Capítulo 4

Criptografia Convencional

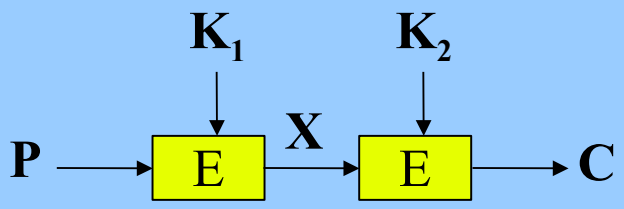
Algoritmos

Plano de Curso

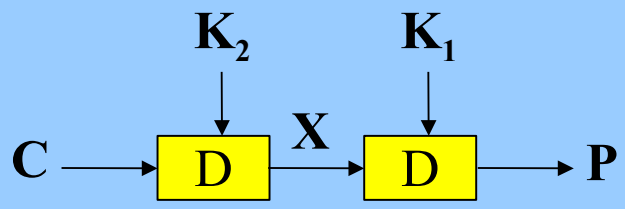
- Triplo DES
- IDEA
- Blowfish
- RC5
- CAST-128
- RC2
- Características dos Cifradores de Bloco Simétrico Avançados

Cifração Dupla

Cifrar $C=E_{K_2}[E_{K_1}[P]]$



Decifrar $P=D_{K_1}[D_{K_2}[C]]$



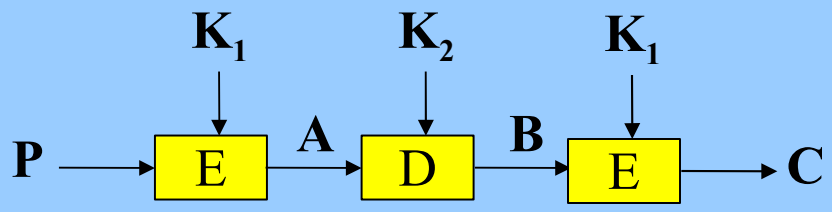
<u>56 x 2=112 bits ?</u>

$$(2^{64})!=10^{34380000000000000000} > (10^{10^{20}})$$

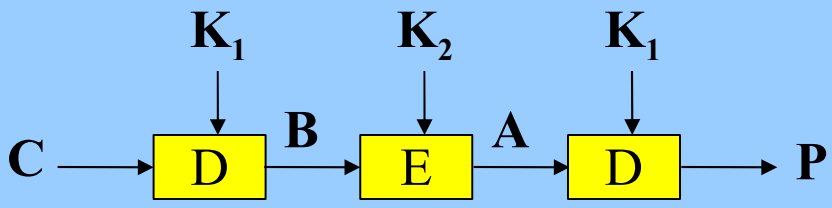
$$2^{56} < 10^{17}$$

Cifração Tripla

Cifrar



Decifrar



IDEA

Algoritmo Internacional de Criptografia de Dados

Desenvolvido por: Xuejia Lai e James Massey
Instituto Federal de Tecnologia Suíço
1990 - Proposta Original
1991 - Revisão

Objetivo:
Substituir o DES

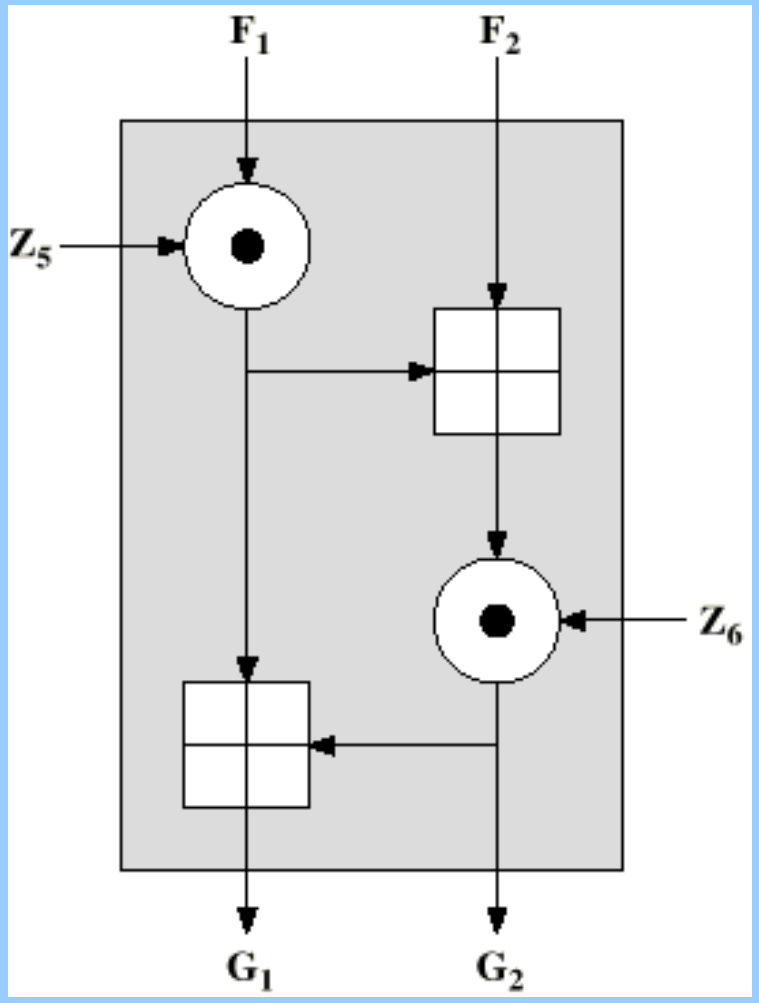
Características:
- Comprimento do Bloco: 64 bits
- Comprimento da Chave: 128 bits
- Confusão: Três Diferentes Operações
- Difusão

Operações:
$\oplus$ - XOR
$\boxplus$ - Adição mod 2^{16}
$\odot$ - Multiplicações mod $2^{16}+1$

