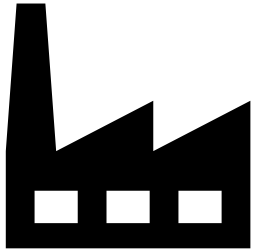
FPGA Implementations of Message Authentication Codes based on Ascon-*p*

MUSTAFA KHAIRALLAH AND SRINIVASAN YADHUNATHAN
SEAGATE RESEARCH GROUP

06/21/2023

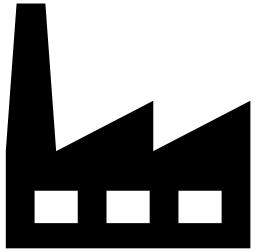
Motivation



- The products must use standardized primitives.
- Certification requires compliance with NIST standards.

- Engineers and system architects need functions that are not covered by the standard.
- Due to cost and effort constraints no new primitives can be implemented.

Motivation



- The products must use standardized primitives.
- Certification requires compliance with NIST standards.

Ascon includes Authenticated Encryption with Associated Data (AEAD) and Hashing schemes.

It does not include a dedicated PRF/Message Authentication Code.

It can be used as a nonce-based MAC by setting the plaintext as empty.

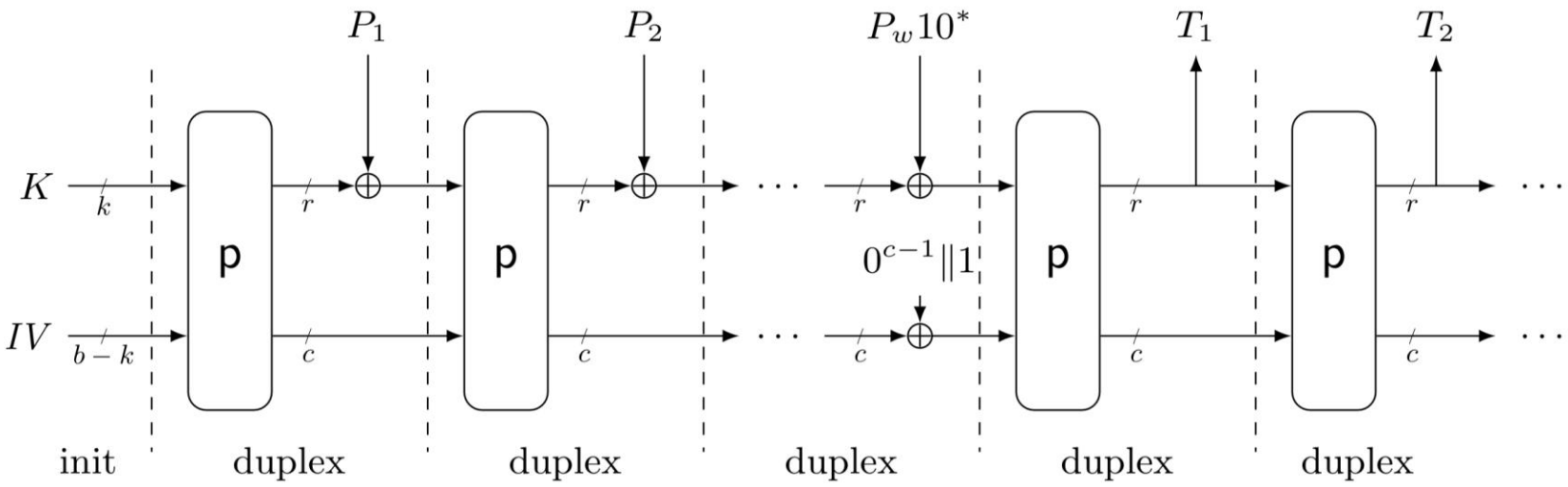
- Engineers and system architects need functions that are not covered by the standard.
- Due to cost and effort constraints, no new primitives can be implemented.

