# Two ways of building round functions for block ciphers

Joan Daemen

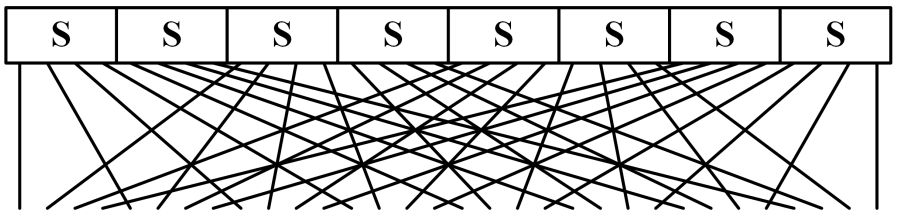
Radboud University

Šibenik summer school 2016

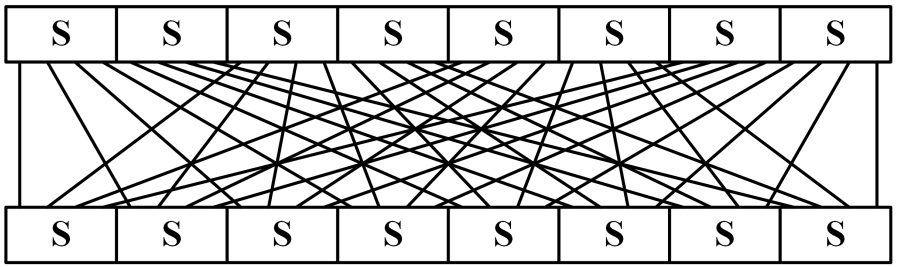
# Outline

# Outline

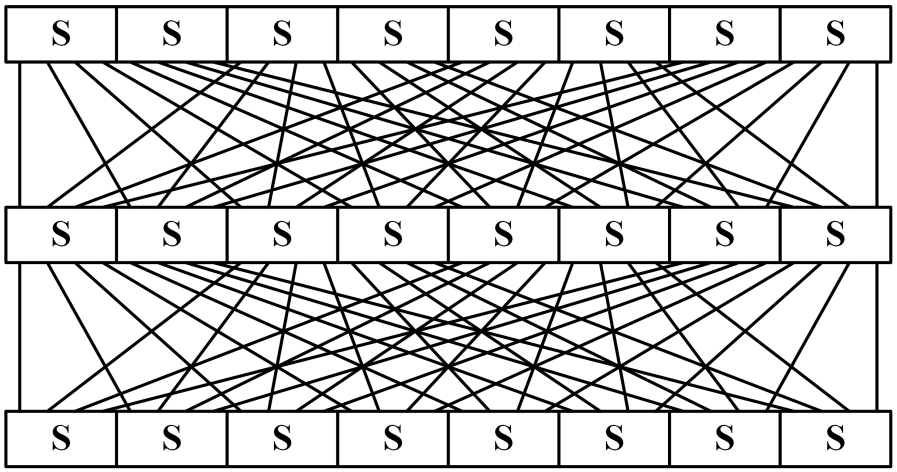
# Product cipher [Claude Shannon, 1949] and SPN



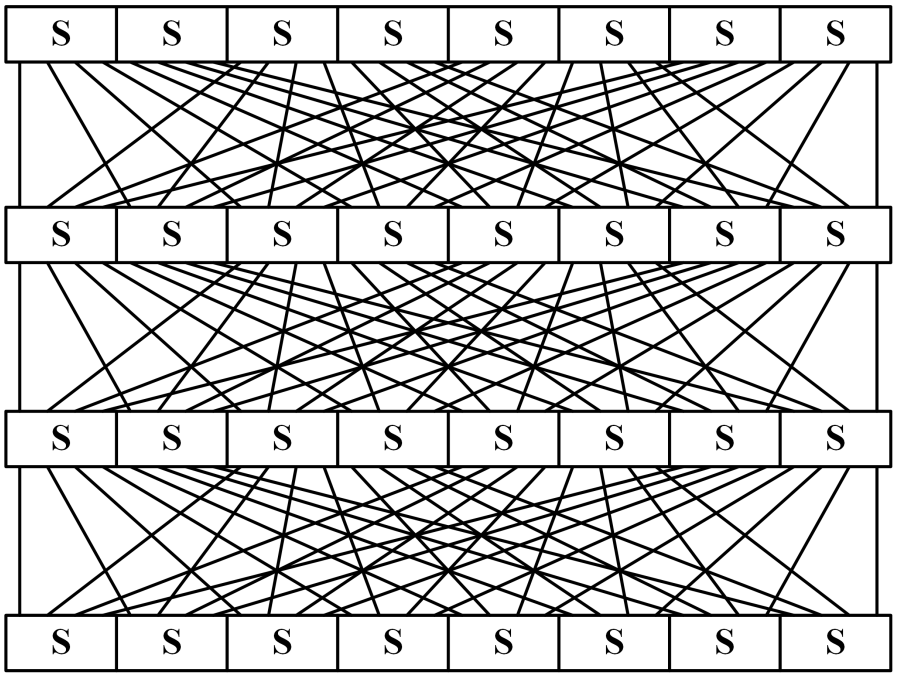
# Product cipher [Claude Shannon, 1949] and SPN



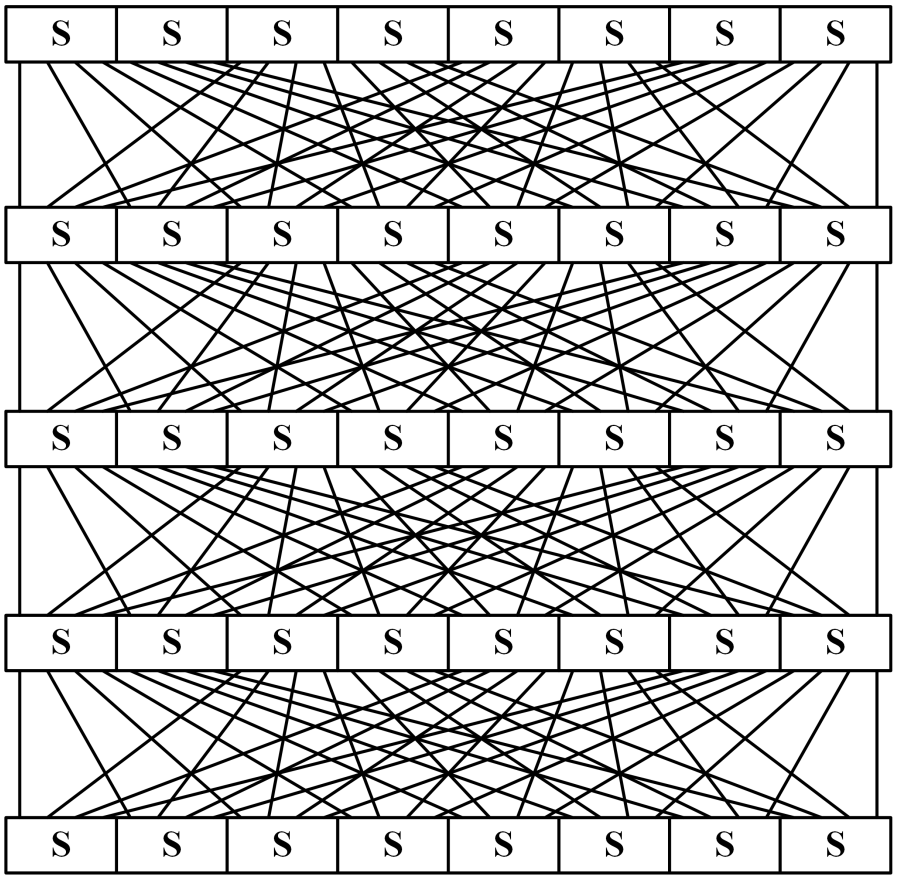
# Product cipher [Claude Shannon, 1949] and SPN



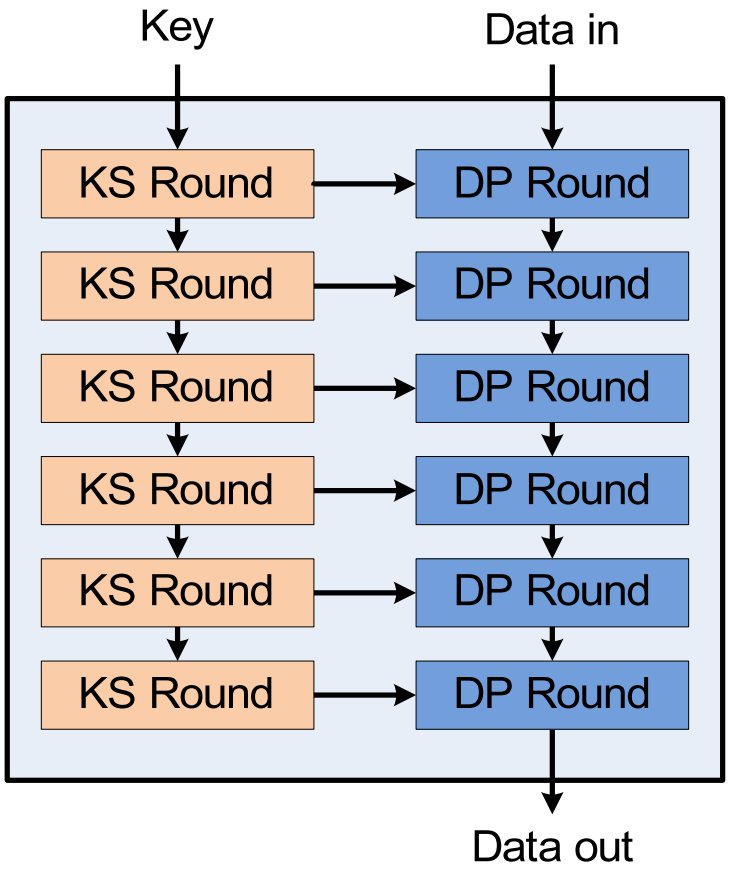
# Product cipher [Claude Shannon, 1949] and SPN



