Data Encryption Standard (DES)

GA18

Nicolas T. Courtois

University College London, UK

DES
history/standardisation/speed

DES

- Federal Standard FIPS 46-3
- Intended to be used to protect all US government communications… First and the only encryption algorithm known for many years.
- Adopted by all the other countries, (incapacity to design their own cryptographic algorithm that would not broken by the NSA ?).
  - Russia: GOST, different S-boxes can be specified.
- Used by almost anyone… - a de facto industry standard.
- 3-DES still used a lot in banking/financial sector (e.g. in bank cards). Replaced by AES slowly, over 20 years (!).

DES

- July 26, 2004:
  NIST announces withdrawal of DES.
- Withdrawn a bit late…

Can be broken in 1 day now…

- <u>Amateur:</u> 2^{55} * 400 cycles CPU,
  - Less than 1 year, 200 PCs, 3Ghz
- <u>Smart:</u> FPGA implementation
  - 1 year on FPGA, cost about 5000 $
  - 1 month if we have 60 K$ etc…
- <u>Large budget:</u> ACICS, DES chips:
  - Few hours with a budget of about 1M $.

[Schneier reports that in the 80s Russia did order 100 000 DES chips from Eastern Germany Robotron]

DES Speed

In cycles on Pentium 3.

- key setup: 883
- encrypt: 472

(59 cycles per byte,

cf. AES-128 = 25 cycles per byte)

Cost of Exhaustive Search ?

2^{55} * 472 cycles on Pentium 3.

Gives 2^{64} cycles (CPU clocks) !

2 GHz => 2^{31} cycles per second.

2^{43} cycles per hour.

2^{47} cycles per day

2^{55} cycles per year, still not enough.

=> Even today we need $2^9 \approx 500$ PCs to break DES in 1 year. (much faster with FPGAs…)

DES Speed – Smart Cards (1)

Low-end smart card:

- Software DES - about 50 ms
- Software 3-DES – about 150 ms

(cf. software AES-128 – about 120 ms)

⇒ Most cards have Hardware DES

⇒ Few μs !!! (even on low-end).

DES Speed – Smart Cards (2)

SLE-66$_{\text{CX680PE}}$, 8051-based.

- Hardware DES - 3.5 μs
- Hardware 3-DES - 5.3 μs

(and hardware AES-128 - 85 μs on recent ST22)
comparatively slow, AES requires much more surface that DES !!!)

=> Several times more (2,3,5,10 times…)
if side-channel attacks are taken into consideration !!

DES
basics

DES basics

- 64-bit blocks (8 bytes)
- effective key size: 56 bit (reduced on purpose by the NSA)
- key written as 64 bits: use 7 bits / byte,

One parity bit.

- Most authors use incompatible bit numberings…
  - (FIPSPUB-46) = 32 – (Matsui numbers)

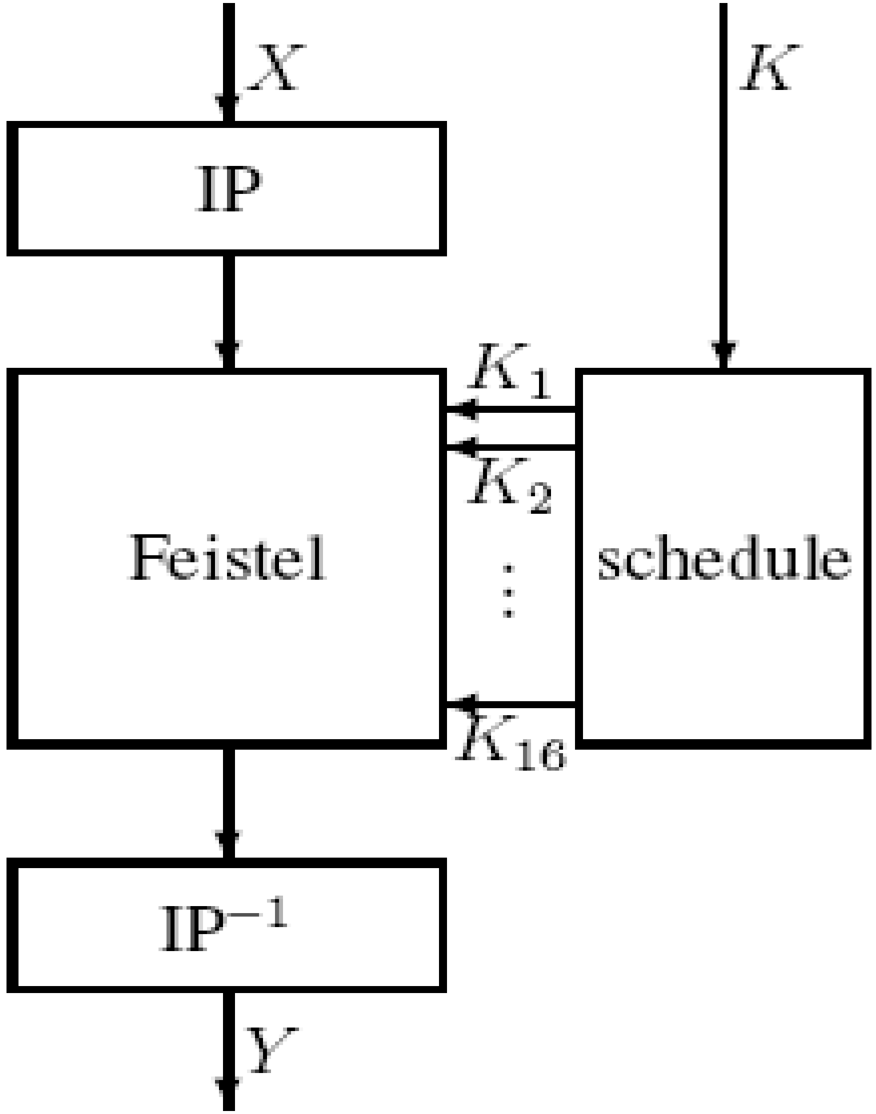
Outline

- Left:

encryption channel

- Right:

Key scheduling:



Outline

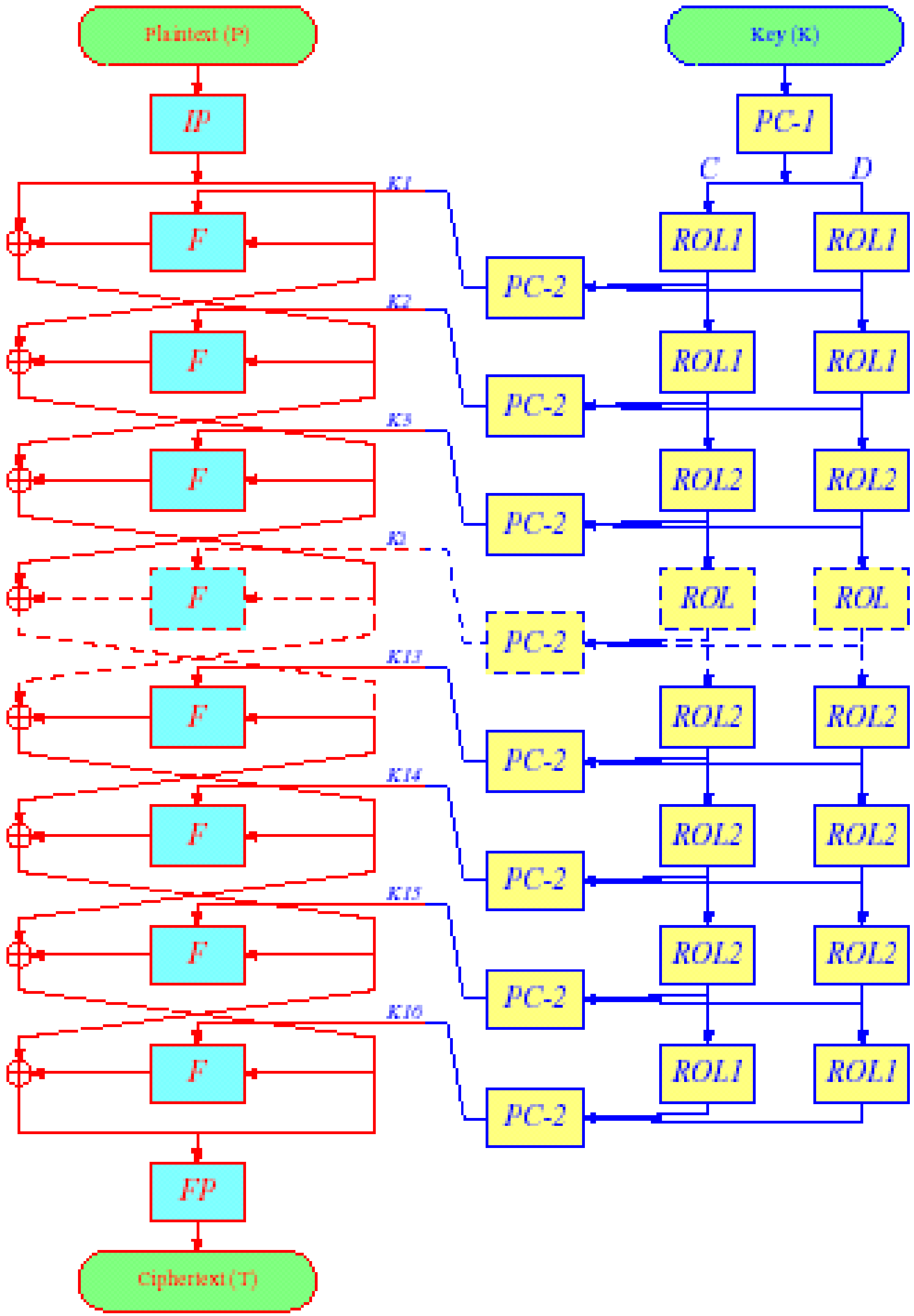
- Left:

encryption channel

- Right:

Key scheduling:

16*48 subsets of 56 bits.



Key Scheduling Details

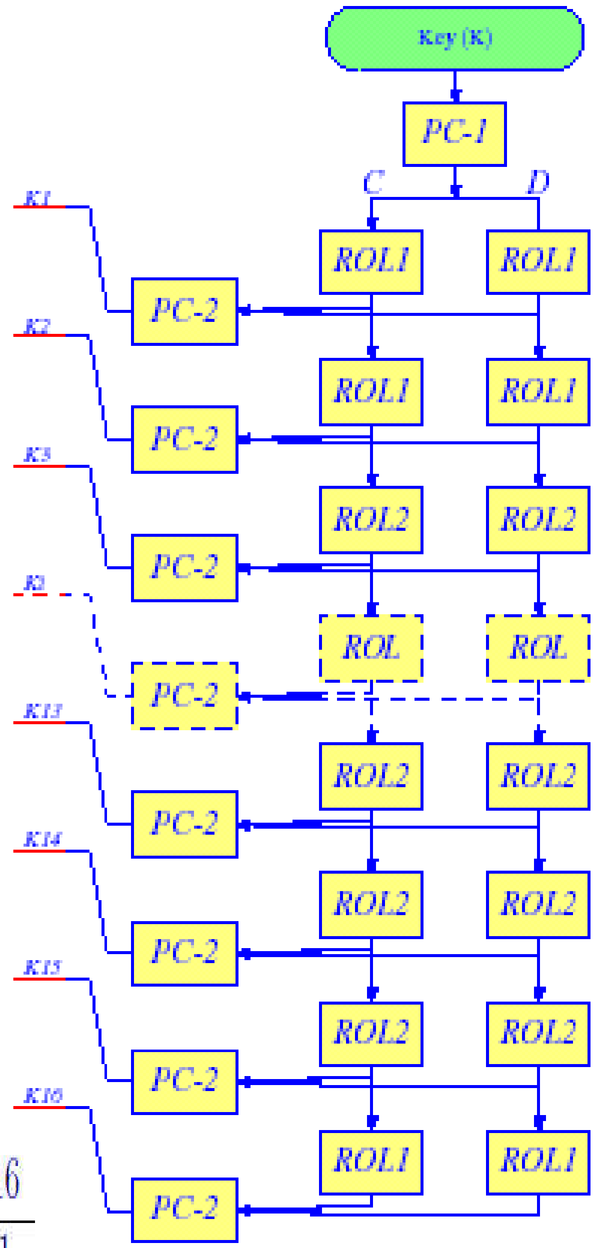
PC1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
above for C_i; below for D_i						
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC2					
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

16*48 subsets of 56 bits.

$K \xrightarrow{\text{PC1}} (C, D)$
for $i = 1$ to 16 **do**
    $C \leftarrow \text{ROL}_{r_i}(C)$
    $D \leftarrow \text{ROL}_{r_i}(D)$
    $K_i \leftarrow \text{PC2}(C, D)$
end for

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
r_i	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1



*Self-Similarity in Key Schedule

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
r_i	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

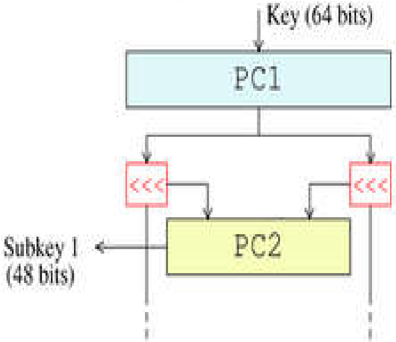
R1 R8 R15

- Can DES key be periodic?
- After step 1= key for R1
- After step 8=key for R8
- After step 15=key for R15
- We have a pattern G
  of length 7 which repeats twice.
- Unhappily G = + 13 mod 28,
- Does NOT have many fixed points.

*******another description [Vaudenay,MOV,etc]

The DES key schedule is done by the following algorithm. We use two registers C and D of 28 bits. The 56 key bits from K are first split into C and D following a fixed bit selection table PC1. Then each round rotates C and D bits by r_i positions depending on the round number i. (The r_i's are also defined by a table.) Then another bit selection table PC2 takes 24 bits from each of the two registers in order to make a round key.