Our Target and Solutions

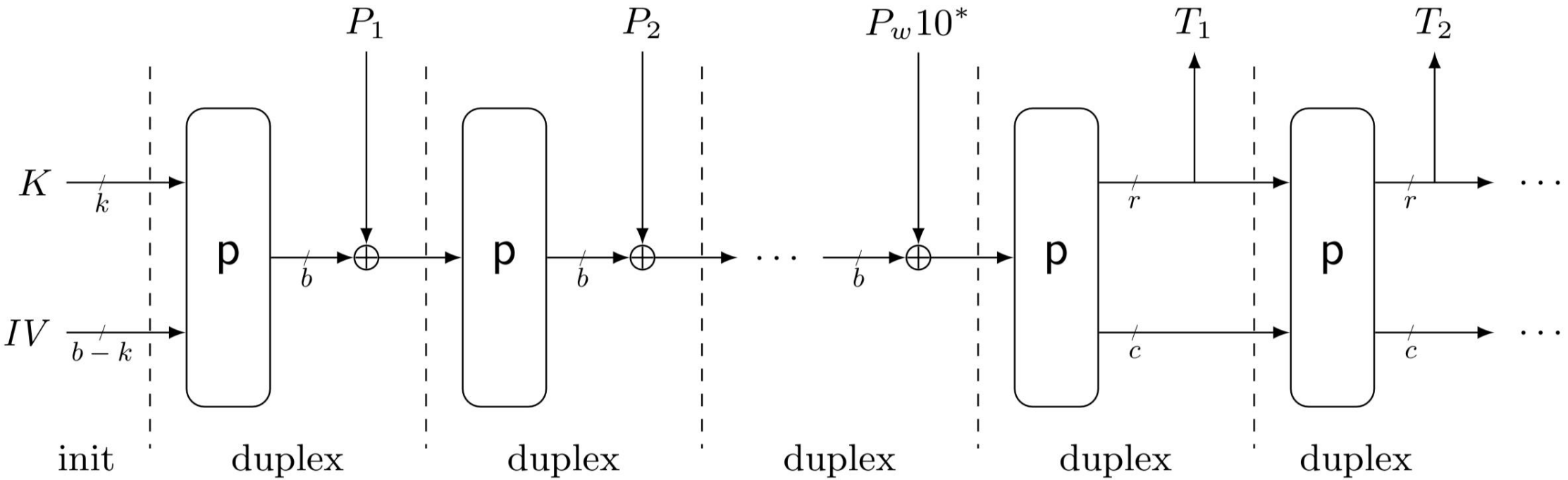
- Unprotected implementations.
- 64-bit and 128-bit security.

Scheme	Parallelizable	Rate	Security
FSKS-64	No	320/12	128
FSKS-128	No	320/12	64
Ascon-PRF-192	No	192/8	128
Ascon-PRF-64	No	64/6	64
Ascon-Farfalle	Yes	320/6	64

Ascon-PRF


P_1
P_2
P_w10^*
T_1
T_2
K
IV
k
$b-k$
r
c
p
$0^{c-1}\|1$
init
duplex
duplex
duplex
duplex
duplex

Full-State Keyed Sponge



Security

B. Mennink, “Understanding the duplex and its security,” Cryptology ePrint Archive, 2022

- The Ascon-PRF security is dominated by the capacity c.
- To target 128-bit security, we can set c=128, r=192 and use 8 rounds per call.
- In order to use 6 rounds per call, we need to set c=256 and r=64.

Security

B. Mennink, “Understanding the duplex and its security,” Cryptology ePrint Archive, 2022

- FSKS security requires DT < 2^c.
- We use 12 rounds for all calls of FSKS, since the full state is updated every time.
- The difference between instances refers to the number of calls needed to generate the tag.