# Product cipher [Claude Shannon, 1949] and SPN

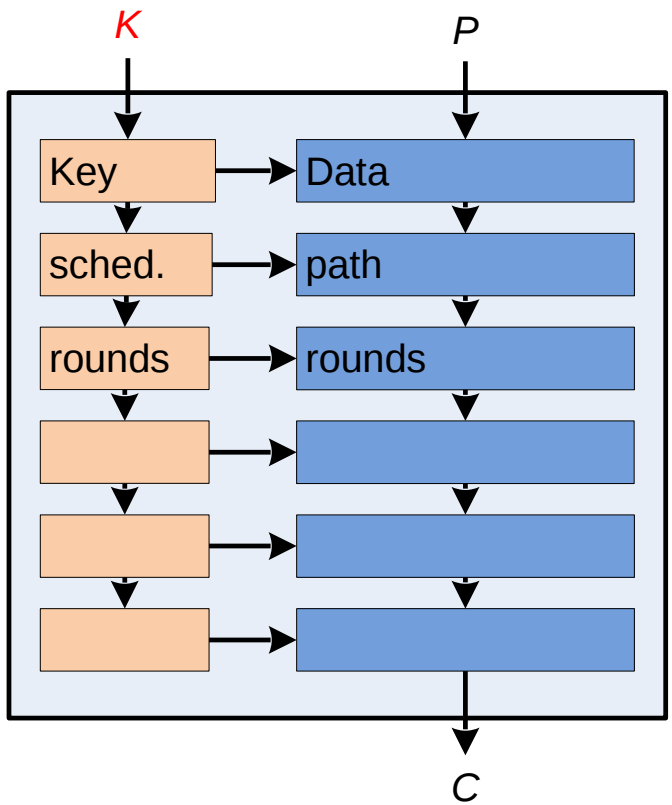


# Iterated block ciphers [DES and later]



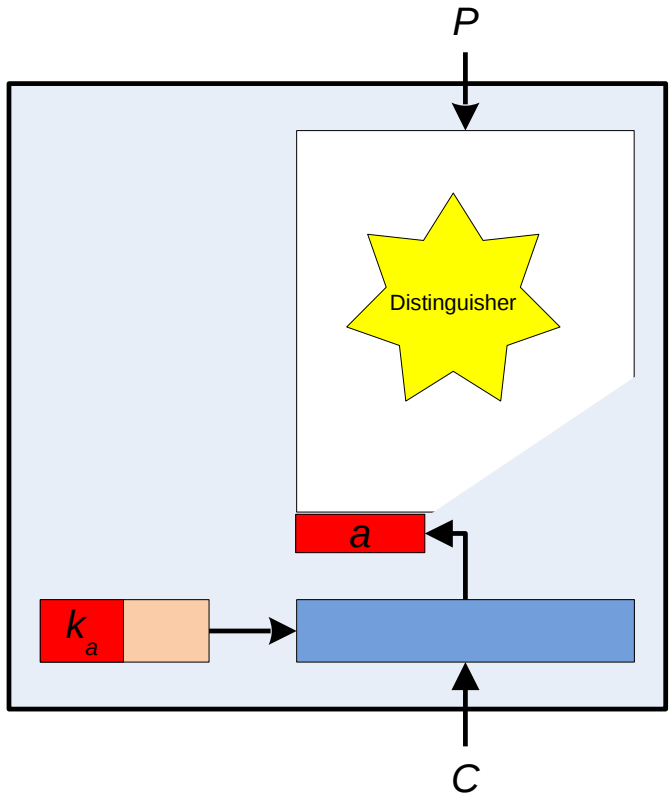
# Statistical attacks

- Exploits *Distinguisher* $\Omega$ over $r-1$ rounds
- Two phases:
  - online: get many $(C_i, P_i)$
  - offline: guess $k_a$
- Wrong guess destroys $\Omega$
- Basic attacks
  - DC: requires $1/DP$ couple
  - LC: requires $1/C^2$ couples
- Many variants ...



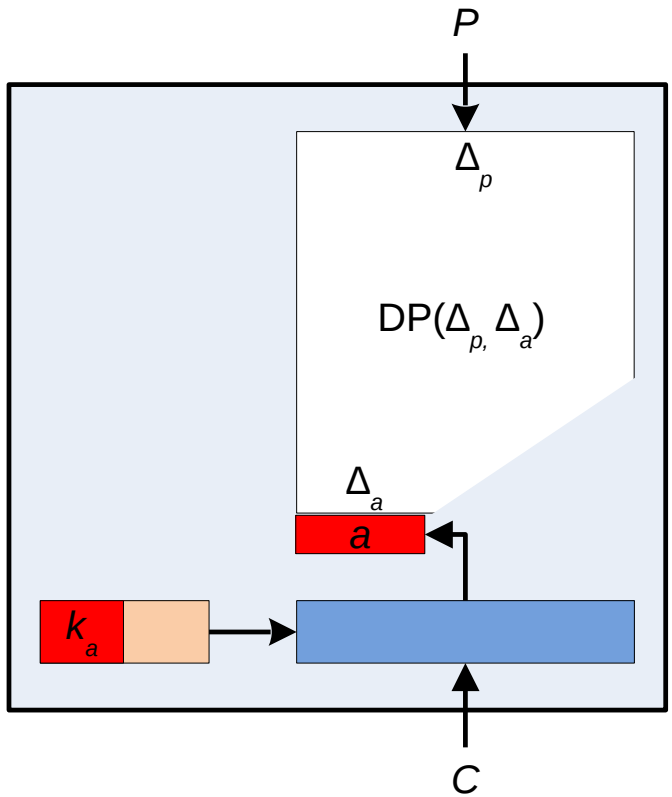
# Statistical attacks

- Exploits *Distinguisher* $\Omega$ over $r-1$ rounds
- Two phases:
  - online: get many $(C_i, P_i)$
  - offline: guess $k_a$
- Wrong guess destroys $\Omega$
- Basic attacks
  - DC: requires $1/DP$ couple
  - LC: requires $1/C^2$ couples
- Many variants ...



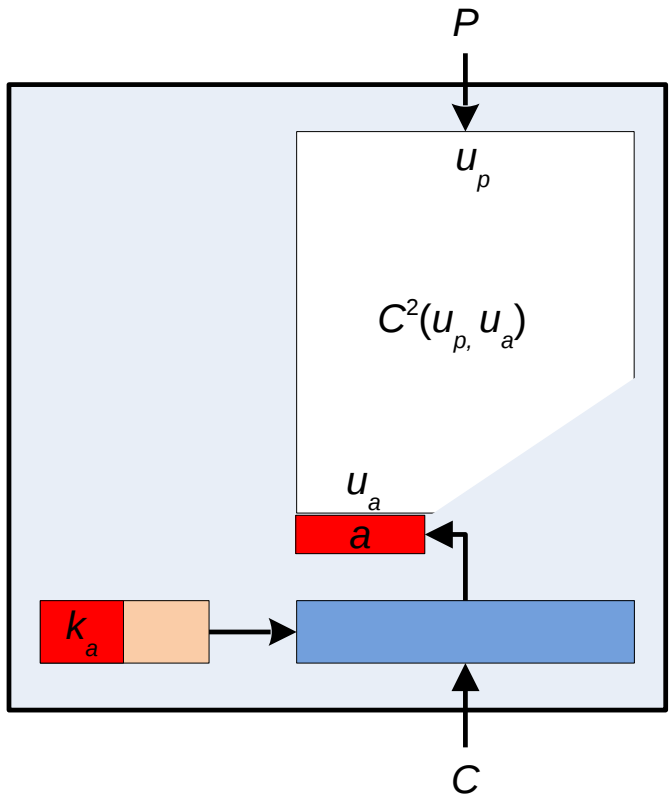
# Statistical attacks

- Exploits *Distinguisher* $\Omega$ over $r-1$ rounds
- Two phases:
  - online: get many $(C_i, P_i)$
  - offline: guess $k_a$
- Wrong guess destroys $\Omega$
- Basic attacks
  - DC: requires $1/DP$ couple
  - LC: requires $1/C^2$ couples
- Many variants ...

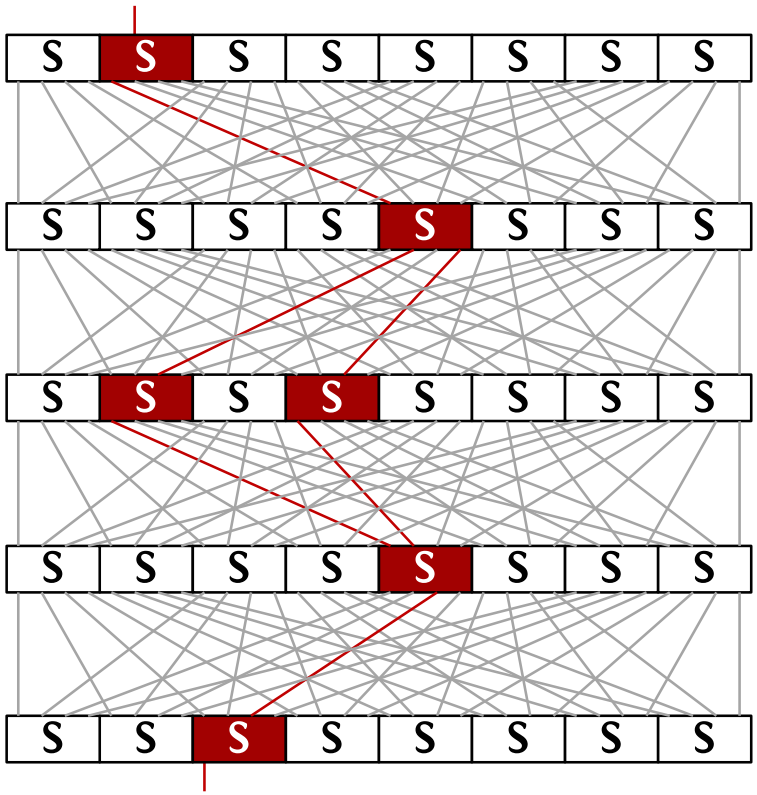


# Statistical attacks

- Exploits *Distinguisher* $\Omega$ over $r-1$ rounds
- Two phases:
  - online: get many $(C_i, P_i)$
  - offline: guess $k_a$
- Wrong guess destroys $\Omega$
- Basic attacks
  - DC: requires $1/DP$ couple
  - LC: requires $1/C^2$ couples
- Many variants ...

