LABORATOIRE HUBERT CURIEN
UMR • CNRS • 5516 • SAINT-ETIENNE

LOW-AREA IMPLEMENTATION OF PHOTON-BEETLE

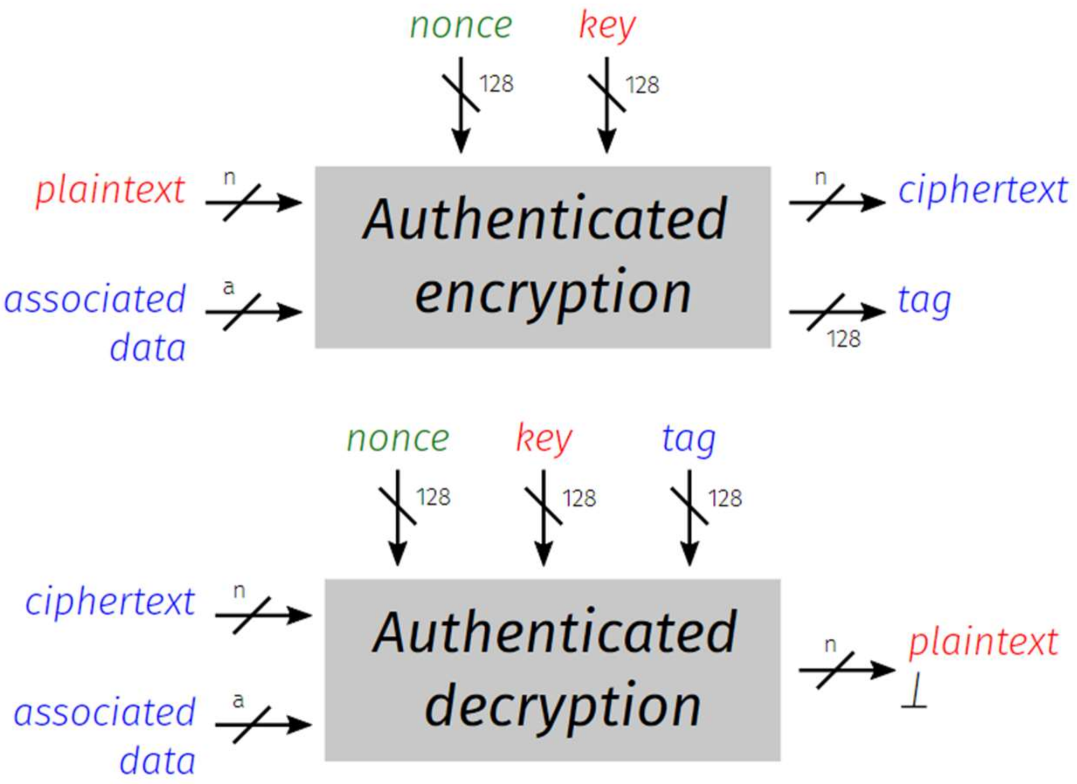
Cryptarchi 2022

Pierre-Antoine TISSOT, Carlos-Andres LARA-NINO
Laboratoire Hubert Curien UMR5516, F-42023 St-Etienne, France

Photon-Beetle – Context

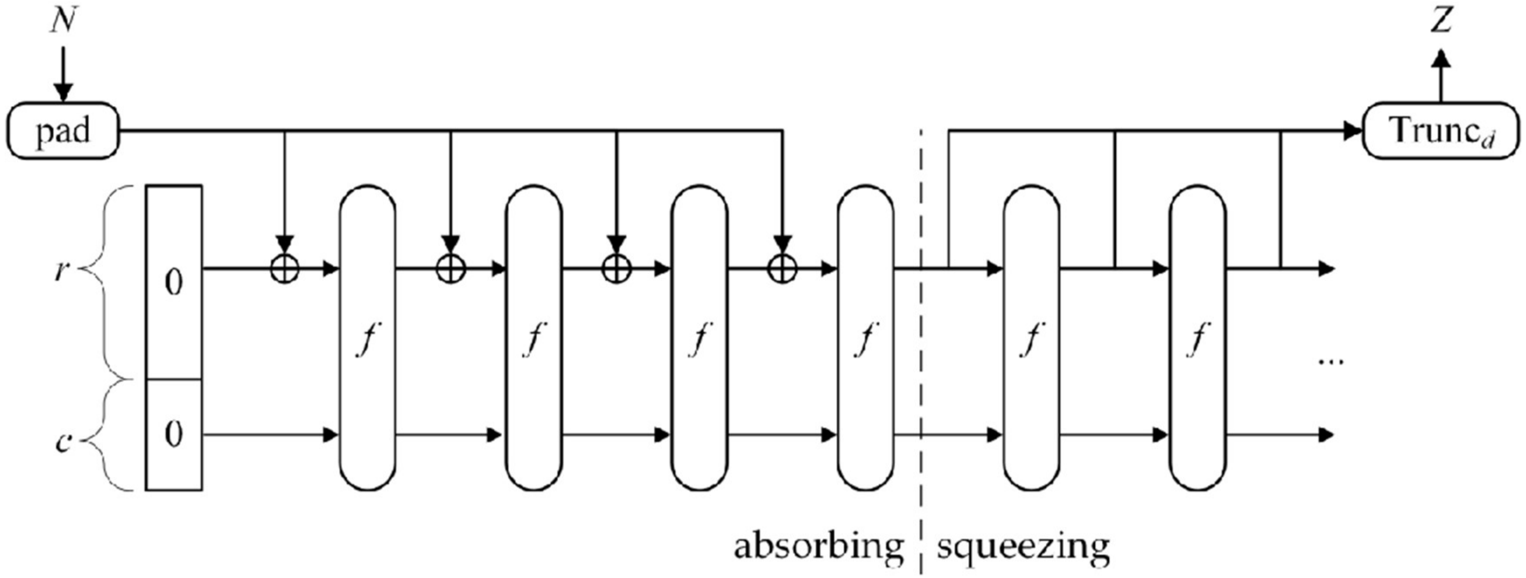
- 2018 NIST Lightweight Cipher Finalist
- Authenticated encryption and hash family
  - *Sponge-based mode Beetle*
  - *PHOTON Hash permutation*
- Hardware implementation
- Robustness against Side-Channel Attacks