MA - Estrutura da Multiplicação/Adição

Difusão

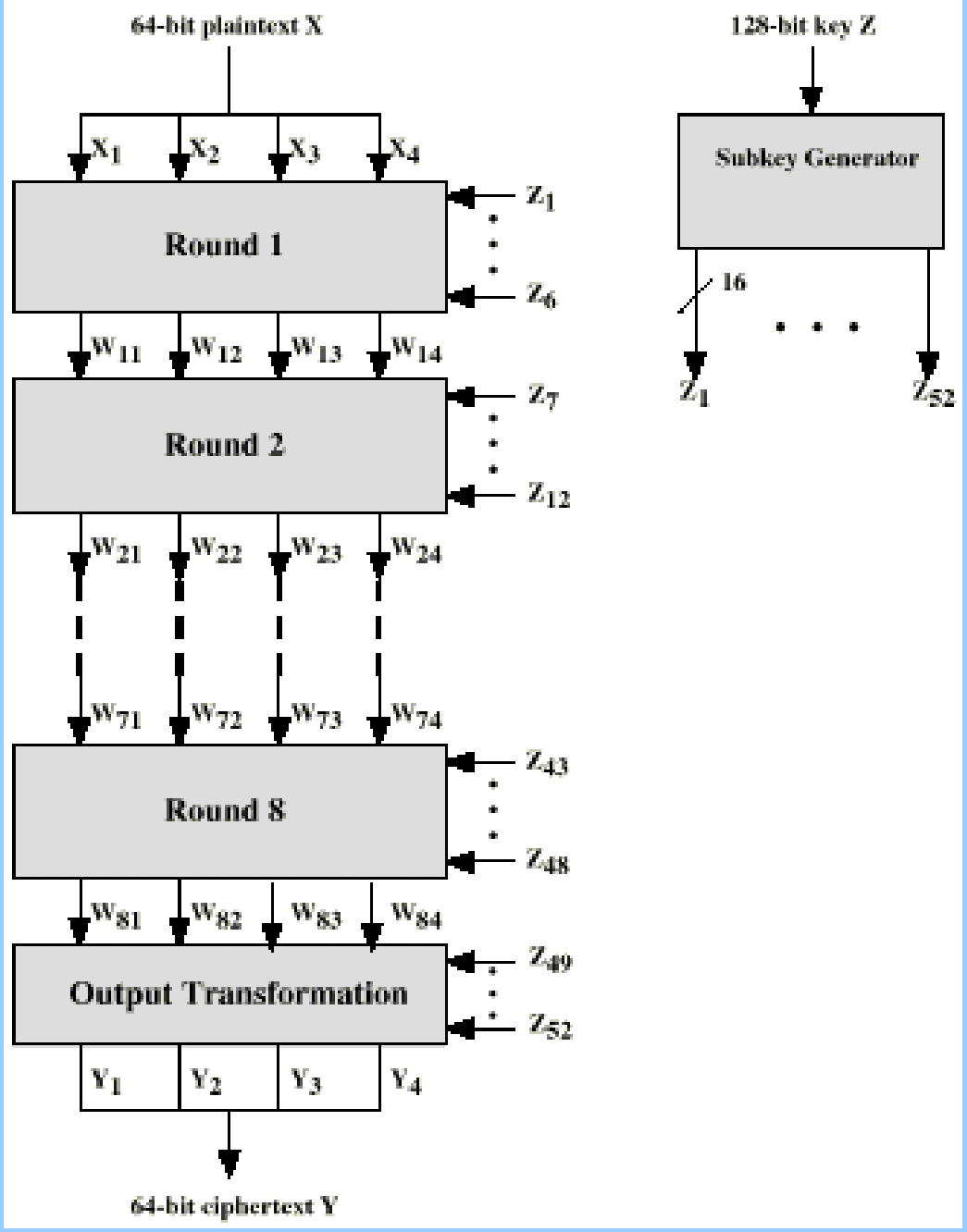
Cada bit de saída depende:
- Todo bit de entrada
- Todo bit da chave



Considerações de Implementação

- Software
  - Sub-blocos naturais (8, 16 ou 32 bits)
  - Uso de simples operações
- Hardware
  - Similaridade entre encriptar e decriptar
  - Estrutura Regular - facilitar VLSI