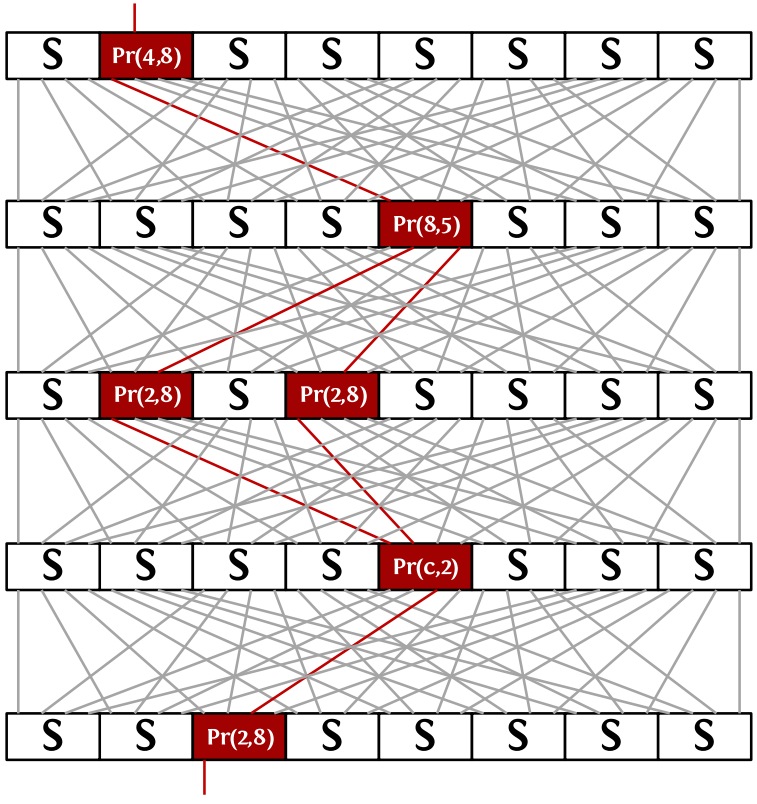


# Distinguisher: difference propagation




- Differential trail: $\mathrm{DP}(Q) \approx \prod_i \mathrm{DP}(\mathrm{Sbox}_i)$ and $w(Q) = \sum_i w(\mathrm{Sbox}_i)$
- Differential: $\mathrm{DP}(\Delta_p, \Delta_a) = \sum_{\Delta_p \to Q \to \Delta_a} \mathrm{DP}(Q)$

# Distinguisher: difference propagation




- Differential trail: $\mathrm{DP}(Q) \approx \prod_i \mathrm{DP}(\mathrm{Sbox}_i)$ and $w(Q) = \sum_i w(\mathrm{Sbox}_i)$
- Differential: $\mathrm{DP}(\Delta_p, \Delta_a) = \sum_{\Delta_p \to Q \to \Delta_a} \mathrm{DP}(Q)$

# Outline

# Boolean function

- Mapping from GF($2^n$) to GF(2)
- Input is a vector $x = (x_1, x_2, \ldots x_n)$
- Algebraic expression:

$$y = x_1x_2 + x_1x_3x_4 + x_2x_4 + 1$$

- Truth table: $2^n$ bit array or *vector*:

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $y$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $y$ |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |

# Correlation between two Boolean functions

$$C(f,g) = 2\Pr\left(f(x) = g(x)\right) - 1$$

Real-valued counterpart of a Boolean function:

$$\hat{f}(x) = (-1)^{f(x)}$$

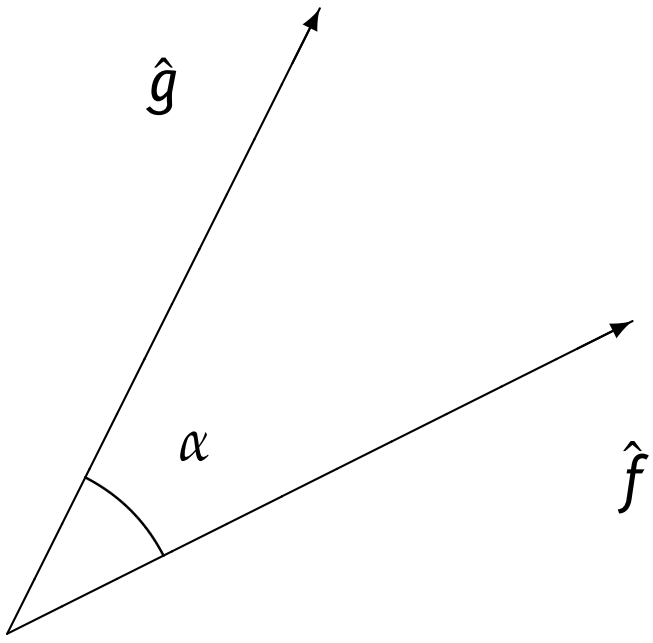
We define an inner product:

$$< \hat{f}, \hat{g} >= \sum_{x} \hat{f}(x)\hat{g}(x)$$

...and norm $||\hat{f}|| = \sqrt{< \hat{f}, \hat{f} >}$
The correlation now becomes

$$C(f,g) = \frac{< \hat{f}, \hat{g} >}{||\hat{f}|| \cdot ||\hat{g}||}$$

# Correlation between Boolean functions geometrically




$$\mathrm{C}(f,g) = \cos\alpha$$

Vector space: $\mathbb{R}^{2^n}$

# Linear functions and selection vectors

- Linear Boolean function with *mask* $w$: $w^T x$
- If $u \neq v$: $< (-1)^{u^T x}, (-1)^{v^T x} >= 0$
- Linear functions form an orthogonal basis of $\mathbb{R}^{2^n}$

| $x$: | $x_0$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ | $x_7$ | $x_8$ | $x_9$ | $x_{10}$ | $x_{11}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $w$: | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |

$w^T x$: $\quad x_1 \quad + \quad x_4 + x_5 \quad + \quad x_8$

# Spectrum of a Boolean function

We can represent $\hat{f}(x)$ with respect to the basis of linear functions:

$$\hat{f}(x) = \sum_{w} F(w)(-1)^{w^{\mathsf{T}}x}$$

with coordinates given by:

$$F(w) = 2^{-n} \sum_{x} \hat{f}(x)(-1)^{w^{\mathsf{T}}x}$$

- This is called the Walsh-Hadamard transform $F(w) = \mathcal{W}(f(x))$
- So simply: $F(w) = C(f(x), w^{\mathsf{T}}x)$
- Orthogonal transformation in $\mathbb{R}^{2^n}$
- Consequence: Parseval's Theorem $\sum F(w)^2 = 1$

# Adding Boolean functions in GF(2)

- Let $h(x) = f(x) + g(x)$
  - From $\hat{h}(x) = \hat{f}(x)\hat{g}(x)$ follows $H(w) = \sum_v F(v+w)G(v)$
- Spectrum of sum equals convolution of spectra
- Special cases:
  - Constant function: $g(x) = 1 : H(w) = -F(w)$
  - Linear function: $g(x) = u^{\mathsf{T}}x : H(w) = F(w+u)$
  - Disjunct functions $f$ and $g$ : $H(v+w) = F(v)G(w)$

# Multiplying Boolean functions in GF(2)

Let $h(x) = f(x)g(x)$. Then:

$$\hat{h}(x) = \frac{1}{2}\left(1 + \hat{f}(x) + \hat{g}(x) - \hat{f}(x)\hat{g}(x)\right)$$

From this it follows

$$\mathcal{W}(fg) = \frac{1}{2}\left(\delta(w) + \mathcal{W}(f) + \mathcal{W}(g) + \mathcal{W}(f+g)\right)$$

with $\delta(w) = 1$ iff $w = 0$

# Correlation matrices [Daemen 1994]

- $m$-bit vector Boolean function: $h(x) = (h_1(x), h_2(x) \ldots h_m(x))$
- Correlation matrix $C^h$:
  - $2^m$ rows and $2^n$ columns
  - element at row $u$, column $v$: $C(u^\mathsf{T} h(x), v^\mathsf{T} x)$
- Homomorphism:

$$\begin{array}{ccc} x & \xrightarrow{\quad h \quad} & y = h(x) \\ \Updownarrow \mathcal{L} & & \Updownarrow \mathcal{L} \\ X \text{ with } X_u = (-1)^{x^\mathsf{T} u} & \xrightarrow{\quad C^{(h)} \quad} & Y = C^{(h)} X \end{array}$$

- If $h$ is permutation: $C^{(h^{-1})} = \left(C^{(h)}\right)^{-1} = \left(C^{(h)}\right)^\mathsf{T}$

# Correlation matrices of special functions

- Adding a constant: $f(x) = x + k$

$$C_{u,u} = (-1)^{u^{\mathsf{T}}k} \text{ and } C_{u,v\neq u} = 0$$

- Linear function: $f(x) = Mx$

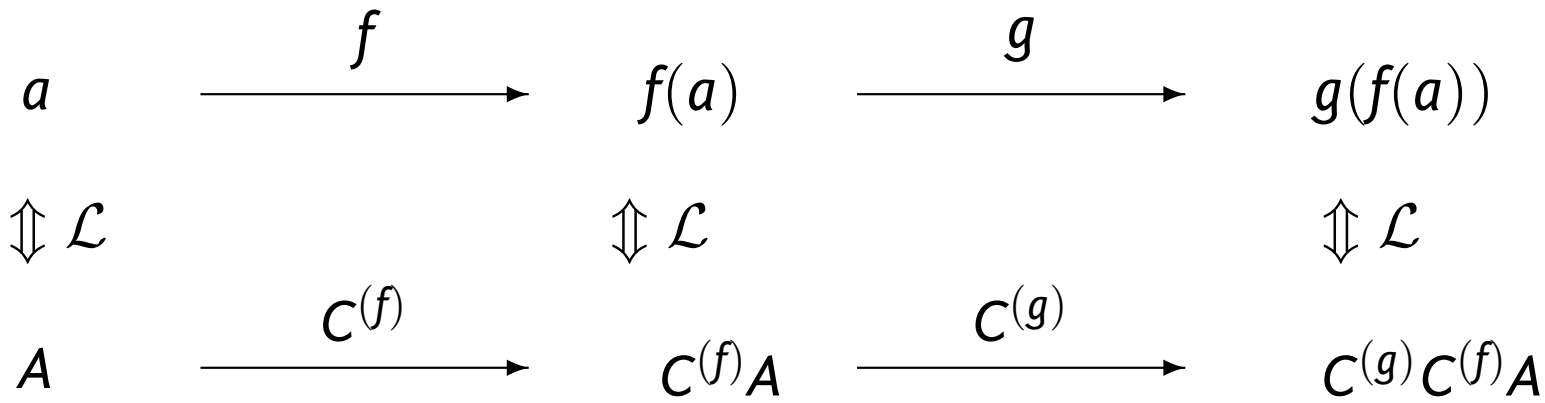
$$C_{u,w} = 1 \text{ iff } M^{\mathsf{T}}u = w \text{ and } 0 \text{ otherwise}$$