Authenticated ciphers with Associated Data

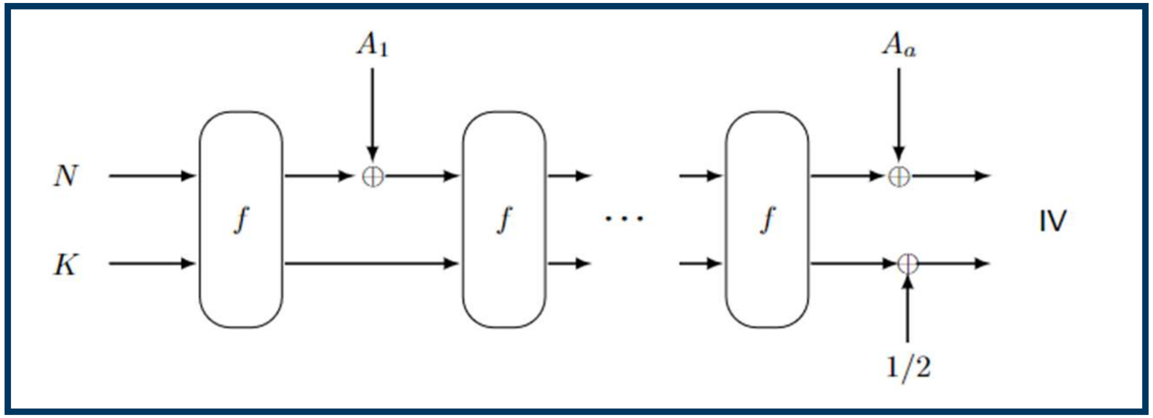


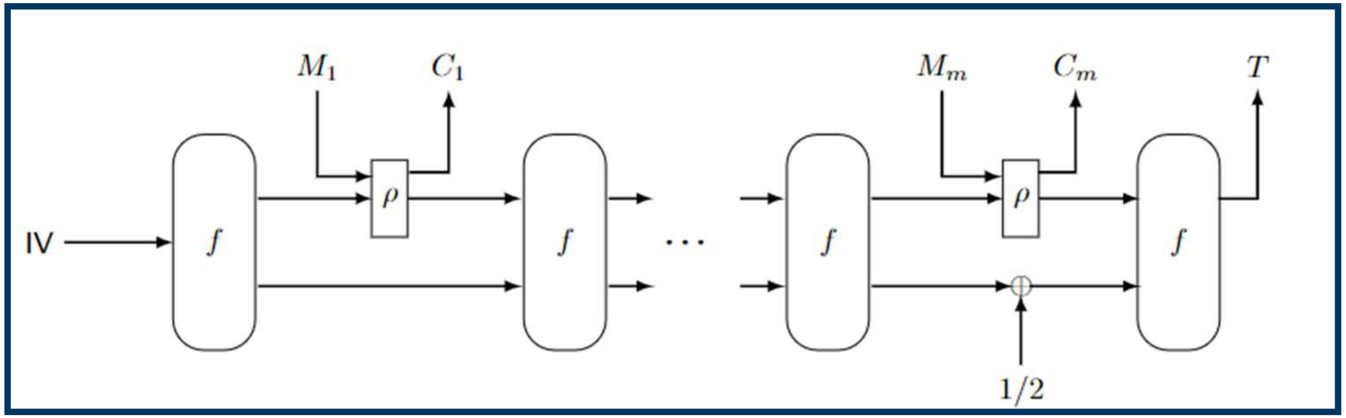
The information can be secret, transmitted, or public.

