

DES

Wyk ad 7 marca 2005

# Szyfry Blokowe I

*Blok danych binarnych:*

$$\mathbf{x} = (x_0, x_1, \dots, x_{N-1}) \in \mathbb{Z}_{2,N}$$

*możemy traktować  $\mathbf{x}$  jako wektor i jako reprezentację liczby całkowitej:*

$$\|\mathbf{x}\| = \sum_{0 \leq i < N} x_i 2^{N-i-1}$$

*Na przykład, dla  $N=4$  reprezentacja binarna liczby całkowitej:*

$$(0,0,0,0) = 0;$$

$$(0,0,0,1) = 1;$$

.....;

$$(1,1,1,1) = 15$$

## Szyfry blokowe II

Przykady innych reprezentacji informacji:

ASCII: (7 bitowa) :

0	1	2	.....	A	B	C	.....	a	b	...
4	49	50		65	66	67		97	98	

Szesnastkowa (4 bity / cyfra ):

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

0 1 2 3 4 5 6 7 8 9 **A** **B** **C** **D** **E** **F**

Inne: EBCDIC (64 bitowe), UNICODE (16, 24)

# ASCII - EBCDIC

DEC	HEX	ASCII	EBCDIC
26	1A	SUB Substitute	CC Cursor Control
27	1B	ESC Escape	CU1 Customer Use 1
28	1C	FS File Separator (IS)	IFS Interchange File Separator
7	1D	GS Group Separator (IS)	IGS Interchange Group Separator
8	1E	RS Record Separator (IS)	IRS Interchange Record Separator
9	1F	US Unit Separator (IS)	IUS Interchange Unit Separator
10	20	SP Space	DS. Digit

# Szyfry blokowe III

Szyfrem blokowym  $\pi$  nazywamy (pseudo-losowe)  
odwzorowanie:

$$\pi: \mathbf{x} \rightarrow \mathbf{y} = \pi(\mathbf{x})$$

$$\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$$

$$\mathbf{y} = (y_0, y_1, \dots, y_{N-1})$$

*Odwzorowanie może zależeć od parametru  $K$ :*

$$\pi(K, x) = y; K \in Z_{2,M}$$

*(nazywanego kluczem szyfru)*

*W ogólności  $N \neq M$*

# Szyfry blokowe IV

## elementy szyfru (1)

Przesuni cie w lewo (cykliczne):

$$\lambda: (x_0, x_1, \dots, x_{N-1}) \rightarrow (x_1, \dots, x_{N-1}, x_0)$$

Przesuni cie w prawo (cykliczne):

$$\rho: (x_0, x_1, \dots, x_{N-1}) \rightarrow (x_{N-1}, x_0, \dots, x_{N-2})$$

Dodawanie (z przeniesieniem):

$$\sigma: \mathbf{x} \rightarrow \mathbf{y} \quad \|\mathbf{y}\| = \|\mathbf{x}\| + 1 \text{ (modulo } 2^N)$$



# Szyfry blokowe VI

## elementy szyfru (3)

Permutacja bitów  $\tau$  (szczególny przypadek transformacji liniowej:

$$\pi_{\tau} : (x_0, x_1, \dots, x_{N-1}) \rightarrow (x_{\tau(0)}, x_{\tau(1)}, \dots, x_{\tau(N-1)})$$

*Przykład (N=4):*

$$\tau(0) = 3, \tau(1) = 2, \tau(2) = 0, \tau(3) = 1$$

$$L = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

# Szyfry blokowe VII

## elementy szyfru (4)

Transformacja afiniczna:

$L$  – nieosobliwa transformacja liniowa,

$$a \in \mathbb{Z}_{2,N}$$

$$\pi: x \rightarrow Lx + a$$

Odwzorowanie  $\pi: \mathbb{Z}_{2,N} \rightarrow \mathbb{Z}_{2,N}$

nazywamy inwolucją gdy:

$$\pi^2 = I$$

( $I$  - to samo )

# Szyfry blokowe VIII

## Założenia I (Funkcjonalno )

1. Służy do zapewnienia poufności przekazu informacji.
2. Bloki informacji długości  $N$  dzielimy na części o długości  $N$  (ewentualnie ostatni blok uzupełniamy zerami lub losowymi bitami).
3. Potrafimy uzgodnić pomiędzy komunikującymi się stronami i zachować w tajemnicy klucz szyfru.

# Szyfry blokowe IX

## Zasady Kerckhoffa II

- Przeciwnik zna algorytm  $\pi$ .
- Przeciwnik zna rozmiar przestrzeni kluczy  $M$ .
- Przeciwnik nie zna klucza  $K$ .