- Parallel composition: $b = (b_1, b_2, \ldots) = (h_1(a_1), h_2(a_2), \ldots) = h(a)$

$$C^{(h)}_{u,w} = \prod_i C^{(h_i)}_{u_{(i)},w_{(i)}}$$

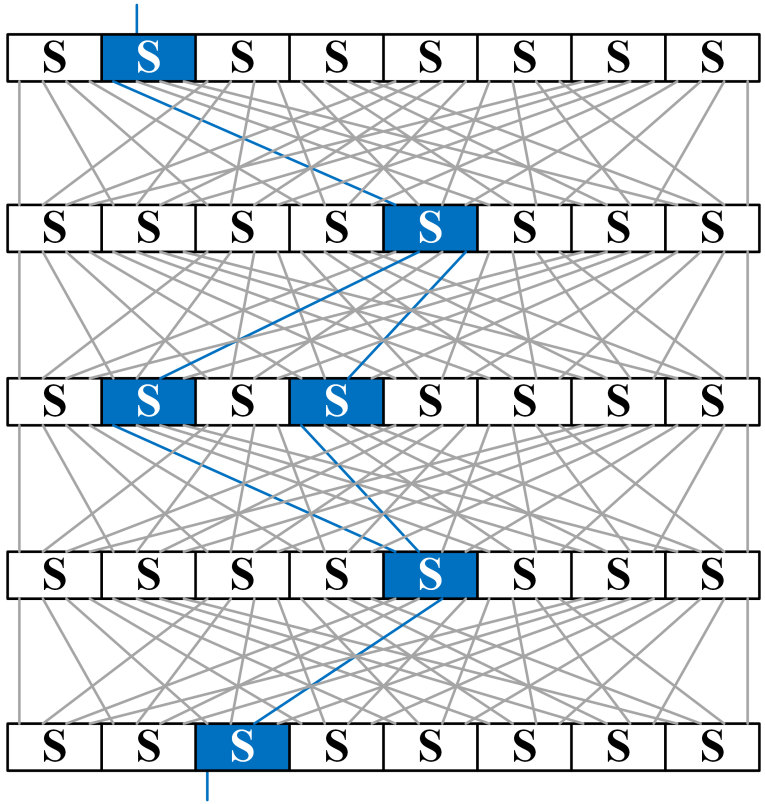
  - If $w_i = 0$ then $C^{(h_i)}_{u_{(i)},w_{(i)}} = 1$
  - $C^{(h)}_{u,w}$ is product of correlation over active S-boxes

# Correlation matrices: serial composition




$$C^{(g\circ f)}(u,v) = \sum_w C^{(g)}(u,w)C^{(f)}(w,v)$$

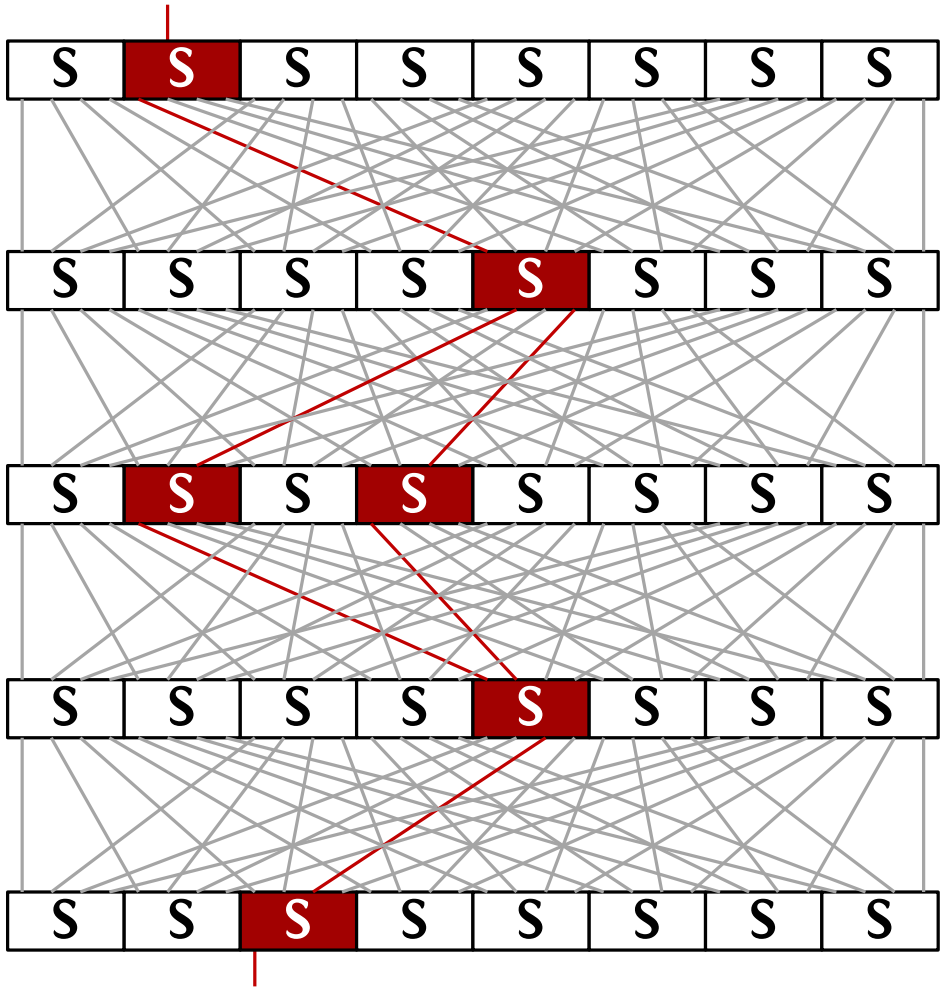
# Linear trails and correlation



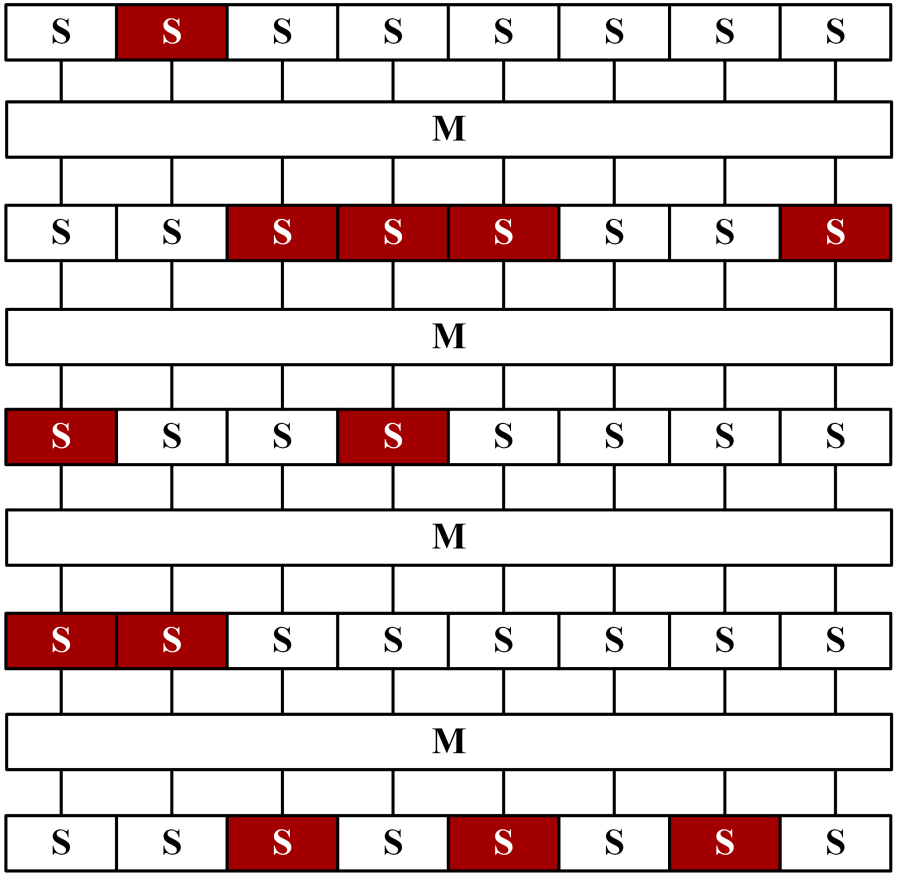

- Linear trail: $\mathrm{C_p}(Q) = \prod_i C(\mathrm{Sbox}_i)$
- Correlation: $C(u^\mathsf{T}\beta(a), w^\mathsf{T}a) = \sum_{w \to Q \to u} \mathrm{C_p}(Q)$

# Outline

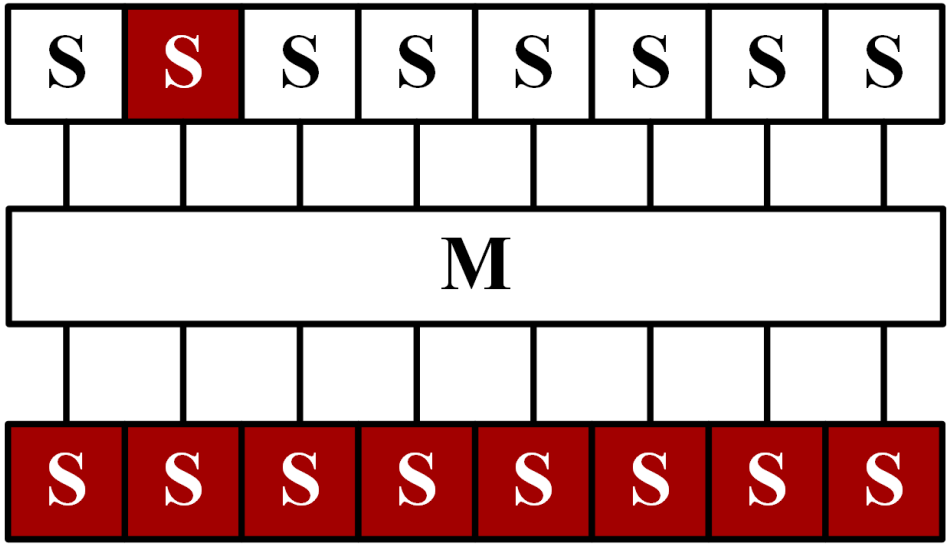
# Replacing the permutation in SPN by a mixing layer



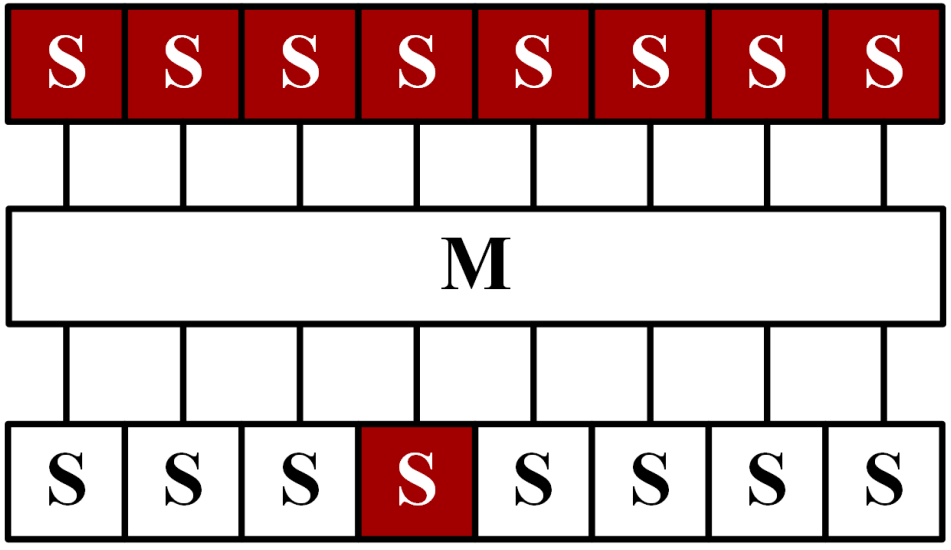
# Replacing the permutation in SPN by a mixing layer