1: $K \xrightarrow{\text{PC1}} (C, D)$
2: **for** $i = 1$ to 16 **do**
3: $\quad C \leftarrow \text{ROL}_{r_i}(C)$
4: $\quad D \leftarrow \text{ROL}_{r_i}(D)$
5: $\quad K_i \leftarrow \text{PC2}(C, D)$
6: **end for**



PC2					
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

PC1						
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
above for C_i; below for D_i						
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Here ROL_r is a circular rotation of r bits to the left. The r_i's are defined by

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
r_i	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Note that the sum of all r_i's is 28 so that we can generate the round keys in the decryption ordering by starting with the same C and D and by running the loop backwards.

Irregular Swap

At last round:

- encryption and decryption are identical - except order of keys.

Cheaper to implement in HW (reuse the same circuit)

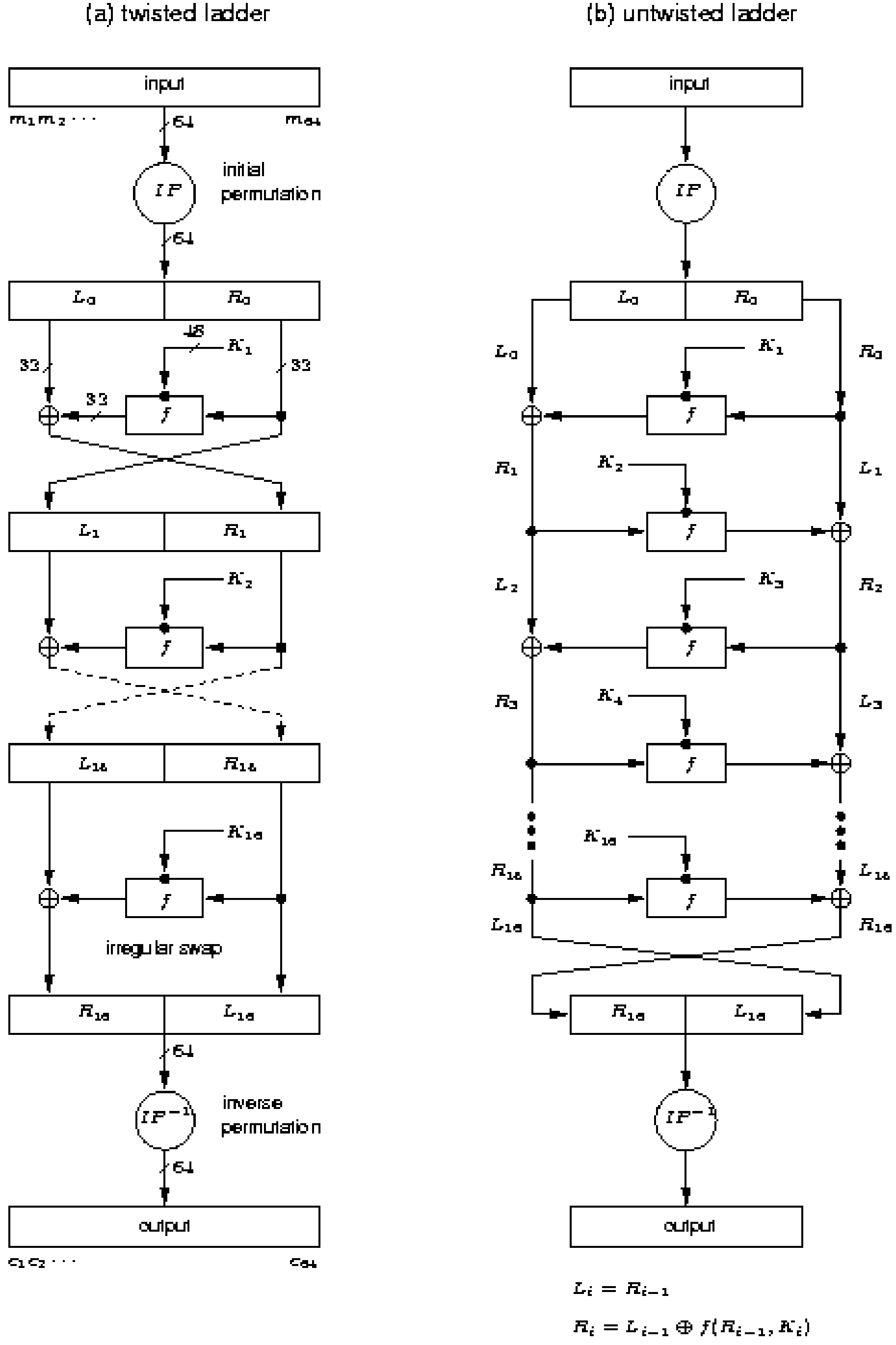


Figure 7.9: DES computation path.

The Initial Permutation

IP,

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

FP = IP^{-1}.

IP^{-1}							
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

Legend: First output bit is input bit 58. (FIPSPUB numbering)

- Why IP is used ? Nobody really knows.
- Makes software implementation harder and a bit slower…
- Makes no difference for the attacker and can be ignored.

Feistel Scheme

- First described by Horst Feistel in 1971.
- Invertible transformation.

- Luby-Rackoff theory:

Relative security proofs…

PRF => PRP

-in fact cannot be applied:

one round is NOT a PRF.

-avoids generic attacks.

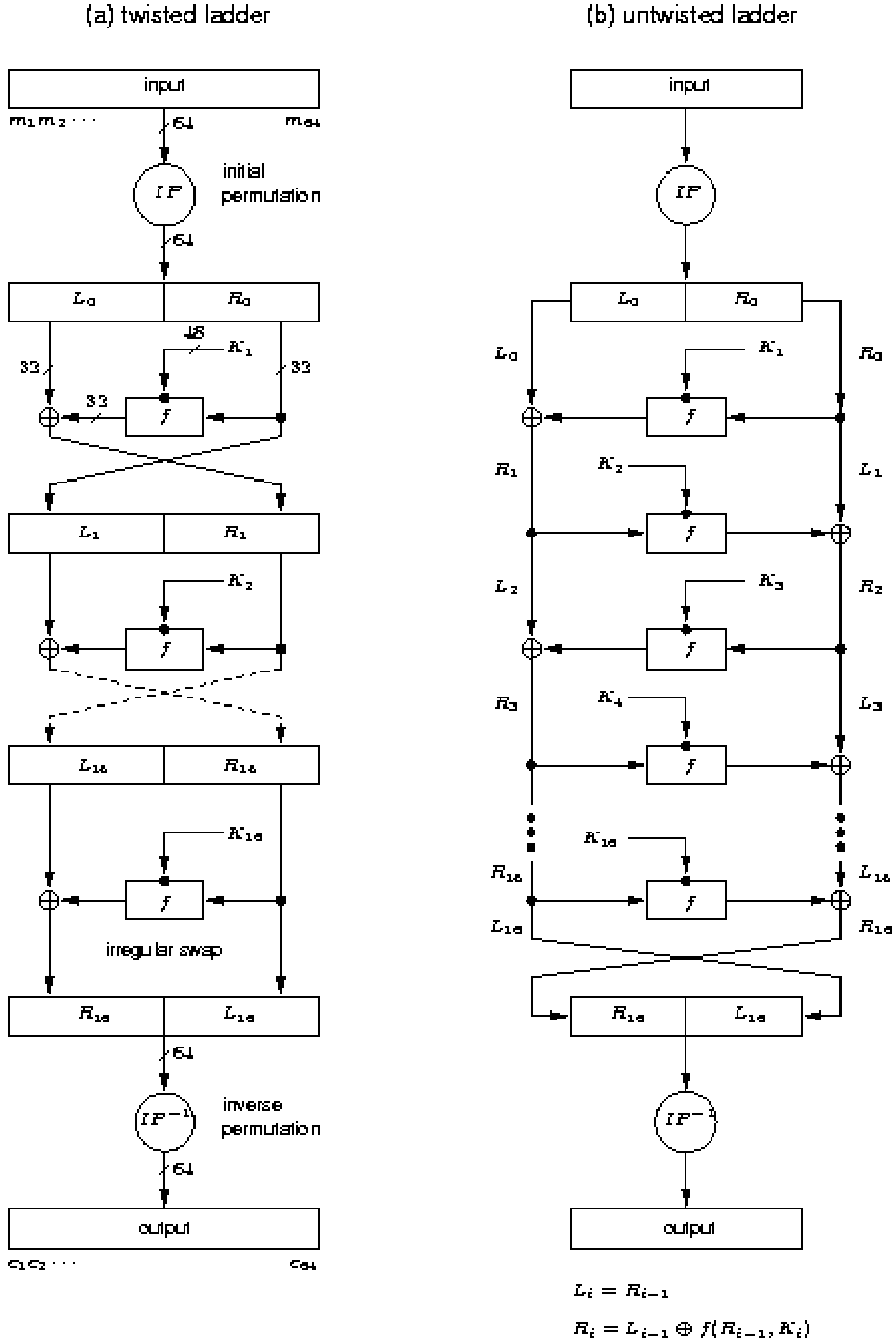


Figure 7.9: DES computation path.

Un-Twisted Feistel

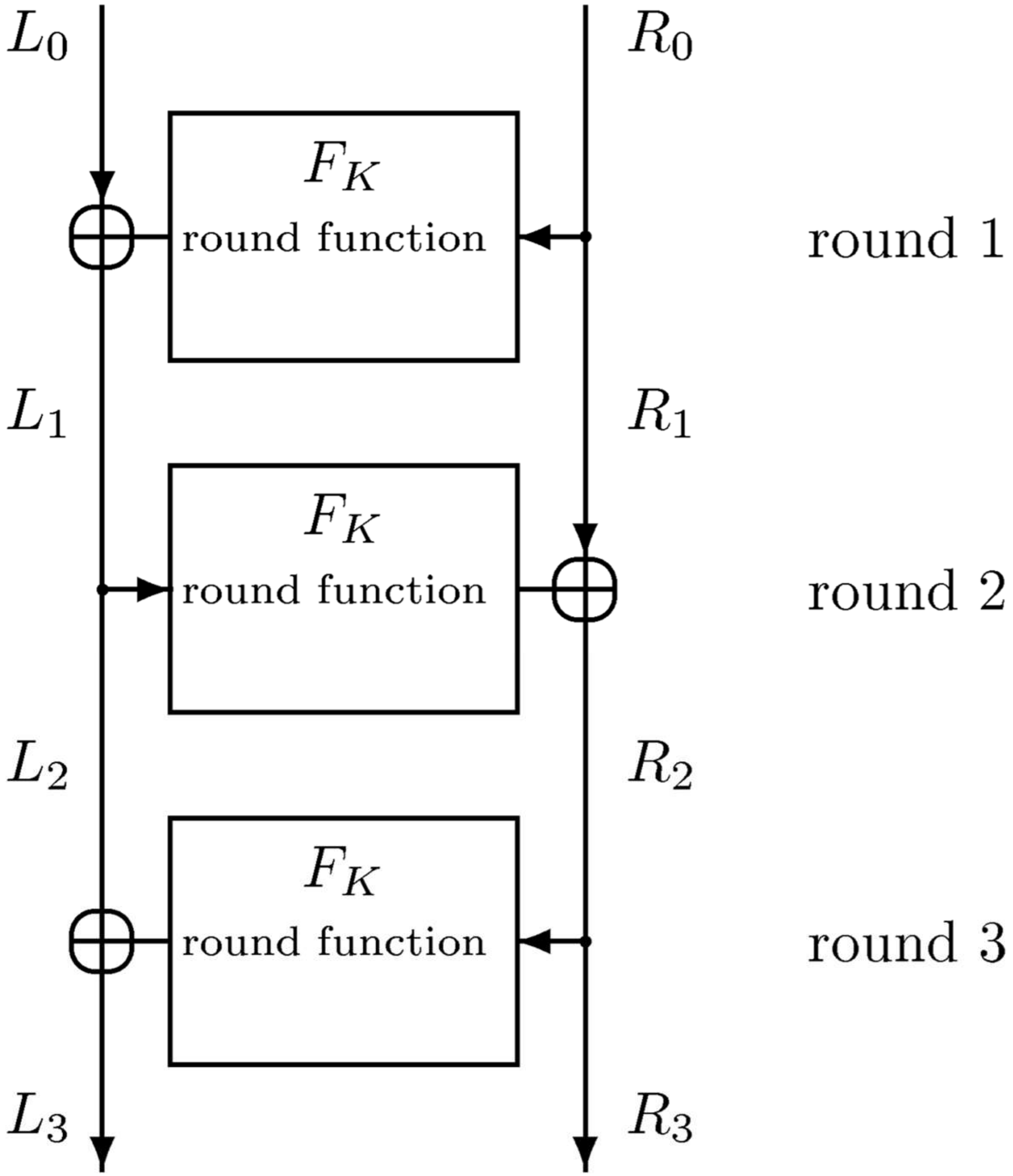
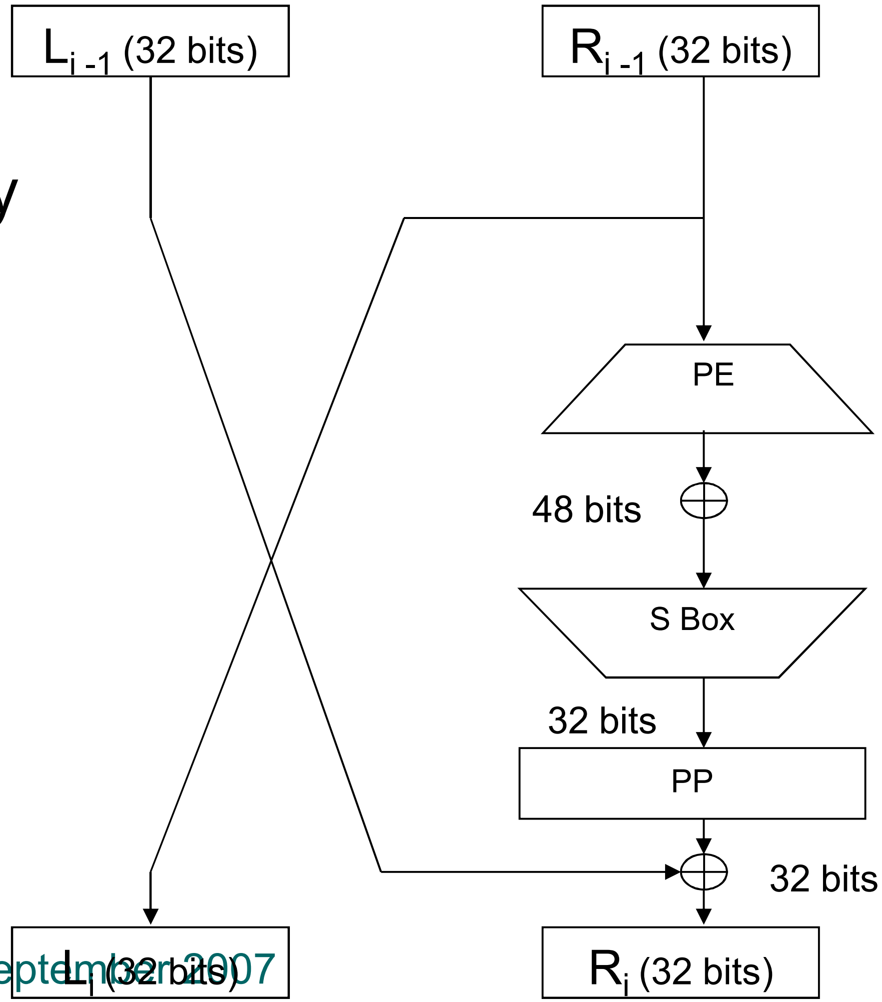