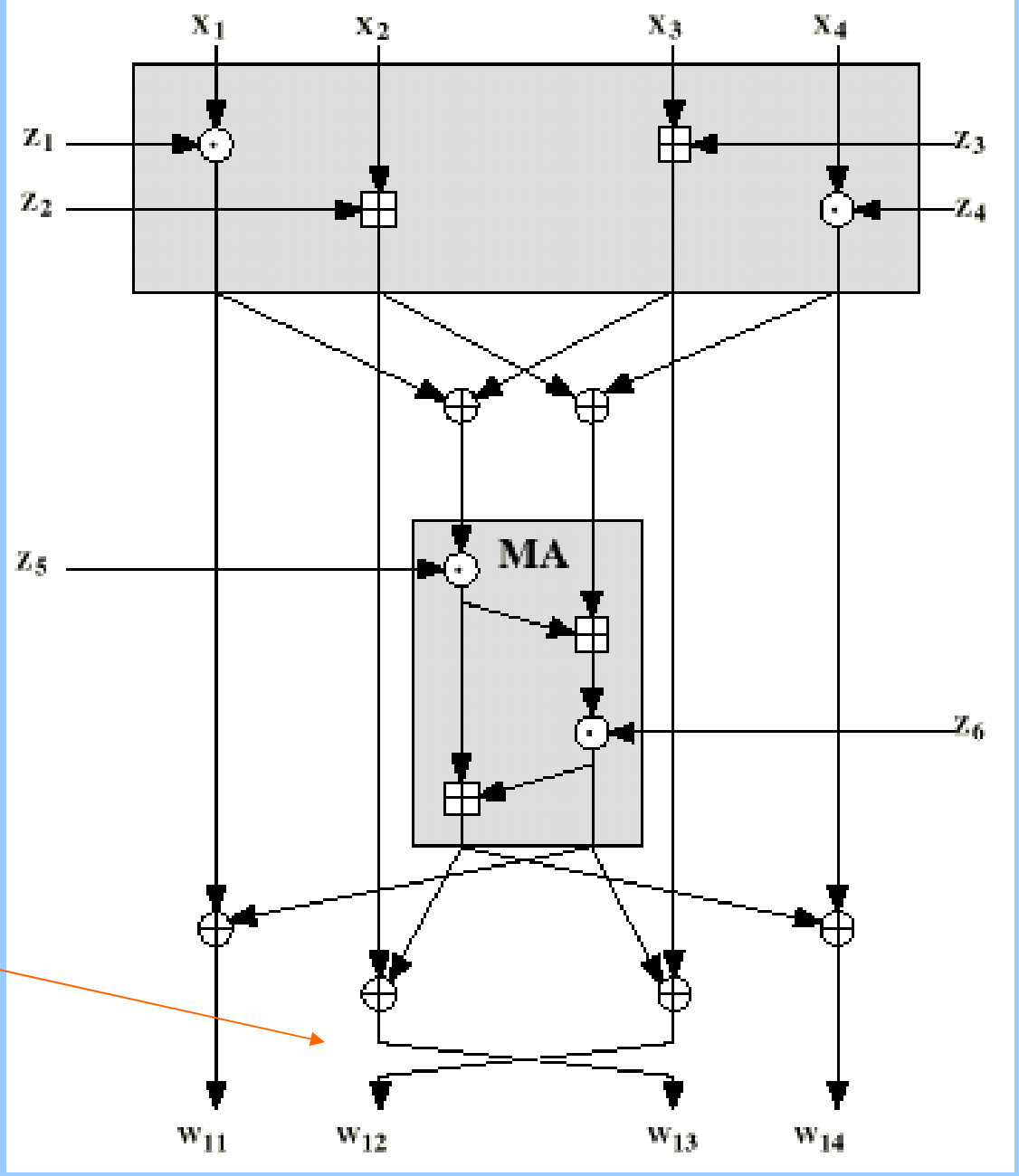
Estrutura Geral do IDEA



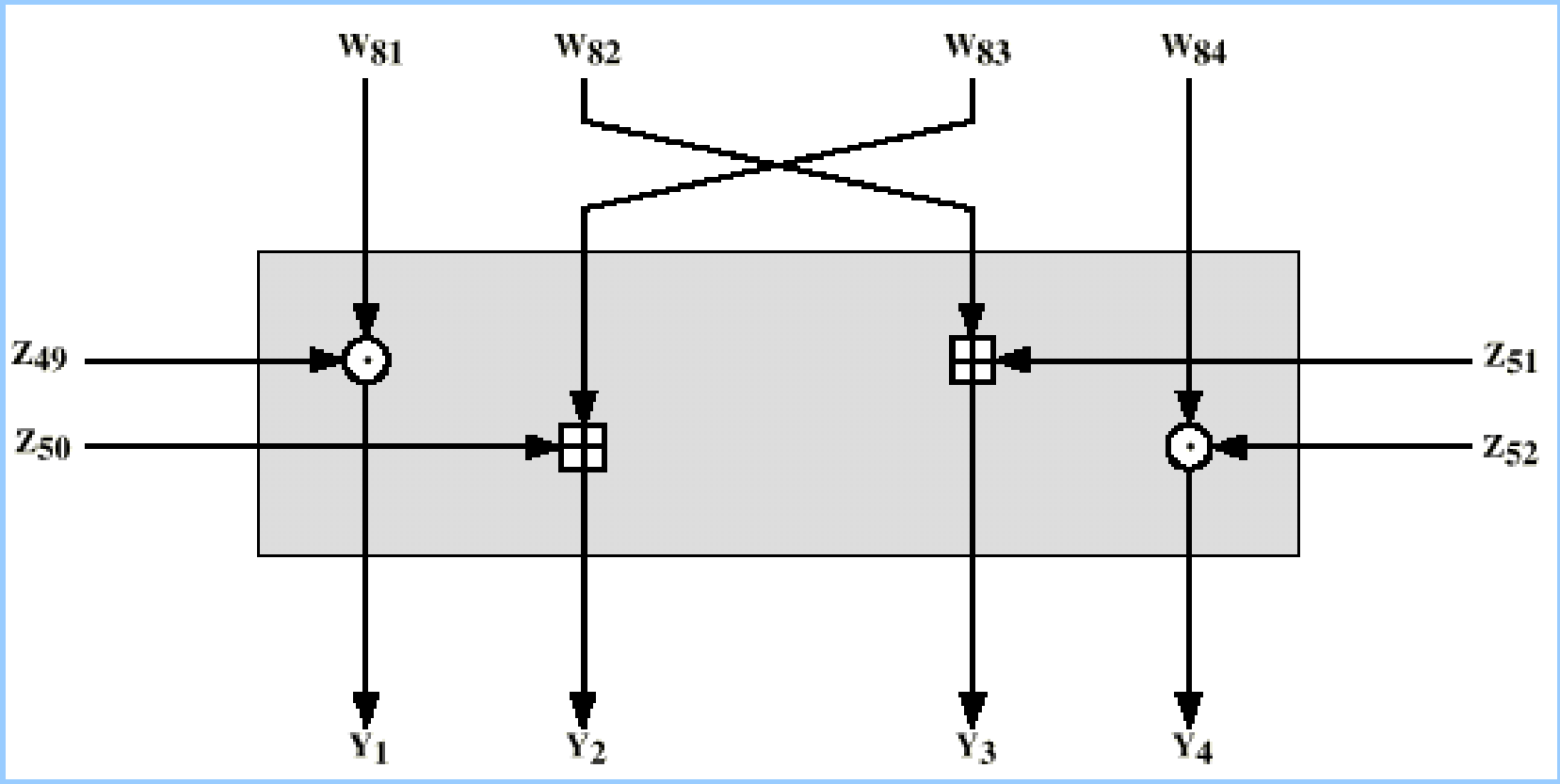
Primeira Fase do IDEA

Desvia-se da estrutura clássica de Feistel

+ resistente a criptoanálise diferencial

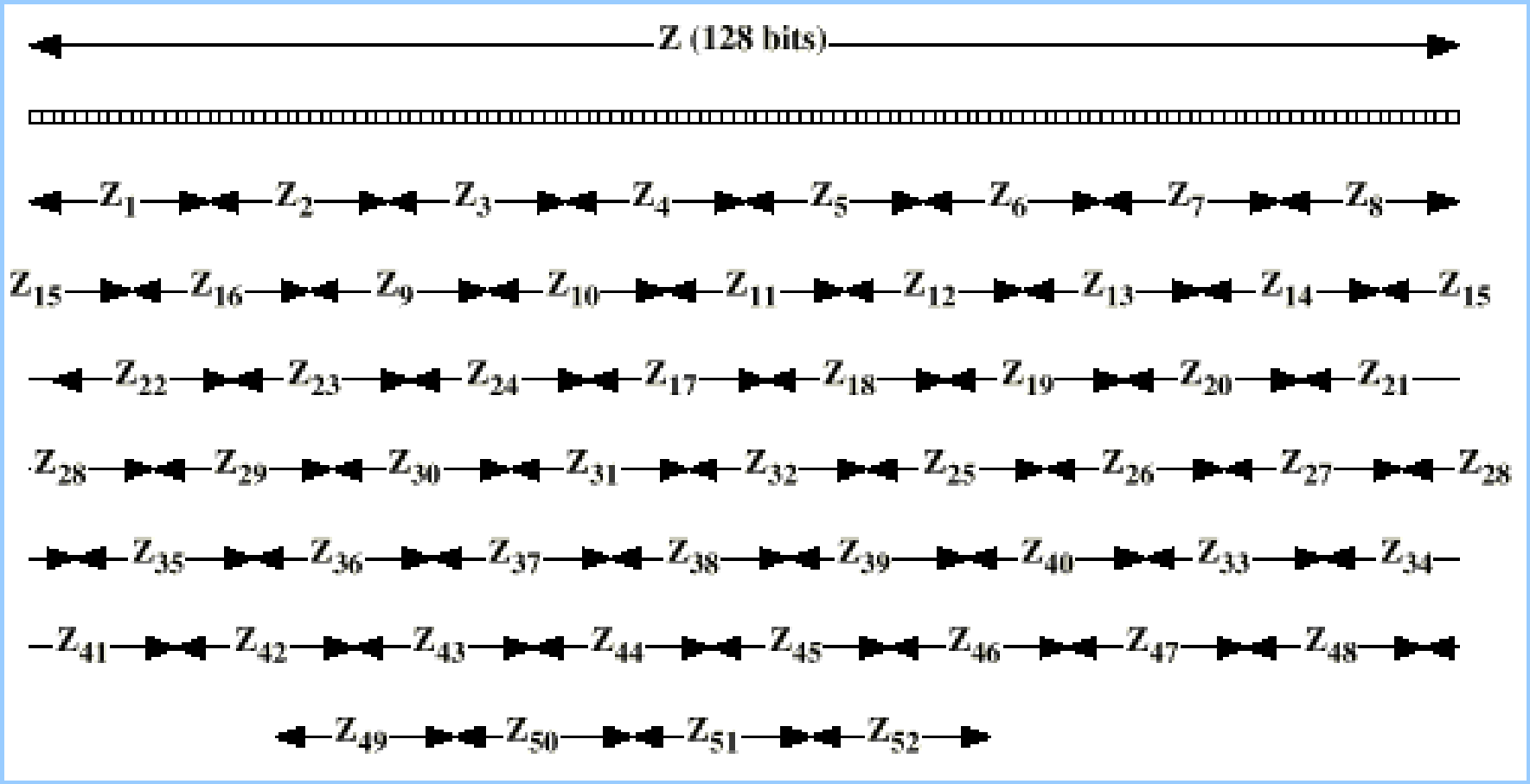


Transformação da Saída do IDEA

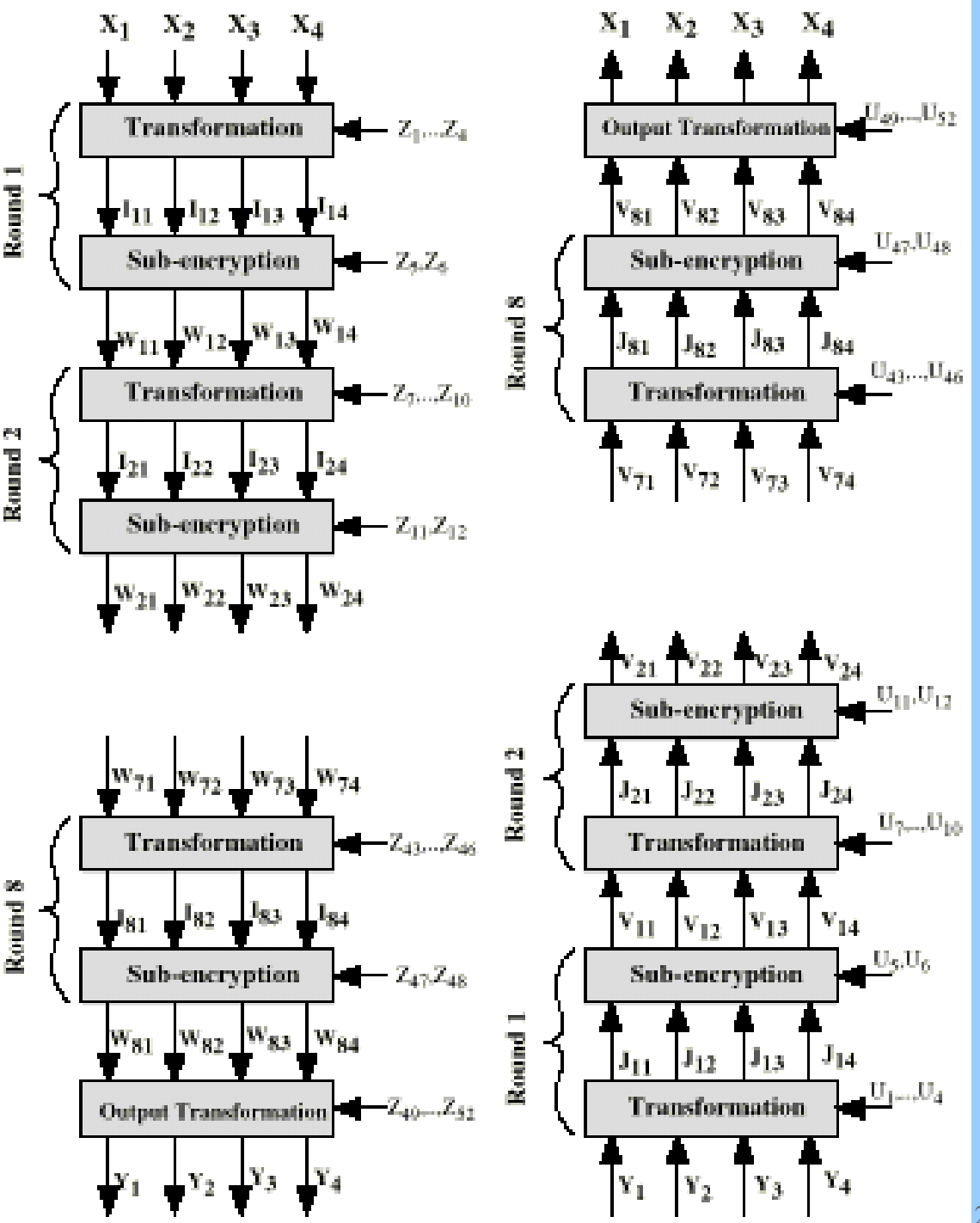


Subchaves do IDEA

25 bits de deslocamento circular a esquerda



Ciframento e Deciframento do IDEA



Formação das Subchaves

Cifrar			Decifrar	
Fase	**Designação**	**Equivalente a**	**Designação**	**Equivalente a**
1	Z_1,Z_2,Z_3,Z_4,Z_5,Z_6	Z[1..96]	U_1,U_2,U_3,U_4,U_5,U_6	$Z_{49}^{-1},Z_{50},Z_{51},Z_{52}^{-1},Z_{47},Z_{48}$
2	$Z_7,Z_8,Z_9,Z_{10},Z_{11},Z_{12}$	Z[97..128,26..89]	$U_7,U_8,U_9,U_{10},U_{11},U_{12}$	$Z_{43}^{-1},Z_{45},Z_{44},Z_{46}^{-1},Z_{41},Z_{42}$
