BLOCK CIPHERS

FOCUS ON THE LINEAR LAYER (FEAT. PRIDE)

Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun,
Gregor Leander, Christof Paar, ***Tolga Yalçın***

(Royal Holloway, University of London | Infineon AG | HGI, Ruhr University Bochum)

Block Ciphers!

Block Ciphers!

Focus: Linear Layer

Block Ciphers!

Focus: Linear Layer

PRIDE

Block Ciphers!

Focus: Linear Layer

PRIDE

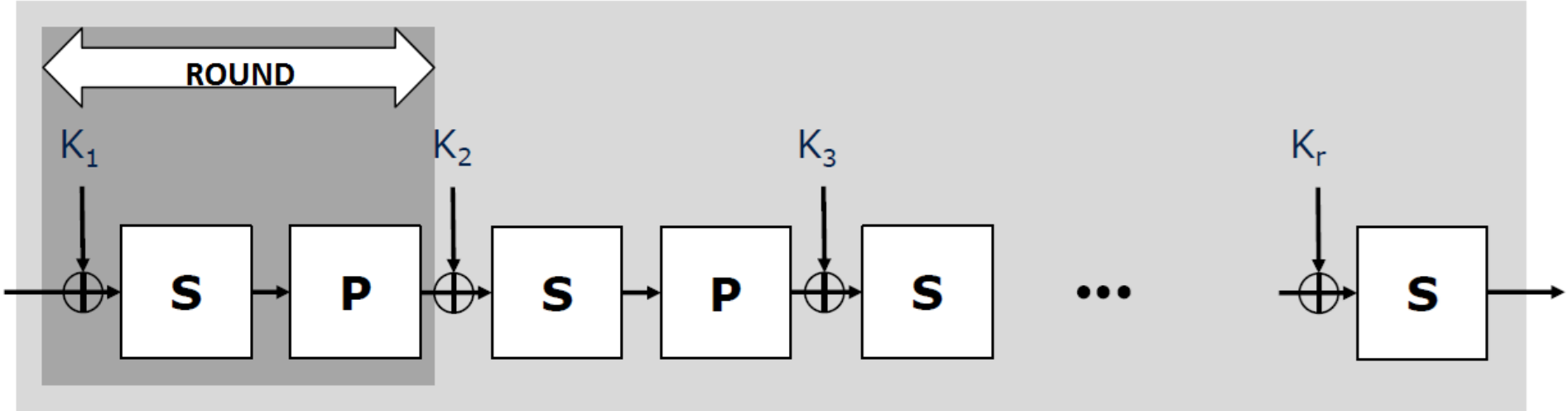
BLOCK CIPHERS!

- One of most prominently used cryptographic primitives
  - A large portion of data encrypted using block ciphers
  - With the rise of ubiquitous computing: Lightweight block ciphers!
- Two main design strategies
  - Constructions without Sbox (ARX)
  - Sbox-based constructions

BLOCK CIPHERS!
SBOX-BASED CONSTRUCTIONS

- Feistel ciphers
  - Most prominent example: DES
- Substitution-permutations networks (SPN)
  - Most prominent example: AES
    - Strong security
    - Demonstrated that well-chosen linear layer facilitates security argument
    - Good results on software & hardware platforms: Not especially light!
    - Inspired many cipher designs
  - Many lightweight proposals (targeting hardware)
    - PRESENT, mCrypton, LED, PRINCE, ...

BLOCK CIPHERS!
SPN CONSTRUCTION



- Non-linear layer (Sboxes)
  - Very well-studied: Many papers, designs, ...
- Linear layer
  - Understudied...

Block Ciphers! ✓

Focus: Linear Layer

PRIDE

FOCUS: LINEAR LAYER

- The main role of the linear layer: ***Diffusion!***
- Desired properties
  - High and fast dependency
  - High number of active Sboxes
  - Efficiency
    - Especially in software (often guarantees hardware)

FOCUS: LINEAR LAYER

TWO DESIGN APPROACHES

- Ad hoc constructions
  - Prominent examples – Serpent, SHA-3
  - Secure, efficient
  - Not easy
  - Not satisfactory from scientific point-of-view
- Wide-trail strategy
  - MDS codes: Efficient examples (serial, hardware) – PHOTON, LED
  - Secure, usually costly
  - Easier
  - Hardly any trade-offs known

FOCUS: LINEAR LAYER
OPEN PROBLEMS

- Observations
  - Not many ***general*** linear layer constructions known
    - Allows to choose between a large variety of trade-offs
  - Existing linear layer solutions generally target low area/latency/power
    - Costly in software platforms (speed/code size)

FOCUS: LINEAR LAYER

NEW DIRECTIONS

- A new methodology to construct good linear layers!
- Optimize for software!