Figure 1: An "un-twisted" Feistel ladder

*1 Round (twisted)

F_K

- Expansion
- XOR with key
- S-boxes
- Permutation



DES Round Function

F_K

- Expansion
- XOR with key
- S-boxes
- Permutation

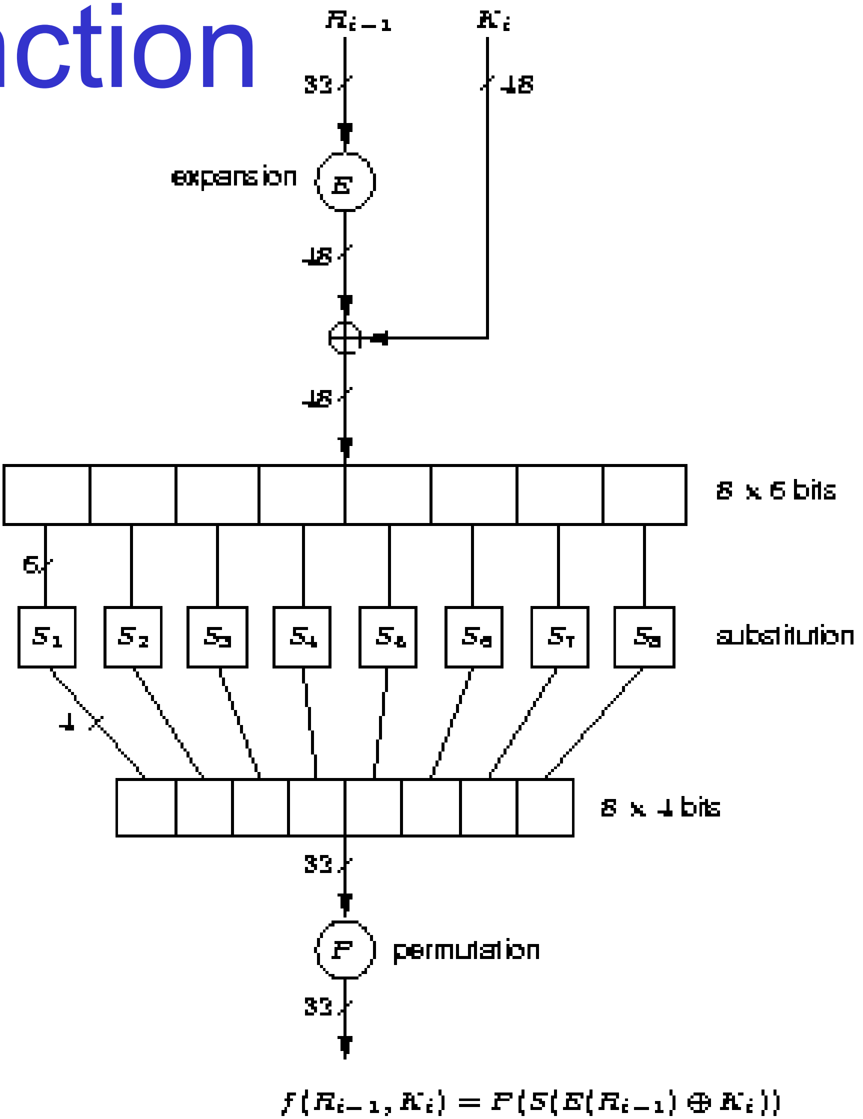
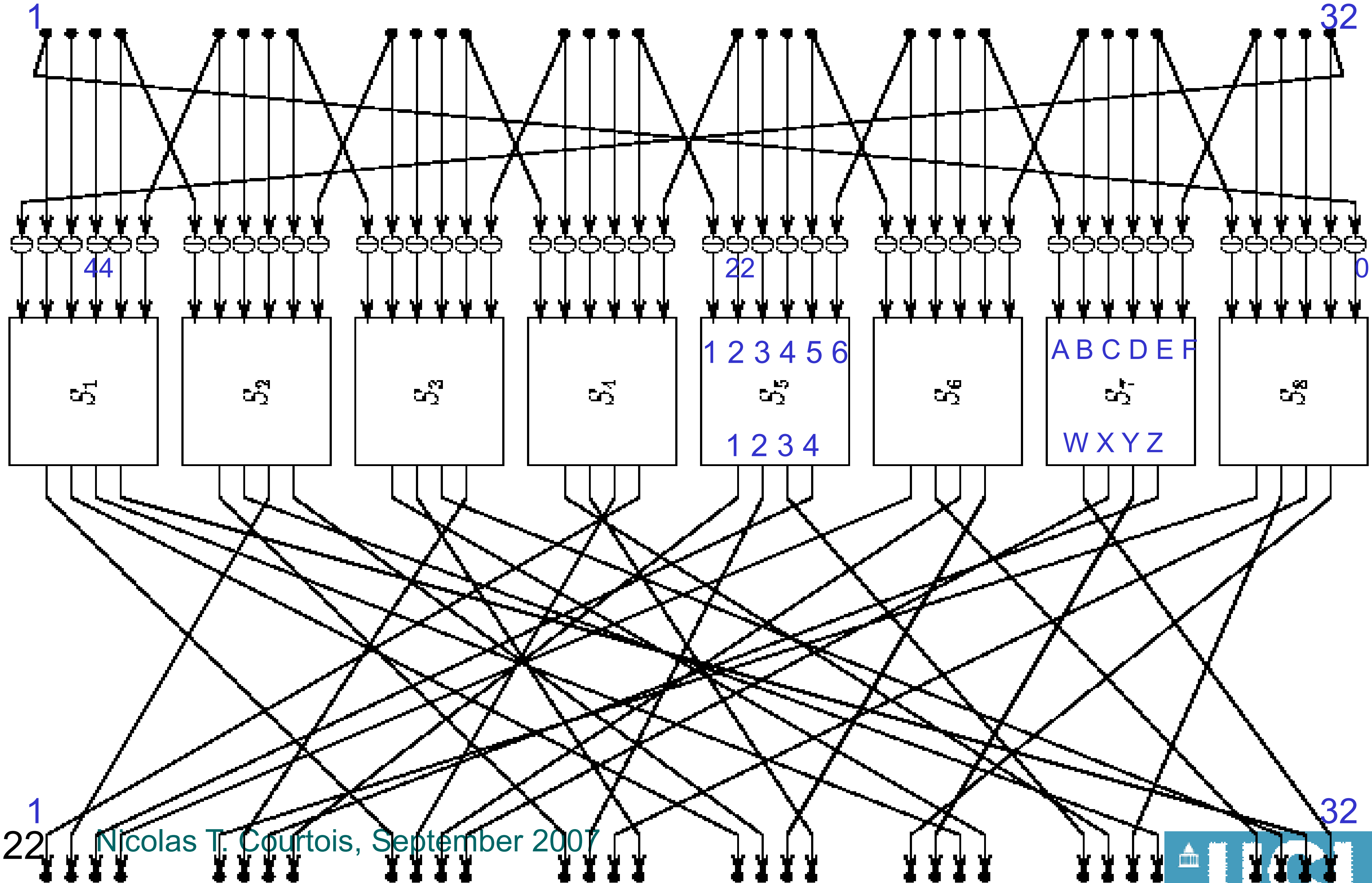


Figure 7.10: *DES inner function f.*

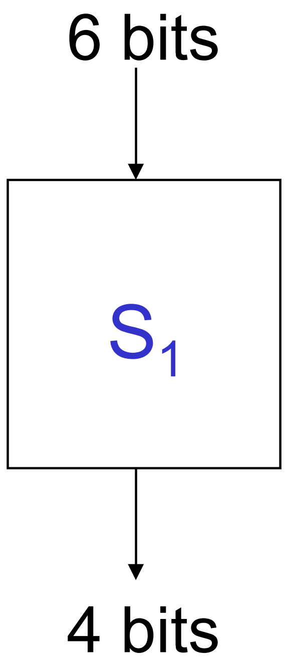
Another view:



DES design

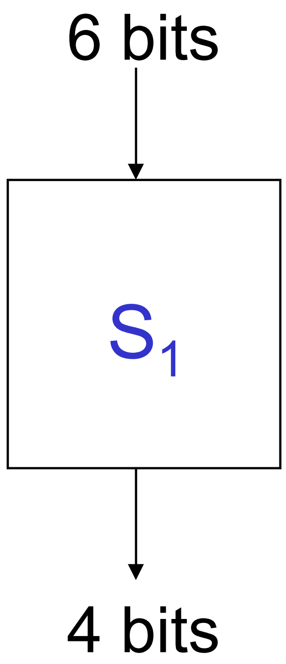
8 DES S-Boxes

row	column number															
	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]
S_1																
[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
S_2																
[0]	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
[1]	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
[2]	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
[3]	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3																
[0]	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
[1]	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
[2]	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
[3]	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4																
[0]	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
[1]	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
[2]	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
[3]	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
S_5																
[0]	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
[1]	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
[2]	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
[3]	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S_6																
[0]	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
[1]	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
[2]	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
[3]	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7																
[0]	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
[1]	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
[2]	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
[3]	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8																
[0]	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
[1]	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
[2]	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
[3]	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11



DES Boxes – S-box 1 / 8

S_1															
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



- Input: 110110 Output: 0111 (7 in decimal)
  - Row is: 10, (2 in decimal)
  - Column is: 1011, (11 in decimal)

DES Design

- IBM S-boxes were designed by IBM.
  Design criteria published,and re-published by Coppersmith etc.
  Presumably incomplete.

- Real S-boxes were done by the NSA, acknowledged publicly in 2000 by Coppersmith (I was there).

DES Boxes

8 S-boxes IBM, modified by the NSA.

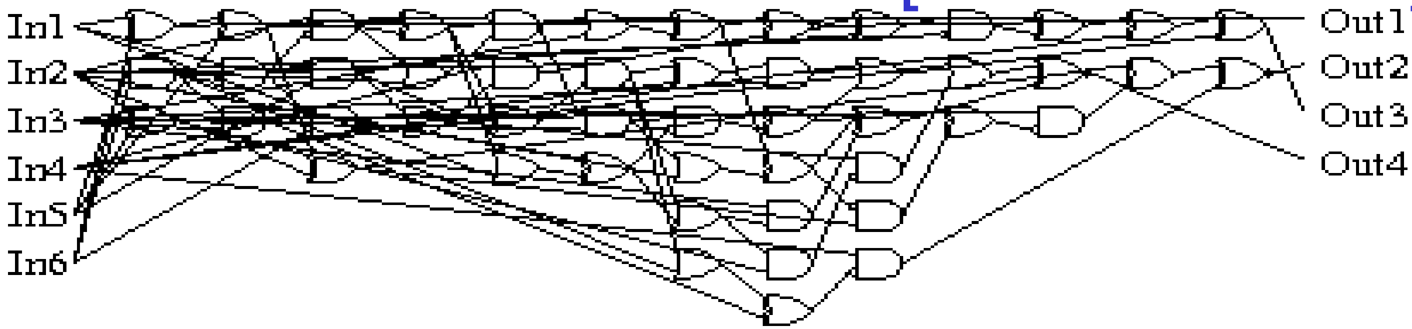
- The whole DES should be implemented on a single IC [with 1974 technology].

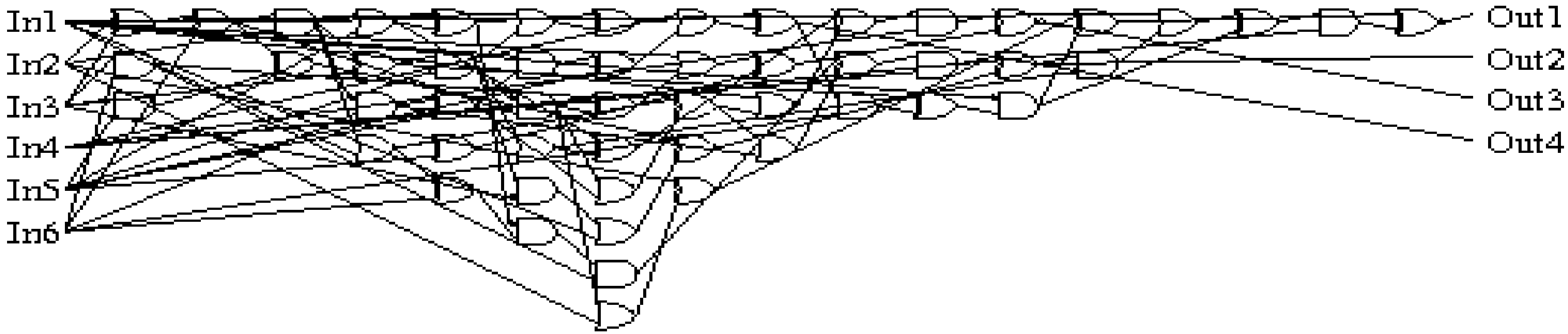
  => Each S-box should be implemented with 47 gates [NAND gates? 47??? NEVER SEEN one].