# Szyfry blokowe X

## Strategie przeciwnika

- Wykraść, znaleźć, wymusić lub kupić klucz  $K$ . (Wykorzysta nieostrość korespondentów).
- Przechwycić szyfrogram  $y$  i wypróbować wszystkie możliwe klucze ( $2^M$ ), a otrzymać sensowny tekst jawny  $x$ .
- Zdobyć szyfrogram  $y = \pi(K, x)$ , odpowiadający mu tekst jawny  $x$  i wypróbować wszystkie możliwe klucze (Przeszukanie przestrzeni kluczy.).
- Zbadać korelacje pomiędzy tekstem jawnym  $x$  szyfrogramem i kluczem. Na tej podstawie próbować odtworzyć klucz. Taki atak na ogół wymaga dostępu do bardzo wielu par  $x, y$ .
- Zbudować katalog wszystkich możliwych bloków o długości  $N$  i badać statystyczne korelacje szyfrogramów i tekstów jawnych. Na tej podstawie może czasem zgadnąć czego dotyczy korespondencja.

# Szyfry blokowe XI

## Zasady konstrukcji I

- Rozmiar bloku  $N$  powinien być na tyle duży, aby uniemożliwiło zbudowanie katalogu wszystkich możliwych bloków.
- Długość  $M$  klucza  $K$  powinna uniemożliwić przeszukanie przestrzeni kluczy.
- Algorytm  $\pi$  powinien być na tyle skomplikowany, aby uniemożliwiło analityczne i statystyczne badania korelacji tekstu jawnego i szyfrogramu. Zasadniczo wynik obliczeń algorytmu  $\pi$  nie powinien różnić się statystycznie od rezultatów pochodzących z generatora losowego.

# Szyfry blokowe XII

## Zasady konstrukcji II

- Claude E. Shannon (1949) Podstawy teoretyczne: dyfuzja + konfuzja.
- Konstrukcja szyfru blokowego: Superpozycja i iteracja elementarnych operacji na bitach.
- Horst Feistel , IBM (1973) „Feistel network”, S-boxes (Lucifer).

# DES I

## Historia

- 1972 – USA – prace nad standardem ochrony informacji na potrzeby administracji, oraz na dla bezpiecznego przekazu informacji finansowych. By o to spowodowane rozwojem usług bankowych, upowszechnieniem komputerów, oraz sieci telekomunikacyjnych.
- 1974 - II konkurs NBS (na pierwszy rok wczesniej nie wpłynęła żadna oferta) – przyjęto projekt IBM DES. Projekt zmodyfikowany we współpracy z NSA.
- 1977 – DES przyjęty w USA jako standard federalny
- 2002 DES został ostatecznie zastąpiony przez AES

# DES II

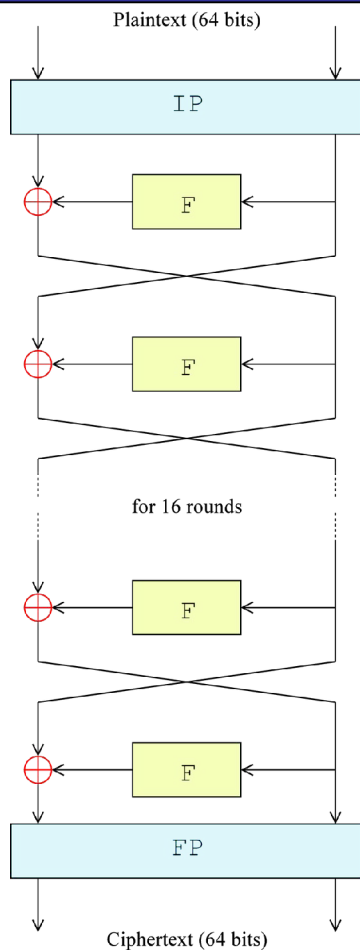
## w asno ci

- Szyfr blokowy, symetryczny, długość bloku 64 bity.
- Długość klucza 56 bitów (64 bity, w tym 8 bitów parzystości bez kryptograficznego znaczenia).
- Ten sam klucz służy do szyfrowania i deszyfrowania.
- Zaimplementowany na praktycznie każdej platformie sprzętowej (w tym na kartach elektronicznych) i w postaci obwodów scalonych.
- Ciągłe użycie w bankowości i w niezliczonych innych zastosowaniach.
- Wariant 3des, z kluczem o długości 112 bitów jest nadal uważany za bezpieczny.