FOCUS: LINEAR LAYER
CONSTRUCTION PRINCIPLE

Block interleaving construction

Given k [$2n$, n, d] codes over F_2^b, construct a code with same parameters over F_2^{kb}

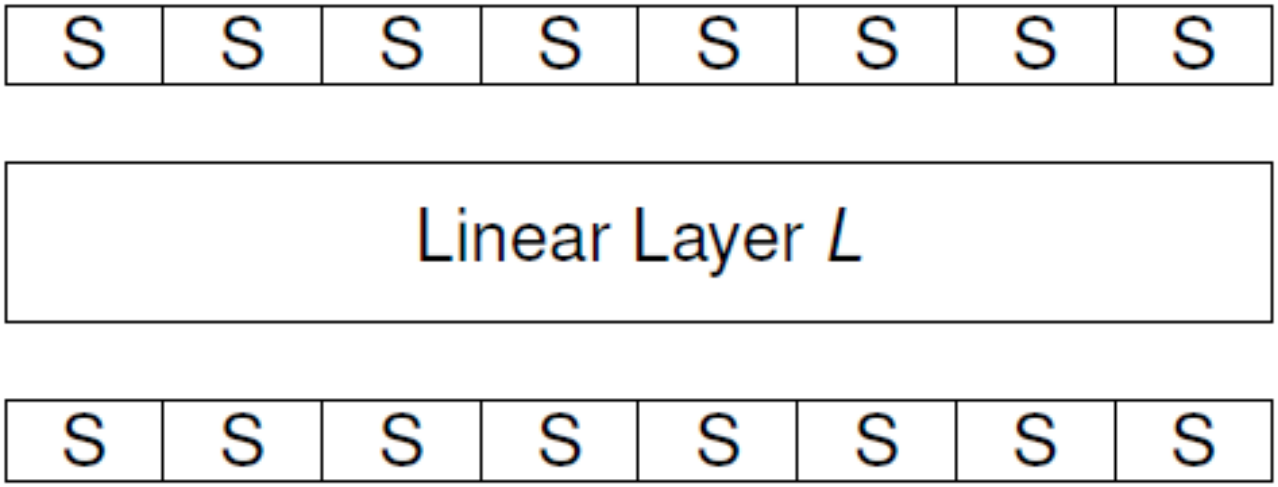
- k generator matrices of form (I | L_i) construct matrix (I | L)

$$L = P \circ \begin{pmatrix} L_1 & & \\ & \ddots & \\ & & L_k \end{pmatrix} \circ P^{-1}$$

  - P is a bit permutation (Recall: ShiftRows)

