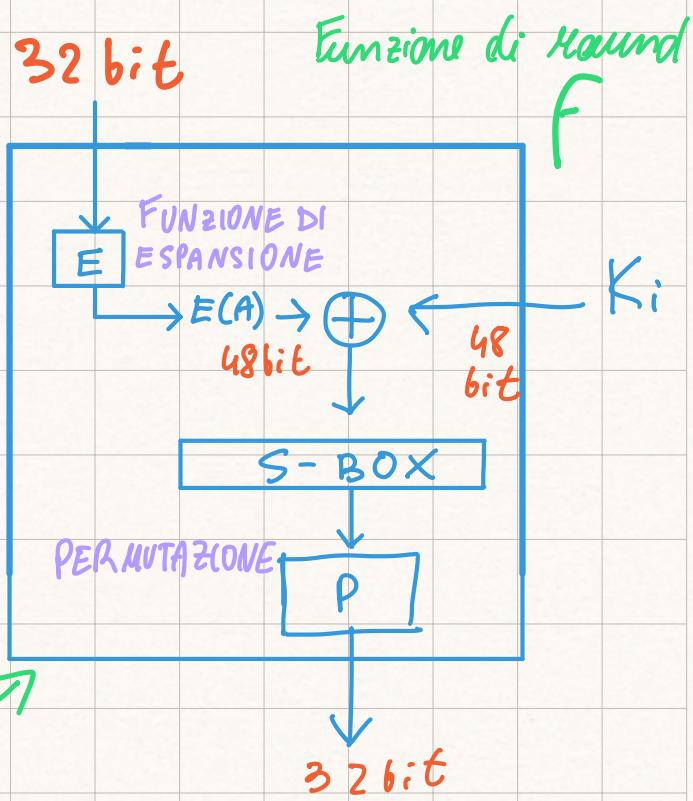


Il DES è costituito da :

- 16 iterazioni: dove in ogni iterazione è implementata con una rete Feistel
- la chiave da usare in ogni iterazione viene generata in questo modo:

A partire dalla chiave di 56 bit data al DES vengono generate 16 sottochiavi da 48 bit

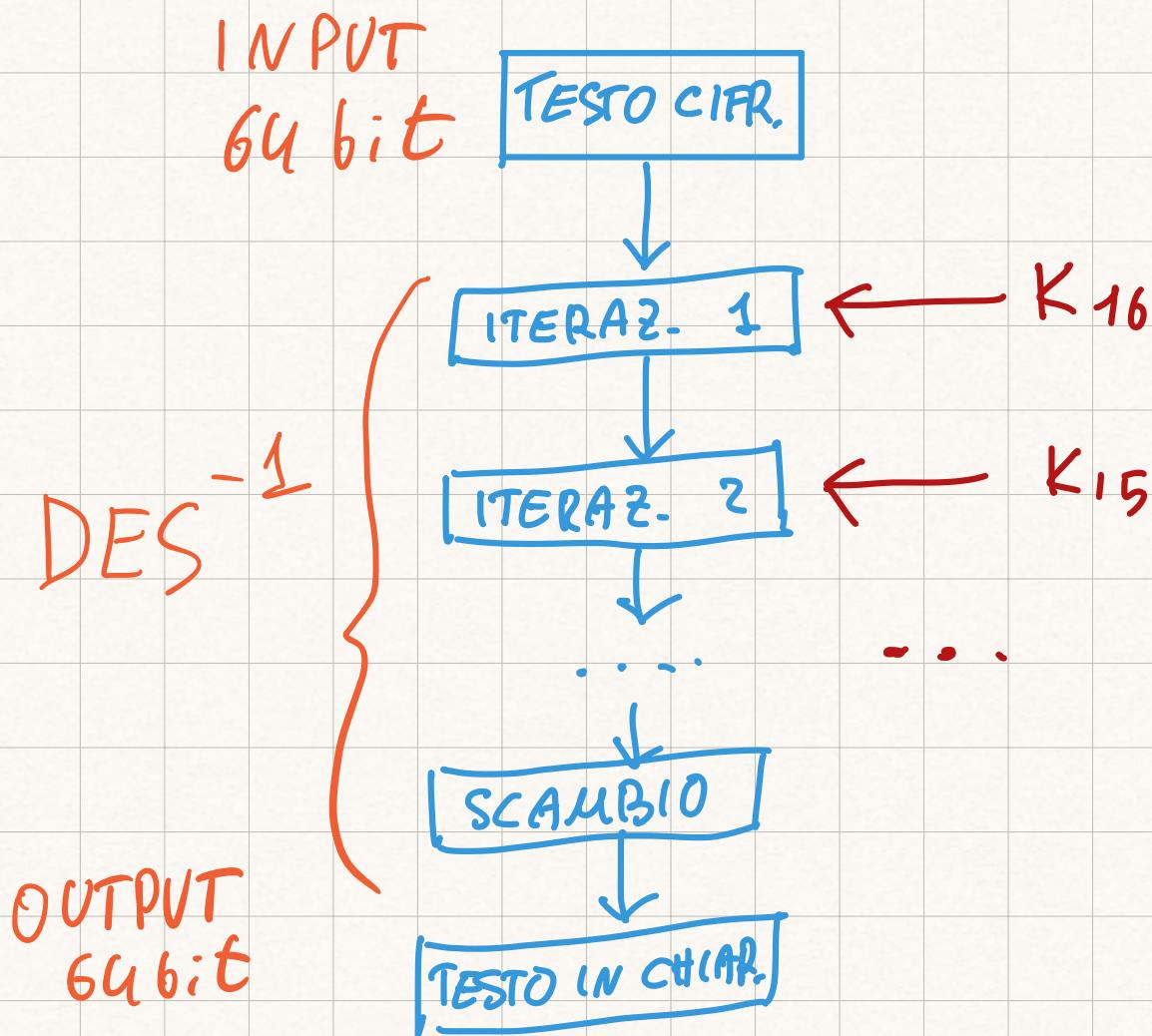
S-BOX: converte input a 6 bit in input a 4 bit. Dividiamo i 48 bit in 8 input da 6 bit e concateniamo questi 6 bit in 4 bit. Così l'output diventa 32 bit.