Sponge Construction



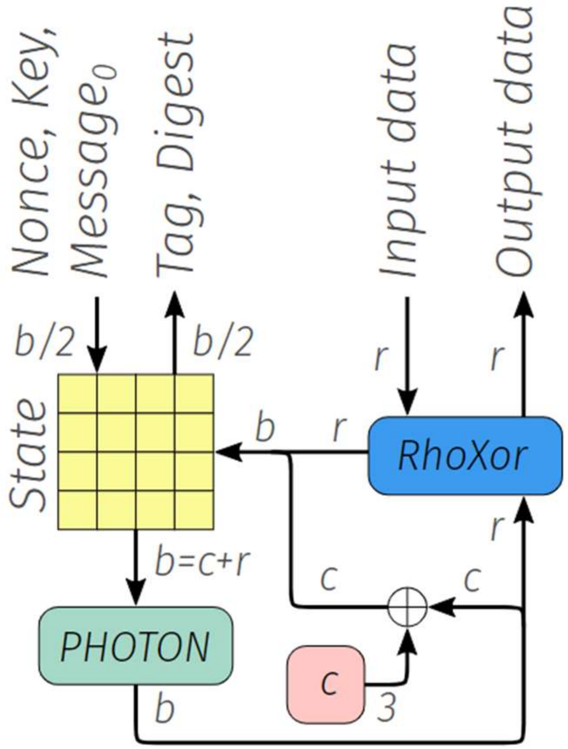
PHOTON-Beetle-AEAD



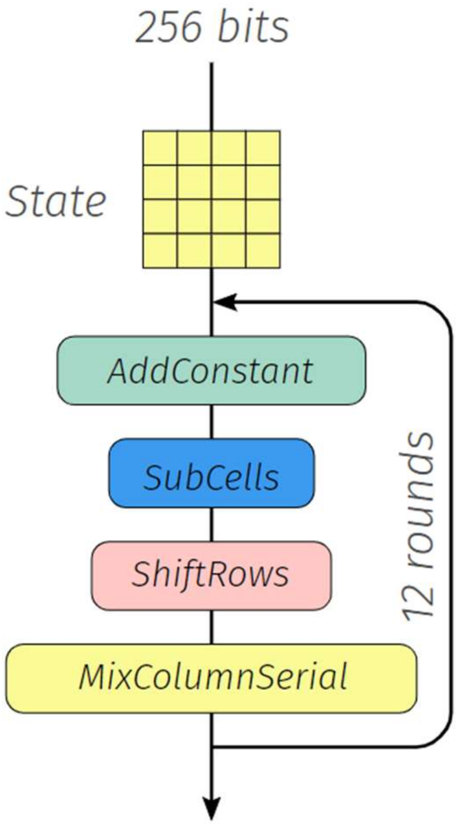


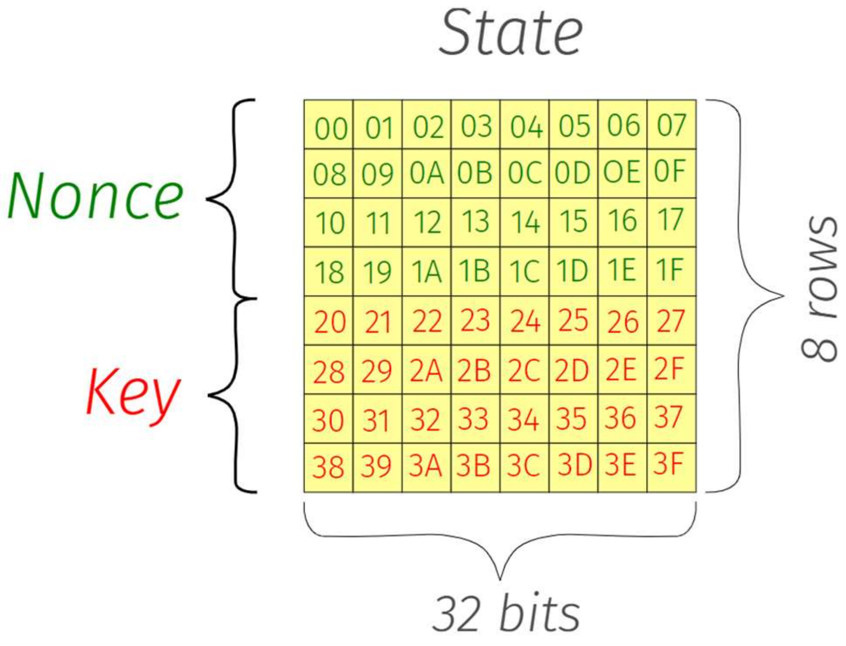
PHOTON-Beetle-AEAD + Hash

- Parameters
  - *NONCE : 128 bits*
  - *KEY : 128 bits*
  - *State : 256 bits*
  - *Rate*
    - AEAD : 32 bits, 128 bits
    - Hash : 32 bits
  - *Capacity : State – Rate*
  - *Tag : 128 bits*
  - *Hash : 256 bits*
- Rate of 32 bits selected to create a unified architecture

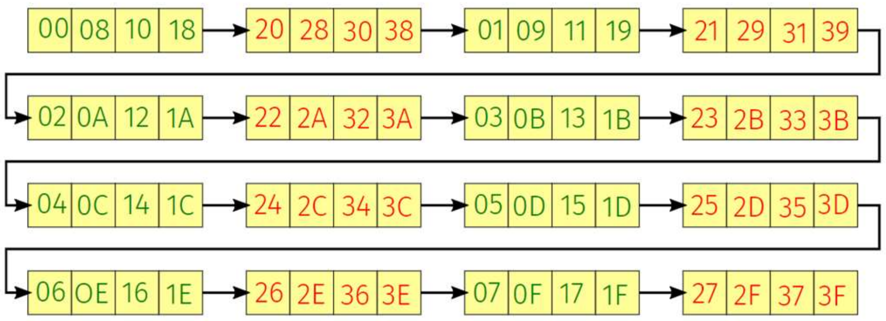


PHOTON-256





00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F



00 08 10 18 → 20 28 30 38 → 01 09 11 19 → 21 29 31 39

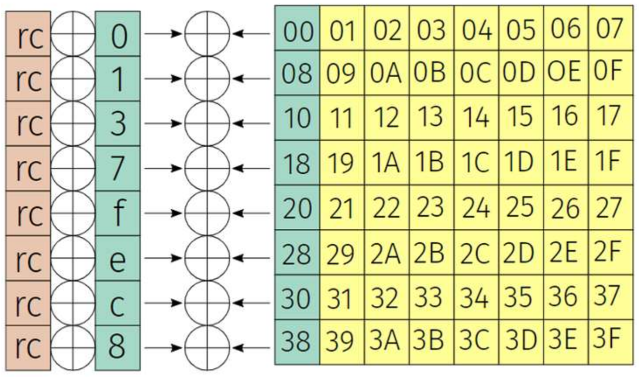
02 0A 12 1A → 22 2A 32 3A → 03 0B 13 1B → 23 2B 33 3B

04 0C 14 1C → 24 2C 34 3C → 05 0D 15 1D → 25 2D 35 3D

06 0E 16 1E → 26 2E 36 3E → 07 0F 17 1F → 27 2F 37 3F

P_{256} Round Functions (1)