FOCUS: LINEAR LAYER

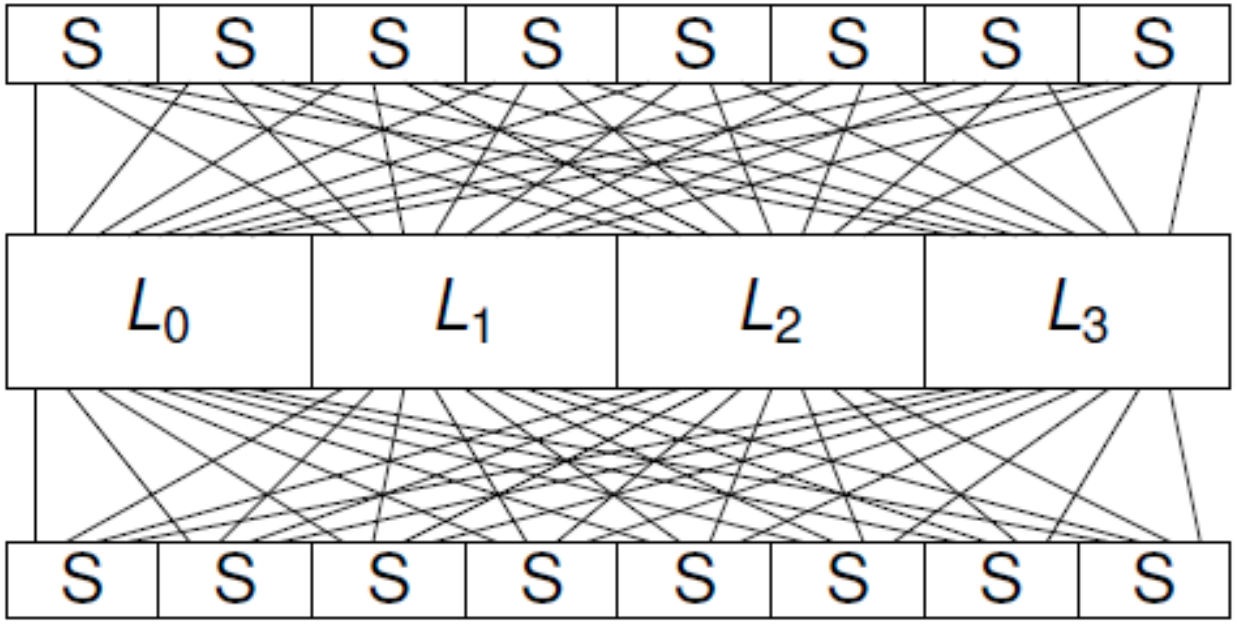
EXAMPLE



- 8 Sboxes of 4-bit each
  - $k = 4$, $n = 8$, $b = 1$, $d = 4$
- Split L in 4 parts, L_i, each 8 x 8 matrix
  - Four [16, 8, 4] codes over F_2, construct a code over F_2^4