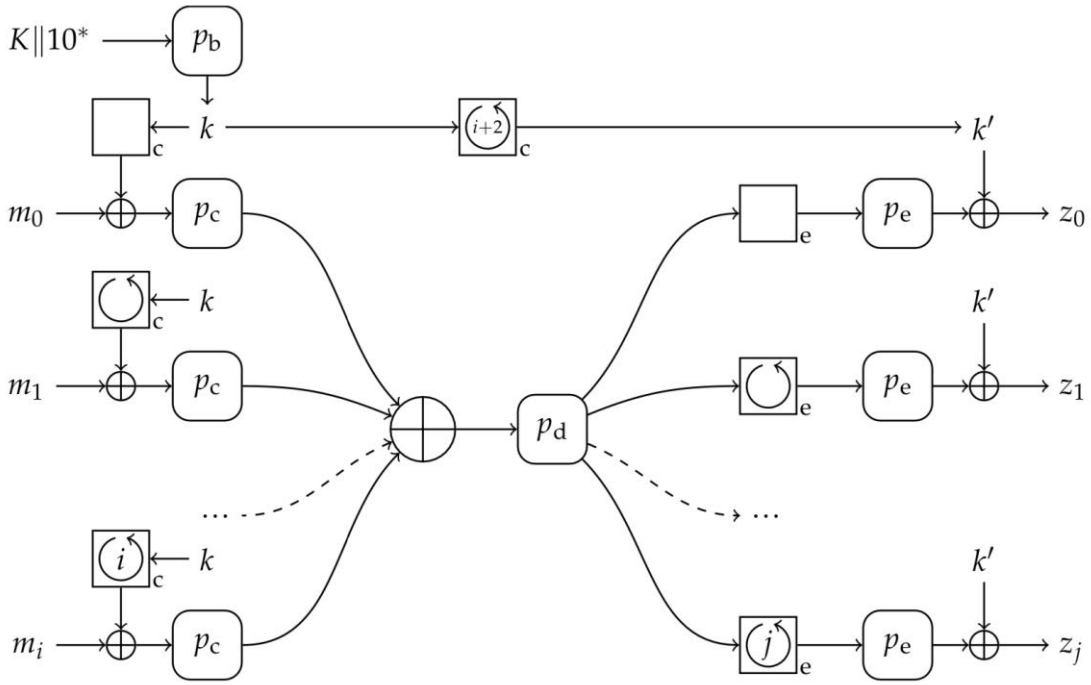
Implementation Strategy

- Iterative implementation.
- Different unrolling levels.
- The number of permutation rounds depends on the security level targeted and the rate of the scheme.

- The Ascon family uses 6 rounds for 64-bit security with rate 64.
- The Ascon family uses 8 rounds for 128-bit security with rate 128.
- The Ascon family uses 12 rounds for initialization and finalization.

Farfalle

G. Bertoni, J. Daemen, S. Hoffert, M. Peeters, G. Van Assche, and R. Van Keer, "Farfalle: parallel permutation-based cryptography," IACR Transactions on Symmetric Cryptology



- 6 rounds for all instances of the permutation (except key derivation).
- A 320-bit Galois LFSR for the compression rolling function.
- The extraction rolling function is not needed in the MAC use case.

Implementation Strategy

- 6-stage pipelined implementation.
- The permutation can process six 320-bit blocks simultaneously.

Implementation results on Xilinx Artix-7 FPGA

	Rounds/ Cycle	LUTs	FFs	Power (Watts)	Period (ns)	Cycles (1600 Bytes)	Cycles (16640 Bytes)	Throughput (Mbps, short)	Throughput (Mbps, long)	Energy (nJ, short)	Energy (nJ, long)
FSKS-64	1	1007	391	0.15	5	516	5028	77.52	82.74	387.00	3771.00
	2	1658	390	0.189	5	258	2514	155.04	165.47	243.81	2375.73
	3	2167	391	0.249	7	172	1676	166.11	177.29	299.80	2921.27
	4	2376	289	0.3	7.5	129	1257	206.72	220.63	290.25	2828.25
	6	3334	390	0.39	10	86	838	232.56	248.21	335.40	3268.20
	12	6170	389	0.464	20	43	419	232.56	248.21	399.04	3888.32
FSKS-128	1	1082	391	0.152	5	504	5016	79.37	82.93	383.04	3812.16
	2	1728	390	0.189	5	252	2508	158.73	165.87	238.14	2370.06
	3	2232	391	0.25	7	168	1672	170.07	177.72	294.00	2926.00
	4	2751	389	0.3	7.5	126	1254	211.64	221.16	283.50	2821.50
	6	3370	390	0.396	10	84	836	238.10	248.80	332.64	3310.56
	12	6207	389	0.47	20	42	418	238.10	248.80	394.80	3929.20
Ascon-PRF-192	1	1031	391	0.149	5	558	5571	71.77	74.68	415.21	4150.15
	2	1693	390	0.19	5	279	2786	143.54	149.35	264.73	2646.07
	4	2421	389	0.308	7.5	140	1393	191.39	199.14	321.86	3217.06
	8	4312	388	0.44	13	70	697	220.83	229.78	398.49	3983.03
Ascon-PRF-64	1	826	391	0.144	5	1224	12504	32.68	33.27	881.28	9002.88
	2	1521	390	0.187	5	612	6252	65.36	66.54	572.22	5845.62
	3	2004	391	0.247	7	408	4168	70.03	71.29	705.43	7206.47
	6	3185	390	0.376	10	204	2084	98.04	99.81	767.04	7835.84
Ascon-Farfalle	6	3863	2634	0.43	4	58	434	862.07	1198.16	99.76	746.48

Efficiency

Scheme	Throughput/LUT	Energy/LUT.bit
FSKS-64	0.099 (2 rounds)	8.94×10^{-6} (4 rounds)
FSKS-128	0.096 (2 rounds)	9.84×10^{-6} (4 rounds)
Ascon-PRF-192	0.088 (2 rounds)	9.98×10^{-6} (4 rounds)
Ascon-PRF-64	0.043 (2 rounds)	1.85×10^{-5} (3 rounds)
Ascon-Farfalle	0.31	1.45×10^{-6}

Final notes

- Farfalle offers a very interesting efficient PRF based on the Ascon permutation.
- The drawback is that this efficiency comes from parallelism and is for long messages.
- A standard Ascon implementation is serial.
- For a serial implementation, FSKS is still more efficient than Ascon-PRF even with more rounds per permutation.
- The only hidden cost is that FSKS requires more XORs for absorption.