EFFETTO VALANGA: le S-BOX sono progettate in modo che se viene cambiato un bit di input, ne varieranno almeno 2 in output. Il DES facendone uso, dimostra un grande effetto valanga.

Decifratura DES

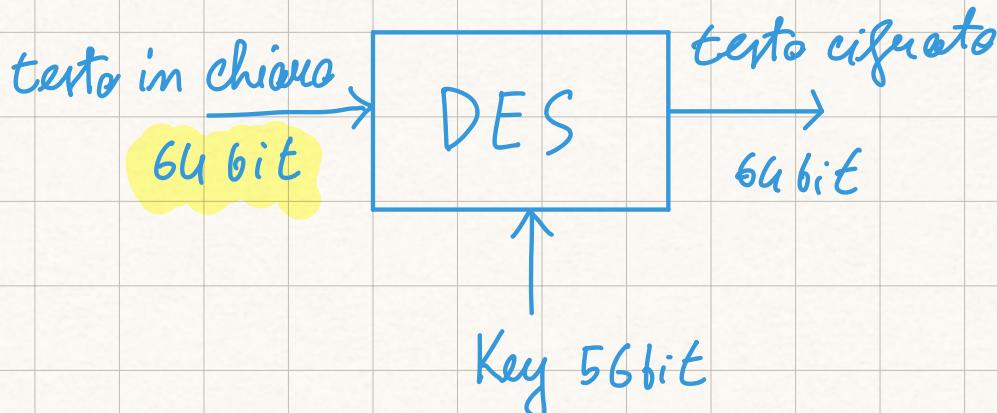
Viene usato lo stesso algoritmo ma impiegando le chiavi nell'ordine inverso.



Osservazioni:

- Progettazione non "limpida". Si suppone che alcune scelte siano state guidate dal governo. ipotesi confermate successivamente
- Attacchi di forza bruta oggi sono fattibili nei confronti del DES.

Modalità operative



il problema del DES è che, nella sua forma semplice permette di cifrare un testo in chiaro solo se è di 64 bit.

Vogliamo estenderne questo limite. Per estenderlo usiamo le modalità operative:

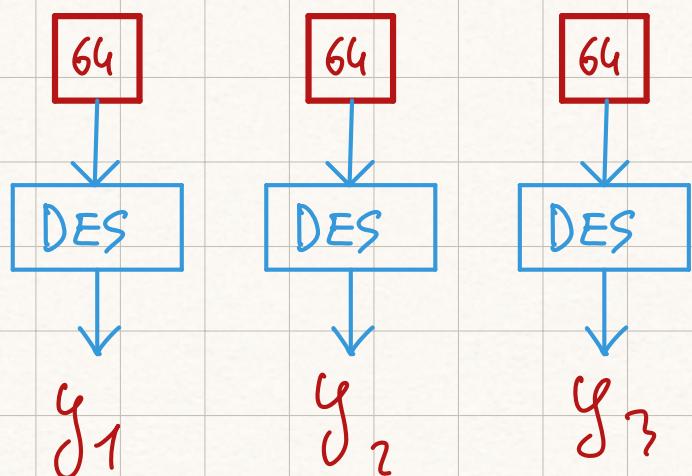
- ECB : electronic codeblock chaining
- CBC : cipher block chaining
- CFB : cipher feedback
- OFB : output feedback
- CTR : counter

ELECTRONIC CODEBLOCK CHAINING (ECB)

$$M = M_1 \ M_2 \ \dots \ M_l$$

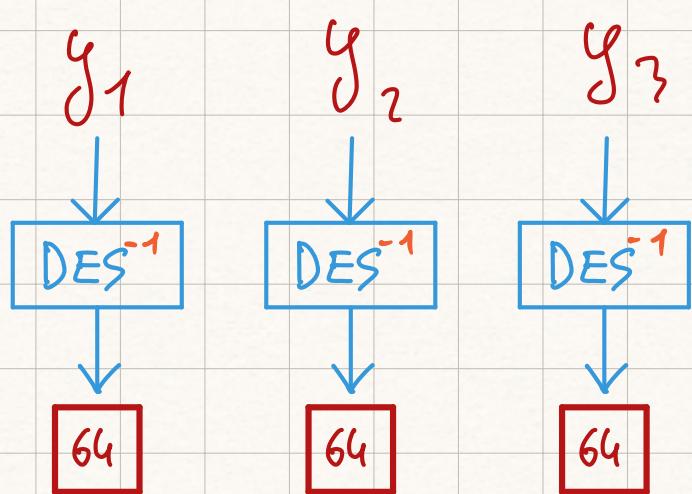
$$X = x_1 \ x_2 \ \dots \ x_m$$

CIFRATURA



Messaggio cifrato = $y_1 \ y_2 \ y_3 \ \dots \ y_m$

DECIFRATURA



il messaggio viene diviso in m blocchi da 64 bit

PARALLELIZZAZIONE:

CIFRATURA ✓

DECIFRATURA ✓

- se la lunghezza dell'input non è un multiplo di 64, si fa uso del PADDING.
- non essendo dipendenza tra i blocchi è possibile memorare attacchi per sostituzione
- se un blocco si ripete otteniamo sempre lo stesso cifrato. Se per esempio cifriamo i colori di una immagine bitmap, le zone di calore