FOCUS: LINEAR LAYER

EXAMPLE



- Suitable for software?
  - Not too much at first glance

FOCUS: LINEAR LAYER

EXAMPLE

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

FOCUS: LINEAR LAYER

EXAMPLE

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

FOCUS: LINEAR LAYER

EXAMPLE

7	15	23	31
6	14	22	30
5	13	21	29
4	12	20	28
3	11	19	27
2	10	18	26
Sbox			
Sbox			

FOCUS: LINEAR LAYER

EXAMPLE

7	15	23	31
6	14	22	30
5	13	21	29
4	12	20	28
3	11	19	27
Sbox			
Sbox			
Sbox			

FOCUS: LINEAR LAYER

EXAMPLE

7	15	23	31
6	14	22	30
5	13	21	29
4	12	20	28
Sbox			
Sbox			
Sbox			
Sbox			

FOCUS: LINEAR LAYER

EXAMPLE

7	15	23	31
6	14	22	30
5	13	21	29
Sbox			
Sbox			
Sbox			
Sbox			
Sbox			

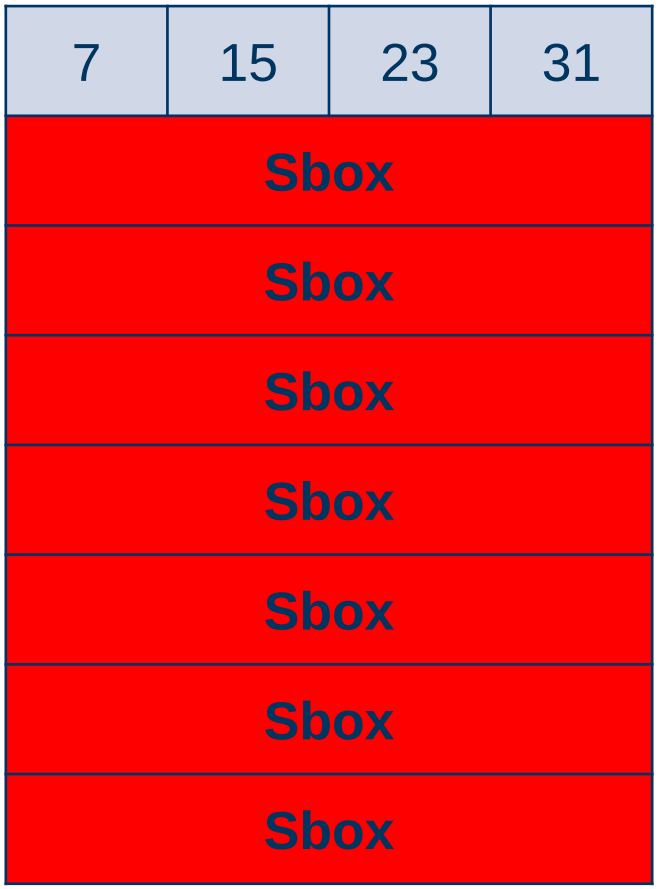
FOCUS: LINEAR LAYER

EXAMPLE

7	15	23	31
6	14	22	30
Sbox			
Sbox			
Sbox			
Sbox			