# DES III - struktura

DES jest z o eniem 33 transformacji:

$$\text{DES} = \text{IP}^{-1} \times \pi_{T(16)} \times \theta \times \pi_{T(15)} \times \dots \times \theta \times \pi_{T(1)} \times \text{IP}$$

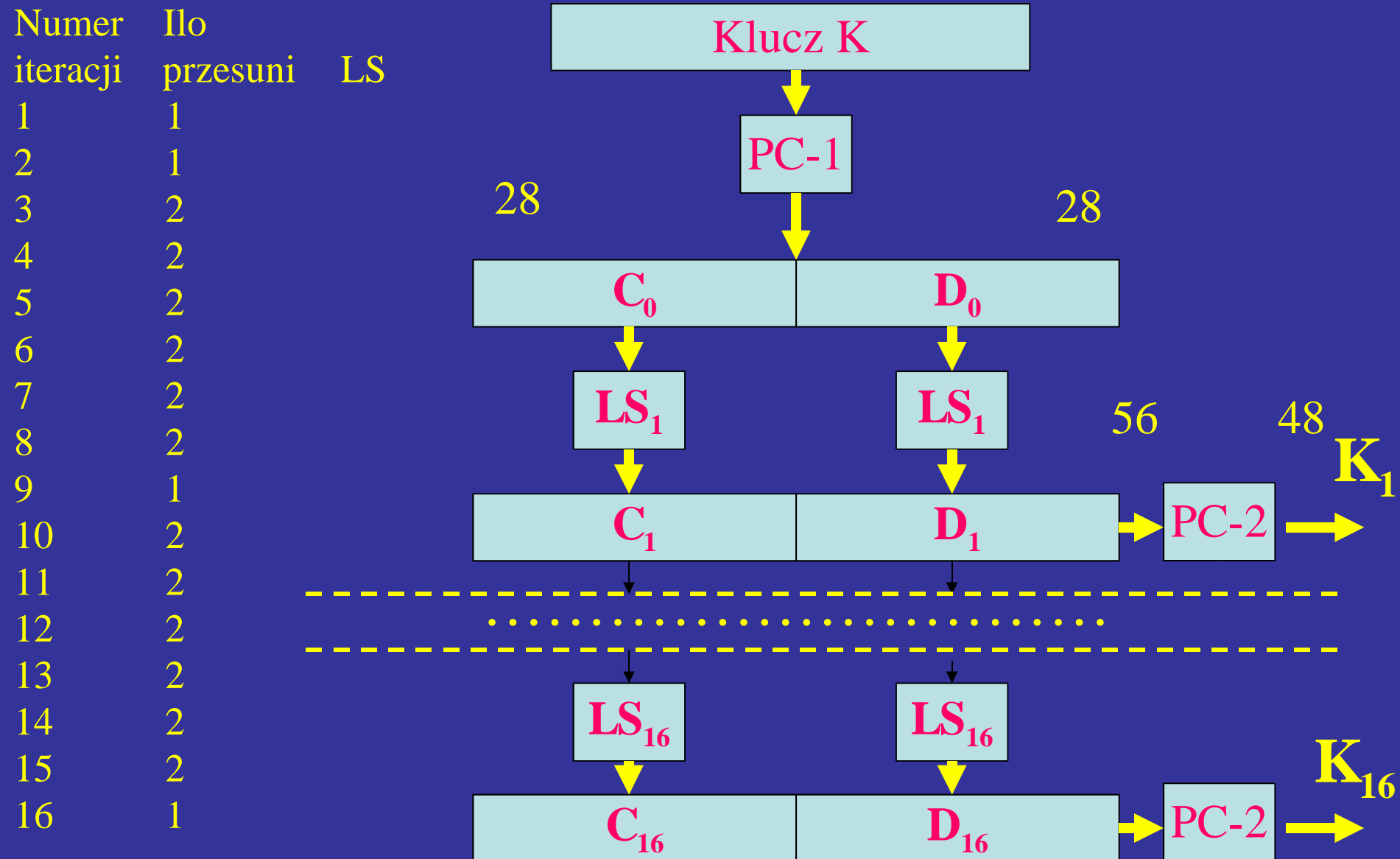


Algorytm – dwa elementy:  
2. Przetwarzanie danych  
3. Przygotowanie kluczy

Przetwarzanie danych:

- Permutacja wst pna IP
- Szesna cie iteracji
- Permutacja ko cowa  $\text{IP}^{-1}$

# DES IV: przygotowanie kluczy



# DES V Przygotowanie kluczy

## PC-1

57 49 41 33 25 17 9  
1 58 50 42 34 26 18  
10 2 59 51 43 35 27  
19 11 3 60 52 44 36

Bity  $C_i$

bity 8, 16, 24, 32, 40,  
48, 56 i 64 nie s  
u ywane do generacji  
kluczy

Bity  $D_i$

## PC-2

14 17 11 24 1 5  
3 28 15 6 21 10  
23 19 12 4 26 8  
16 7 27 20 13 2  
41 52 31 37 47 55  
30 40 51 45 33 48  
44 49 39 56 34 53  
46 42 50 36 29 32