3	$Z_{13},Z_{14},Z_{15},Z_{16},Z_{17},Z_{18}$	Z[90..128,1..25,51..82]	$U_{13},U_{14},U_{15},U_{16},U_{17},U_{18}$	$Z_{37}^{-1},Z_{39},Z_{38},Z_{40}^{-1},Z_{35},Z_{36}$
4	$Z_{19},Z_{20},Z_{21},Z_{22},Z_{23},Z_{24}$	Z[83..128,1..50]	$U_{19},U_{20},U_{21},U_{22},U_{23},U_{24}$	$Z_{31}^{-1},Z_{33},Z_{32},Z_{34}^{-1}, Z_{29},Z_{30}$
5	$Z_{25},Z_{26},Z_{27},Z_{28},Z_{29},Z_{30}$	Z[76..128,1..43]	$U_{25},U_{26},U_{27},U_{28},U_{29},U_{30}$	$Z_{25}^{-1},Z_{27},Z_{26},Z_{28}^{-1},Z_{23},Z_{24}$
6	$Z_{31},Z_{32},Z_{33},Z_{34},Z_{35},Z_{36}$	Z[44..75,101..128,1..36]	$U_{31},U_{32},U_{33},U_{34},U_{35},U_{36}$	$Z_{19}^{-1},Z_{21},Z_{20},Z_{22}^{-1},Z_{17},Z_{18}$
