Noekeon

Joan Daemen*, Gilles Van Assche*,
Michael Peeters* and Vincent Rijmen**

***Proton World, Brussels**
****COSIC, Leuven**

Outline

- **Noekeon design philosophy and properties**
- **Round transformation and components**
- **Key schedule modes**
- **Resistance against cryptanalysis**
  - **Propagation analysis**
- **Implementation aspects**
- **The inverse cipher**
- **Surprising properties of Noekeon**
- **Conclusions**

Noekeon Design Philosophy

- **Security: resistance against known types of cryptanalysis and implementation attacks**
- **and Efficiency: fast and compact in software and dedicated hardware**
- **through Symmetry:**
  - **iterated cipher with one single, round transformation**
  - **bit-wise Boolean operations and cyclic shifts only**
  - **same round key for each round: *working key***
  - **inverse cipher is (almost) equal to the cipher**

Noekeon Properties

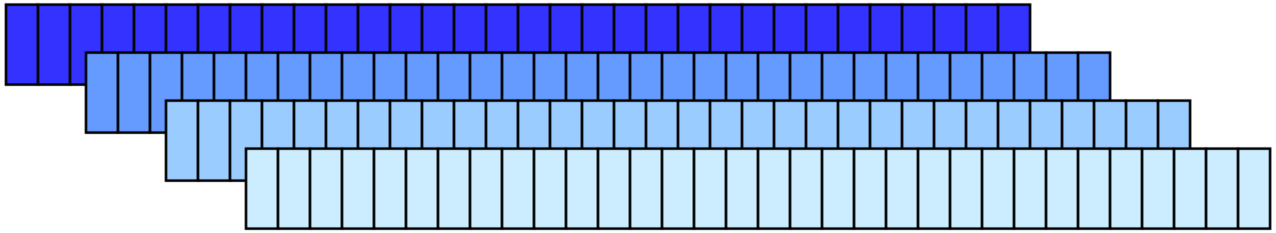
- **Block Cipher**
  - **128-bit key**
  - **128-bit block**
- **Substitution-linear transformation network in bit-slice mode**
  - **inspired by 3-Way [Da93] and BaseKing [Da95]**
  - **very similar to Serpent [BAK98]**
- **Optional key schedule**
  - **key schedule only needed when related-key attacks can be mounted**

Round Transformation

- Noekeon has 16 equal rounds
- Round transformation consists of 5 steps:
  - Round constant addition
  - Theta: diffusion and key addition
  - Pi1: permutation
  - Gamma: non-linearity
  - Pi2: permutation
- Output transformation:
  - Theta