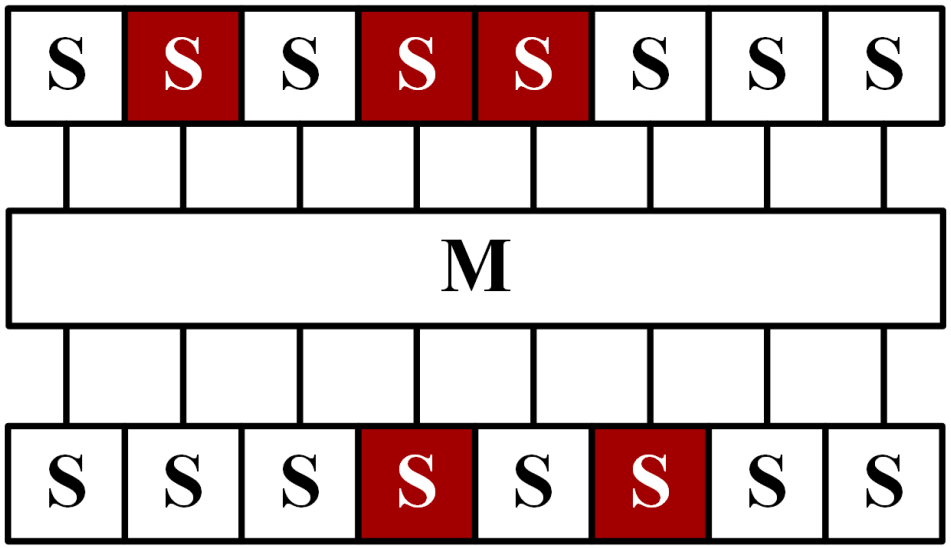
# Mixing layer criterion: Branch number $\mathcal{B}$



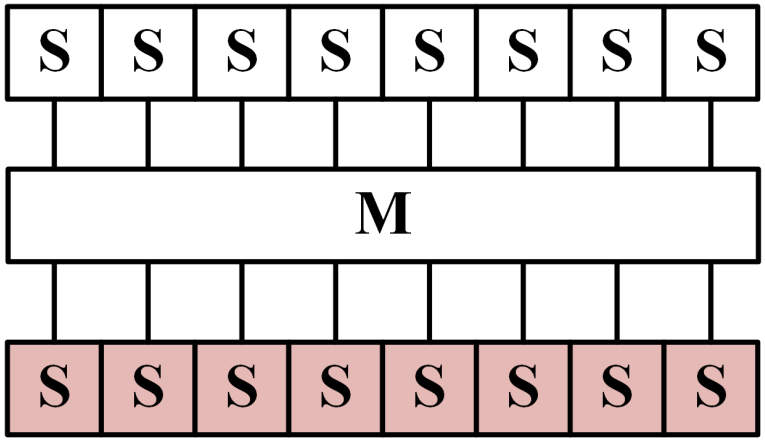
# Mixing layer criterion: Branch number $\mathcal{B}$

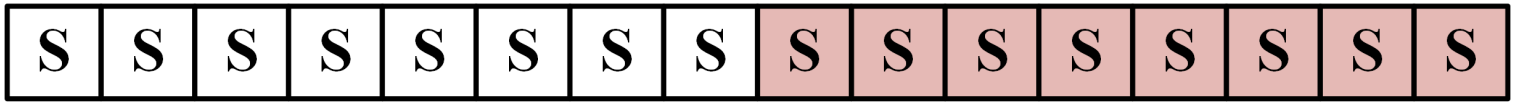


# Mixing layer criterion: Branch number $\mathcal{B}$

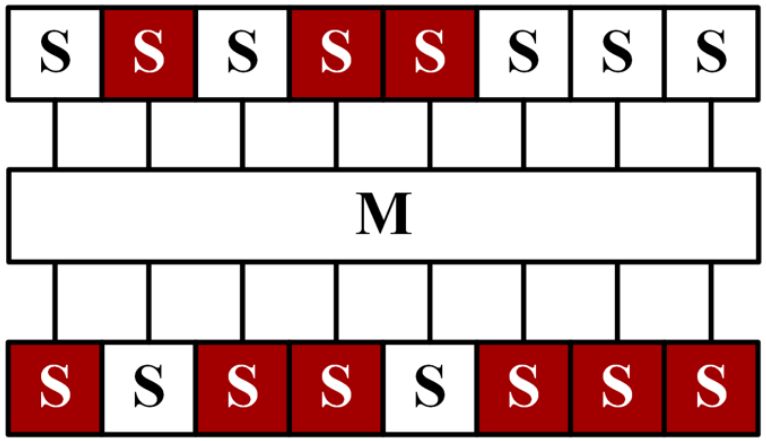


# Mixing layer and error-correcting codes

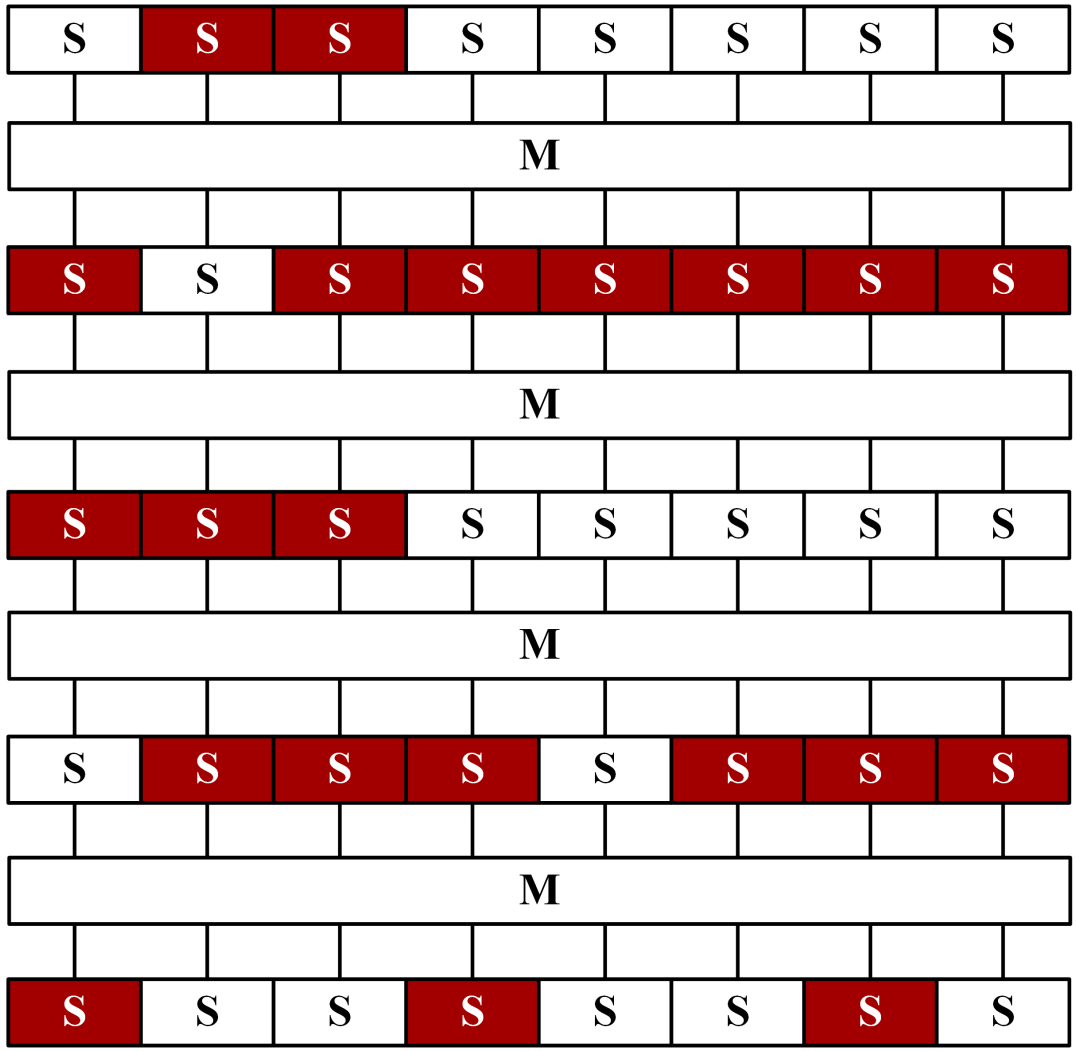


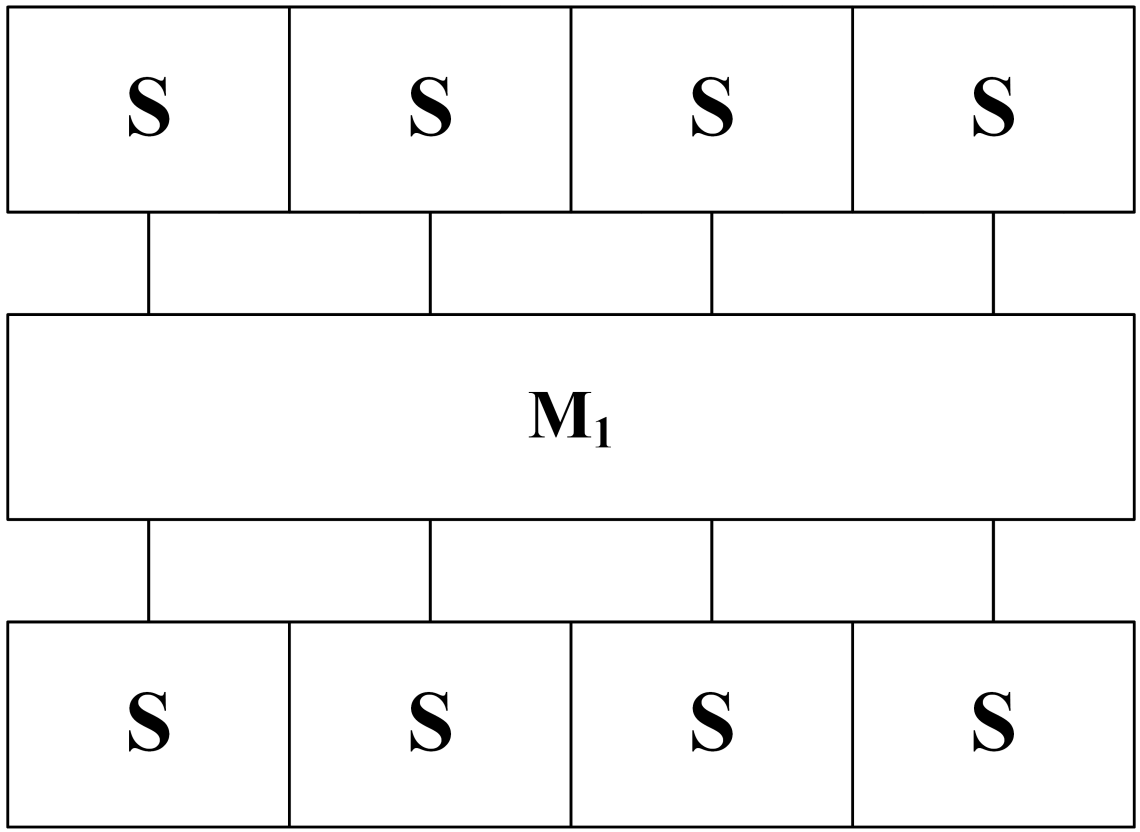
# Mixing layer and error-correcting codes