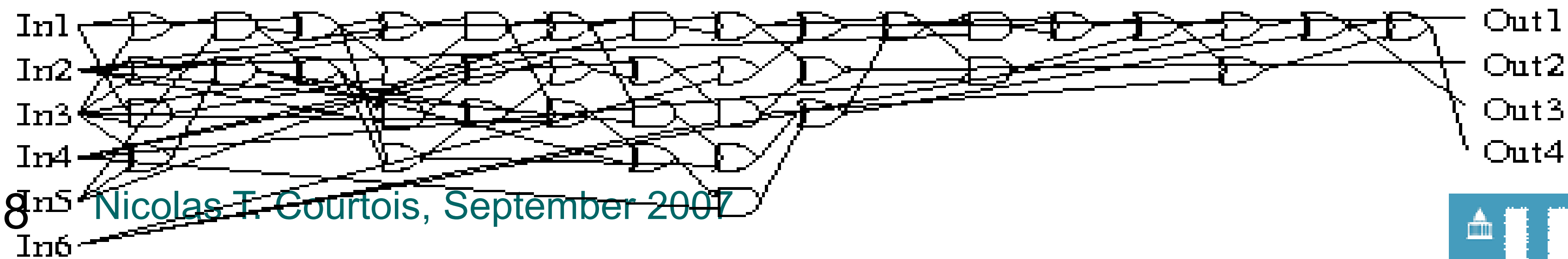
$\Rightarrow$ Fix two outer bits – permutation.

- No output should be too close to a linear function of inputs. [LC]. Coppersmith[C'2000]: A better criterion would be "no linear combination of outputs…"

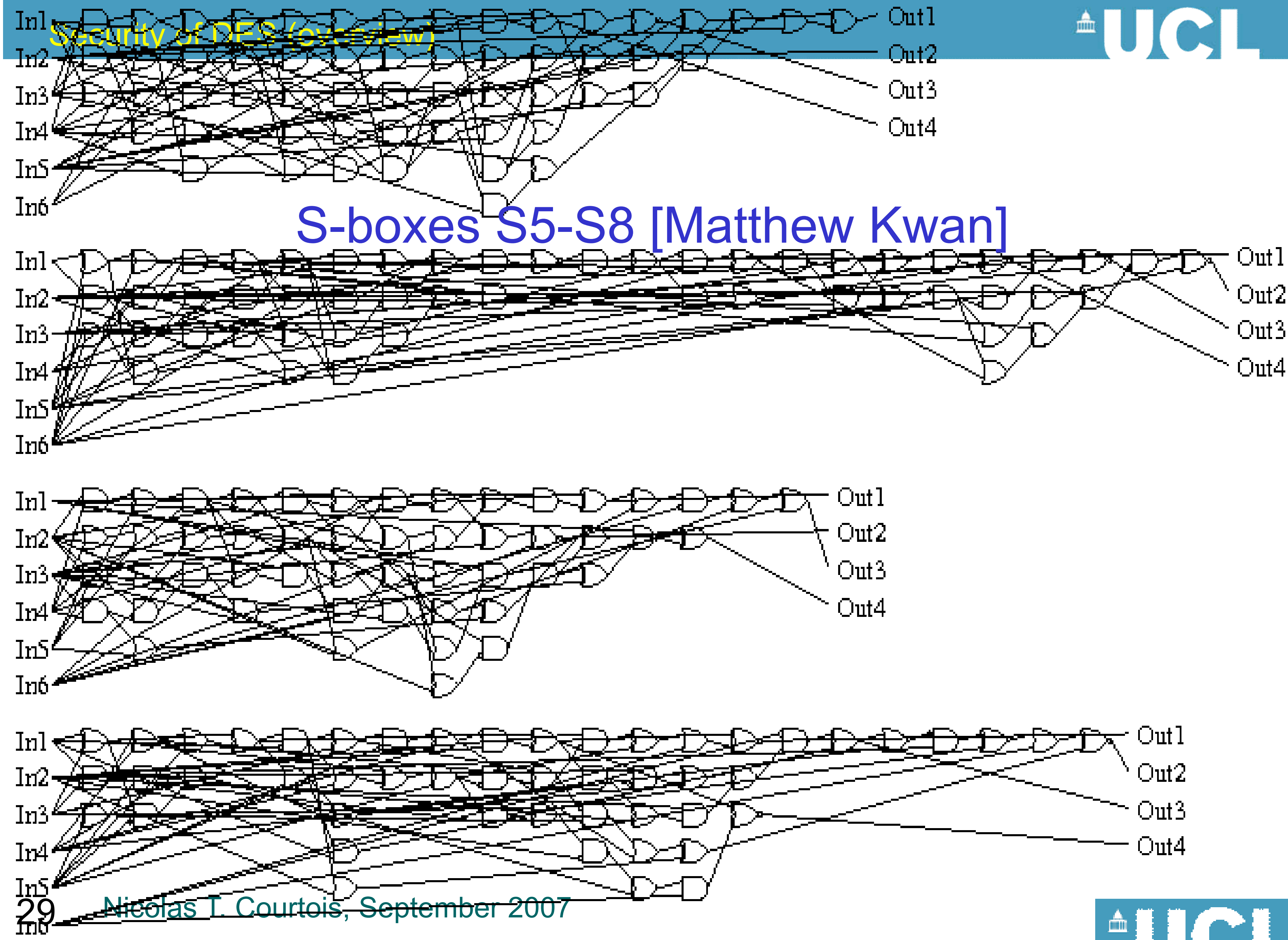
S-boxes S1-S4 [Matthew Kwan]







S-boxes S5-S8 [Matthew Kwan]



***DES Implementation [2013]

- 17% less gates still, by Roman Rusakov
- Bitslice
  - average of 44.125 gates per S-box
    (NB. they found several solutions with the same gate count)
  - vs. 53.375 for Kwan (his XNOR=>2gates).

  - cf. www.openwall.com/lists/john-users/2011/06/22/1
  - or the source code of John the Ripper

DES Boxes

- Change one bit in the input => at least two outputs change.
- Same for S(x) and S(x+001100).
- Some other…

Coppersmith 2000:

$$\mathrm{Prob}(\Delta_{\mathrm{out}} = 0 | \Delta_{\mathrm{in}}) \leq \frac{8}{32}.$$

preventing annihilation of differential perturbations!

$$A: \quad I[17] \oplus O[3,8,14,25] = K[22] \qquad 12/64$$

DES Design

- NSA trapdoor ? Never found.
  Maybe did not know how to embed one in such a construction.
- Is DES a group? Not at all.

DES
early attacks

Chronology on DES

- Complementation property.
- No real attacks, lots of speculations until 1991 (work has been classified?).
- Davies-Murphy attack [1982-1995] …….LC
- Shamir Paper [1985]………LC
- Differential Cryptanalysis [1991]
- Linear Cryptanalysis: Gilbert and Matsui [1992-93]

Weak key of DES

- Does not matter.
- Tells us things about structure of DES.
- 4 weak keys:
  - 0101 0101 0101 0101
  - FEFE FEFE FEFE FEFE
  - 1F1F 1F1F 1F1F 1F1F
  - E0E0 E0E0 E0E0 E0E0
- For each of these there are 2^{32} fixed points.

“Early LC” - Shamir 1985

On the Security of DES

Adi Shamir
Applied Mathematics
The Weizmann Institute
Rehovot, Israel

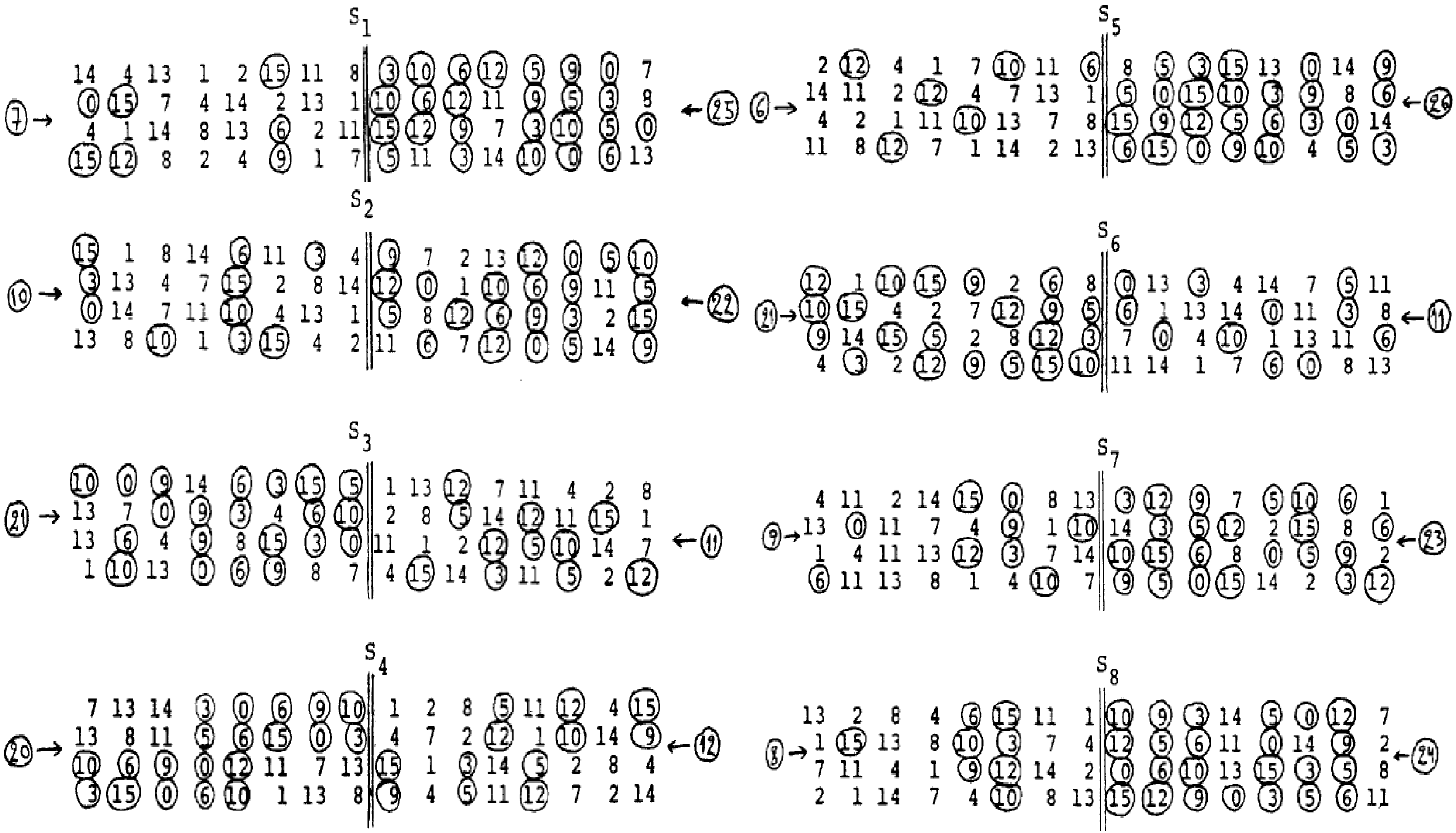
(abstract)

The purpose of this note is to describe some anomalies found in the structure of the S-boxes in the Data Encryption Standard. These anomalies are potentially dangerous, but so far they have not led to any successful cryptanalytic attack.

Mystery thing.

Related to LC published 8 years later.

** Shamir 1985



Shamir 1985

On the Security of DES

Adi Shamir
Applied Mathematics
The Weizmann Institute
Rehovot, Israel
(abstract)

$x\_2 \approx y\_1 \oplus y\_2 \oplus y\_3 \oplus y\_4$.

Common to all S-boxes !!!!

Mystery only partially explained by Coppersmith...

S5: the strongest linear bias in DES, used in LC.