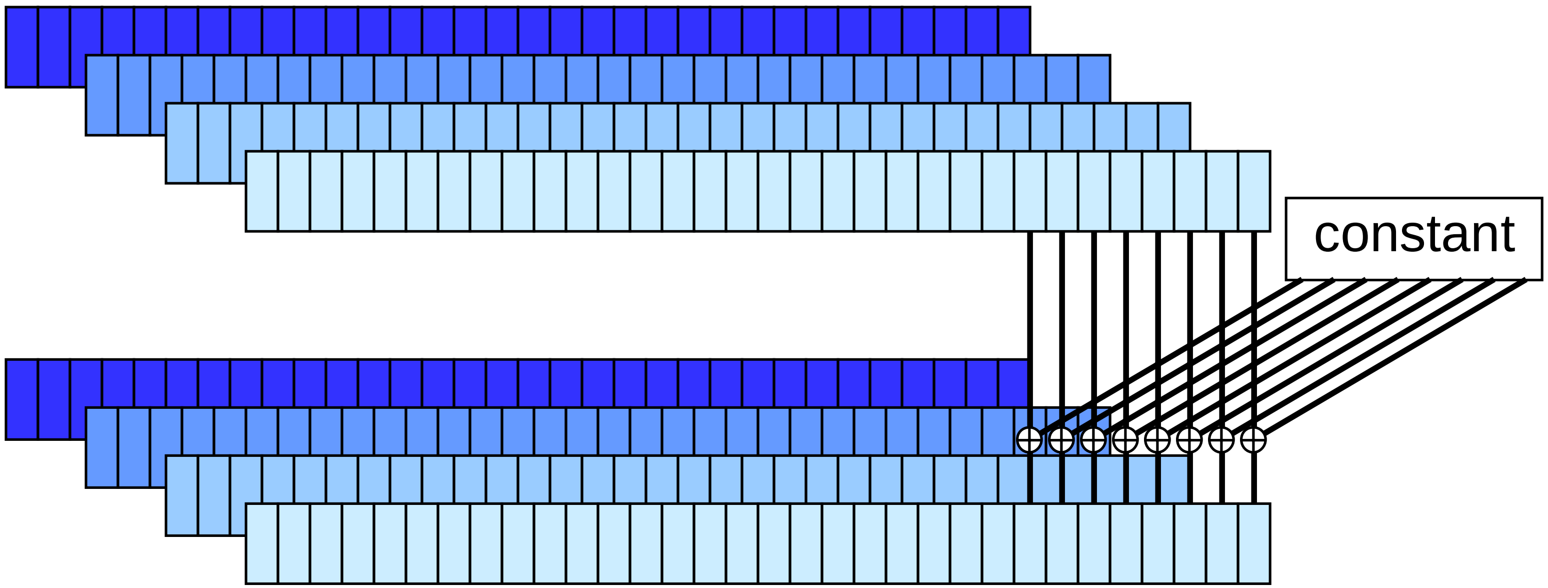
The Noekeon State

- **All round transformations operate on a state consisting of 4 32-bit words: a_0, a_1, a_2, a_3**



Round Constant Addition

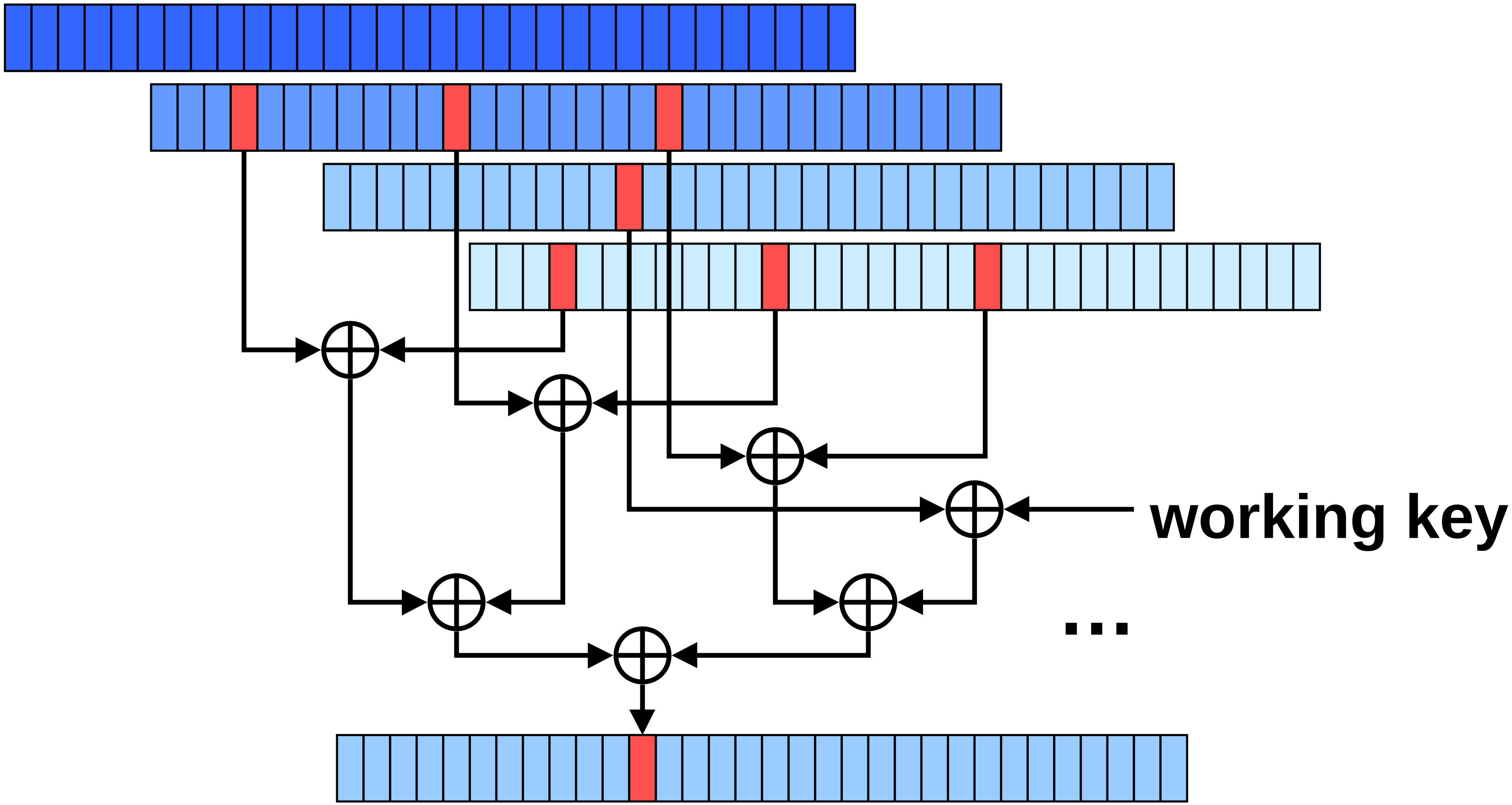
- **Break symmetry between the words and between the rounds**