Przesuni cie o jeden bit w lewo: stary bit nr 2 staje si bitem nr 1,  
a stary bit nr 1 staje si bitem ostatnim, nr 28.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28  
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 1

# DES VI – Permutacja wst pna

*IP*

58 50 42 34 26 18 10 2

60 52 44 36 28 20 12 4

62 54 46 38 30 22 14 6

64 56 48 40 32 24 16 8

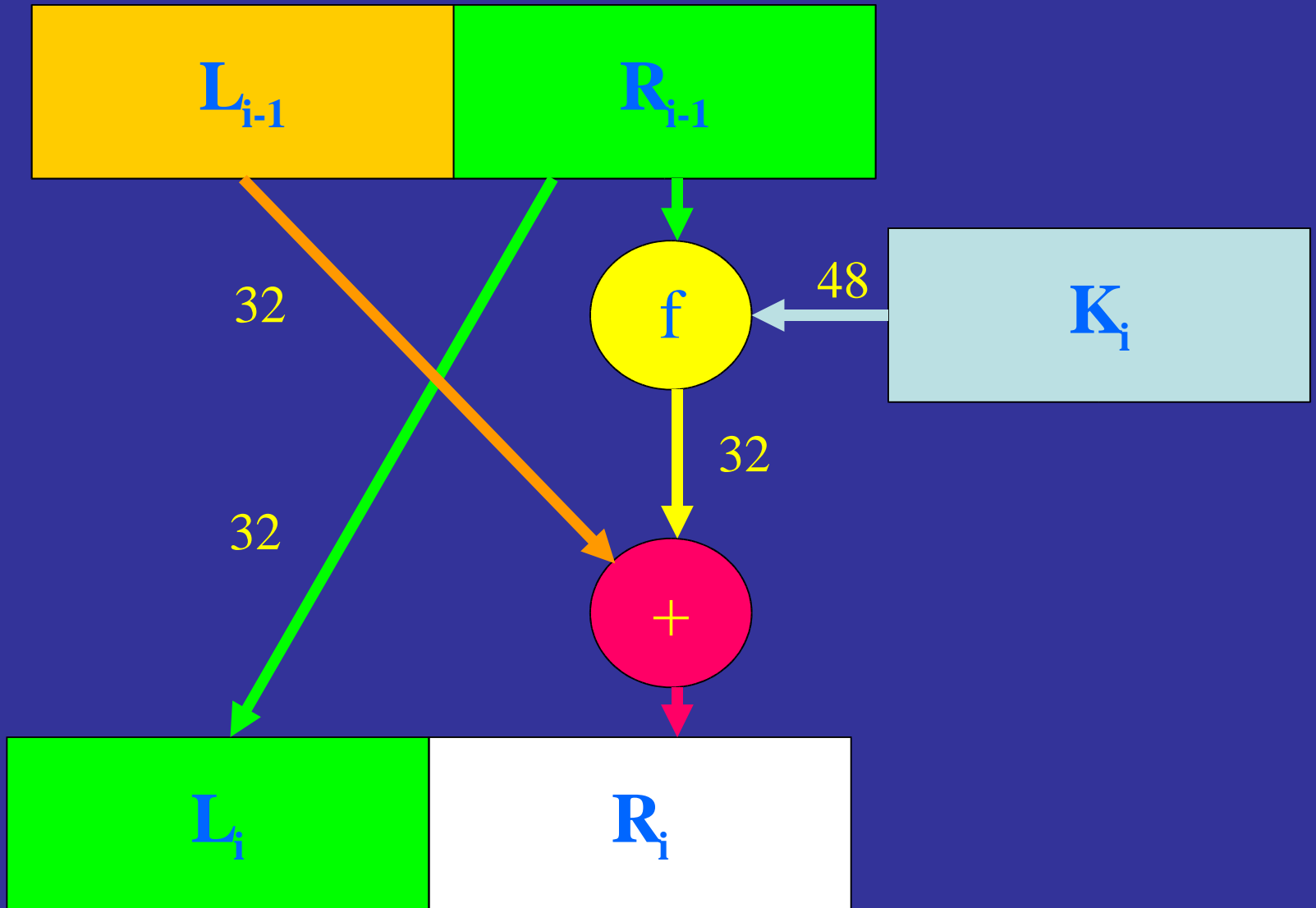
57 49 41 33 25 17 9 1

59 51 43 35 27 19 11 3

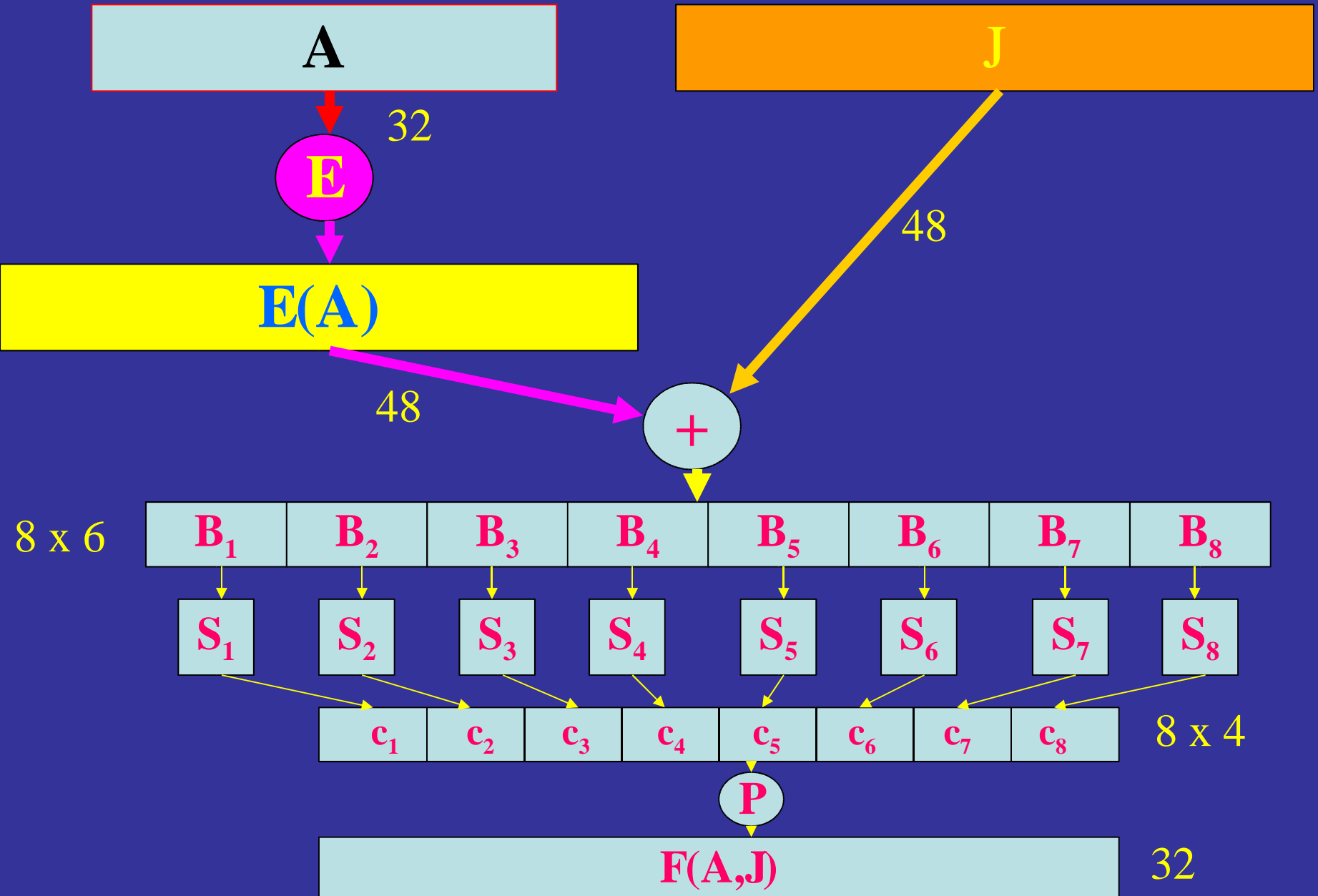
61 53 45 37 29 21 13 5

63 55 47 39 31 23 15 7

# DES VII – Jedna runda iteraciji



# DES VIII Funkcja f



# DES IX s-boxes

## S1

14 4 13 1 2 15 11 8 3 10 6 12 5 9 0 7  
0 15 7 4 14 2 13 1 10 6 12 11 9 5 3 8  
4 1 14 8 13 6 2 11 15 12 9 7 3 10 5 0  
15 12 8 2 4 9 1 7 5 11 3 14 10 0 6 13

## S2

15 1 8 14 6 11 3 4 9 7 2 13 12 0 5 10  
3 13 4 7 15 2 8 14 12 0 1 10 6 9 11 5  
0 14 7 11 10 4 13 1 5 8 12 6 9 3 2 15  
13 8 10 1 3 15 4 2 11 6 7 12 0 5 14 9