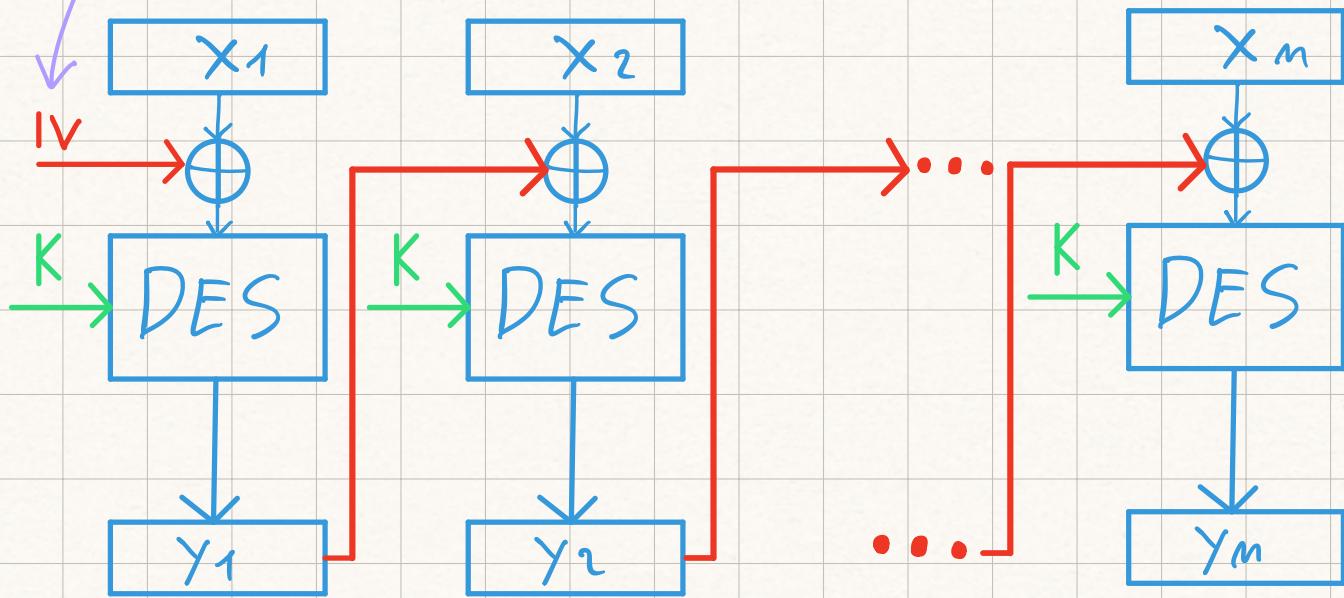
unifattimi saranno cifrati tutti allo stesso modo, rendendo evidenti quale immagine vi si cela dietro.

CIPHER BLOCK CHAINING

(CBC)

VETTORE DI INIZIAZIONE:
SCELTO A CASO,
PUBBLICO O NASCOSTO

CIFRATURA



Messaggio cifrato = $y_1 y_2 y_3 \dots y_m$

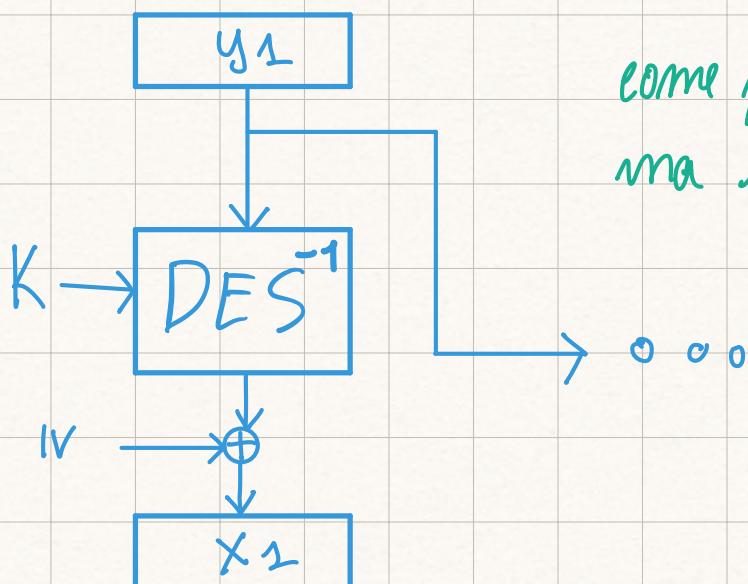
$M = m_1 m_2 \dots m_l$ il messaggio viene
diviso in m blocchi da
64 bit

$$X = x_1 x_2 \dots x_m$$

Funziona in modo simile a quanto visto prima con l'unica differenza che l'input da dare al DES ne viene fatto il PAD :

- 1) Nel caso del 1° blocco, l'input viene dato in XOR con un PAD costituito dal **vettore di inizializzazione**.
- 2) Per i successivi blocchi: viene usato come PAD l'output del DES precedente.

DECIFRATURA



come per la cifratura,
ma tutto invertito.

PARALLELIZZAZIONE:	
CIFRATURA	✗
DECIFRATURA	✓

- c'è dipendenza fra i blocchi quindi non possibili attacchi di riordino.