Sbox			
Sbox			

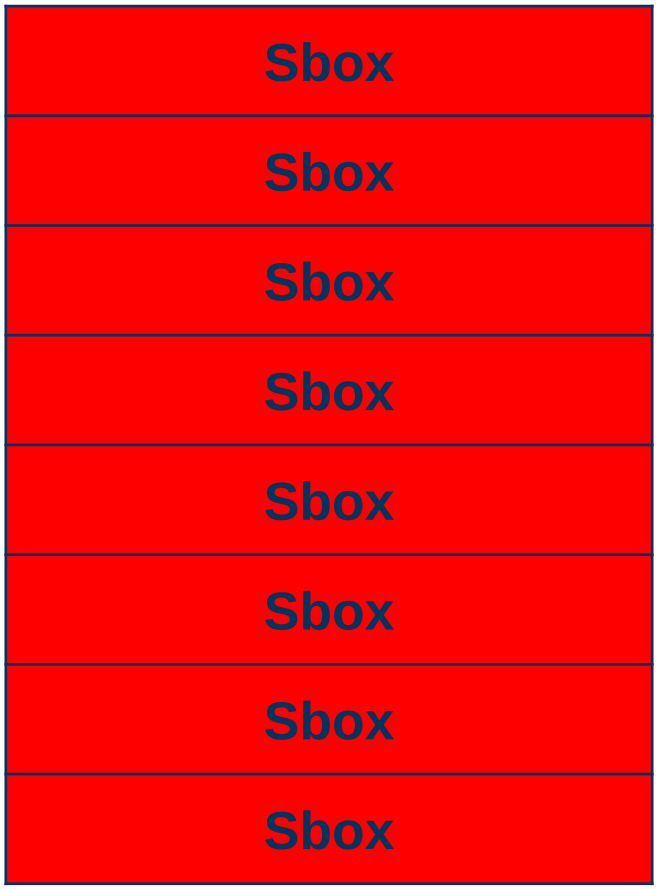
FOCUS: LINEAR LAYER

EXAMPLE



FOCUS: LINEAR LAYER

EXAMPLE



FOCUS: LINEAR LAYER

EXAMPLE

L_0	15	23	31
	14	22	30
	13	21	29
	12	20	28
	11	19	27
	10	18	26
	9	17	25
	8	16	24

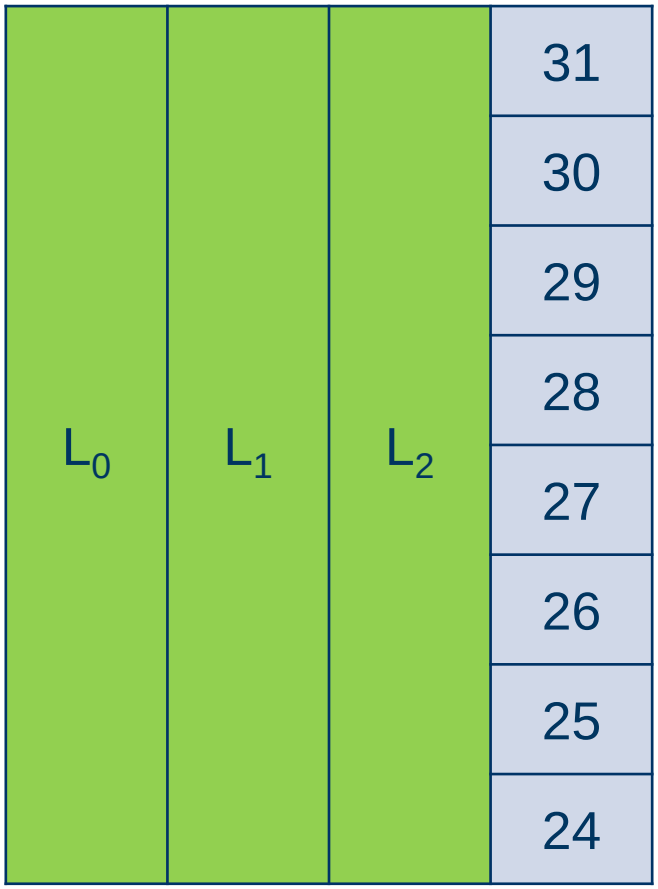
FOCUS: LINEAR LAYER

EXAMPLE

L_0	L_1	23	31
		22	30
		21	29
		20	28
		19	27
		18	26
		17	25
		16	24

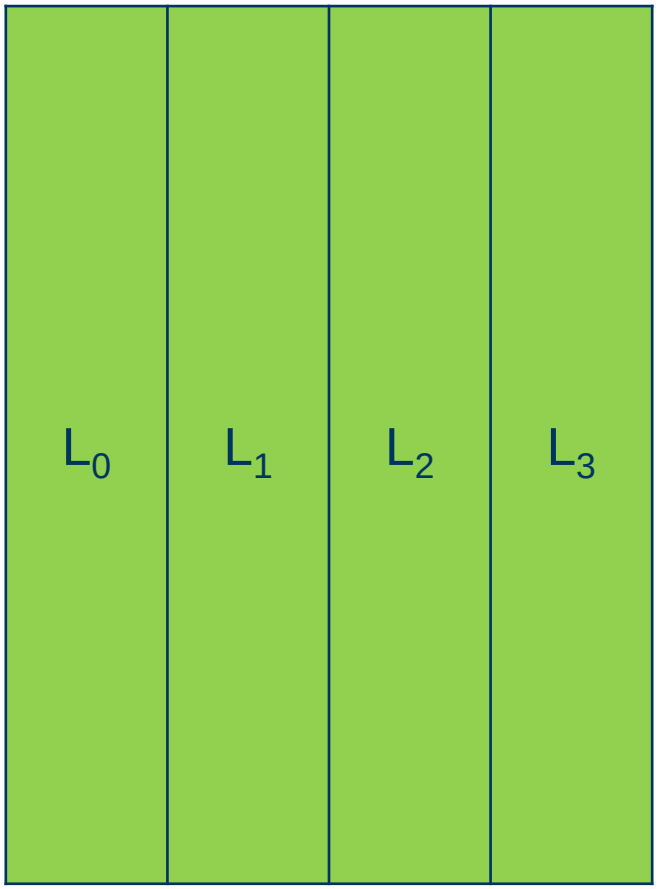
FOCUS: LINEAR LAYER

EXAMPLE



FOCUS: LINEAR LAYER

EXAMPLE



FOCUS: LINEAR LAYER

HOW TO CHOOSE L_i

- Looking for “the cheapest implementation of a given linear layer”?

FOCUS: LINEAR LAYER

HOW TO CHOOSE L_i

- Looking for “the cheapest implementation of a given linear layer”?

NO!

FOCUS: LINEAR LAYER

HOW TO CHOOSE L_i

- Looking for “the cheapest implementation of a given linear layer”?

NO!