## S3

10 0 9 14 6 3 15 5 1 13 12 7 11 4 2 8  
13 7 0 9 3 4 6 10 2 8 5 14 12 11 15 1  
13 6 4 9 8 15 3 0 11 1 2 12 5 10 14 7  
1 10 13 0 6 9 8 7 4 15 14 3 11 5 2 12

## S4

7 13 14 3 0 6 9 10 1 2 8 5 11 12 4 15  
13 8 11 5 6 15 0 3 4 7 2 12 1 10 14 9  
10 6 9 0 12 11 7 13 15 1 3 14 5 2 8 4  
3 15 0 6 10 1 13 8 9 4 5 11 12 7 2 14

## S5

2 12 4 1 7 10 11 6 8 5 3 15 13 0 14 9  
14 11 2 12 4 7 13 1 5 0 15 10 3 9 8 6  
4 2 1 11 10 13 7 8 15 9 12 5 6 3 0 14  
11 8 12 7 1 14 2 13 6 15 0 9 10 4 5 3

## S6

12 1 10 15 9 2 6 8 0 13 3 4 14 7 5 11  
10 15 4 2 7 12 9 5 6 1 13 14 0 11 3 8  
9 14 15 5 2 8 12 3 7 0 4 10 1 13 11 6  
4 3 2 12 9 5 15 10 11 14 1 7 6 0 8 13

## S7

4 11 2 14 15 0 8 13 3 12 9 7 5 10 6 1  
13 0 11 7 4 9 1 10 14 3 5 12 2 15 8 6  
1 4 11 13 12 3 7 14 10 15 6 8 0 5 9 2  
6 11 13 8 1 4 10 7 9 5 0 15 14 2 3 12

## S8

13 2 8 4 6 15 11 1 10 9 3 14 5 0 12 7  
1 15 13 8 10 3 7 4 12 5 6 11 0 14 9 2  
7 11 4 1 9 12 14 2 0 6 10 13 15 3 5 8  
2 1 14 7 4 10 8 13 15 12 9 0 3 5 6 11

# DES X s-box - struktura

*S1*

Kolumna

Wiersz

Nr	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Input:  $z = (z_0, z_1, z_2, z_3, z_4, z_5)$ ,

$(z_0, z_5) = \{0, 1, 2, 3\}$  – reprezentacja dwójkowa numeru wiersza

$(z_1, z_2, z_3, z_4) = \{0..15\}$  – reprezentacja dwójkowa numeru kolumny

Przyk ad:  $z = (100111)$  wiersz  $(1, 1) = 3$ ; kolumna  $(0011) = 3$ ;

Wynik: 2 (0010)

# DES XI Funkcja P

16 7 20 21  
29 12 28 17  
1 15 23 26  
5 18 31 10  
2 8 24 14  
32 27 3 9  
19 13 30 6  
22 11 4 25

# DES XII – permutacja ko cowa

*IP<sup>-1</sup>*

40 8 48 16 56 24 64 32

39 7 47 15 55 23 63 31

38 6 46 14 54 22 62 30

37 5 45 13 53 21 61 29

36 4 44 12 52 20 60 28

35 3 43 11 51 19 59 27

34 2 42 10 50 18 58 26

33 1 41 9 49 17 57 25

# DES XIII – przyk ad efektu lawinowego

P1 = 12345677 (ASCII)

P2 = 12345676 (ASCII)

K = 1234567890abcd (HEX)

K = 00010010 00110100 01010110 01111000 10010000 10101011 11001101 (BIN)

C1 = DES<sub>K</sub>(P1)

C2 = DES<sub>K</sub>(P2)

P1 = 00110001 00110010 00110011 00110100 00110101 00110110 00110111 00110111

C1 = 01000110 00101101 10101110 10011101 01001110 10111001 11101000 01000111

P2 = 00110001 00110010 00110011 00110100 00110101 00110110 00110111 00110110

C2 = 01001100 10100000 01010100 10101001 00101110 01101100 11100001 10000110

## DES XIV deszyfrowanie

- Dane – szyfrogram
- Ten sam klucz
- Ten sam algorytm, ale z odwrotno kolejno ci kluczy  $K_i$ :  $K_{16} \dots K_1$
- Wynik – tekst jawny