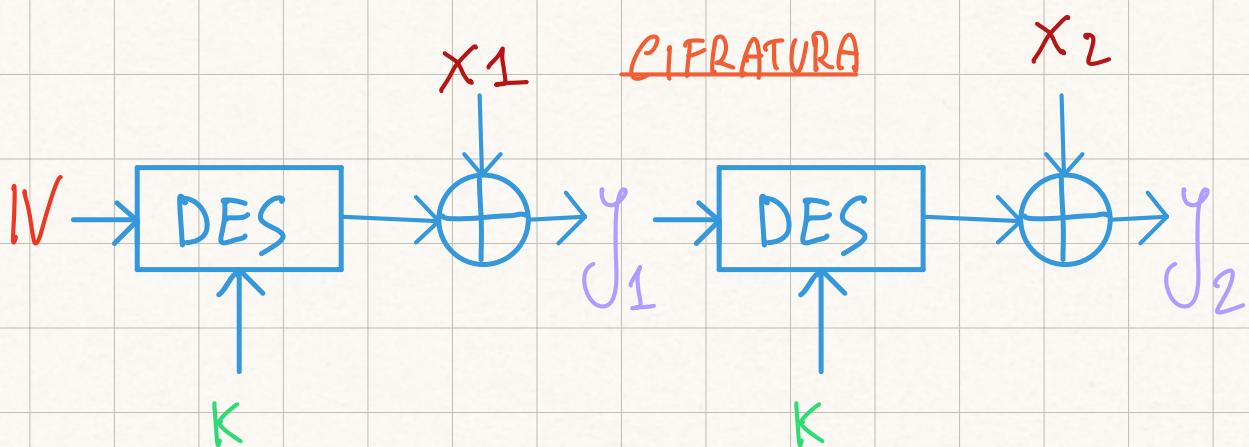
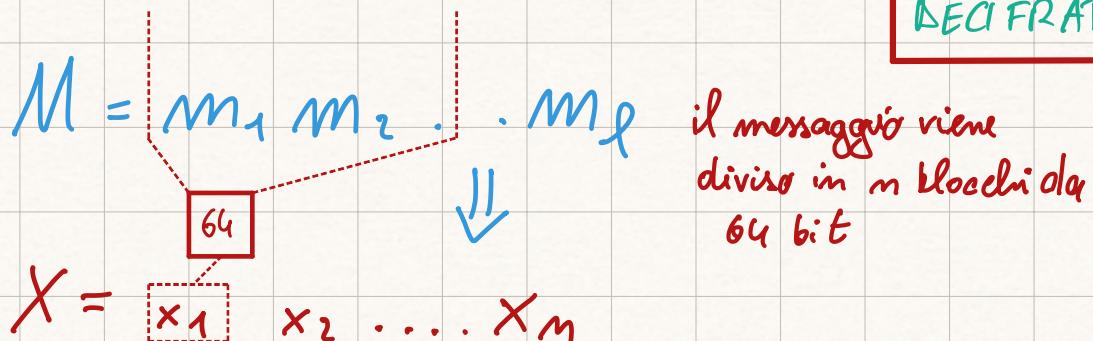
• questa dipendenza può comportare anche una propagazione degli errori.

CIPHER FEEDBACK (CFB)

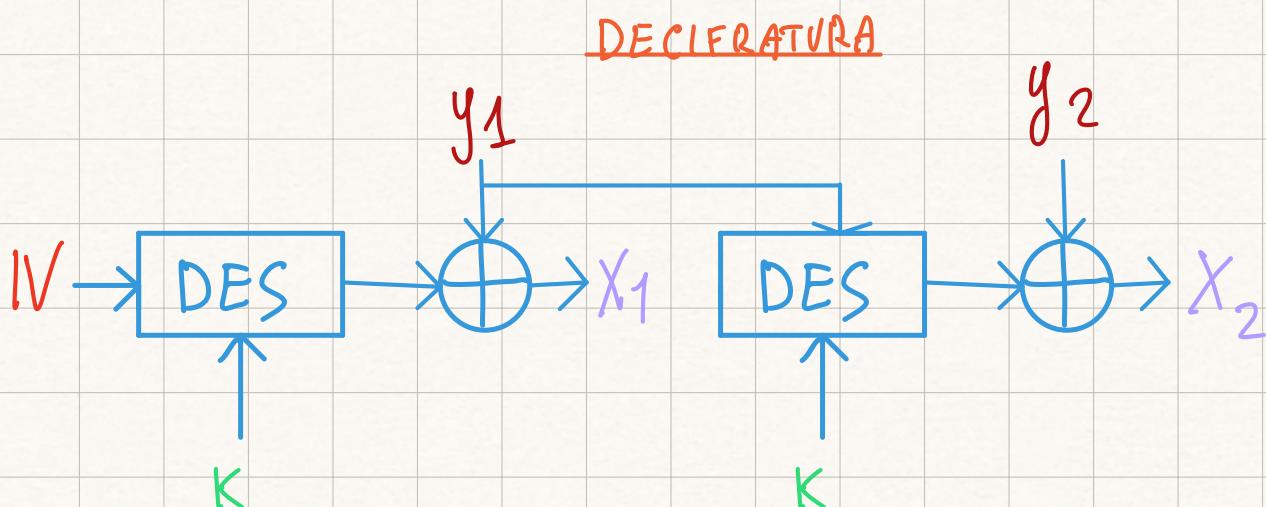
PARALLELIZZAZIONE:

CIFRATURA X

DECIFRATURA ✓



Messaggio cifrato = $y_1 y_2 y_3 \dots y_m$



OUTPUT FEEDBACK

(OFB)

$$M = m_1 m_2 \dots m_l$$

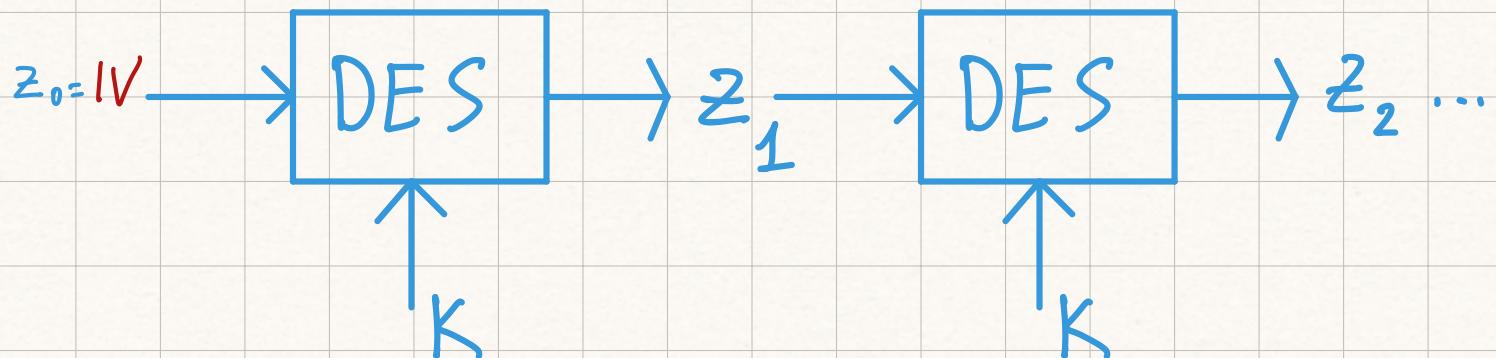
↓

$$X = \boxed{x_1} x_2 \dots x_m$$

64

il messaggio viene diviso in m blocchi da 64 bit

Messaggio cifrato = $y_1 y_2 y_3 \dots y_m$



andiamo prima a generare una sequenza di z indipendente dal messaggio. Costruiamo poi il cifrato di output in questo modo:

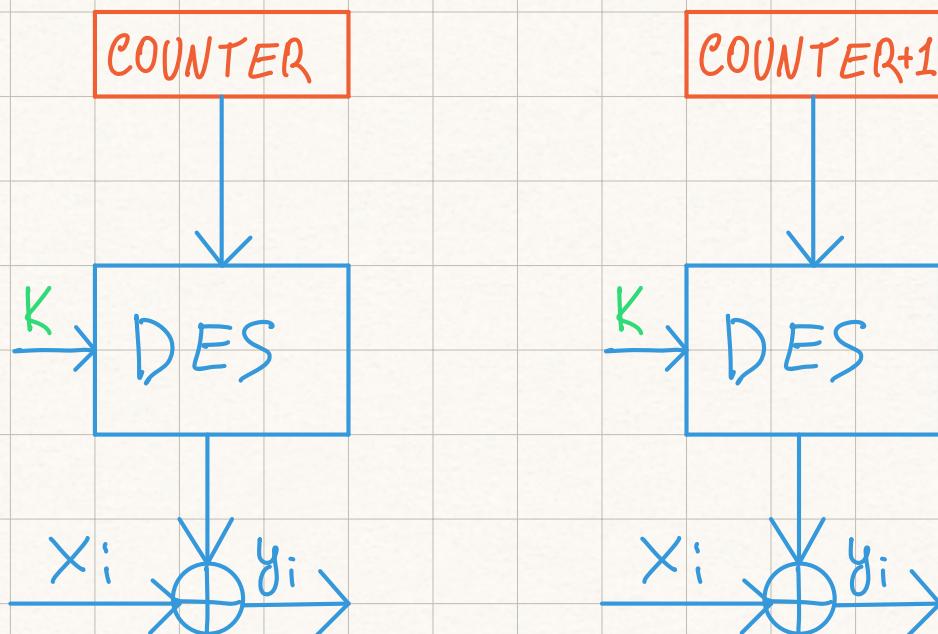
$$y_i = X_i \oplus z_i$$

COUNTER (CTR)

$M = m_1 m_2 \dots m_l$ il messaggio viene diviso in m blocchi da 64 bit
 $X = x_1 x_2 \dots x_m$

Messaggio cifrato = $y_1 y_2 y_3 \dots y_m$

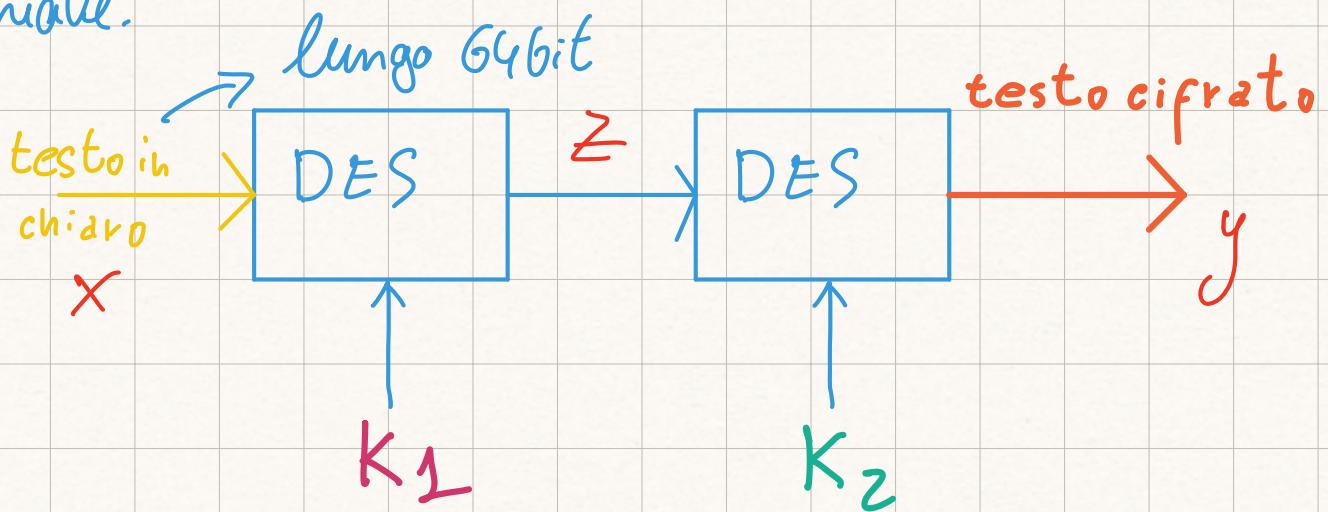
questa modalità operativa usa un contatore di 64 bit.



⚠ bisogna evitare il riuso dello stesso counter e della stessa chiave, altrimenti si veranno a creare delle correlazioni tra i blocchi:

DES DOPPIO

- Il DES ha come limite il fatto che la chiave è lunga solo 56 bit.
- Per "allungare" la chiave poniamo utilizzare il DES due volte (DES DOPPIO) dove, nella seconda applicazione, andiamo ad usare un'altra chiave.