Theta

- **Linear transformation in 3 steps:**
  - **modification of odd words**
  - **addition of *working key***
  - **modification of even words**
- **Symmetry within the state words:**
  - **all bits are treated in the same way**
- **High average diffusion**
- **Involution**

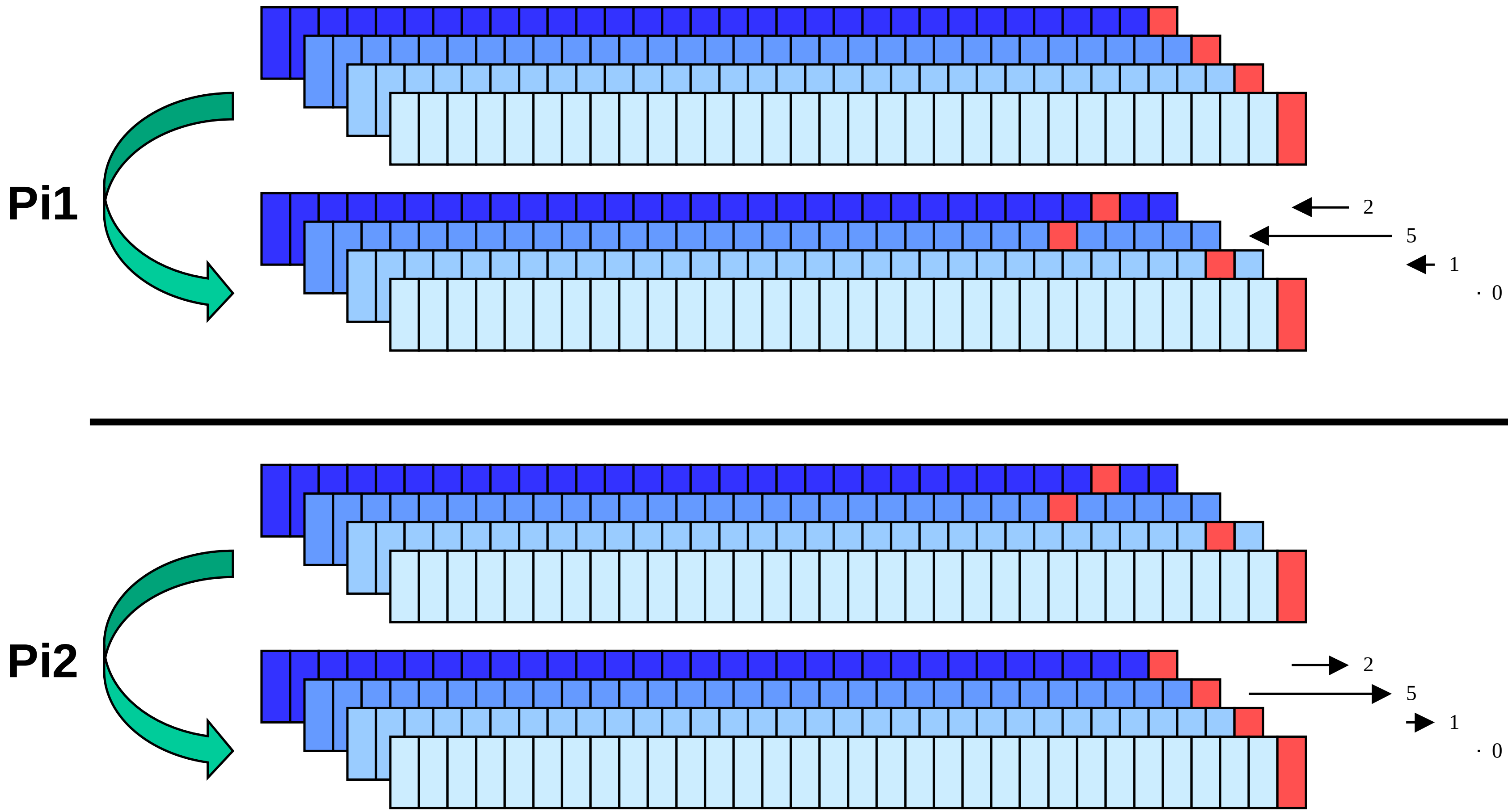
Theta Illustrated



Pi1 and Pi2

- Cyclic shift of words a_1, a_2, a_3
- Symmetry within the state words:
  - all bits in a word are treated in the same way
- Give high multiple-round diffusion in combination with Theta and Gamma
- Pi1 and Pi2 are each others inverse:
  - Pi1 shifts are 1, 5 and 2 to the left
  - Pi2 shifts are 1, 5 and 2 to the right