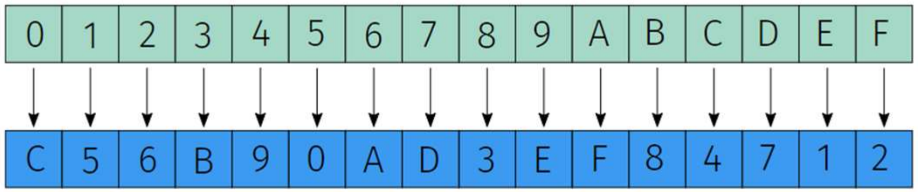
AddConstant



rc	⊕	0	→ ⊕ ←	00	01	02	03	04	05	06	07
rc	⊕	1	→ ⊕ ←	08	09	0A	0B	0C	0D	0E	0F
rc	⊕	3	→ ⊕ ←	10	11	12	13	14	15	16	17
rc	⊕	7	→ ⊕ ←	18	19	1A	1B	1C	1D	1E	1F
rc	⊕	f	→ ⊕ ←	20	21	22	23	24	25	26	27
rc	⊕	e	→ ⊕ ←	28	29	2A	2B	2C	2D	2E	2F
rc	⊕	c	→ ⊕ ←	30	31	32	33	34	35	36	37
rc	⊕	8	→ ⊕ ←	38	39	3A	3B	3C	3D	3E	3F

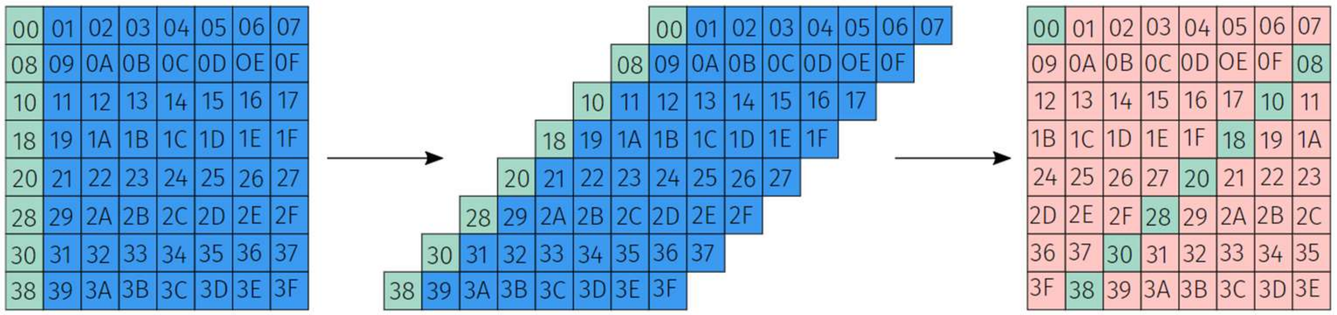
r	0	1	2	3	4	5	6	7	8	9	10	11
rc	1	3	7	e	d	b	6	c	9	2	5	a

Substitution Box



0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

ShiftRows



00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F

→

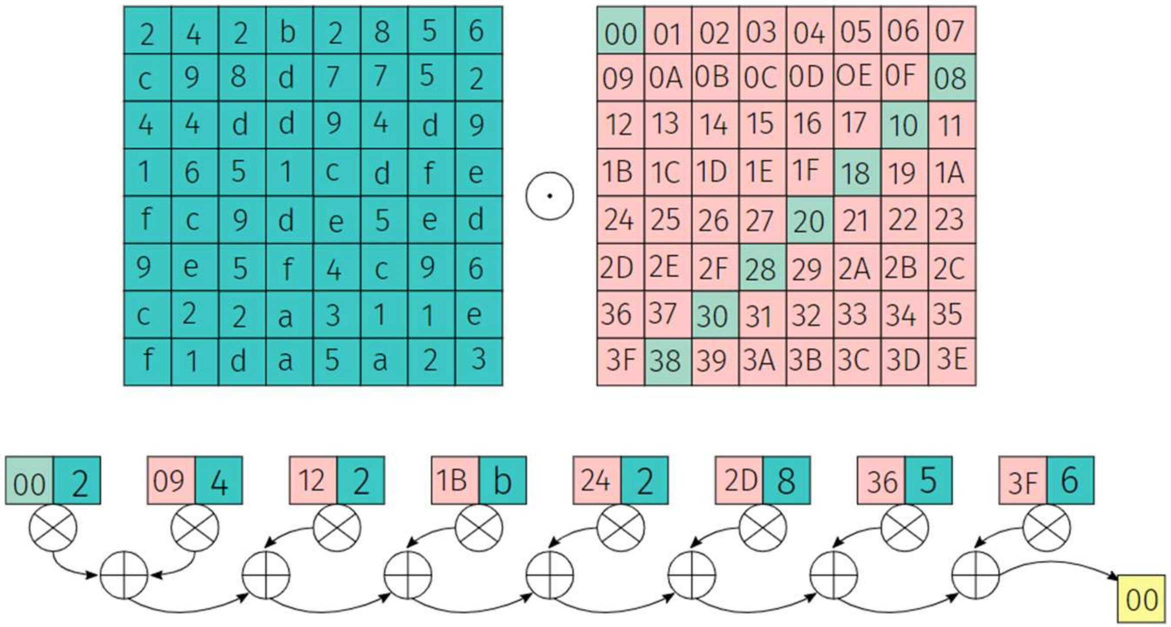
							00	01	02	03	04	05	06	07
						08	09	0A	0B	0C	0D	0E	0F	
					10	11	12	13	14	15	16	17		
				18	19	1A	1B	1C	1D	1E	1F			
			20	21	22	23	24	25	26	27				
		28	29	2A	2B	2C	2D	2E	2F					
	30	31	32	33	34	35	36	37						
38	39	3A	3B	3C	3D	3E	3F							