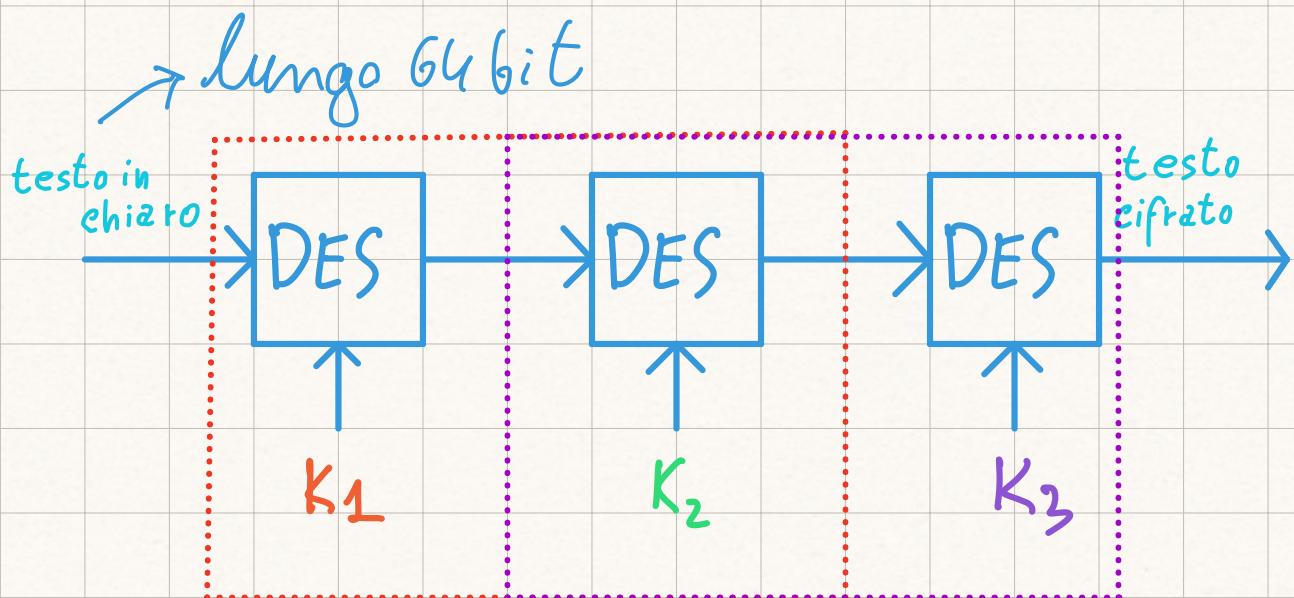
la chiave sarà lunga $K_1 + K_2$ bit, ovvero
 $56 + 56 = 112$ bit.

PROBLEMI:

1. se tuorriamo un K_3 , t.e. $\text{Des}_{K_3}(\cdot) = \text{Des}_{K_2}(\text{Des}_{K_1}(\cdot))$ allora la doppia cifratura equivorrrebbe ad una cifratura singola.
2. Possibili attacchi:
 - Meet in the Middle; si conoscono l'input X e l'output y, ma non le chiavi (K_1, K_2)

Cifra X provando tutte le 2^{56} poss. chiavi K_1
Decifra Y provando tutte le 2^{56} poss. chiavi K_2
Se trova un match per Z, allora ho trovato la chiave!!!

DES TRIPPLICATO



lunghezza della chiave $K_1 + K_2 + K_3 = 56 + 56 + 56 = 168$ bit.