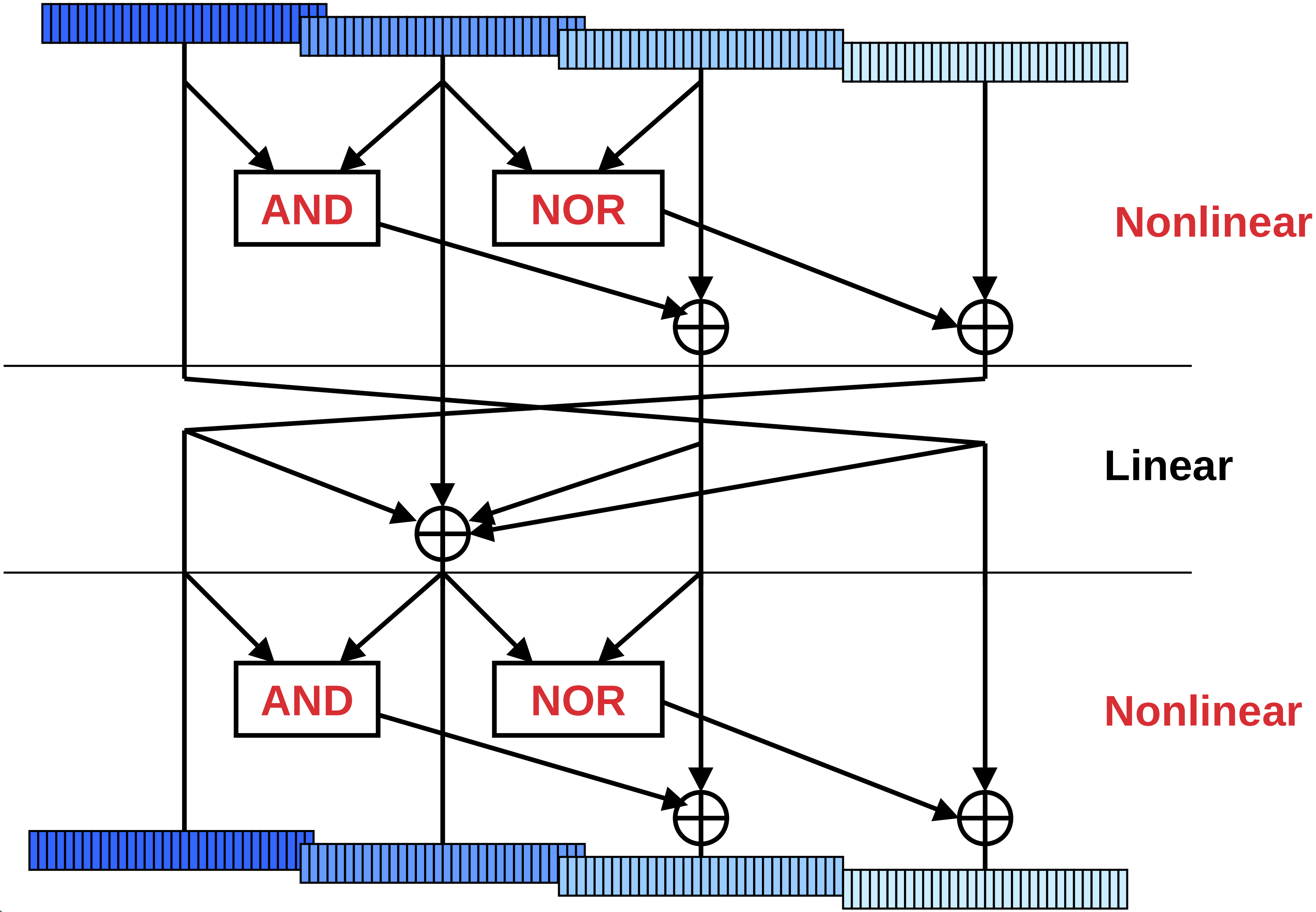
Pi1 and Pi2



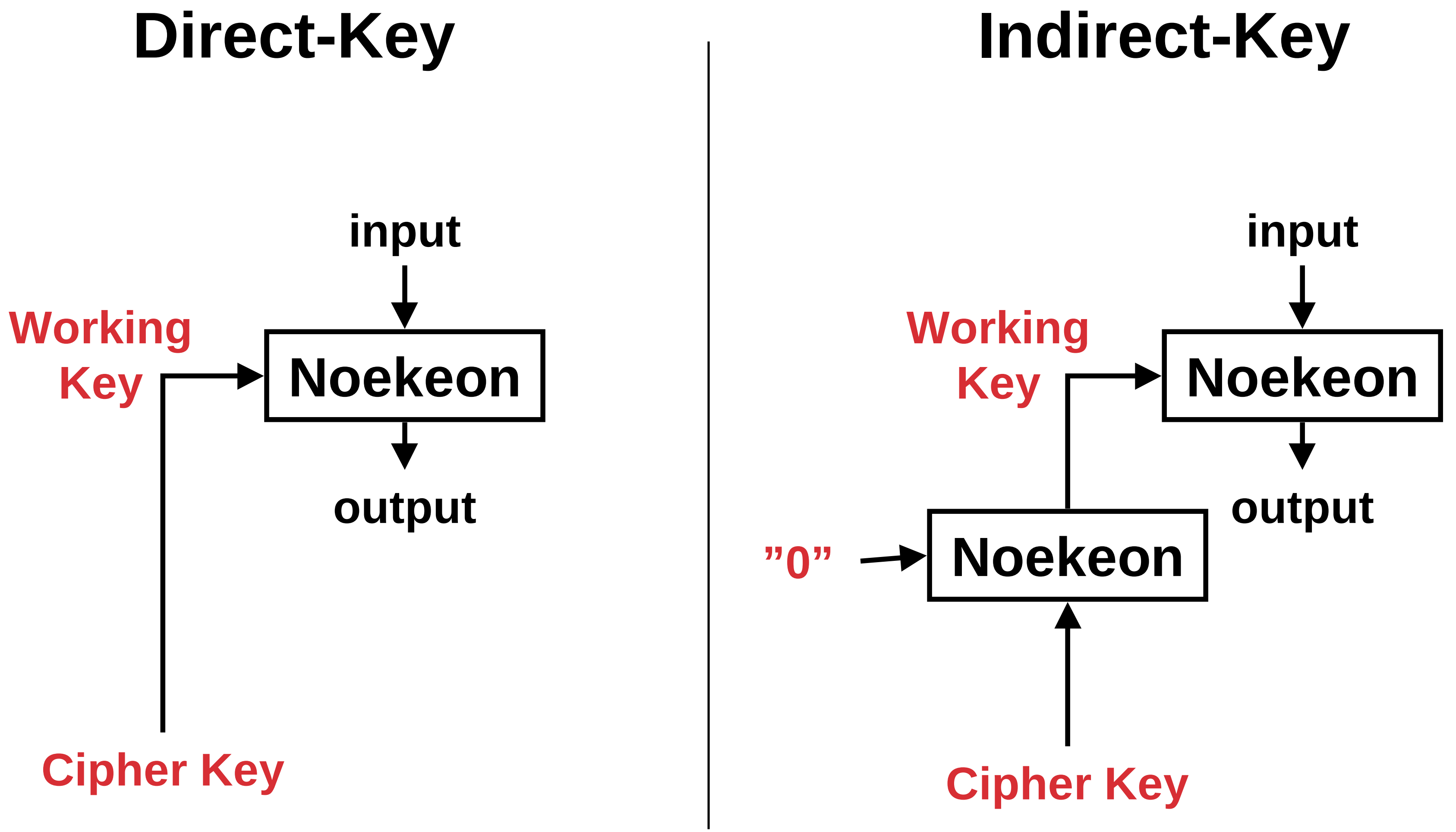
Gamma

- **Nonlinear transformation in 3 steps:**
  - **simple nonlinear transformation**
  - **simple linear transformation**
  - **simple nonlinear transformation**
- **Symmetry within the state words:**
  - **32 times the same 4-bit S-box**
- **Good nonlinear properties**
- **Involution**