Instead...

FOCUS: LINEAR LAYER

HOW TO CHOOSE L_i

- Looking for “the cheapest implementation of a given linear layer”?

NO!

Instead...

- “Which linear layers can be implemented with N instructions”?
  - In turn it gives us also...
    - # of *clock cycles* for speed
    - #of *bytes* for code size

FOCUS: LINEAR LAYER

HOW TO CHOOSE L_i

- Interleaving helps there!
- Focus on smaller linear layers
  - Reduces the search space

Block Ciphers! ✓

Focus: Linear Layer ✓

PRIDE

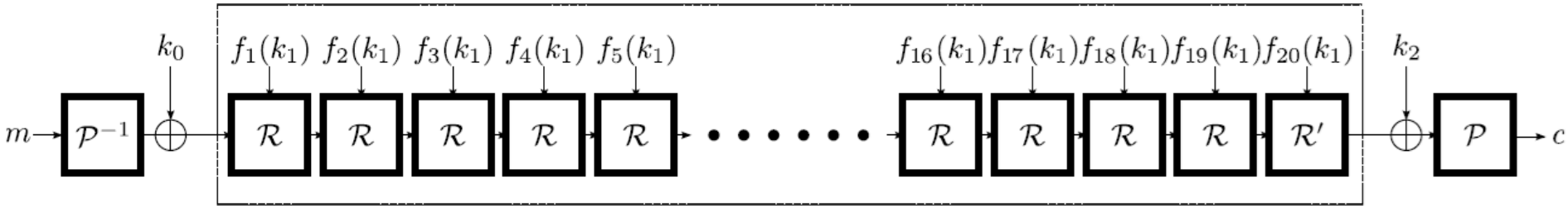
PRIDE
TARGET

- Software-oriented for widely-used embedded microprocessors
- Design target is Atmel AVR 8-bit instruction set
  - For easier and fairer comparison
- Traditional design methods for easier security analysis
- Benchmark: SPECK-64/128 cipher of NSA*
  - ARX design followed in SPECK