→

00	01	02	03	04	05	06	07
09	0A	0B	0C	0D	0E	0F	08
12	13	14	15	16	17	10	11
1B	1C	1D	1E	1F	18	19	1A
24	25	26	27	20	21	22	23
2D	2E	2F	28	29	2A	2B	2C
36	37	30	31	32	33	34	35
3F	38	39	3A	3B	3C	3D	3E

P_{256} Round Functions (2)

MixColumnSerial



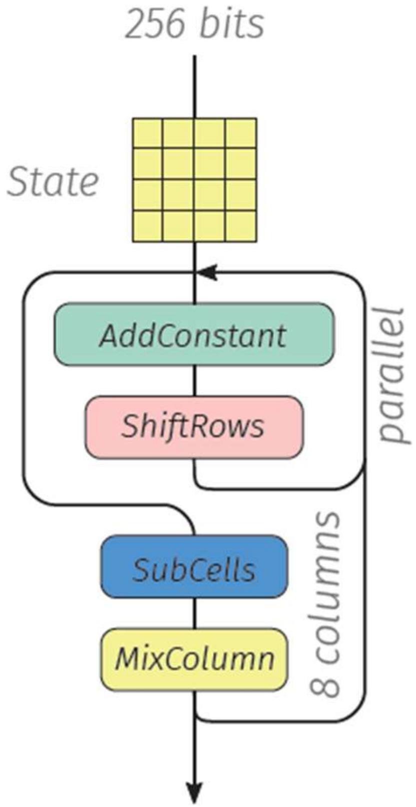
2	4	2	b	2	8	5	6
c	9	8	d	7	7	5	2
4	4	d	d	9	4	d	9
1	6	5	1	c	d	f	e
f	c	9	d	e	5	e	d
9	e	5	f	4	c	9	6
c	2	2	a	3	1	1	e
f	1	d	a	5	a	2	3

$\odot$

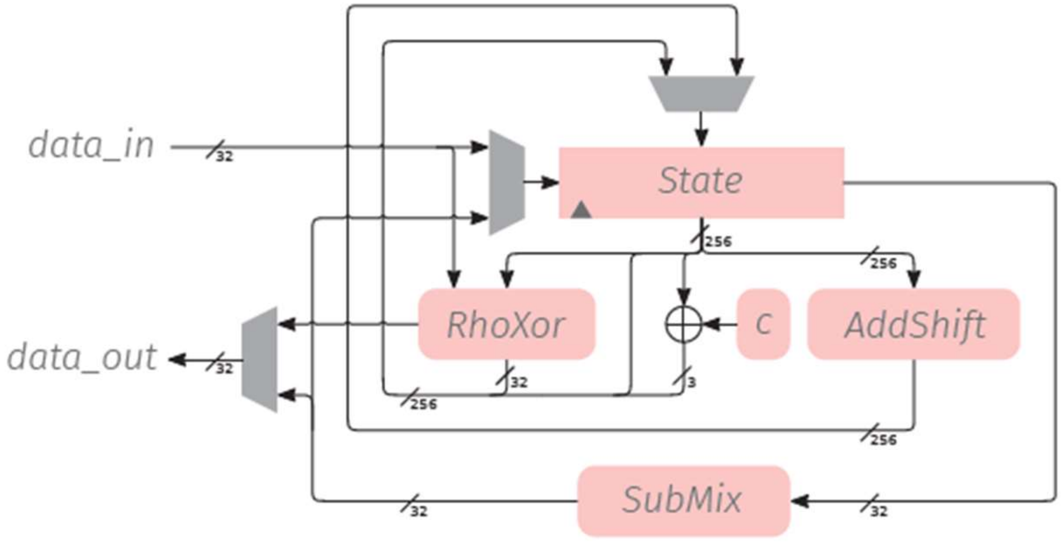
00	01	02	03	04	05	06	07
09	0A	0B	0C	0D	OE	0F	08
12	13	14	15	16	17	10	11
1B	1C	1D	1E	1F	18	19	1A
24	25	26	27	20	21	22	23
2D	2E	2F	28	29	2A	2B	2C
36	37	30	31	32	33	34	35
3F	38	39	3A	3B	3C	3D	3E