7	$Z_{37},Z_{38},Z_{39},Z_{40},Z_{41},Z_{42}$	Z[37..100,126..128,1..29]	$U_{37},U_{38},U_{39},U_{40},U_{41},U_{42}$	$Z_{13}^{-1},Z_{15},Z_{14},Z_{16}^{-1},Z_{11},Z_{12}$
8	$Z_{43},Z_{44},Z_{45},Z_{46},Z_{47},Z_{48}$	Z[30..125]	$U_{43},U_{44},U_{45},U_{46},U_{47},U_{48}$	$Z_7^{-1},Z_9,Z_8,Z_{10}^{-1},Z_5,Z_6$
Transf.	$Z_{49},Z_{50},Z_{51},Z_{52}$	Z[23..86]	$U_{49},U_{50},U_{51},U_{52}$	$Z_1^{-1},Z_2,Z_3,Z_4^{-1}$

$$Z_j \odot Z_j^{-1} = 1$$

$$-Z_j \boxplus Z_j = 0$$

RC5

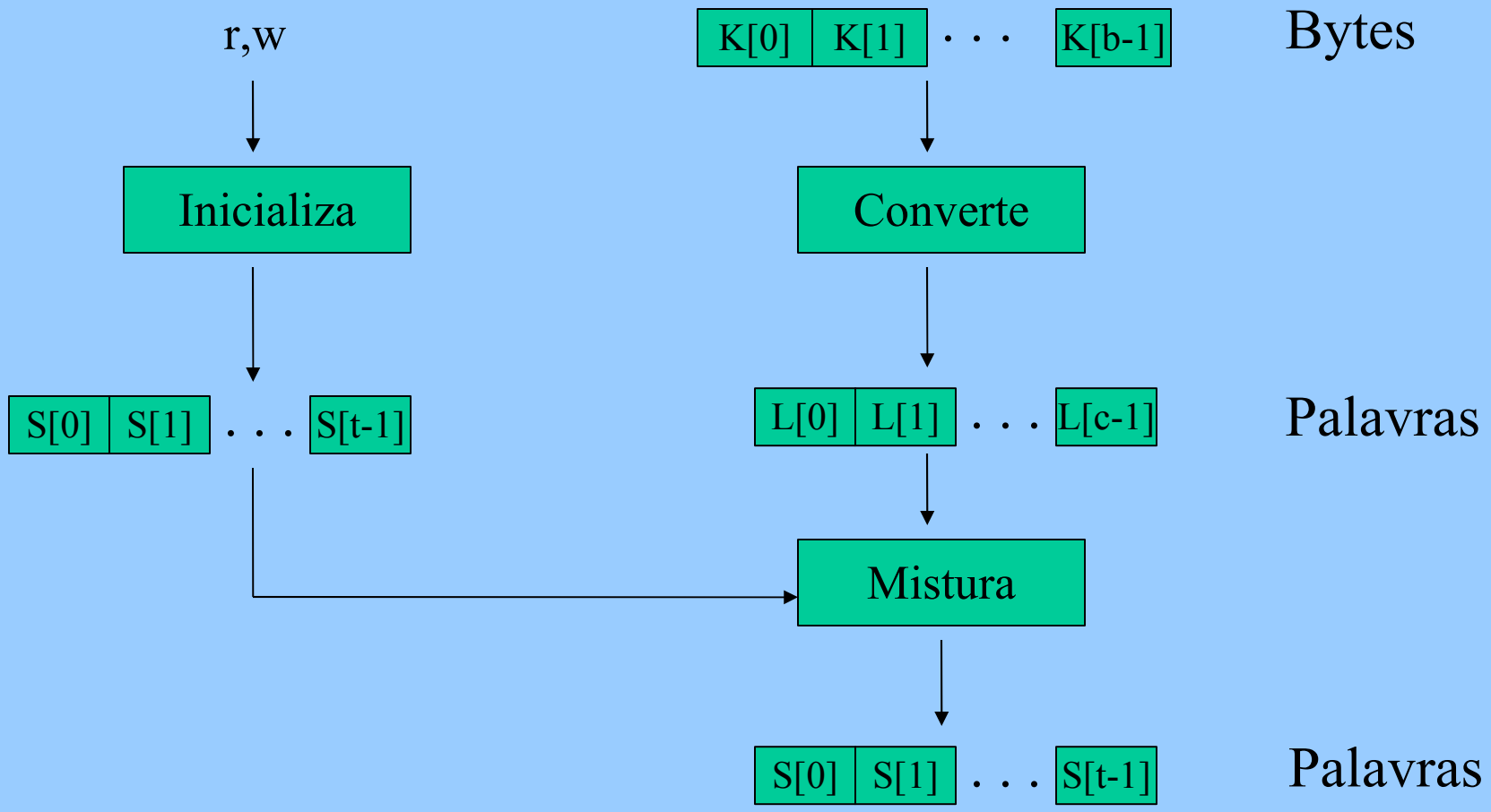
Desenvolvido por: Ron Rivest, 1994

Características:
- Adequado a hardware e software
- Rápido
- Adaptável a diferentes CPUs
- Número variável de rodadas
- Tamanho variável da chave
- Simples
- Necessita de pouca memória
- Alta segurança
- Rotação dependente dos dados
- Tamanho do Bloco: 32, 64 ou 128