Davies-style attacks

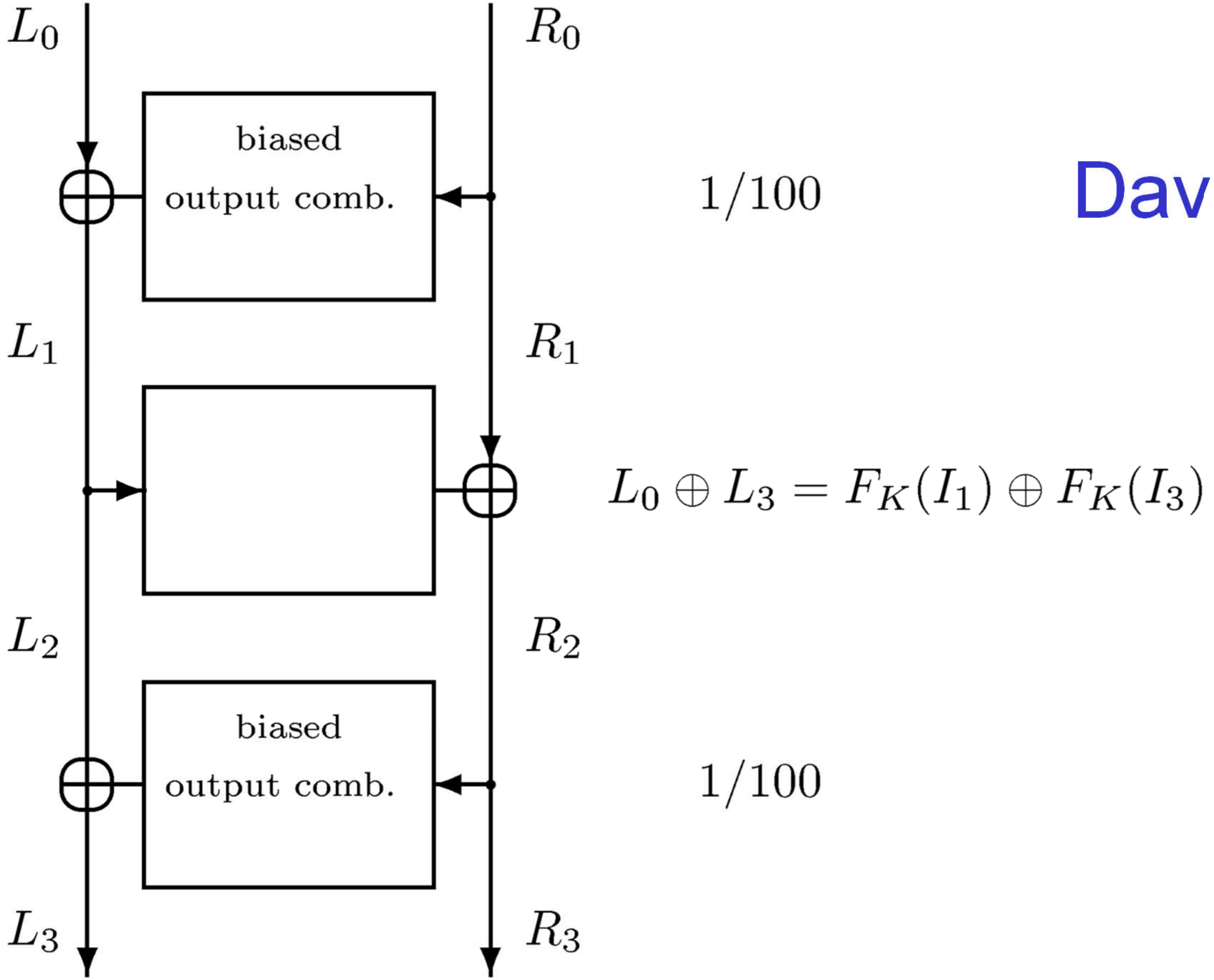
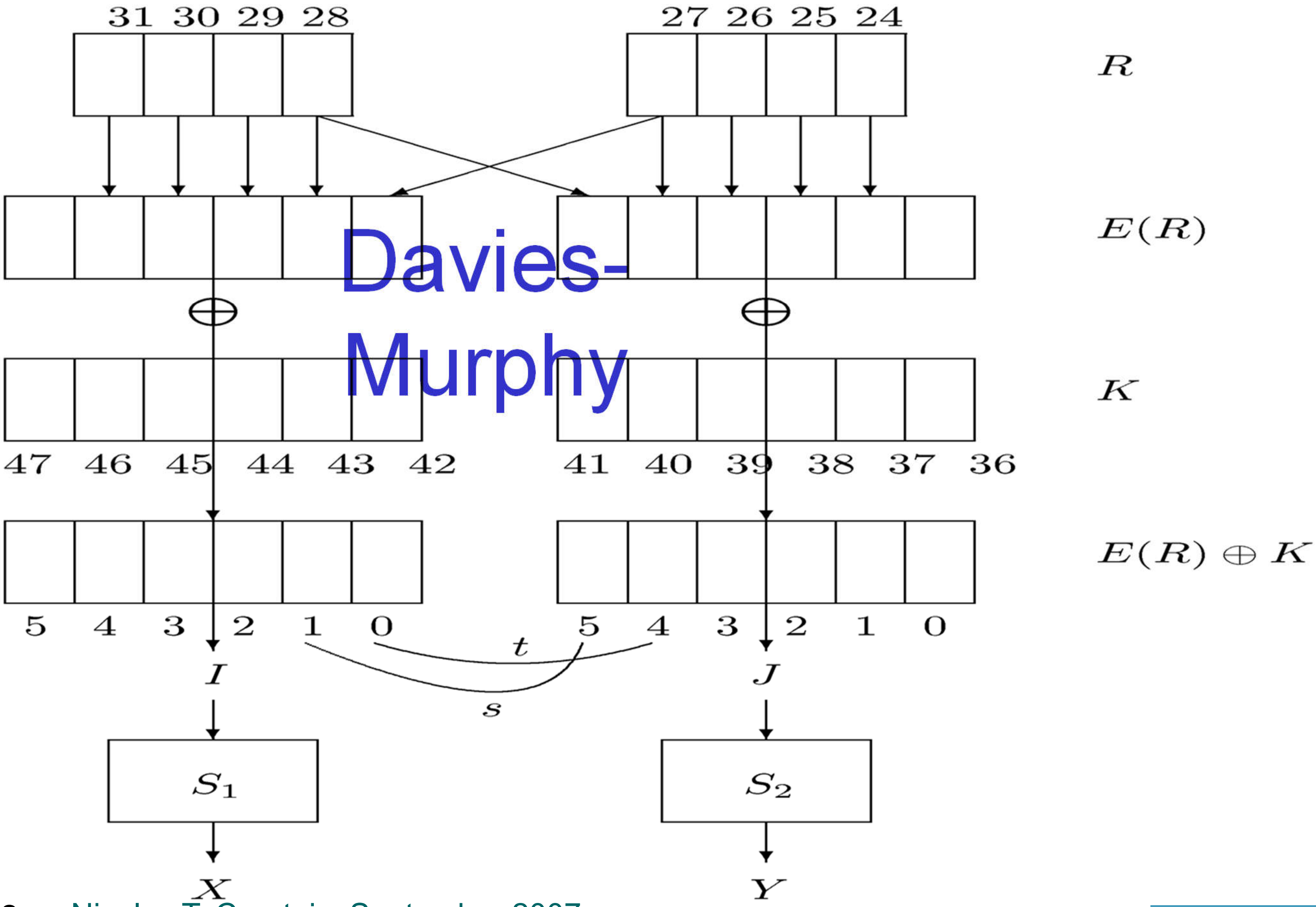


Figure 1: Davies-style attack


31 30 29 28
27 26 25 24
R
E(R)
Davies-
Murphy
K
47 46 45 44 43 42
41 40 39 38 37 36
E(R) ⊕ K
5 4 3 2 1 0
5 4 3 2 1 0
t
s
I
J
S1
S2
X
Y

Davies-Murphy - example

S_1	S_2															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
1	5	5	4	3	4	4	3	4	3	4	6	4	4	5	3	3
2	2	2	4	6	4	4	6	4	6	4	0	4	4	2	6	6
3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	3	3	4	5	4	4	5	4	5	4	2	4	4	3	5	5
6	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
7	5	5	4	3	4	4	3	4	3	4	6	4	4	5	3	3
8	5	5	4	3	4	4	3	4	3	4	6	4	4	5	3	3
9	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
10	6	6	4	2	4	4	2	4	2	4	8	4	4	6	2	2
11	3	3	4	5	4	4	5	4	5	4	2	4	4	3	5	5
12	5	5	4	3	4	4	3	4	3	4	6	4	4	5	3	3
13	3	3	4	5	4	4	5	4	5	4	2	4	4	3	5	5
14	3	3	4	5	4	4	5	4	5	4	2	4	4	3	5	5
15	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

DES
and LC

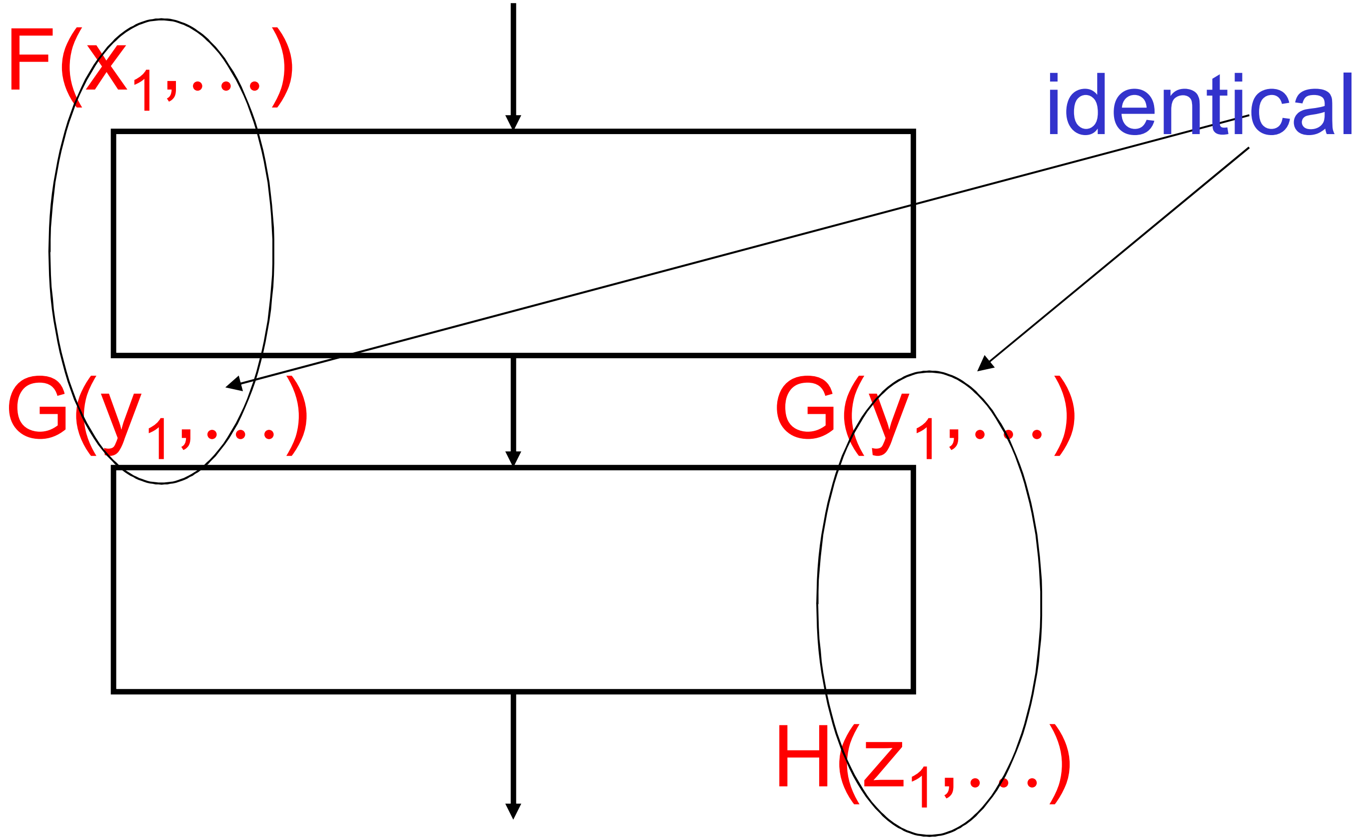
Linear Cryptanalysis = LC

Not known by Coppersmith/NSA ?

- [Gilbert and Tardy-Corfdir, FEAL, Crypto'92]
- [Matsui and DES, EuroCrypt'93]
- Biham at Eurocrypt'94: shows that the earlier Davies and Murphy DES attack method [1982-1995] is "essentially" a linear attack (!).
- Shamir [Crypto'85]: already exhibits a strong linear characteristic for each DES S-box.

Linear Cryptanalysis

Combine I/O Equations.



Linear Cryptanalysis

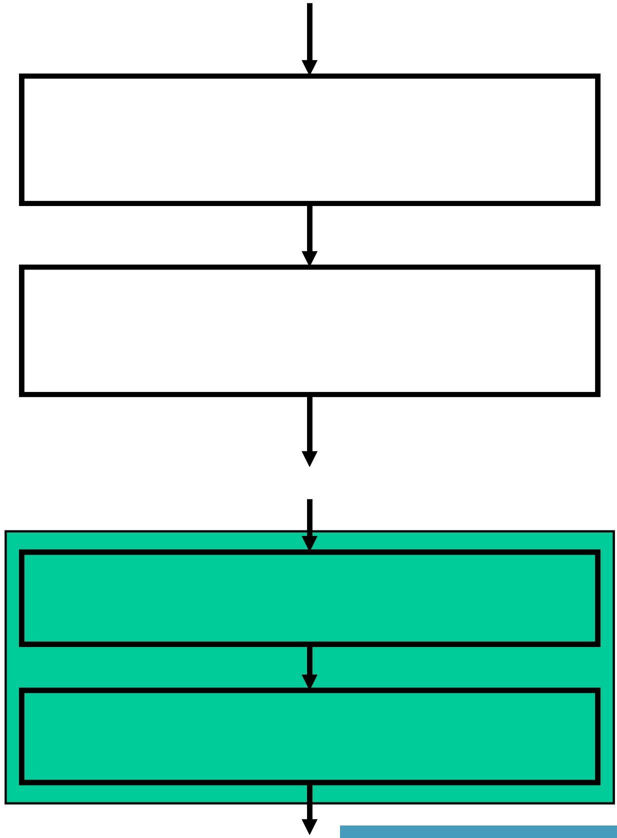
Add I/O Equations => get another I/O Equation.

$F(x_1,\ldots)\oplus G(y_1,\ldots) = 0$ with P=…

$\oplus$

$G(y_1,\ldots)\oplus H(z_1,\ldots) = 0$ with P=…

$F(x_1,\ldots)\oplus H(z_1,\ldots) = 0$ with P=…



Linear Cryptanalysis

Piling-up Lemma [Matsui]

$$p \;=\; p_1\, p_2 + (1-p_1)(1-p_2) \;=\; \tfrac{1}{2} + 2(p_1 - \tfrac{1}{2})\,(p_2 - \tfrac{1}{2})$$

Imbalances : $I = 2\,|\,p_1 - \tfrac{1}{2}\,|$

They do multiply !!!

Search for LC – Matsui 1993

Table of size

$2^6 * 2^4$

find the strongest bias

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	4	−2	2	−2	2	−4	0	4	0	2	−2	2	−2	0	−4
3	0	−2	6	−2	−2	4	−4	0	0	−2	6	−2	−2	4	−4
4	2	−2	0	0	2	−2	0	0	2	2	4	−4	−2	−2	0
5	2	2	−4	0	10	−6	−4	0	2	−10	0	4	−2	2	4
6	−2	−4	−6	−2	−4	2	0	0	−2	0	−2	−6	−8	2	0
7	2	0	2	−2	8	6	0	−4	6	0	−6	−2	0	−6	−4
8	0	2	6	0	0	−2	−6	−2	2	4	−12	2	6	−4	4
9	−4	6	−2	0	−4	−6	−6	6	−2	0	−4	2	−6	−8	−4
10	4	0	0	−2	−6	2	2	2	2	−2	2	4	−4	−4	0
11	4	4	4	6	2	−2	−2	−2	−2	−2	2	0	−8	−4	0
12	2	0	−2	0	2	4	10	−2	4	−2	−8	−2	4	−6	−4
13	6	0	2	0	−2	4	−10	−2	0	−2	4	−2	8	−6	0
14	−2	−2	0	−2	4	0	2	−2	0	4	2	−4	6	−2	−4
15	−2	−2	8	6	4	0	2	2	4	8	−2	8	−6	2	0
16	2	−2	0	0	−2	−6	−8	0	−2	−2	−4	0	2	10	−20
17	2	−2	0	4	2	−2	−4	4	2	2	0	−8	−6	2	4
18	−2	0	−2	2	−4	−2	−8	4	6	4	6	−2	4	−6	0
19	−6	0	2	−2	4	2	0	4	−6	4	2	−6	4	−2	0
20	4	−4	0	0	0	0	0	−4	−4	4	4	0	4	−4	0
21	4	0	−4	−4	4	−8	−8	0	0	−4	4	8	4	0	4
22	0	6	6	2	−2	4	0	4	0	6	2	2	2	0	0
23	4	−6	−2	6	−2	−4	4	4	−4	−6	2	−2	2	0	4
24	6	0	2	4	−10	−4	2	2	0	−2	0	2	4	−2	−4
25	2	4	−6	0	−2	4	−2	6	8	6	4	10	0	2	−4
26	2	2	−8	−2	4	0	2	−2	0	4	2	0	−2	−2	0
27	2	6	−4	−6	0	0	2	6	8	0	−2	−4	−6	−2	0
28	0	−2	2	4	0	−6	2	−2	6	−4	0	2	−2	0	0
29	4	−2	6	−8	0	−2	2	10	−2	−8	−8	2	2	0	4
30	−4	−8	0	−2	−2	−2	2	−2	2	−2	6	4	4	4	0
31	−4	8	−8	2	−6	−6	−2	−2	2	−2	−2	−8	0	0	−4
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 1. A distribution table of S5 (part).