# $\mathcal{B}$ active S-boxes in each sequence of 2 rounds

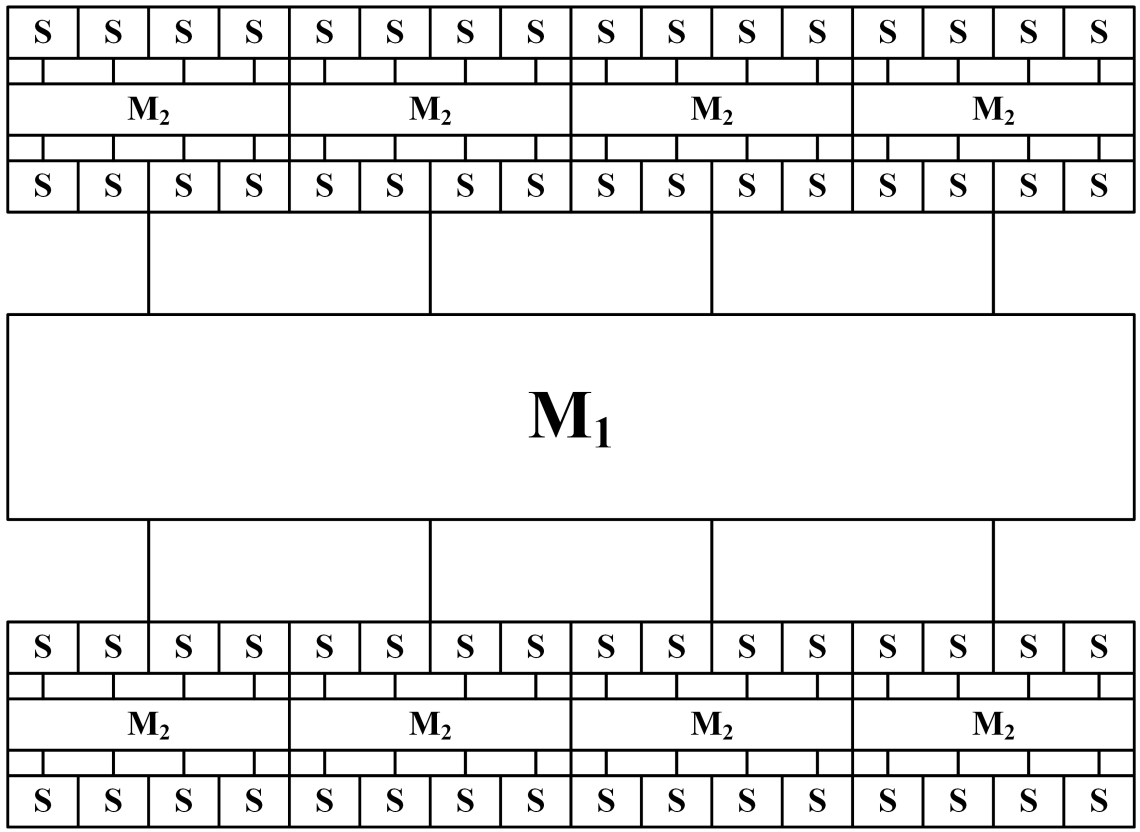


# Recursion: four-round theorem



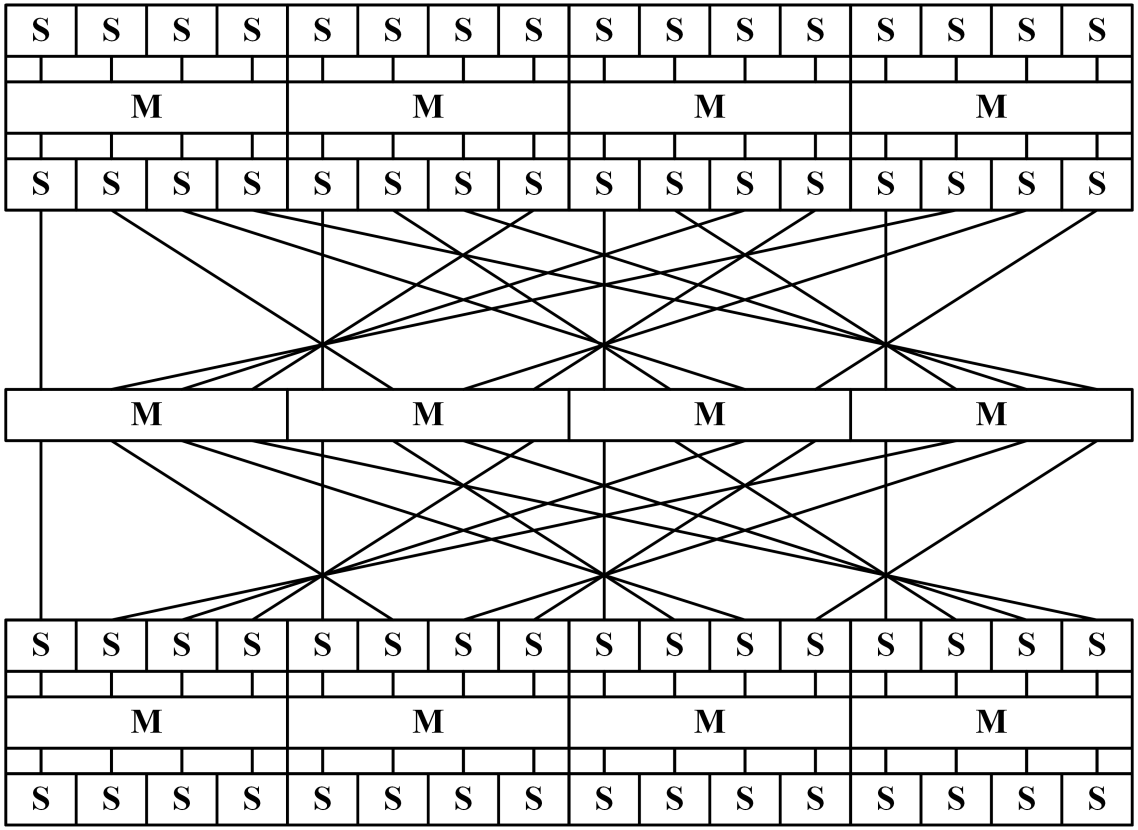

$\mathcal{B}_1 \times \mathcal{B}_2$ active S-boxes per 4 rounds

# Recursion: four-round theorem



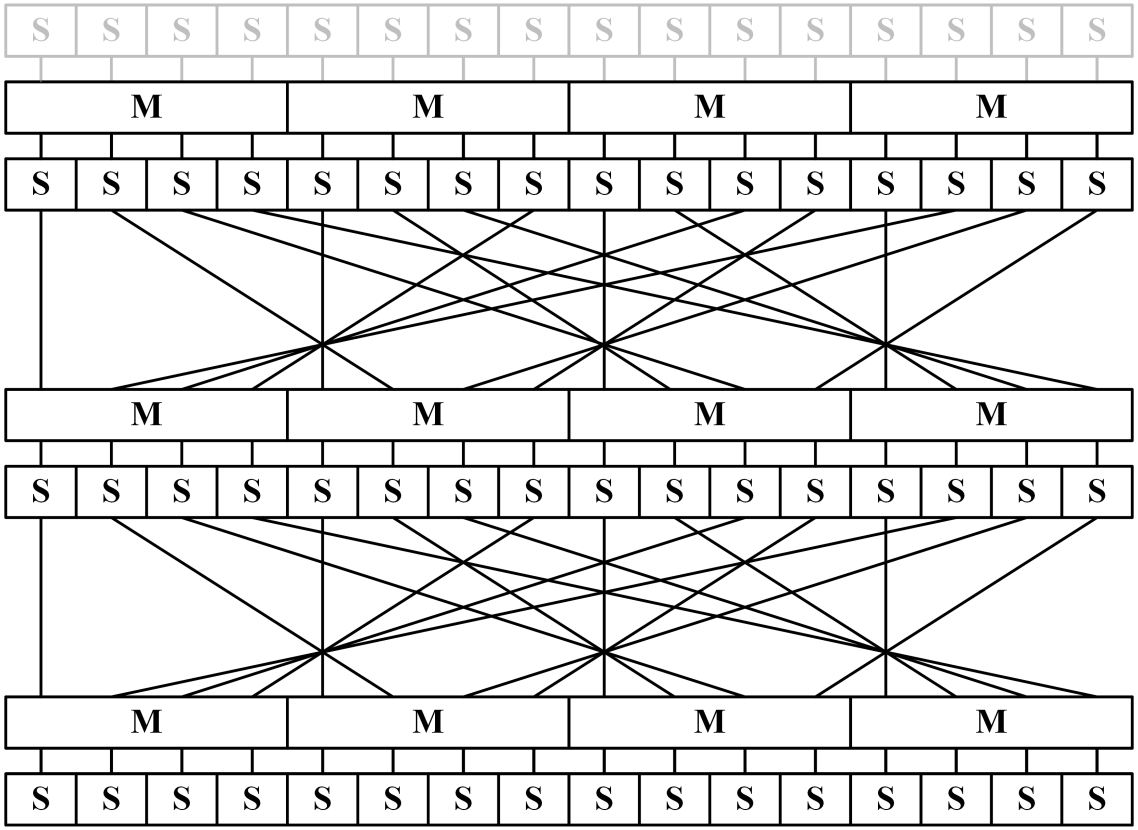

$\mathcal{B}_1 \times \mathcal{B}_2$ active S-boxes per 4 rounds

# Recursion: four-round theorem



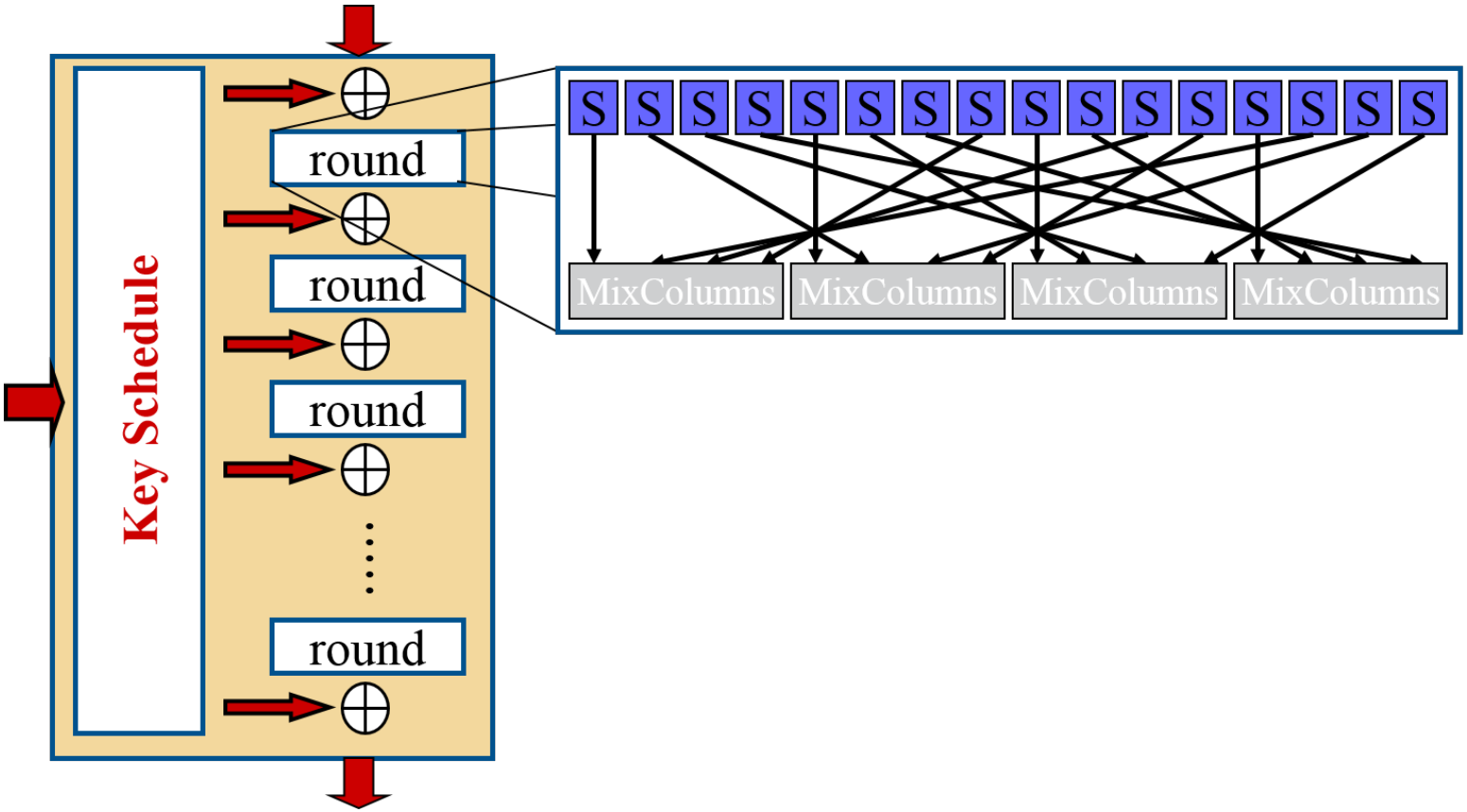

$\mathcal{B}_1 \times \mathcal{B}_2$ active S-boxes per 4 rounds

# Recursion: four-round theorem




$\mathcal{B}_1 \times \mathcal{B}_2$ active S-boxes per 4 rounds

# Rijndael [Daemen, Rijmen 1998]




- Trails: 25 active S-boxes per 4 rounds
- Clustering of trails but not alarming
- Costly S-box and mixing
- Byte-alignment leads to structural properties

# Outline

# Some years earlier: 3-Way and BaseKing [Daemen 1993-1994]

- Only bitwise instructions and shifts
- 4-layer round function alternated with key addition

  $\theta$ mixing
  $\pi_1$ transposition 1: shifts of words
  $\gamma$ non-linear
  $\pi_2$ transposition 2: shifts of words
- Additional $\theta$ at the end
- Round key $=$ cipher key $\oplus$ round constant
- Cipher and inverse same, mod round constants and word order
- 96-bit (3-Way) and 192-bit (BaseKing) ciphers

# The $\gamma$ S-box

| $x$ | 000 | 001 | 010 | 100 | 110 | 101 | 011 | 111 |
|---|---|---|---|---|---|---|---|---|
| $y$ | 111 | 010 | 100 | 001 | 011 | 110 | 101 | 000 |

- $\chi$ of KECCAK, complemented: $y_i = x_i + 1 + (x_{i+1} + 1)x_{i+2}$
- Differentially uniform: all differentials have probability $1/4$
- Uniform correlation: all correlations have amplitude $1/2$
- Positions of non-zero correlations and differentials coincide

# The mixing layer $\theta$: operates on 12-bit slices

$$M_\theta = \begin{pmatrix} 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 & 1 \\ 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 \\ 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 \\ 1 & 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot \\ \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 \\ 1 & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & 1 & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & 1 & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & \cdot & \cdot \\ \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 & 1 & 1 & \cdot & 1 \\ 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 & 1 & 1 & \cdot \\ \cdot & 1 & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & 1 & 1 & 1 \end{pmatrix}$$