Hardware Implementation

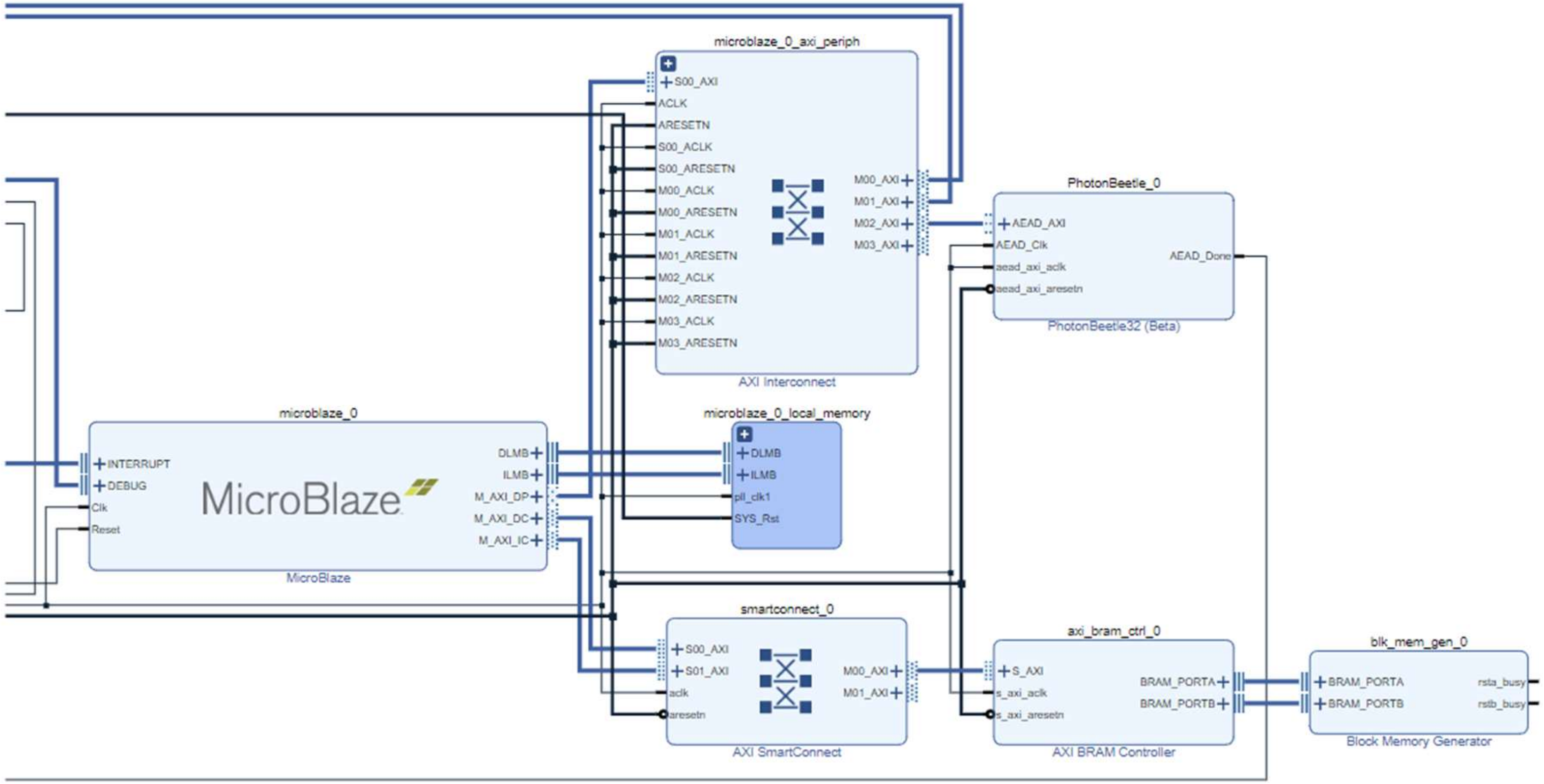
Serialization of P_{256}



Proposed Architecture

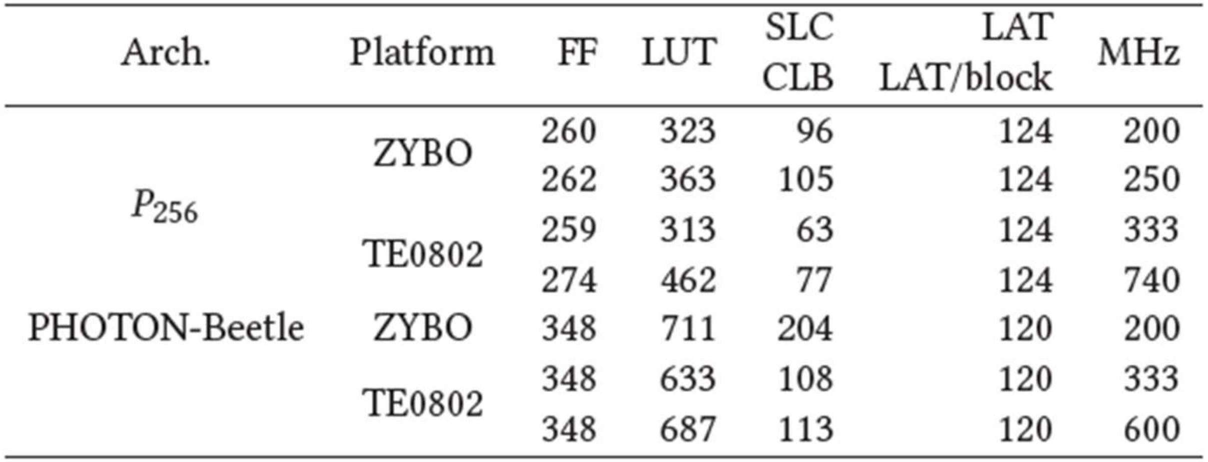


Packing the core as an IP



Experimental evaluation

- Set of test vectors provided to NIST



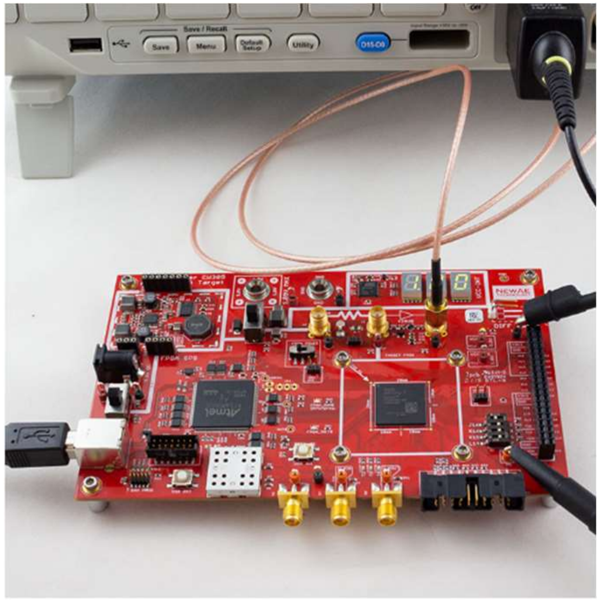
Arch.	Platform	FF	LUT	SLC	LAT	MHz
				CLB	LAT/block	
P_{256}	ZYBO	260	323	96	124	200
		262	363	105	124	250
	TE0802	259	313	63	124	333
		274	462	77	124	740
PHOTON-Beetle	ZYBO	348	711	204	120	200
	TE0802	348	633	108	120	333
		348	687	113	120	600