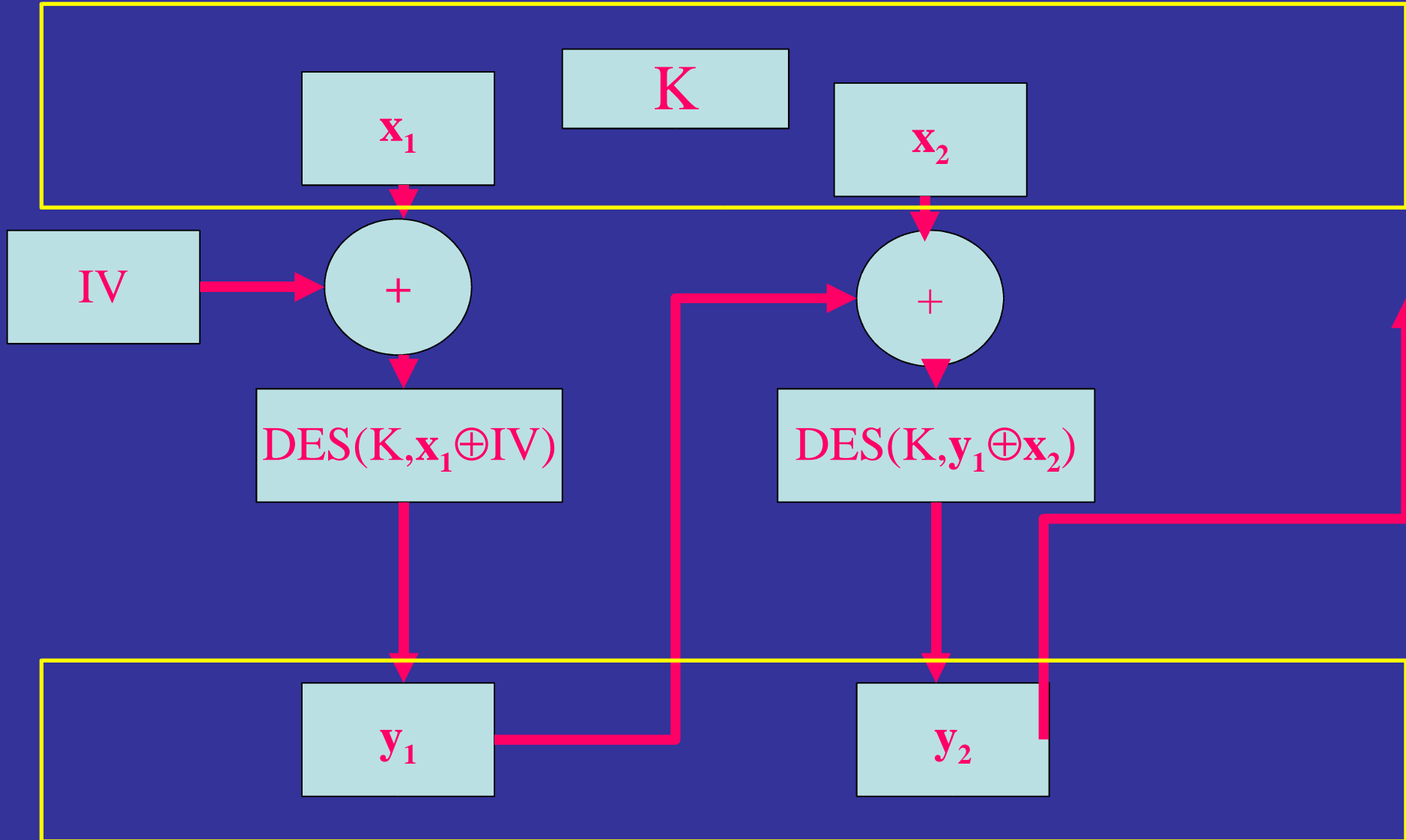
# DES XV Uwagi I

- Kontrowersje otaczające proces projektowania DES
- Długość klucza
- Własności s-boxów
- $DES(\sim K, \sim x) = \sim DES(K, x)$
- DES ma cztery s-aby klucze  $\kappa_i$  :  $DES_{\kappa}(DES_{\kappa}(x)) = x$ ; oraz sześć par pó-s-abych kluczy  $\eta_i, \gamma_i$  :  $DES_{\eta}(DES_{\gamma}(x)) = x$ ;  
Przykład s-abych kluczy (HEX): 0101 0101 0101 0101;  
FEFE FEFE FEFE FEFE; 1F1F 1F1F 0E0E 0E0E; E0E0  
E0E0 F1F1 F1F1
- DES nie tworzy grupy. Gdyby DES tworzy grupę, dla pary kluczy  $K_1, K_2$  istniałby klucz  $K_3$  taki, że dla każdego  $x$   $DES_{K_3}(x) = DES_{K_2}(DES_{K_1}(x))$