Orthogonal: $M_\theta^{-1} = M_\theta^T$, so differences and masks propagate same way

# Diffusion properties of $\theta$

| $\lvert y\rvert \setminus \lvert x\rvert$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | - | - | - | - | - | - | 12 | - | - | - | - |
| 2 | - | - | - | - | - | 60 | - | - | - | 6 | - |
| 3 | - | - | - | - | 180 | - | - | - | 40 | - | - |
| 4 | - | - | - | 255 | - | - | - | 240 | - | - | - |
| 5 | - | - | 180 | - | - | - | 600 | - | - | - | 12 |
| 6 | - | 60 | - | - | - | 804 | - | - | - | 60 | - |
| 7 | 12 | - | - | - | 600 | - | - | - | 180 | - | - |
| 8 | - | - | - | 240 | - | - | - | 255 | - | - | - |
| 9 | - | - | 40 | - | - | - | 180 | - | - | - | - |
| 10 | - | 6 | - | - | - | 60 | - | - | - | - | - |
| 11 | - | - | - | - | 12 | - | - | - | - | - | - |

- (Hamming weight) branch number $\mathcal{B} = 8$
- implies a $[24, 12, 8]$ code: the binary extended Golay code

# Resulting block ciphers

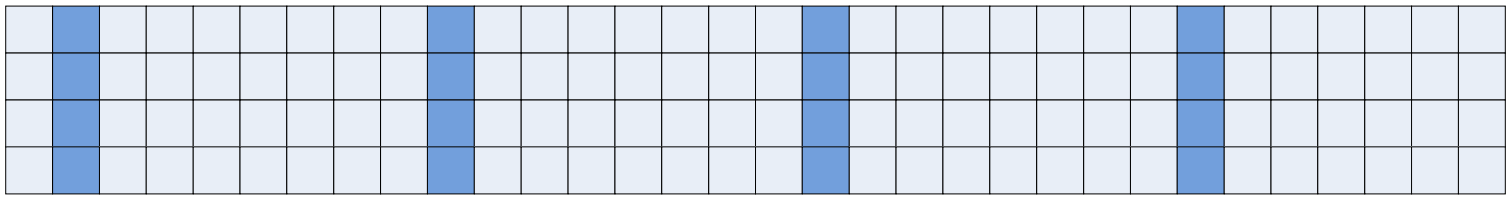
- Two instances:
  - 3-Way: 96-bit block and key
  - BaseKing: 192-bit block and key
- Symmetry
  - equivalence of differential and linear trails
  - propagation $\leftarrow$ same als $\rightarrow$ with order of bits permuted
- Implementation
  - small number of operations per bit
  - same circuit for cipher and inverse
  - suitable for bit-slice

# NOEKEON [Daemen, Peeters, Rijmen and Van Assche, 2000]

- Block cipher
  - 128-bit blocks
  - 128-bit keys
  - security claim: PRP $2^{-128}\mu N$
- *Porting of* 3-WAY *to 128 bits*

See http://gro.noekeon.org/

# The Noekeon state



- Two-dimensional $4 \times \ell$ array
  - 4 *rows*
  - $\ell$ *columns*
- Additional partitioning of the state: *slices*
  - $\ell/4$ slices
- $\ell = 32$

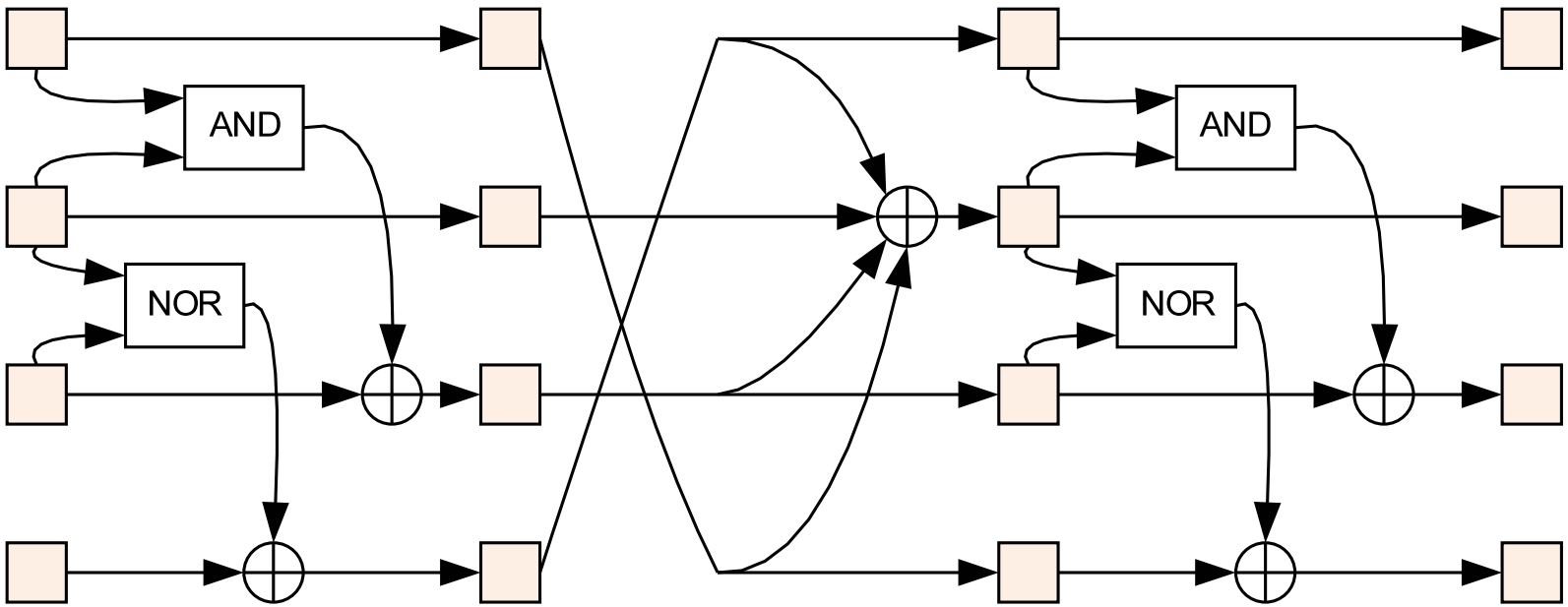
# Round transformation

- $\gamma$: nonlinear layer
  - 4-bit S-box operating on columns
  - Involution
- $\theta$: combines mixing layer and round key addition
  - Linear 16-bit mixing layer operating on slices
  - Involution
- $\pi$: dispersion between slices
  - Rotation of bits within $\ell$-bit rows
  - Two instances that are each others inverse
- $\iota$: round constant addition for asymmetry