* Beaulieu et al, The Simon and Speck Families of Lightweight Block Ciphers, IACR ePrint Archive 2013/404, 2013

PRIDE

KEY FIGURES



- 64-bit block, 128-bit key
- FX construction with pre- and post-whitening
- 20 rounds (first 19 identical)
- Simple key scheduling
  - 128-bit key divided: $k = k_0 \,||\, k_1$ and $k_2 = k_0$
  - f( ) modifies 4 bytes of k_1 per round, using round constants
- P and P^{-1} permutations go away due to bit-sliced design

PRIDE
SUBSTITUTION LAYER

- An involution Sbox used
  - Efficiently implementable 4-bit Sbox
  - 10 instructions per 8 parallel Sboxes
  - Best correlation of any linear approximation: 1/2
  - Maximal probability of a differential: 1/4

PRIDE
LINEAR LAYER

- Use block interleaving construction
- Search for efficient codes
  - Look for 4 efficiently-implementable, linear [32, 16, 4] codes
  - Similar to Sbox search of Ullrich et al.*
    - Search performed on hardware platform instead of software platform
    - Faster search, larger search space

***** Ullrich et al., Finding Optimal Bitsliced Implementations of 4 x 4-bit S-boxes, SKEW 2011

PRIDE
LINEAR LAYER SEARCH ON HARDWARE

- Search in a subset of possible 16 x 16 matrices using an FPGA
  - Limit number of instructions
    - CLC, EOR, MOV, MOVW, CLR, SWAP, ASR, ROR, ROL, LSR, LSL
  - Limit number of used registers
    - 2 state, 4 temporary registers
  - Try all possible combinations of instructions and registers
  - Save the matrices generating appropriate code
    - Out of these, look for the ones with least instructions
  - Ended up with ***36 instructions*** for the whole linear layer!

PRIDE
LINEAR LAYER SEARCH FOR HARDWARE

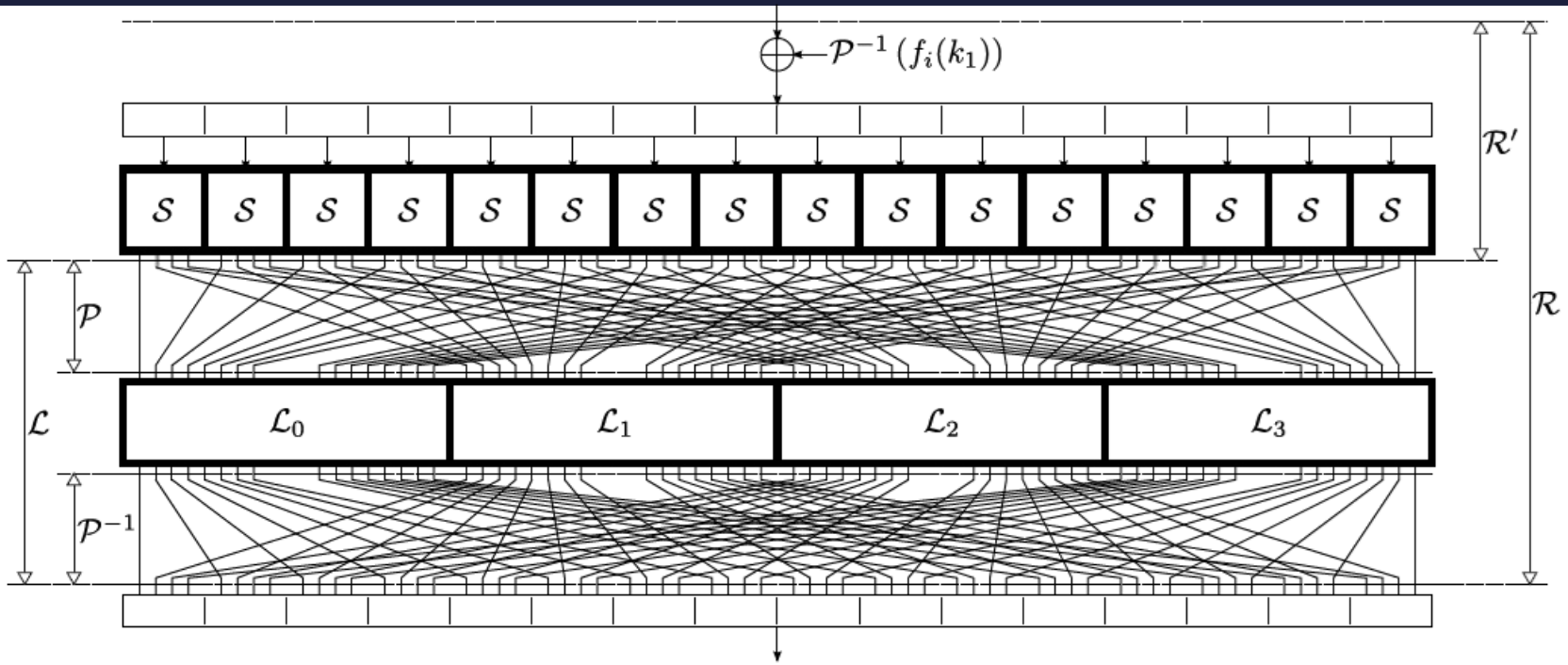
- Search in a subset of possible 16 x 16 matrices using an FPGA
  - Limit number of instructions
    - CLC, EOR, MOV, MOVW, CLR, SWAP, ASR, ROR, ROL, LSR, LSL