Matsui's Favourite

$$A: \quad I[17] \oplus O[3,8,14,25] = K[22] \qquad 12/64$$

$$C: \quad I[3] \oplus O[17] = K[44] \qquad 30/64$$

$$D: \quad I[17] \oplus O[8,14,25] = K[22] \qquad 42/64$$

LC Example: (Untwisted)

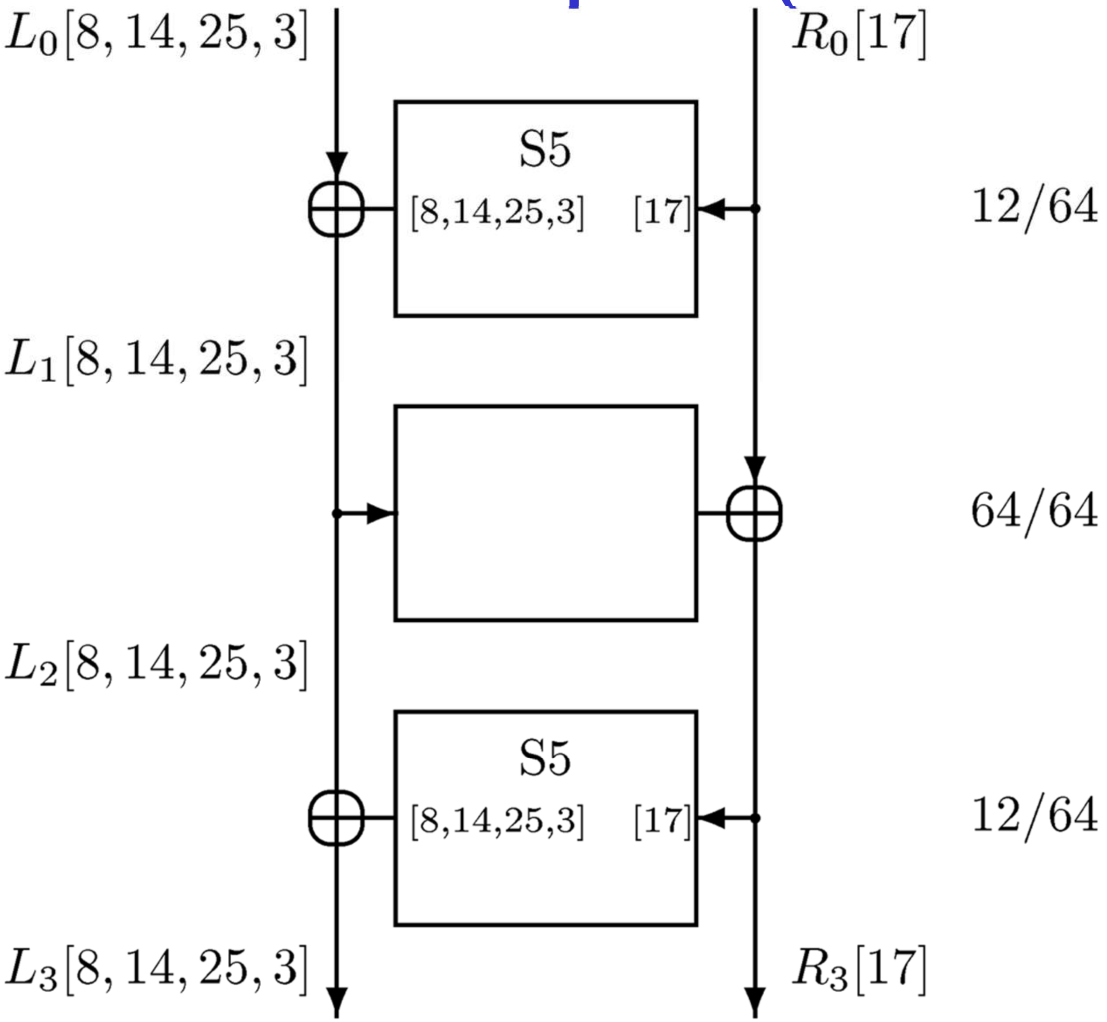


Figure 1: Matsui's Best Linear Approximation on 3 Rounds of DES

Two "Magical" Rules

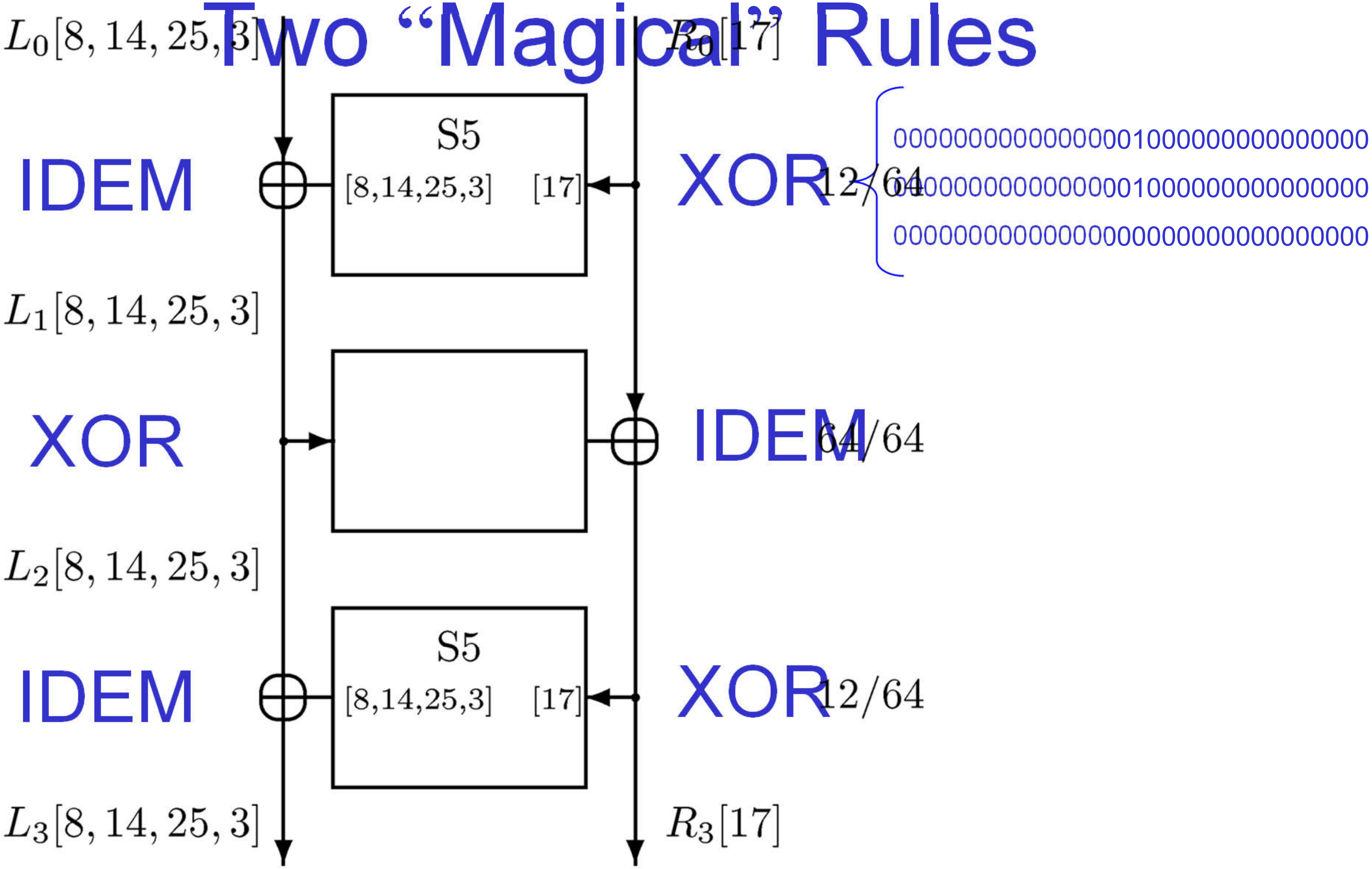


Figure 1: Matsui's Best Linear Approximation on 3 Rounds of DES

Complexity of LC

Decision by majority. Bias = ε.

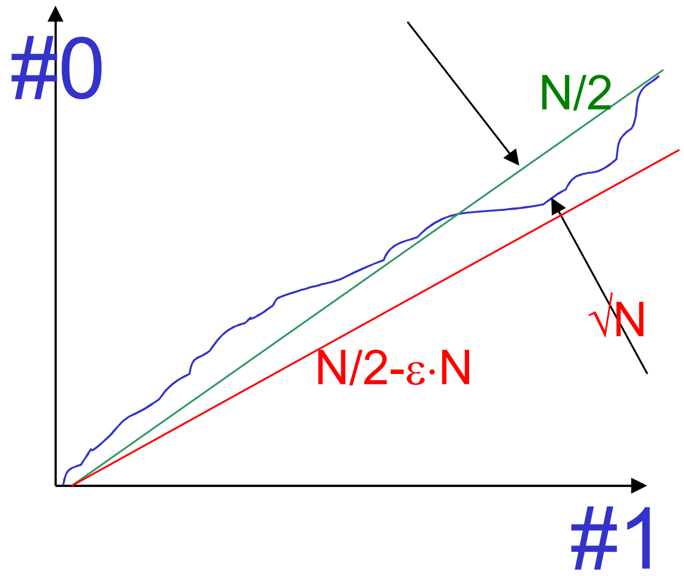
The signal must be stronger than “noise”.

The law of the random walk.

(average N/2, std. dev=$\sqrt{N}$)

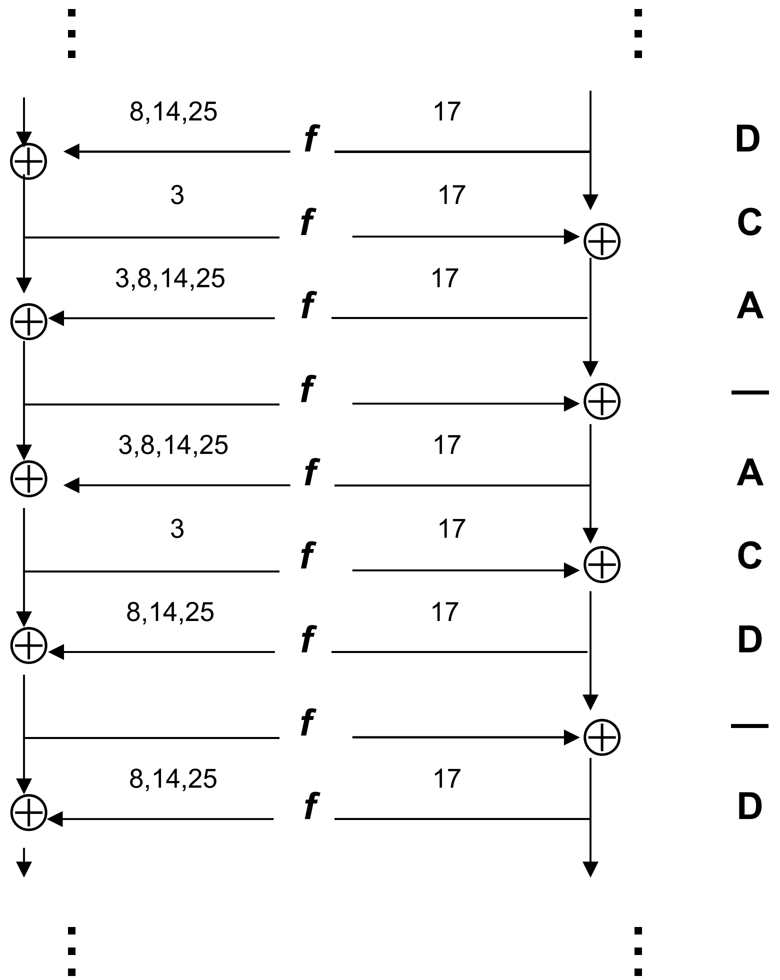
$$\Rightarrow \varepsilon \cdot N \geq \sqrt{N}$$

$\# KP \geq (1 / \text{bias})^2$



Best DES Approximation

(Matsui, 1993)



- cyclic; 14 rounds
- 2-R method

Linear Cryptanalysis

- A statistical known plaintext attack
- Correlation among pt, ct, key bits are exploited:
  - Find a binary equation of pt, ct, key bits (“linear approximation”) which shows a non-trivial correlation among them (“bias”).
  - Collect a large pt-ct sample.
  - Try all key values with the collected pt-ct in the eq. (hence, relatively few key bits must be involved.)
  - Take the key that maximizes the bias as the right key.
- The remaining key bits can be found by brute force or by another LC attack.

Improvements

Apply LC to 16-1 rounds.

Guess some key bits in the last round.

See if the results confirm the guess.

This is called 1R method.

Possible because the “Linear Characteristics” used uses very few I/O bits, that involve very few bits in the last round.