# The round and its inverse

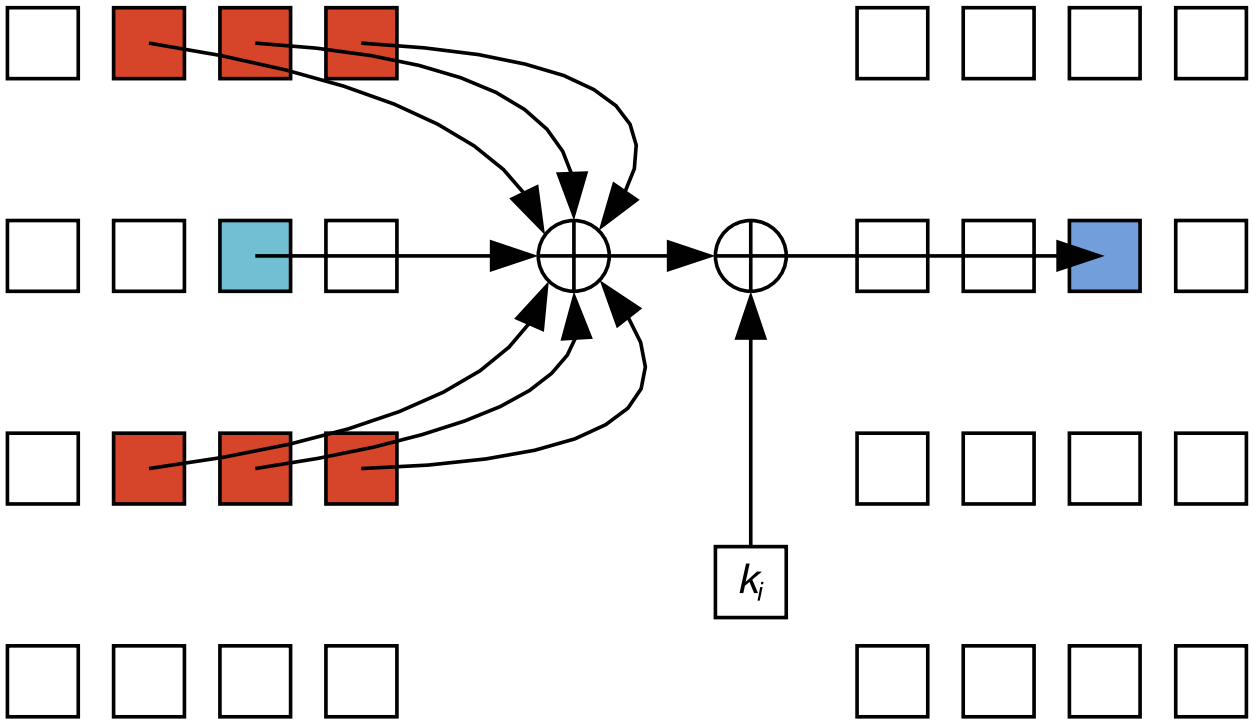
- Round: $\pi_2 \circ \gamma \circ \pi_1 \circ \theta[k]$
- Inverse round:
  - $\theta[k]^{-1} \circ \pi_1^{-1} \circ \gamma^{-1} \circ \pi_2^{-1}$
  - $\theta[k] \circ \pi_2 \circ \gamma \circ \pi_1$
- $\theta[k]$ as final transformation:
  - Regrouping: round of inverse cipher $=$ cipher round
  - round constants prevent involution
- NOEKEON: 16 rounds and a final transformation
  - Inverse cipher equal to cipher itself
  - Asymmetry provided by round constants only

# Nonlinear layer $\gamma$




Two identical nonlinear steps with a linear step in between

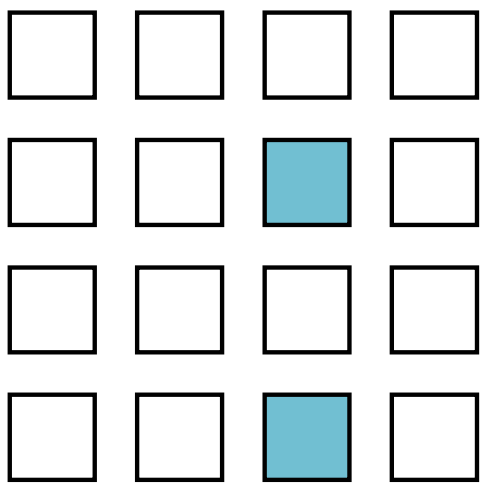
# Mixing layer $\theta$



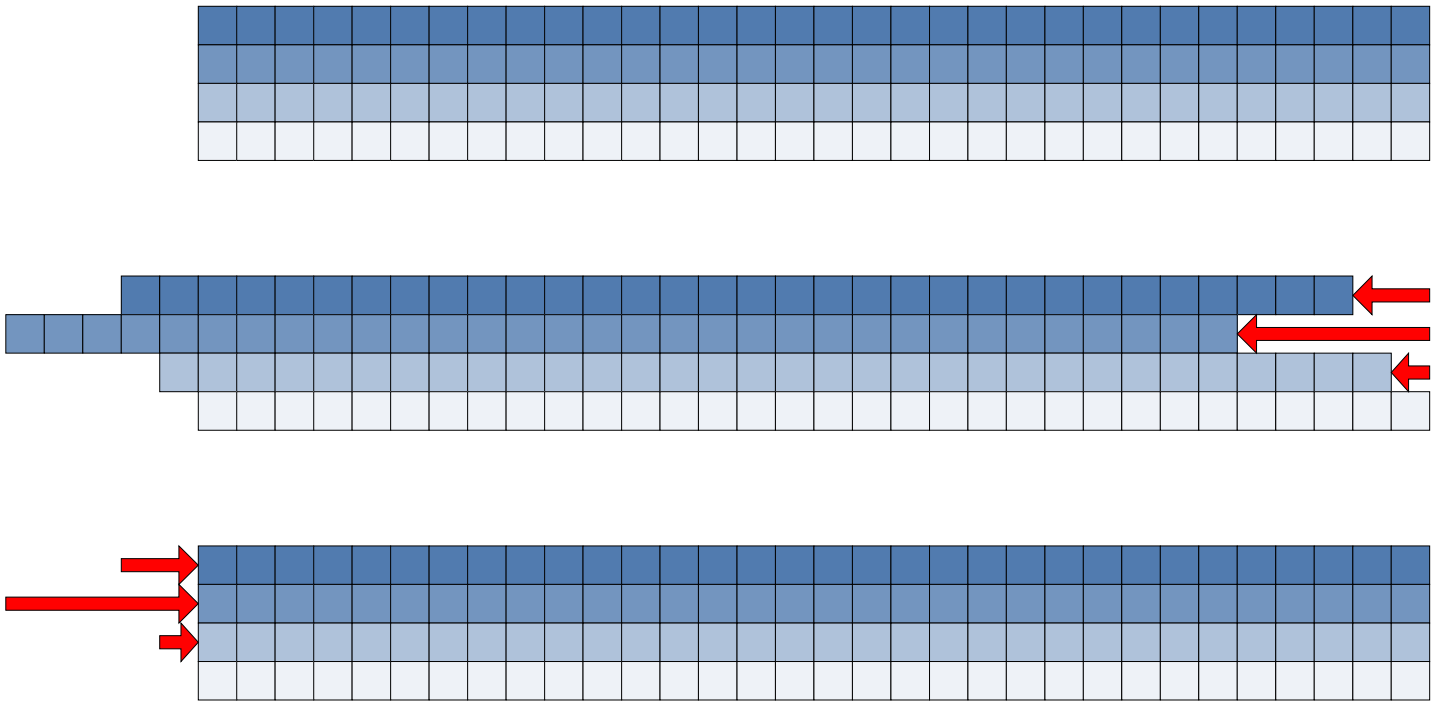

High average diffusion and low cost

# Mixing layer $\theta$ cont'd

- Branch number $\mathcal{B}$ only 4 due to symmetry
- Invariant sparse states in kernel, e.g.:



# Transposition steps $\pi$



- $\pi_1$ and $\pi_2$ are each others inverses

# Trail bounds

- Bounds on 4-round trails
  - Differential trails: probability $\leq 2^{-48}$
  - Linear trails: correlation squared $\leq 2^{-48}$
- *rounds over more than 11 rounds are unusable*
- Powerful bounds thanks to
  - High average diffusion in $\theta$ and $\pi$
  - Kernel addressed in $\gamma$ S-box
- Determining bounds:
  - Non-trivial exercise but one-time effort
  - See http://gro.noekeon.org/Noekeon-spec.pdf

# Lightweight aspect

- Round function: 5 XOR, 1 AND/OR per bit
  - Compare to AES: 16 XOR, 5 AND per bit
- Hardware
  - # gates: $[640 - 1050]$ XOR, 64 AND, 64 NOR, 128 MUX
  - Gate delay: 7 XOR, 1 AND, 1 MUX
  - Coprocessor architecture: speed/area trade-off
- Software: e.g. numbers for ARM7:
  - code size 332 bytes, 44.5 cycles/byte
  - code size 3688 bytes, 30 cycles/byte
  - RAM usage: everything in registers
- Cipher and inverse are equal: re-use of circuit and code

# Outline

# Conclusions

- Wide trail strategy is a way to design round functions
- Strong alignment
  - simple proofs for trail weights
  - other distinguishers more likely
- Weak alignment
  - proofs for trail weights require computer assistance
  - other distinguishers less likely