Parâmetros, Definição e Valores Permitidos
w - Tamanho da palavra - 16, 32 ou 64
r - Número de fases - 0, 1, ..., 255
b - Número de bytes da chave - 0, 1, ..., 255

Versão Nominal
RC5-w/r/b = RC5-32/12/16

Expansão da Chave



Inicializa

$P_w = Odd[(e-1)2^w]$

$Q_w = Odd[(\phi-12^w)]$

$e = 2{,}718281828459$

$\phi = 1{,}618033988749 = (1+\sqrt{5})/2$

w	16	32	64
P_w	B7E1	B7E15163	B7E151628AED2A6B
Q_w	9E37	9E3779B9	9E3779B97F4A7C15

$S[0] = P_w$

Para i = 1 até t-1 faça

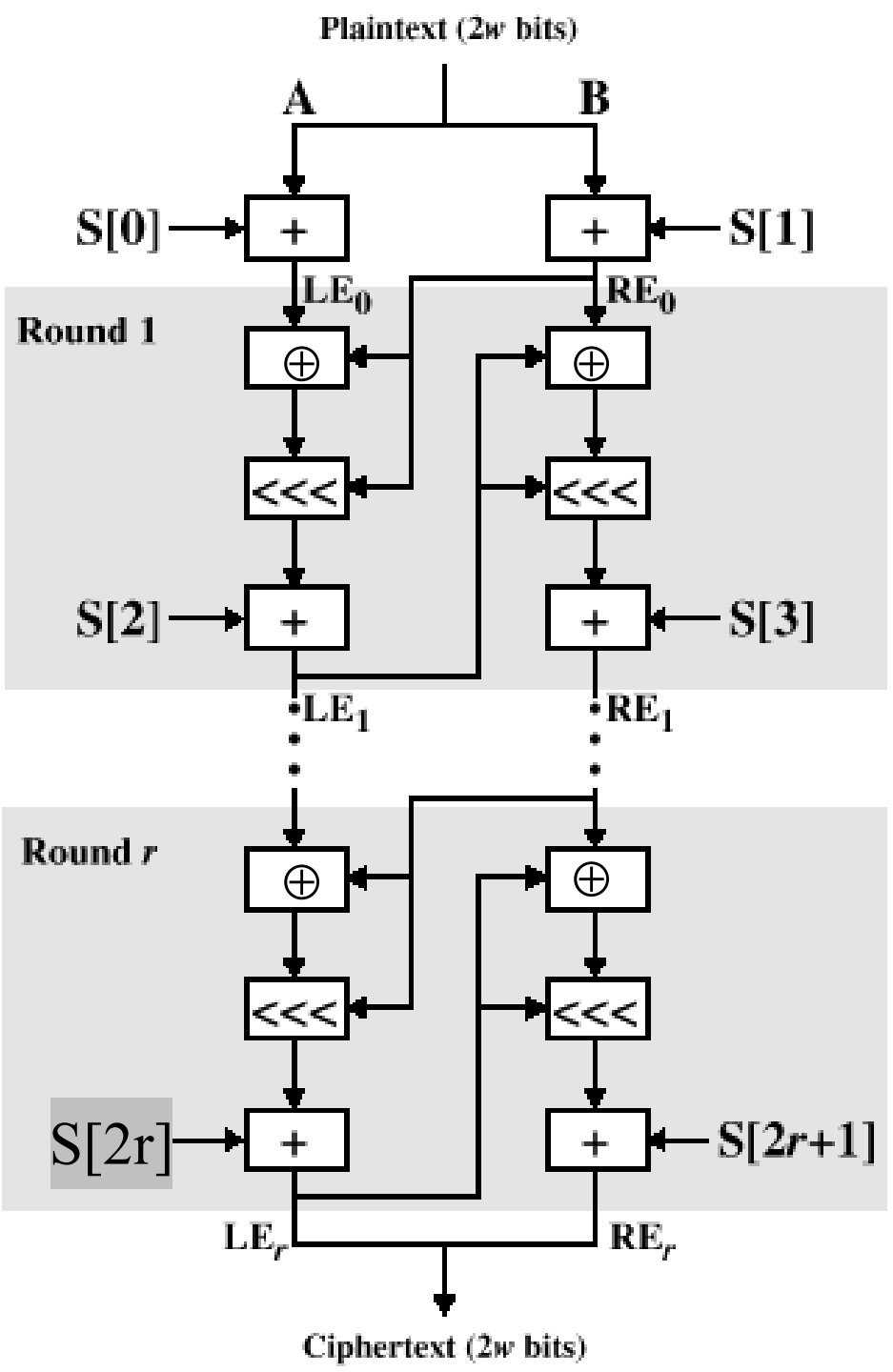
$S[i] = S[i-1] + Q_w$

Mistura

```
i = j = X = Y = 0
Repita 3 x max(t,c) vezes
    S[i] = (S[i] + X + Y)   <<<   3
    X = S[i]
    i = (i+1) mod t
    L[j] = (L[j] + X + Y)   <<<   (X+Y)
    Y = L[j]
    j = (j+1) mod c
Fim Repita
```

Cifrar

LE0 = A + S[0]
RE0 = B + S[1]
Para i = 1 até r Faça
$LE_i = ((LE_{i-1} \oplus RE_{i-1}) <<< RE_{i-1}) + S[2i]$
$RE_i = ((RE_{i-1} \oplus LE_i) <<< LE_i) + S[2i+1]$
Fim Para