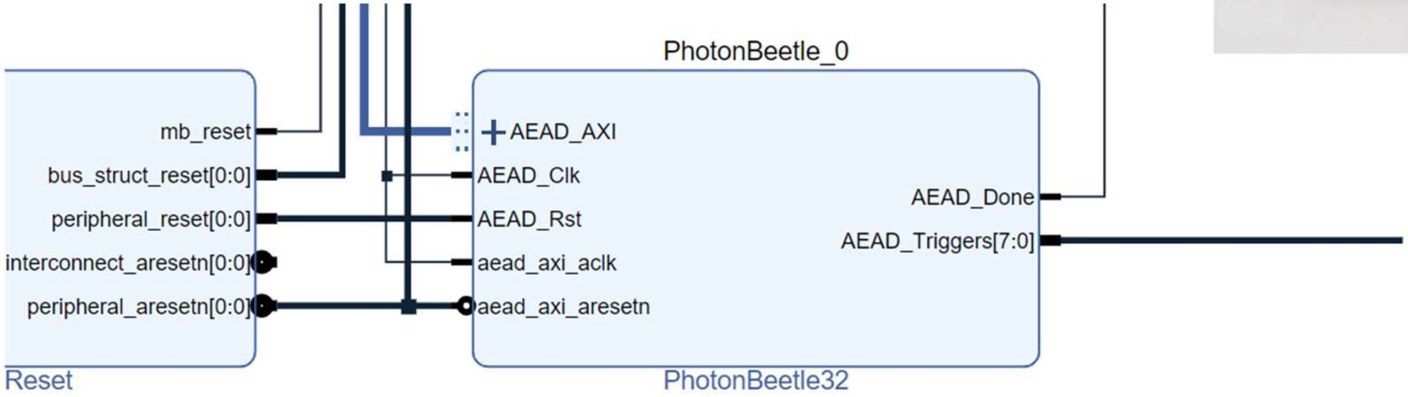
Implementation results

Side-Channel Attack (1)

- Test the robustness against Side-Channel Attack

- Step 1: acquire power consumption traces
  - *Nonce variation*
  - *CW305 with amplified power consumption output*

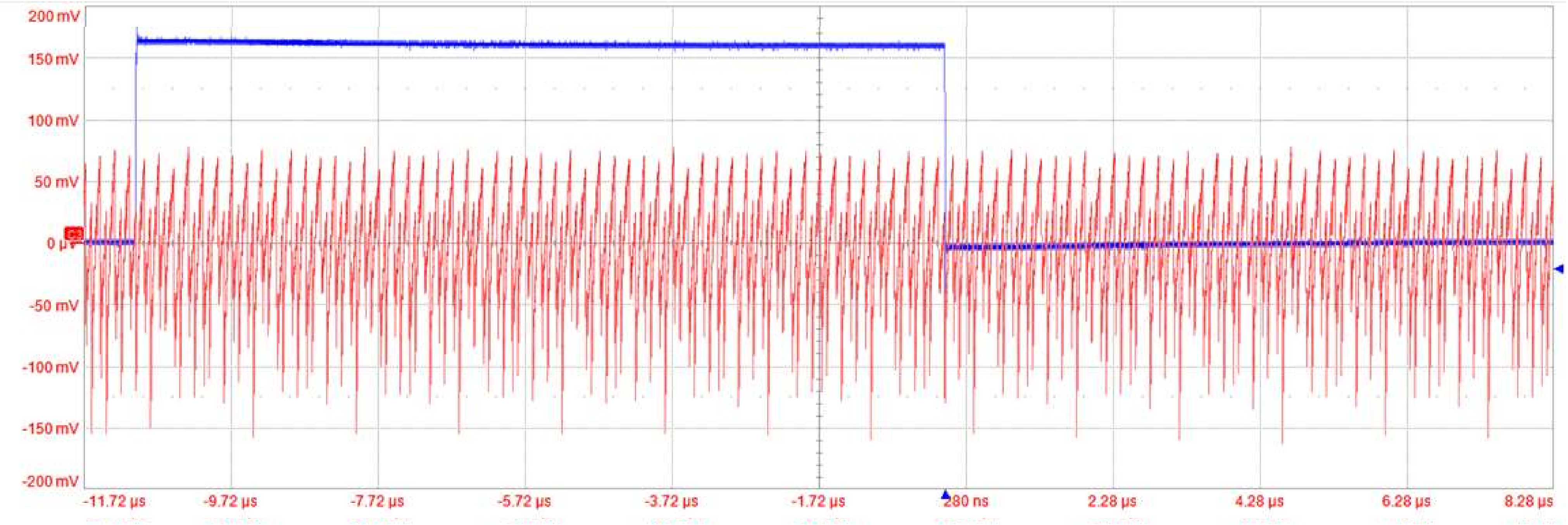




Side-Channel Attack (2)