1R Method

A linear approximation of r-1 rounds:

$P[i_1 \ldots i_a] \oplus X_{r-1}[j_1 \ldots j_b] = K[m_1 \ldots m_c]$

with $p \neq ½$. ($p = 1$ usually not possible)

- $|p - ½|$: the "bias" of the approximation
- (notation: X_i: ciphertext after i rounds;
  $S[\ldots]$: xor of the specified bits of the string S.)

Expressed in terms of the ciphertext:

$P[i_1 \ldots i_a] \oplus F(C, K_r)[j_1 \ldots j_b] = K[m_1 \ldots m_c]$

where F is related to the last round's decryption.

1R Method

- Approximation:

$$P[i_1 \ldots i_a] \oplus F(C, K_r)[j_1 \ldots j_b] = K[m_1 \ldots m_c] \qquad (1)$$

- Collect a large number (N) of pt-ct blocks
- For all possible K_r values, compute the left side of (1). $T^{(i)}$ denoting the # of zeros for the i^{th} candidate, take the K_r value that maximizes the "sample bias" $| T^{(i)} - N/2 |$ as the right key.
- Another bit of key information (that is, $K[m_1 \ldots m_c]$) can be obtained comparing the signs of $(p - ½)$ and $(T^{(i)} - N/2)$.

1R Improved

- 1 bit in the equation => 6 key bits/eqs
- 1 S-box => then 12 key bits / equation.
- 24 due to the symmetry: scrap 1 at the end and at the beginning…
- Remaining: exhaustive search !

- False positives ?
  - E.g. $5 * 2^{(56-24)}$ = easy !

LC of DES

- 8 rounds: 2^{21} known plaintexts
  12 rounds: 2^{33} known plaintexts
  16 rounds: 2^{43} known plaintexts

- First experimental cryptanalysis of the 16-round DES (Matsui, 1994).

- Ordering of the S-boxes were far from optimal against LC.

GLC

Generalised Linear Cryptanalysis = GLC =

[Harpes, Kramer and Massey, Eurocrypt'95]
[related work: Harpes, Jakobsen…]

Concept of non-linear I/O sums.

F(inputs) = F'(outputs)
with some probability…

GLC

[Eurocrypt'95]

Proof of Concept for SPN-type ciphers:

Exhibit a cipher very secure w.r.t. LC but very weak w.r.t. to GLC.

Can be Seen as GLC

[Jakobsen and Knudsen polynomial approximation attacks, Crypto'98, JoC'01]

Another proof of concept for SPNs.

Contrived ciphers secure w.r.t. to all known attacks but in fact very weak…

GLC and Feistel Ciphers?

[Knudsen and Robshaw, EuroCrypt'96]

For some reason decided that…GLC was impossible for Feistel Ciphers. Write that:

"one-round approximations that are non-linear […] cannot be joined together"…

- Content themselves with using non-linear approximations for the first and last round… [cf. also Kaneko and Shimoyama, Crypto'98].

BLC – Courtois 2004

1. Proof of concept: ciphers resistant to DC, LC etc. yet extremely weak w.r.t. the new attack.
2. New non-trivial attacks on DES.
   Some do slightly beat Matsui's best equation.

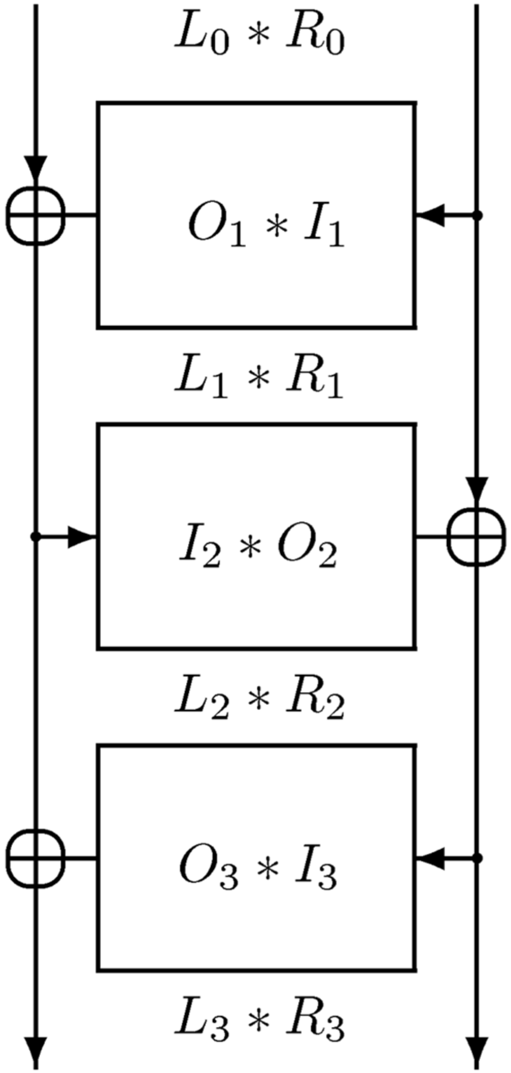
GLC and Feistel Ciphers.

Main Claim:

The structure of Feistel ciphers makes them predisposed to a special subclass of GLC.

BLC = Bi-Linear Cryptanalysis.

Bi-linear Cryptanalysis over GF(2^n)



$$= \cancel{L_1 * R_1} - L_0 * R_0$$

$$= \cancel{\cancel{L_2 * R_2}} - \cancel{L_1 * R_1}$$

$$= L_3 * R_3 - \cancel{\cancel{L_2 * R_2}}$$

Bi-linear Cryptanalysis – Example:

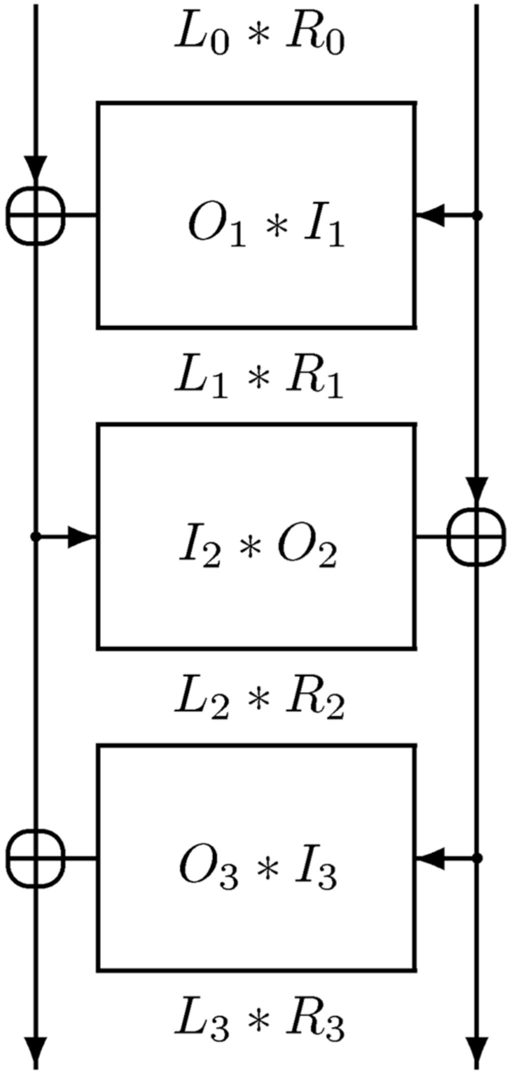
Round function:

$$f_i(X) = K_i \cdot Inv(X) \qquad \text{in } GF(2^n),$$

Then for every round:

$$I_i \cdot O_i = K_i \quad \text{with probability} \quad \left(1 - \frac{1}{2^n}\right)$$

Sum-Up:



$$= \cancel{L_1 * R_1} - L_0 * R_0$$

$$= \cancel{L_2 * R_2} - \cancel{L_1 * R_1}$$

$$= L_3 * R_3 - \cancel{L_2 * R_2}$$

Example - contd.

Whole cipher:

$$L_{N_r} \cdot R_{N_r} \oplus L_0 \cdot R_0 = \sum_{i=1}^{N_r} K_i$$

$$\text{with probability} \quad \left(1 - \frac{1}{2^n}\right)^{N_r}$$

Broken even for 2^n rounds !

What we get:

- Insecure Feistel cipher based on Inverse in $GF(2)^n$.
- Mixes 3 different group operations.
- High non-linearity.
- Satisfies all design criteria.
- Provably secure against DC and LC.
- Yet broken even for 2^n rounds !

DES S-boxes and BLC

Table 1: Selected bi-linear characteristics for DES S-boxes

		equation			remarks and comments
		input	output	input*output	
$S5$	$12/64$	17	$8, 14, 25, 3$		Matsui's equation A
$S5$	$6/64$	17	$8, 14, 25, 3$	$[17] * [8, 14, 25, 3]$	gets better
$S5$	$58/64$			$[17] * [8, 14, 25, 3]$	
$S5$	$\mathbf{61/64}$	$16, 20$	$8, 14, 25, 3$	$[16, 17, 20] * [3]$	the best in DES
$S5$	$47/64$		$8, 14, 25$	$17 * 3$	
$S5$	$17/64$		$8, 14, 25, 3$	$17 * 3$	
$S1$	$30/64$	3	17		Matsui's equation C
$S1$	$15/64$	3	17	$3 * 17$	gets better
$S1$	$47/64$		17	$3 * 17$	

1st Example for DES

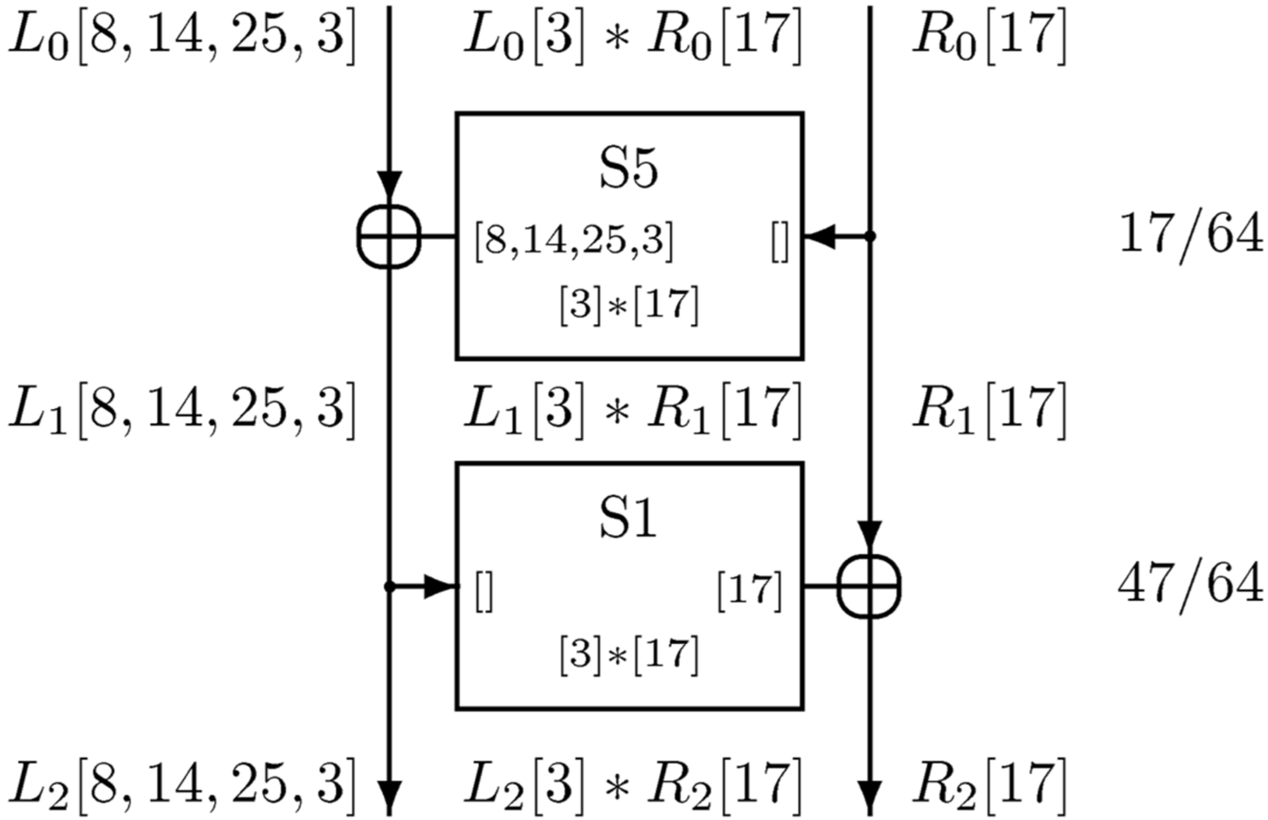


Figure 1: Our first example - an invariant bi-linear attack on DES (∗)

3 Rounds:

$$(*)\quad \begin{array}{l} L_0[3,8,14,25] \oplus L_0[3]R_0[17] \oplus R_0[17] \oplus \\ L_2[3,8,14,25] \oplus L_2[3]R_2[17] \oplus R_2[17] = K[sth] \end{array}$$

$$\frac{1}{2} - 1.76 \cdot 2^{-4}$$

Happens to work also for EVERY OTHER KEY !

Bias varies slightly…

r rounds:

$$(*) \quad \begin{array}{l} L_0[3,8,14,25] \oplus L_0[3]R_0[17] \oplus R_0[17] \oplus \\ L_r[3,8,14,25] \oplus L_r[3]R_2[17] \oplus R_r[17] = K[sth] \end{array}$$

Biased for:

- any key
- any number of rounds

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, …

What we get:

$$(*) \quad \begin{array}{l} L_0[3,8,14,25] \oplus L_0[3]R_0[17] \oplus R_0[17] \oplus \\ L_r[3,8,14,25] \oplus L_r[3]R_2[17] \oplus R_r[17] = K[sth] \end{array}$$

An invariant-based bi-linear attack for DES, for any key, and any number of rounds.

The strongest known invariant attack on DES.

How good it is ?

$$(*)\quad \begin{array}{l} L_0[3,8,14,25] \oplus L_0[3]R_0[17] \oplus R_0[17] \oplus \\ L_r[3,8,14,25] \oplus L_r[3]R_2[17] \oplus R_r[17] = K[sth] \end{array}$$

- Always worse than some other Matsui's equation.
- But never much worse.
- In fact closely related to some prominent equations of Matsui – their difference is a biased Boolean function.