$$\begin{bmatrix} x_3 \\ x_2 \\ x_1 \\ x_0 \\ x_7 \\ x_6 \\ x_5 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} x_7 \\ x_6 \\ x_5 \\ x_4 \\ x_3 \\ x_2 \\ x_1 \\ x_0 \end{bmatrix}$$

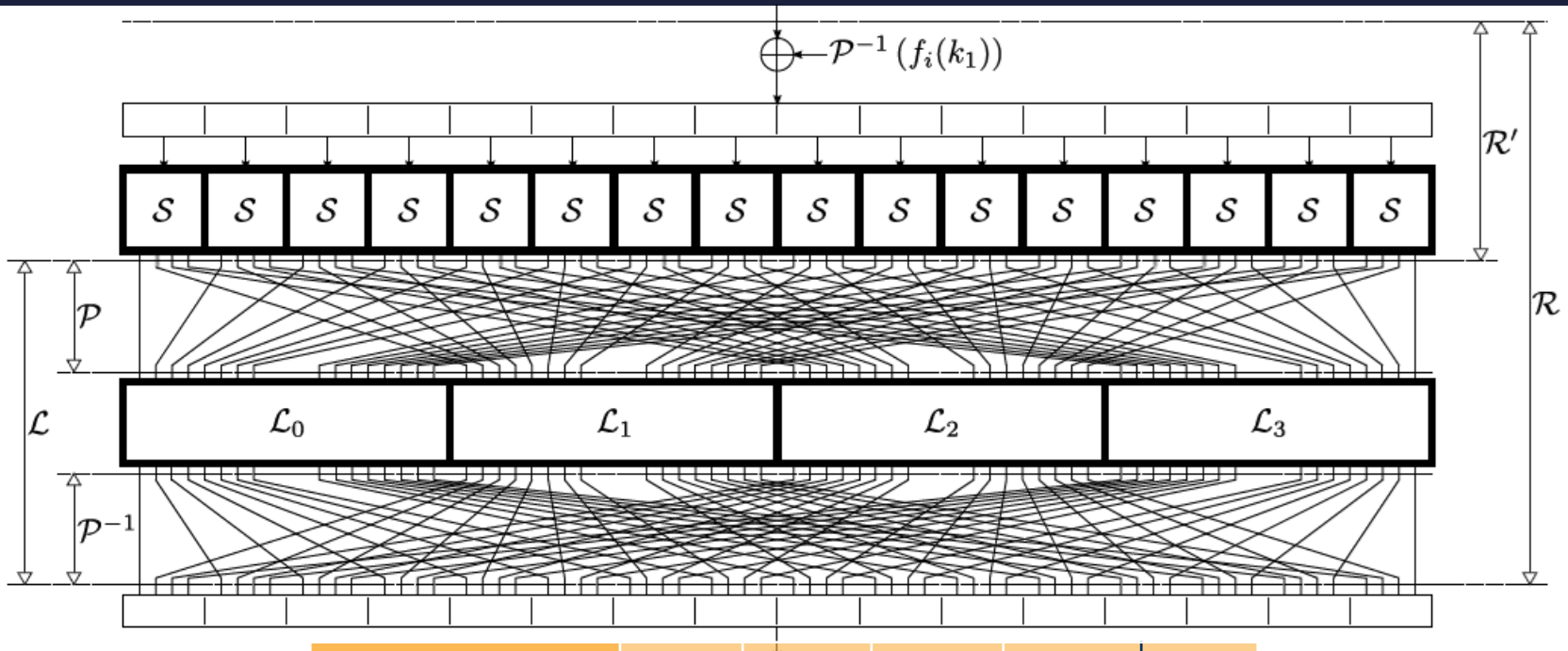
SWAP

PRIDE

ROUND FUNCTION



PRIDE
ROUND FUNCTION



One Round Cost	Key Update	Key Addition	Sbox Layer	Linear Layer	Total
Time (Clock Cycles)	4	8	20	36	68
Code Size (Bytes)	8	16	40	72	136

PRIDE
RESULTS

	AES-128	SERPENT-128	PRESENT-128	CLEFIA-128	SEA-96	NOEKEON-128	PRINCE-128	ITUBee-80	SIMON-64/128*	SPECK-64/96*	SPECK-64/128*	PRIDE
Time (Clock Cycles)	3159	49314	10792	28648	17745	23517	3614	2607	2000	1152	1200	1514
Code Size (Bytes)	1570	7220	660	3046	386	364	1108	716	282	182	186	266

***** Data & key read-write omitted

- Performance on Atmel AVR microcontroller (encryption)
  - *PRIDE decryption:* 1570 clock cycles and 282 bytes
- Results close to SPECK
  - Good results for a “traditional” design

PRIDE
RESULTS

- Also efficient on 16-bit microcontroller (TI – MSP430)*
  - ***PRIDE encryption*: 2424 clock cycles & 358 bytes**
  - ***PRIDE decryption*: 2598 clock cycles & 390 bytes**
  - *AES-128 encryption (size-opt.)*: **4763 clock cycles & 1284 bytes**
  - *AES-128 encryption (speed-opt.)*: **4204 clock cycles & 4900 bytes**
  - *PRESENT-80 encryption (size-opt.)*: **8061 clock cycles & 834 bytes**
  - *PRESENT-80 encryption (speed-opt.)*: **6752 clock cycles & 17072 bytes**
  - *PRINCE encryption*: **2488 clock cycles & 2584 bytes**

***** Dennis Schweer, Efficiency Analysis of Block Ciphers on Sensor Nodes, B.Sc. Thesis, Ruhr University Bochum, 2014

PRIDE
SECURITY

- Linear and differential cryptanalysis performed
- Best possible linear and differential trails generated for 16 rounds
  - No clustering of these optimal trails
- Other attacks (zero-correlation, algebraic, ...)
  - No serious issues
- Further security analysis encouraged!

PRIDE
SECURITY

- Linear and differential cryptanalysis performed
- Best possible linear and differential trails generated for 16 rounds
  - No clustering of these optimal trails
- Other attacks (zero-correlation, algebraic, ...)
  - No serious issues
- Further security analysis encouraged!

Actually, there already are quite a few!

- Zhao et al.: Differential Analysis on Block Cipher PRIDE*
  - Found 16 different 2-round iterative characteristics
  - Constructed several 15-round differentials
  - Based on these, launched differential attack on 18-round PRIDE
  - Data, time, and memory complexity are 2^{60}, 2^{66}, and 2^{64}

***** Zhao et al., Differential Analysis on Block Cipher PRIDE, IACR ePrint Archive 2014/525, 2014

PRIDE
FUTURE DIRECTIONS

- Linear layer
  - Improve hardware search, cover larger space
  - Find more efficient constructions
  - Explore trade-offs
  - Extend to different platforms (PIC, ARM, etc.)

- PRIDE
  - (Even) more security analysis

Block Ciphers! ✓

Focus: Linear Layer ✓

PRIDE ✓

Block Ciphers! ✓

Focus: Linear Layer ✓

PRIDE ✓

Thanks for Listening!

Block Ciphers! ✓

Focus: Linear Layer ✓

PRIDE ✓

Any Questions?