Decifrar

Para i = r até 1 Passo 01 Faça

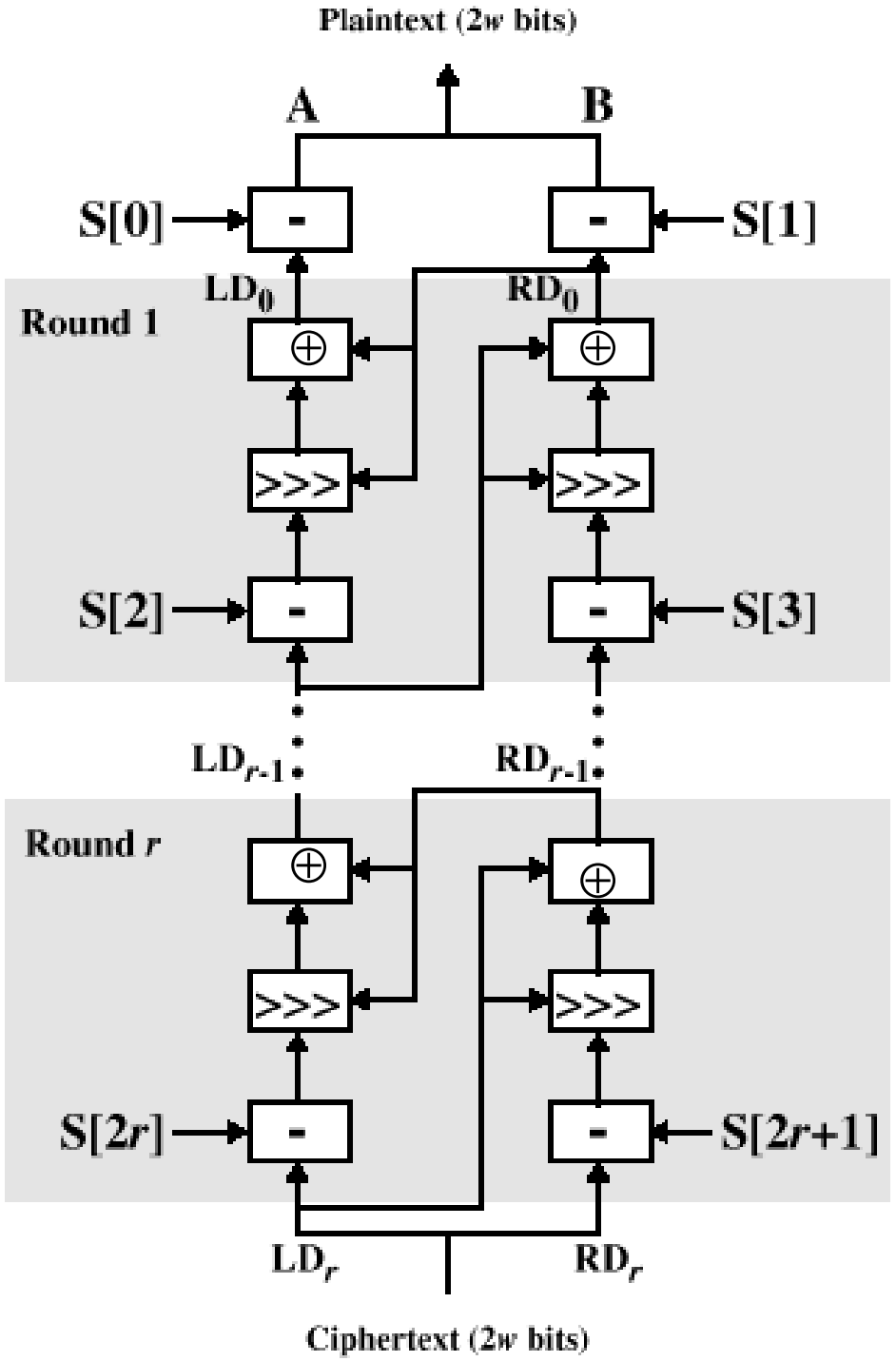
$RD_{i-1} = ((RD_i - S[2i+1] >>> LD_i) \oplus LD_i$

$LD_{i-1} = ((LD_i - S[2i] >>> RD_{i-1}) \oplus RD_{i-1}$

Fim Para

$B = RD_0 - S[1]$

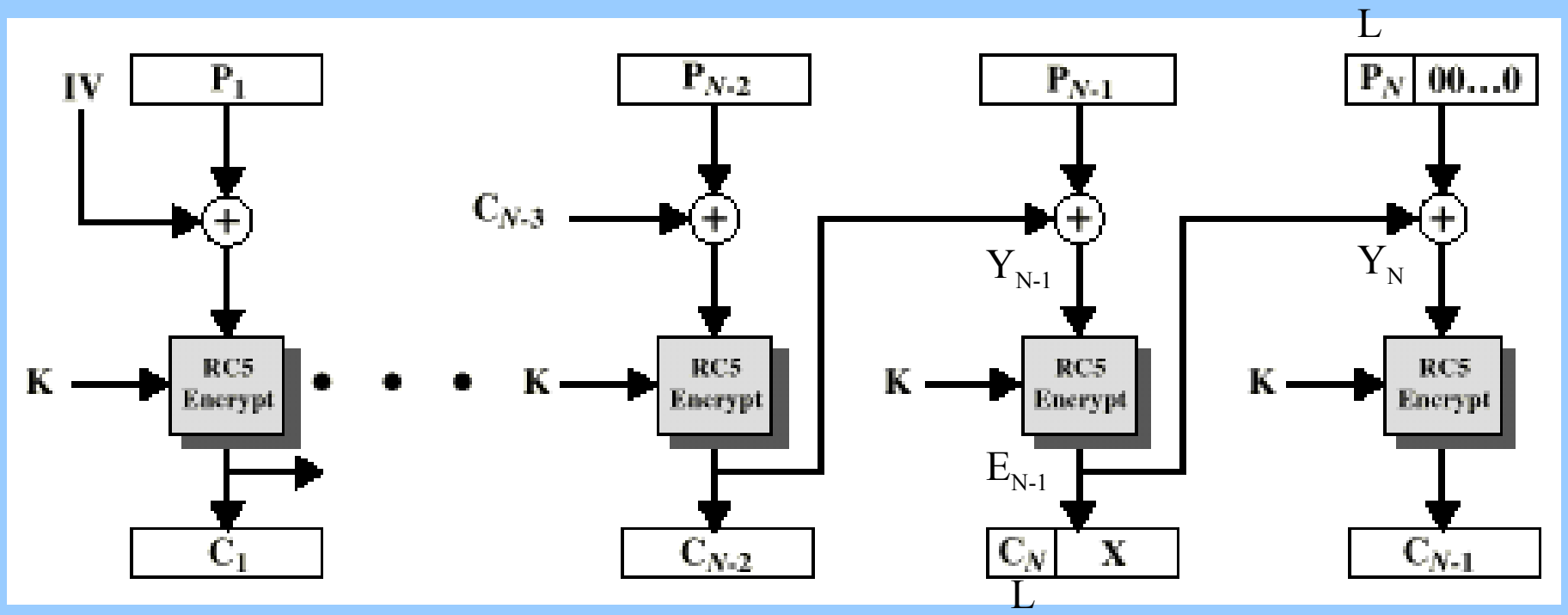
$A = LD_0 - S[0]$



Modos de Operação

- ECB
- CBC
- CBC Pad 00001000
- CTC - Texto Cifrado Roubado

Modo Texto Cifrado Roubado (RFC 2040)