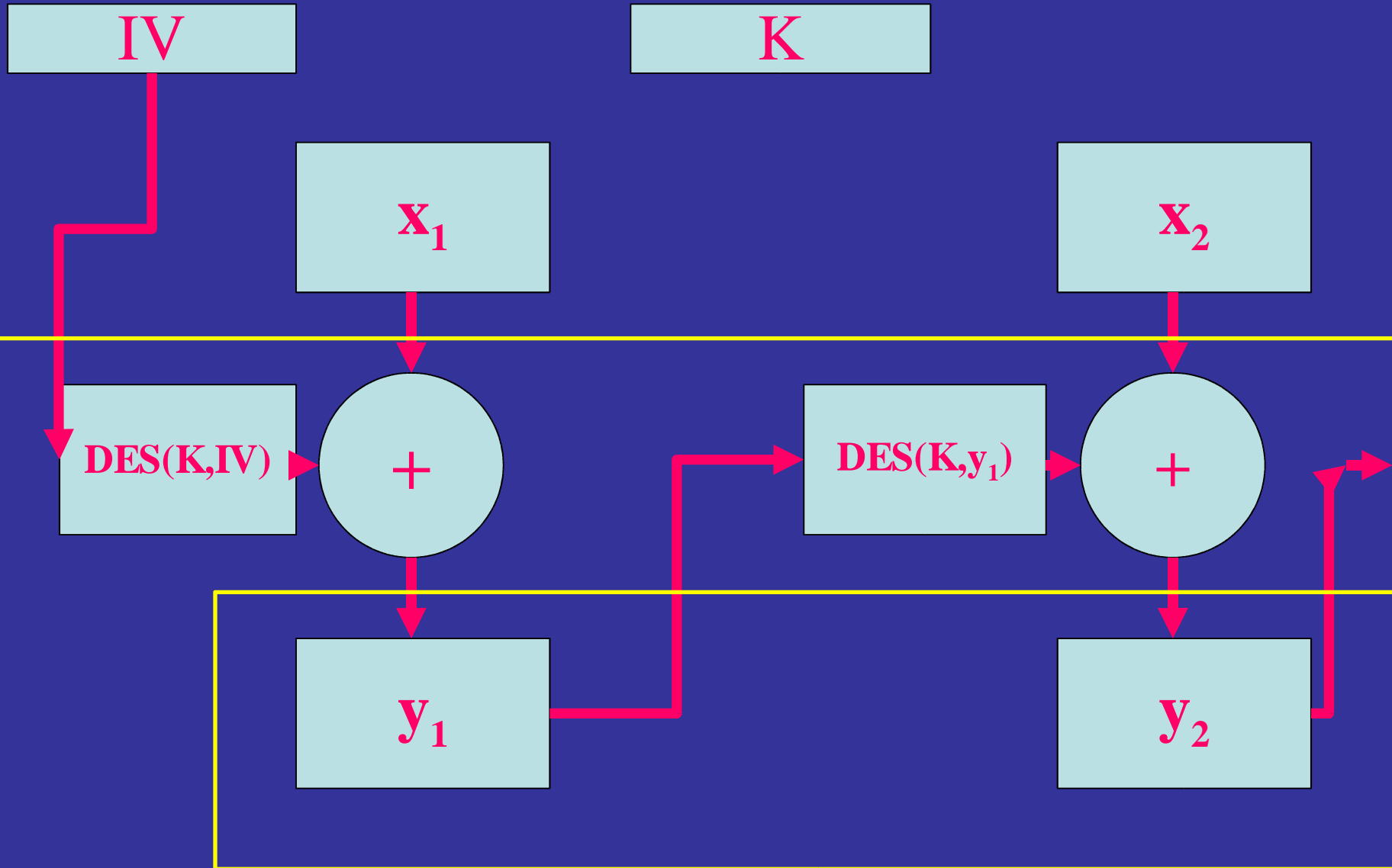
# DES XVI Tryby pracy

- Tryb elektronicznej księgi kodowej (ECB): każdy blok jest szyfrowany oddzielnie
- Tryb wiązania bloków zaszyfrowanych (CBC): dane wejściowe stanowią XOR następujących 64 bitów tekstu jawnego i poprzednich 64 bitów szyfrogramu. Pierwsze dane stanowią wektor inicjalizujący.
- Tryb szyfrowania ze sprzężeniem zwrotnym (CFB): dane wejściowe są przetwarzane po  $l$  bitów za jednym razem. Uprzedni tekst zaszyfrowany służy jako dane wejściowe do algorytmu, który produkuje pseudolosowy ciąg bitów. Ten ciąg służy jako klucz dodawany (XOR) do danych wejściowych.
- Tryb sprzężenia zwrotnego wyjściowego (OFB): synchroniczny szyfr strumieniowy. Strumień klucza wytwarzany jest przez iteracyjne szyfrowanie wektora inicjalizującego  $IV$ . Szyfrogram jest następnie tworzony jako XOR strumienia danych i klucza.

# DES XVII CBC (cipher block chaining)



# DES XVIII CFB (cipher feedback)



# DES XIX Atak