L'attacco meet in the Middle può essere fatto in 2 punti:
- punto 1
- punto 2

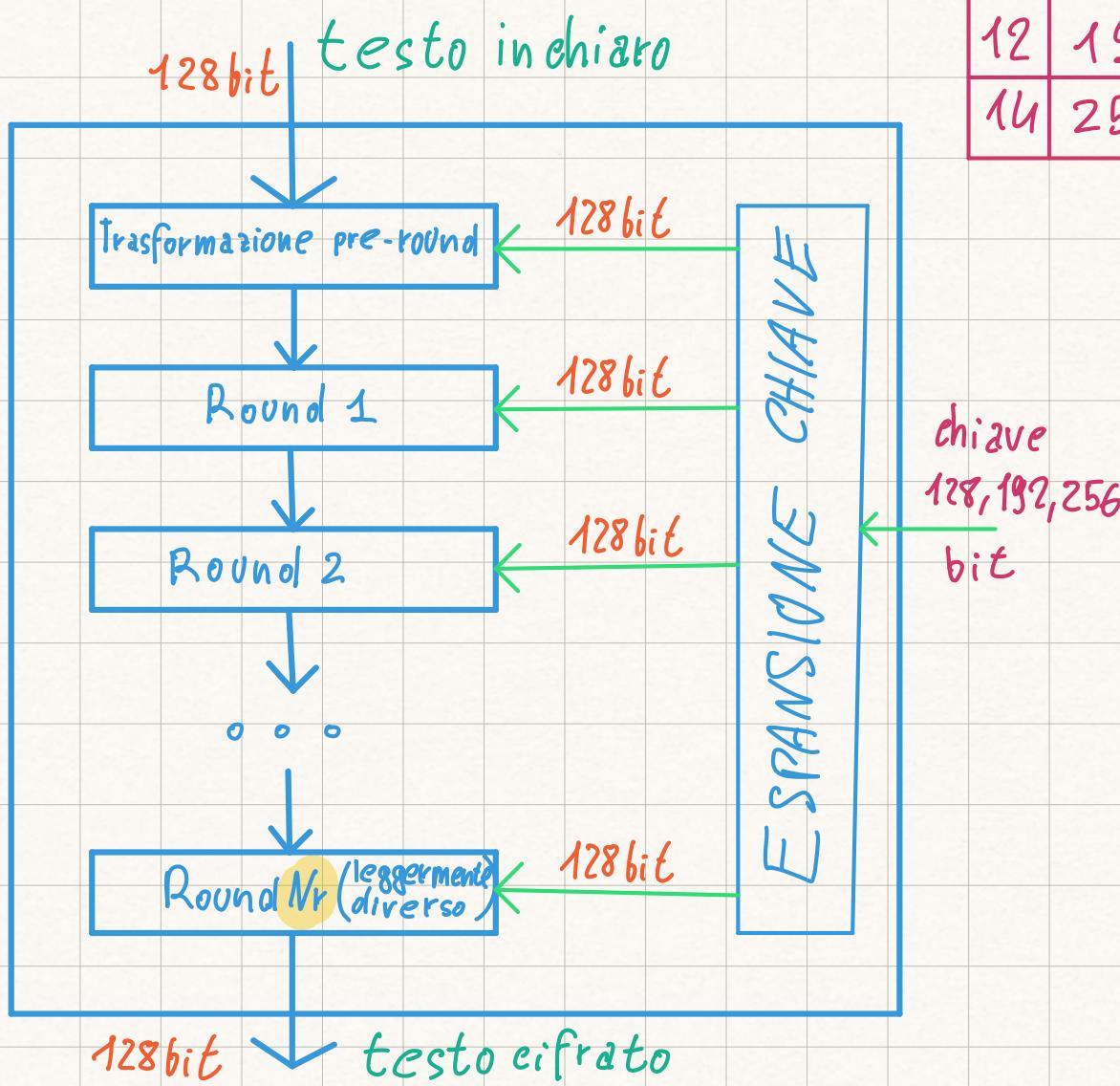
AES

Advanced Encryption Standard.

Pensare è nato:

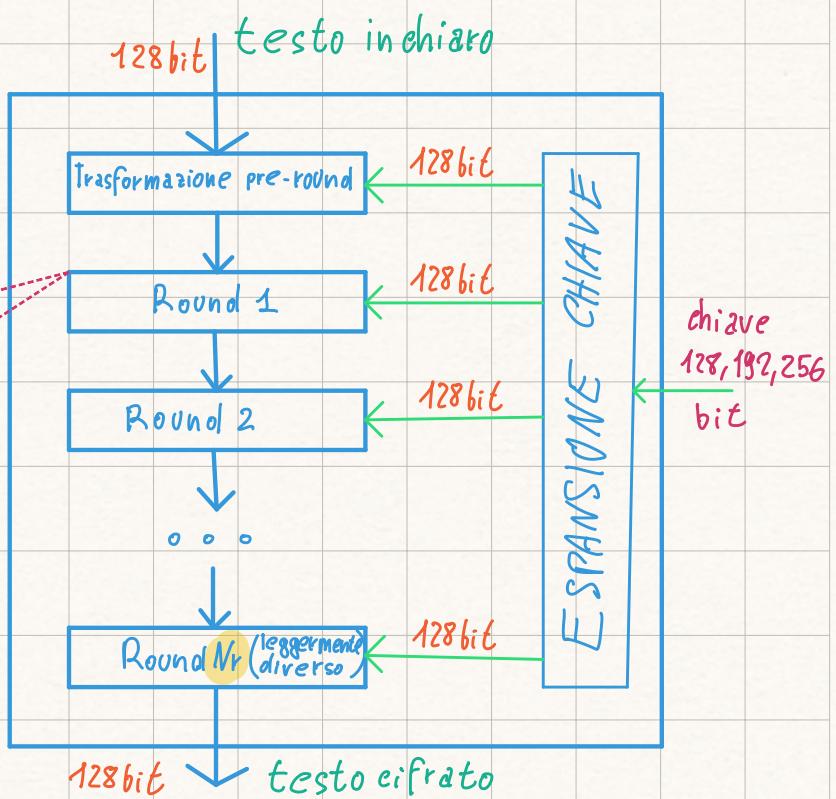
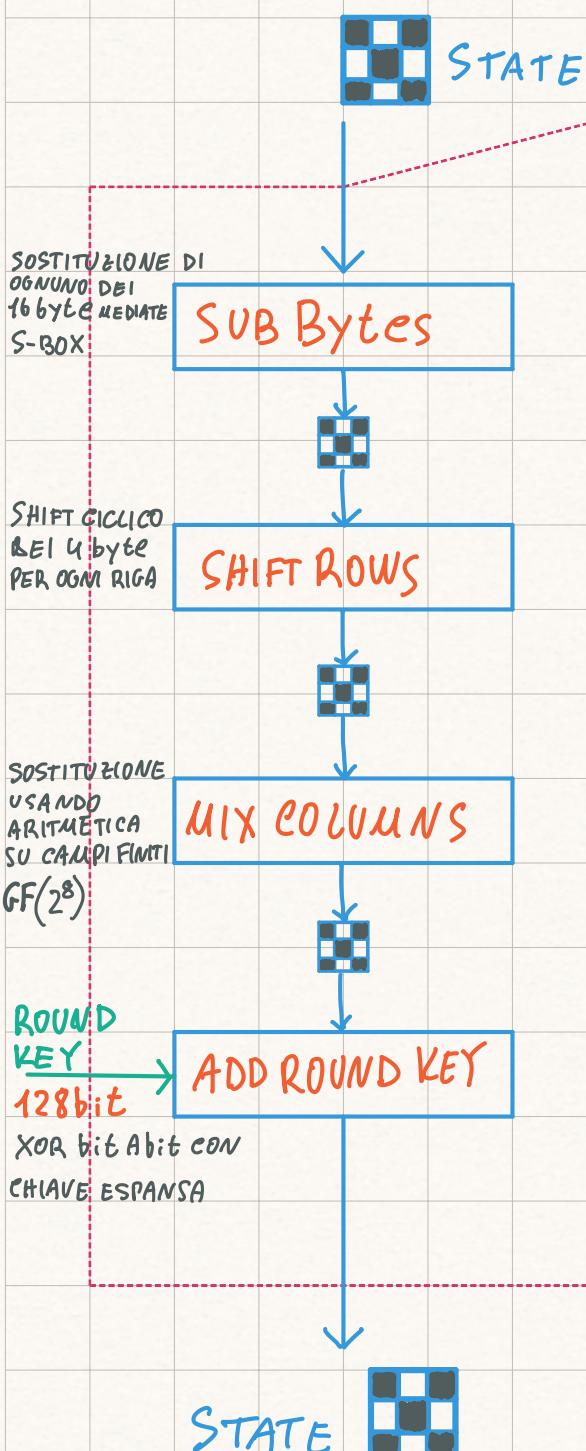
- Il DES aveva dei limiti:
 - chiave di 56 bit
 - blocchi di 64 bit
 - progettazione non "limpida" delle S-BOX.
- Anche il DES DOPPIO/ TRIPLO non era un aiuto.