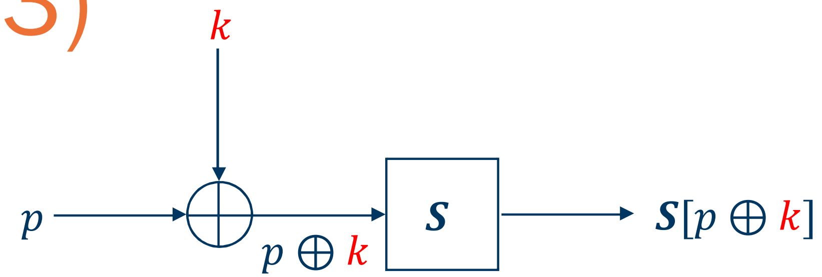
First PhotonBeetle call

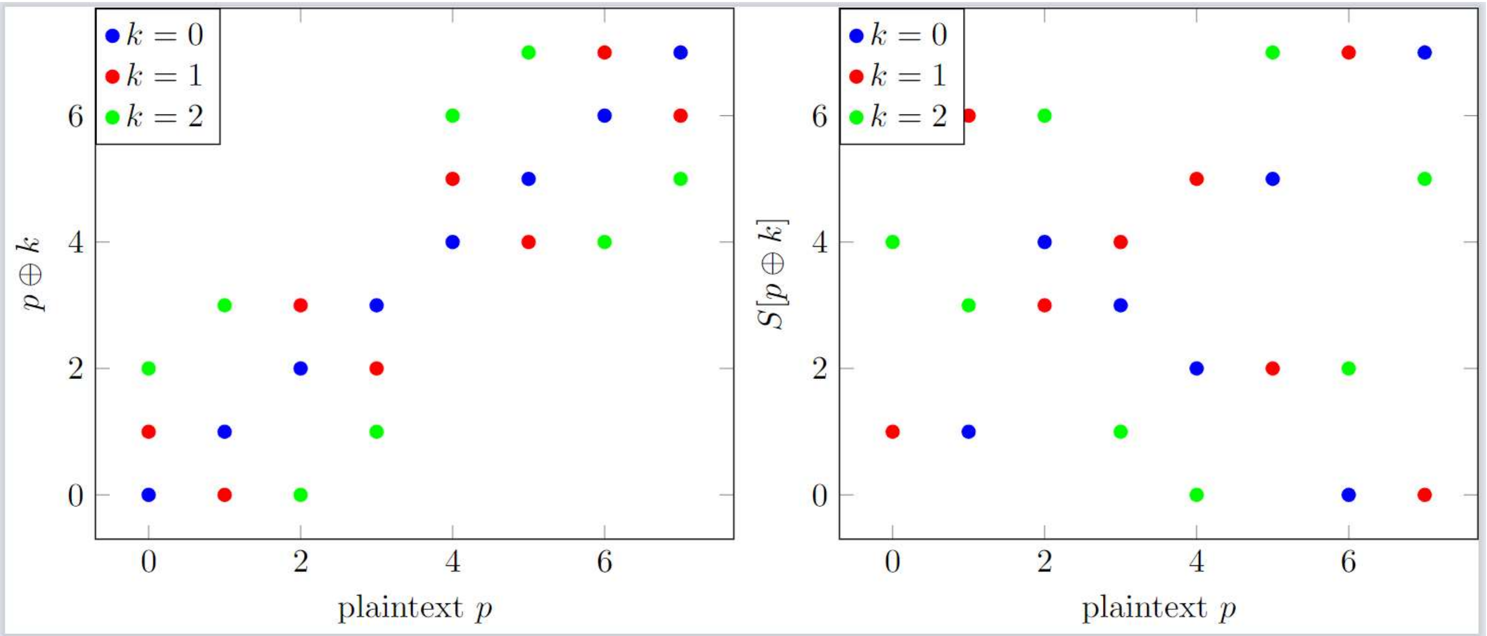
Power consumption of the FPGA



Side-Channel-Attack (3)

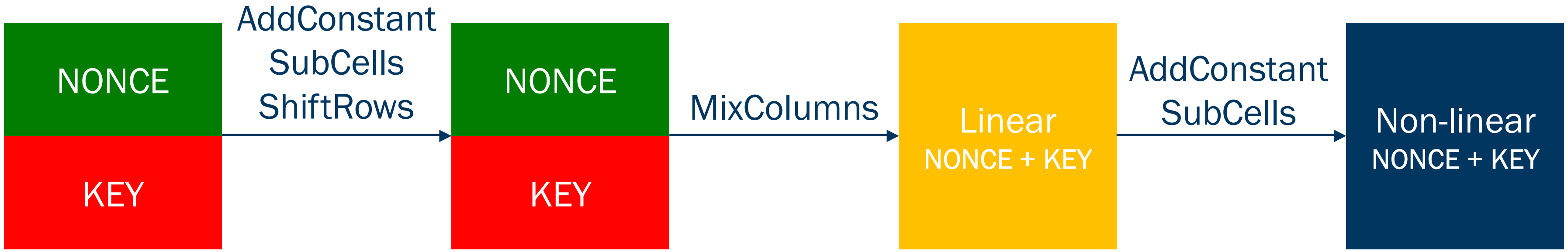
- Step 2: Power Analysis
  - *Classical DPA on block cipher*



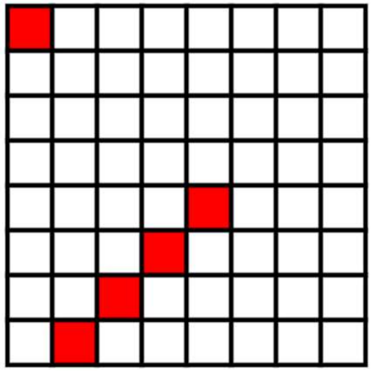


Side-Channel-Attack (4)

- PhotonBeetle



- Each nibble of the nonce relies on 4 nibbles of the key



Conclusion and Future work

- Hardware Implementation
  - *Serialization of the* P_{256}
  - *Core packed as an IP*

- Side-channel attacks
  - *Power analysis on the traces to recover the key*
  - *Robustness evaluation against SCA*

- Protection against SCA
  - *Protected implementation*

THANK YOU

- Weak points
  - *Too slow*
    - Processing each input block using a hash leads to high processing latency
  - *Underlying permutation is only 128 bits*
    - 112 bit s of security against pre-image and colliion attacks according to the original photon assesment
  - *Absorbtion od the key and the nonce*
    - Clear point where the Power analysis can focus
  - *Squeezing of the tag*
    - Symply empties half of the state
- Fix this points would lead to greater delays, inviable for lightweight algorithms