CAST-128 (RFC 2144)

Desenvolvido por: **C**arlisle **A**dams e **S**tafford **T**avares, 1997

Características:
Tamanho de chave Variável: 40 a 128 bits (8 em 8)
Tamanho do bloco: 64 bits
Estrutura clássica de Feistel
Função F depende da fase
Usa duas subchaves em cada fase

Cifrar

$L_0 \| R_0 = P$
Para i = 1 até 16 Faça
    $L_i = R_{i-1}$
    $R_i = L_{i-1} \oplus F_i[R_{i-1}, Km_i, Kr_i]$
Fim Para
$C = R_{16} \| L_{16}$

Decifrar

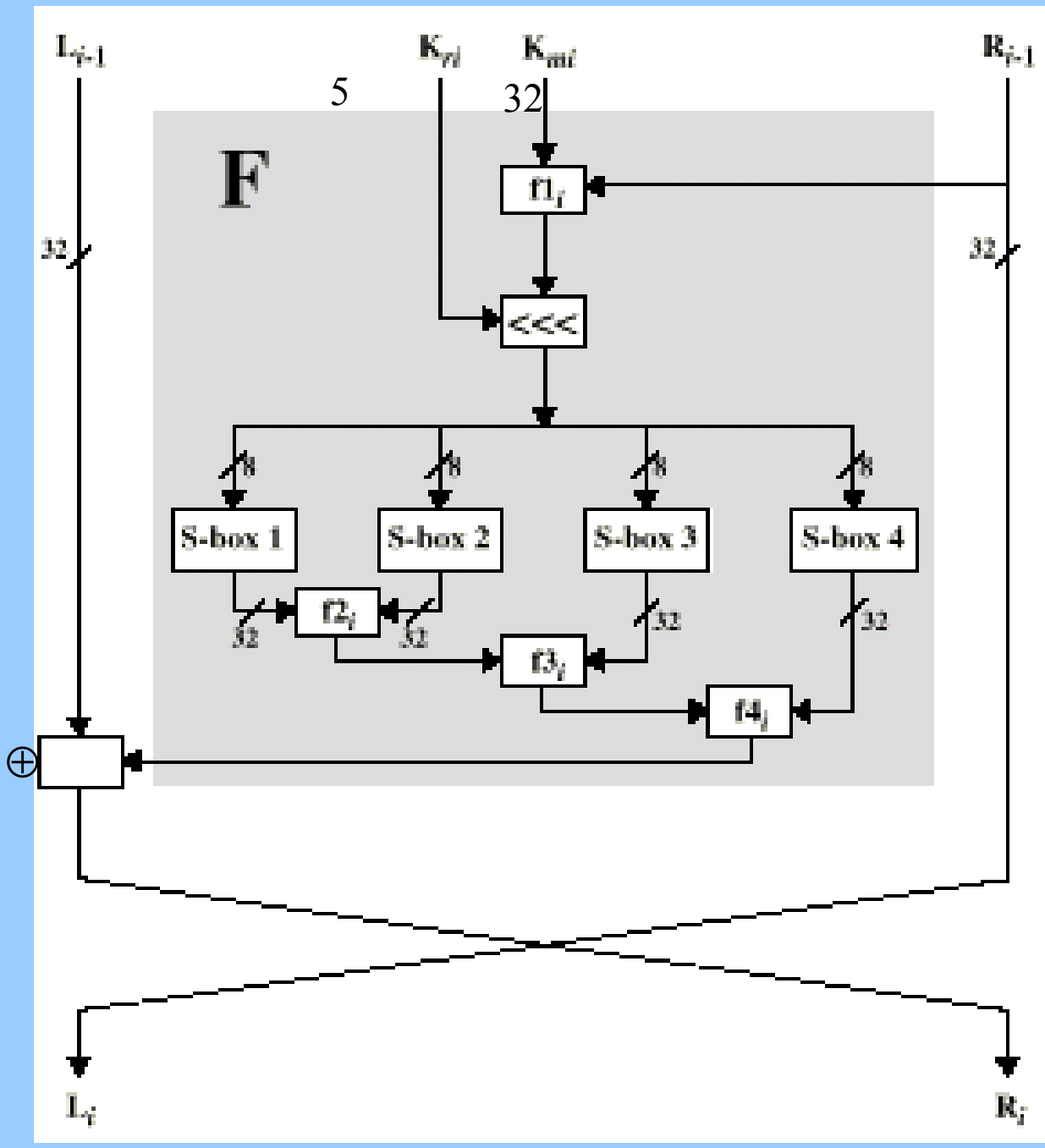
Chave em ordem inversa

Uma simples fase do CAST-128

Rodadas: 1, 4, 10, 13, 16
f1 = +
f2 = ⊕
f3 = -
f4 = +

Rodadas: 2, 5, 8, 11, 14
f1 = ⊕
f2 = -
f3 = +
f4 = ⊕

Rodadas: 3, 6, 9, 12, 15
f1 = -
f2 = +
f3 = ⊕
f4 = -



RC2

Desenvolvido por: Ron Rivest

Características:
Tamanho de chave Variável: 8 a 1024 bits (8 em 8)
Tamanho do bloco: 64 bits
Fácil de implementar em CPUs 16 bits
Usado em S/MIME com chaves de 40, 64 e 128 bits

Características dos Cifradores Simétricos Avançados

- Tamanho variável da chave
- Operações mistas
- Rotação dependente dos dados
- Rotação dependente da Chave
- Caixas S dependente da Chave
- Algoritmo de geração das subchaves
- Função F variável
- Comprimento do bloco variável
- Número variável de fases
- Operação nas duas metades dos dados em cada fase