STRUTTURA



Nr	grandezza chiave
10	128
12	192
14	256

STRUTTURA DI UN ROUND



In ogni ROUND vengono fatte delle operazioni su una matrice di byte chiamata STATE

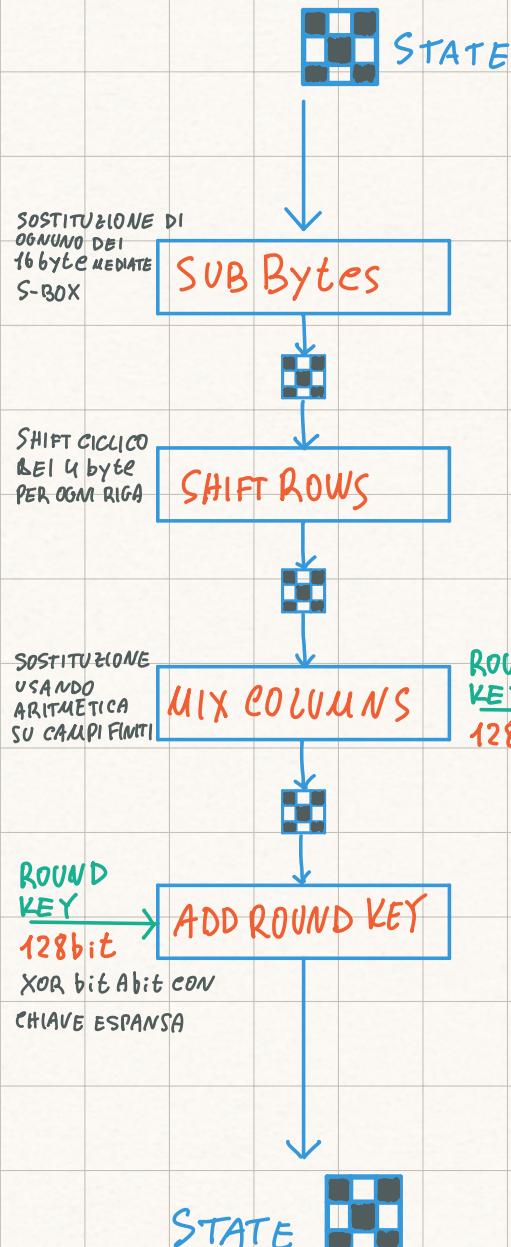
$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

1 byte
blocco è lungo 16 byte = 128bit

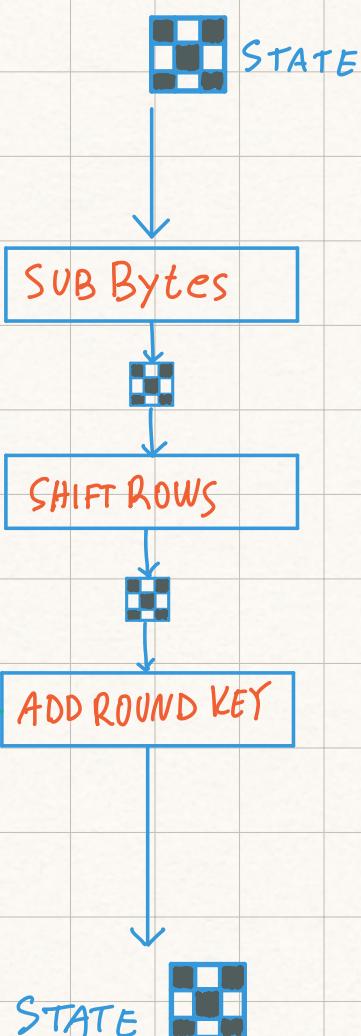
1 byte = 8 bit
1 colonna = $4 \cdot 8 = 32$ bit

ps: anche la chiave è rappresentata mediante una matrice, ma dato che la lunghezza della chiave varia, il numero delle colonne che la compongono cambia.

ROUND NORMALE



ROUND NR



TRASFORMAZIONE PRE-ROUND

ADD ROUNDKEY MIX COLUMNS

prima del 1° ROUND

CONCLUSIONE

DES ed AES fanno parte dei CIFRARI A BLOCCHE e vengono impiegati nella realizzazione di CIFRARI SIMMETRICI.

Invece, per i CIFRARI ASIMMETRICI useremo RSA (Rivest Shamir Adelman)

RSA

Basato su:

- CHIAVE PRIVATA: (n, d)
 - $n = p \cdot q$ con p, q numeri primi
 - d
- CHIAVE PUBBLICA: (n, e)
 - n calcolato come prima
 - e è un numero tale che
 $\text{MCD}(e, (p-1)(q-1)) = 1$
 - $e \cdot d = 1 \pmod{(p-1)(q-1)}$

PER CIFRARE

calcolo il cifrato del messaggio facendo
 $C = M^e \pmod{n}$

PER DECIFRARE

$$M = C^d \pmod{n}$$

si basa sul concetto di funzioni "one way"
funzioni facili da calcolare ma difficili da invertire.
la funzione one way presa qui in considerazione è
quella della fattorizzazione di un numero
ottenuto come il quadrato di due n. primi.