Gamma Illustrated



Key Schedule Modes



Resistance Against Cryptanalysis

- Linear and differential cryptanalysis: propagation analysis
- Truncated differentials
- Interpolation attacks
- Symmetry properties and slide attacks
- Weak keys
- Related-key attacks
  - use indirect-key mode
- Hidden weaknesses and Trapdoors

Propagation Analysis

- **Identification of all 4-round trails with less than 24 active S-boxes ("< 24")**
  - **differential trails: *characteristics***
  - **linear trails: *linear approximations***
- **In the small set of 4-round trails found:**
  - **no differential trails with prob. $> 2^{-48}$**
  - **no linear trails with correlation $> 2^{-24}$**
- **For the full cipher this means:**
  - **DC: no 12-round differential trails with prob. $> 2^{-144}$**
  - **LC: no 12-round linear trails with correlation $> 2^{-72}$**

Propagation Analysis

- **Step 1: recording all 2-round trails (< 18)**
  - **non-trivial exercise!**
  - **made feasible by exploiting symmetry properties in component transformations**
- **Step 2: covering space of 4-round trails (< 24)**
  - **by chaining pairs of recorded 2-round trails (≥ 6)**
  - **the few 2-round trails (< 6) are treated separately**

Table of 2-round Trails

	1	2	3	4	5	6	7	8
1							4	
2		2				14	4	8
3			6		28	12	70	108
4				163	32	178	328	1,493
5			28	32	617	1,283	3,762	6,261
6		14	12	179	1,283	9,101	15,341	54,660
7	4	4	70	328	3,762	15,341	93,668	273,344
8		8	108	1,493	6,261	54,660	273,344	1,249,658
9		1	357	1,972	21,036	129,640	838,646	4,378,578
10		41	305	5,038	44,593	353,545	2,380,721	?
11	1	52	899	9,356	97,629	853,003	?	?
12		113	1,273	18,489	205,194	2,085,751	?	?
13	5	66	1,947	33,605	444,745	4,827,996	?	?
14		149	3,338	63,611	897,923	?	?	?
15		109	5,852	112,168	?	?	?	?
16		199	8,222	?	?	?	?	?

X: number of active S-boxes in round 1, Y: number of active S-boxes in round 2

Hardware Suitability

- **Ultra compact: small number of gates**
  - **1050 XOR**
  - **64 AND**
  - **64 NOR**
  - **128 MUX**
- **High speed: small gate delay**
  - **7 XOR**
  - **1 AND**
  - **1 MUX**

Software Performance

- Very well suited for 32-bit processors
- Pentium II: 525 cycles (49 Mbit/s @ 200 MHz)
- Well suited to other word lengths of form 2^m
- ARM7 (RISC core):

	code size (bytes)	# cycles	bit rate @ 28.56MHz
Min. size	332	712	5.1 Mbit/s
Max speed	3688	475	7.7 Mbit/s

No RAM usage

Protection Against DPA

- Noekeon is a fixed sequence of operations
  - counters timing attack and SPA
- State splitting as applied to BaseKing in our FSE 2000 paper
  - counters first-order DPA (extendable to also counter higher-order DPA) ...
  - at relatively low CPU cost, thanks to few non-linear operations
- In direct-key mode:
  - counters key schedule attacks

The Inverse Cipher

- **The inverse cipher is equal to the cipher**
  - **with the exception of the round constant addition**
- **Because**
  - **Theta and Gamma are involutions**
  - **Pi1 and Pi2 are each others inverses**
- **Cipher and inverse use same hardware circuit or program**

The Unbearable Weakness of Noekeon

- **All round keys are the same!**
- **The linear part of the round has order 2!**
- **The nonlinear part of the round has order 2!**
- **If the round constants are removed:**
  - **all rounds are equal!**
  - **there is a symmetry within the words!**
  - **the cipher and its inverse are equal!**
- **The only non-linearity is provided by some binary ANDs (order 2)!**

➔ **Actual weaknesses? We don't think so…**

Noekeon:

- is ultra compact and fast in hardware,
- runs fast even in DPA-resistant implementations,
- has very low RAM usage in software,
- takes very small amount of code,
- is very efficient on a wide range of platforms,
- so simple that it can be memorized by an average person!