How good is BLC ?

Conjecture: BLC cannot be much better than some existing linear attack. Heuristic, detailed argumentation in the extended version of the paper.

----- BUT ------

BLC can be strictly better than LC.

BLC better than LC for DES

$$L_0[3,8,14,25] \oplus L_0[3]R_0[16,17,20] \oplus R_0[17] \oplus$$
$$L_{11}[3,8,14,25] \oplus L_{11}[3]R_{11}[16,17,20] \oplus R_{11}[17] =$$
$$K[sth] + K[sth']L_0[3] + K[sth'']L_{11}[3]$$

Better than the best existing linear attack of Matsui
for 3, 7, 11, 15, ... rounds.

Ex: LC 11 rounds: $\frac{1}{2} \pm 1.91 \cdot 2^{-16}$

BLC 11 rounds: $\frac{1}{2} \pm 1.2 \cdot 2^{-15}$

DC
of DES

DC

Differential Cryptanalysis = DC.
[1991]

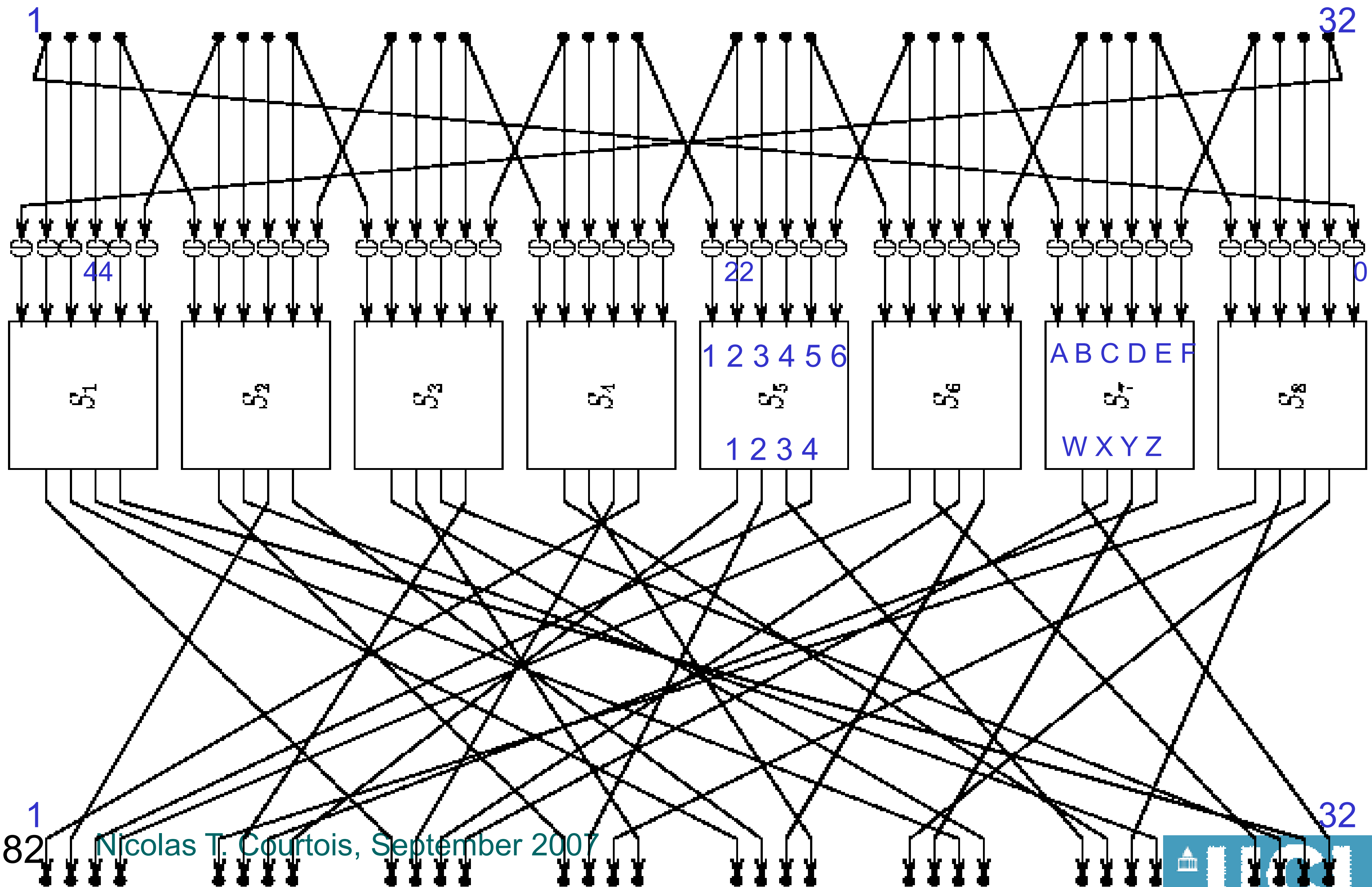
- Very powerful
- Known by Coppersmith, optimised against, random S-boxes are weak !
- Shamir's disturbing remark to Coppersmith…
- Russian Des: GOST. S-boxes not published.

DES vs. DC

Critical property: a differential with 4 bits in the middle ‘active’ cannot happen with P<=1/256 or so.

- BTW. If we use outer bits => other boxes will be affected.

Why?



DES vs. DC

Consequences:

- DES with random S-boxes would be very weak w.r.t. DC.
- Best differentials for DES use 3 S-boxes.

DC example

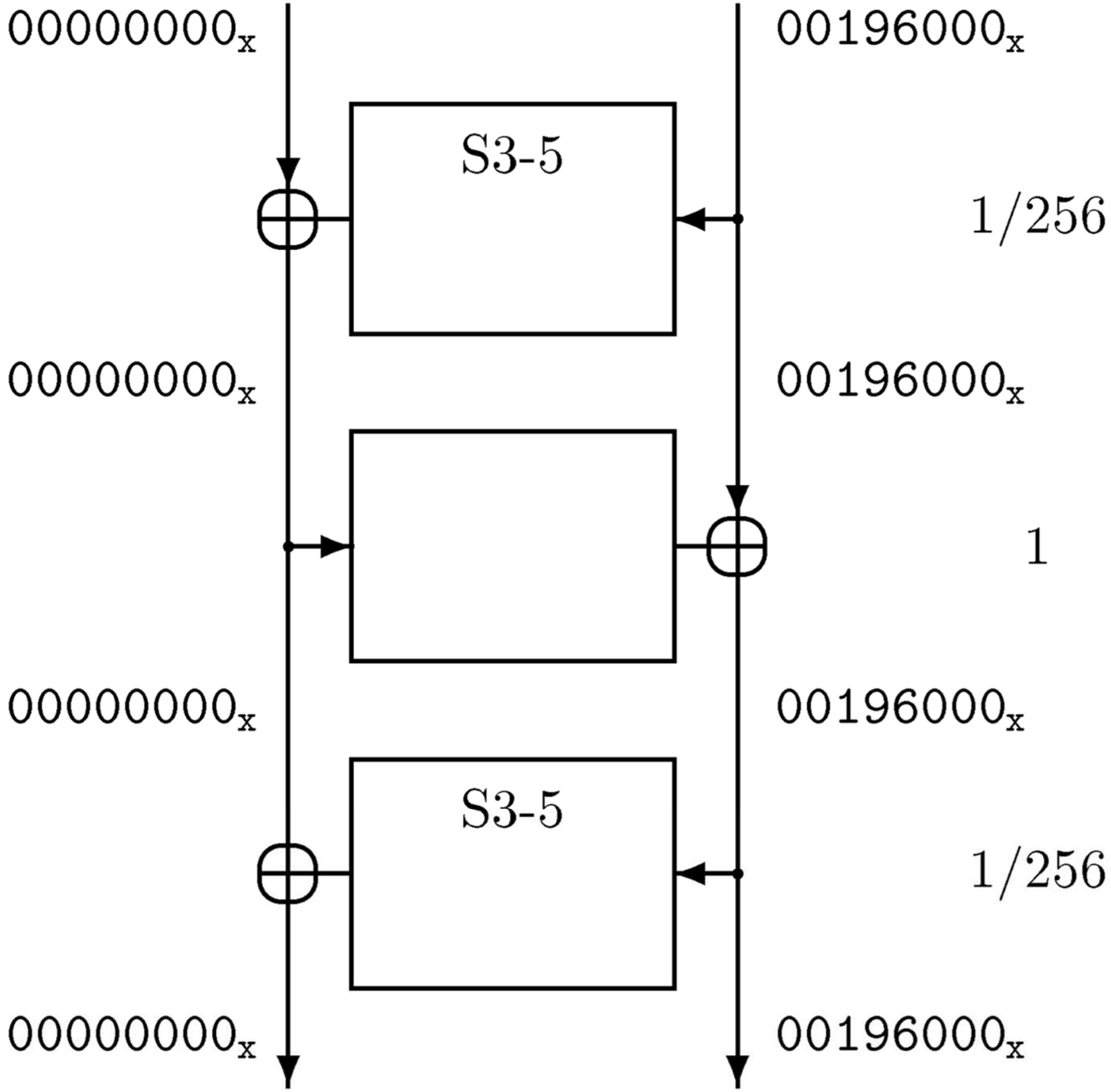


Figure 1: An example of Differential Cryptanalysis

DC complexity

Plaintexts = 1 / probability

No “noise”,

Looking for an exceptional event the almost never happens by itself.

Very strong property that gives a lot of information !

DES
and Algebraic Attacks

[recent work]

Results on DES

Nicolas T. Courtois and Gregory V. Bard: "Algebraic Cryptanalysis of the D.E.S.".

In IMA Cryptography and Coding 2007
18-20 December 2007, Cirencester, UK
eprint.iacr.org/2006/402/

What Can Be Done ?

As of today, we can:

Idea 1+ Method 1:

Recover the key of 5-round DES with
3 known plaintexts faster than brute force.

Idea 2 + Method 2:

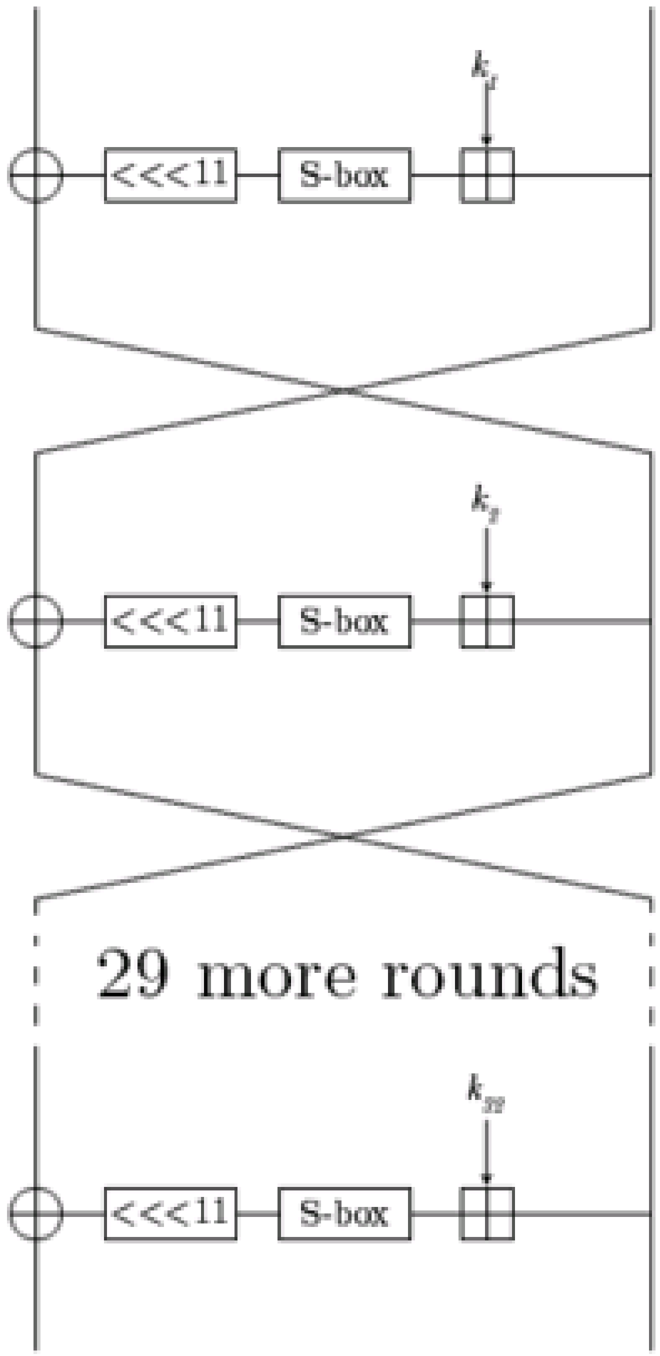
Key recovery for 6-round DES !
1 known plaintext (!).

GOST

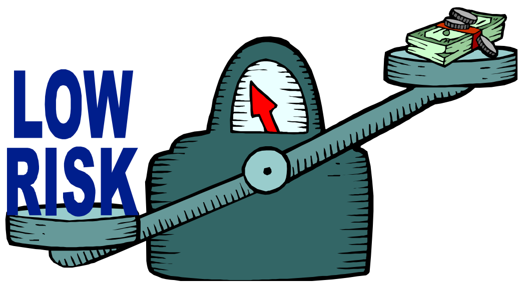
GOST 28147-89

- 64-bit block, 256-bit key, 32 rounds
- Slow diffusion,
  - lack of P-box
- Ultra-simple key schedule
  - 3xdirect, 1xreversed
- 8 secret S-boxes. (354 bits of info)
  - Central Bank of Russia uses these:

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



So What?



Summary

attack method	data complexity		storage complexity	processing complexity
	known	chosen		
exhaustive precomputation	—	1	2^{56}	1 (table lookup)
exhaustive search	1	—	negligible	2^{55}
linear cryptanalysis	2^{43} (85%)	—	for texts	2^{43}
	2^{38} (10%)	—	for texts	2^{50}
differential cryptanalysis	—	2^{47}	for texts	2^{47}
	2^{55}	—	for texts	2^{55}

Table 7.7: *DES strength against various attacks.*

Never was "really" broken [Coppersmith Crypto 2000]