Towards a Lightweight Hardware Implementation of PRIMATEs - A Recent Authenticated Encryption Candidate

Richard Fellner, Philipp Jantscher, Thomas Korak, and Samuel Weiser
thomas.korak@iaik.tugraz.at, {richard.fellner, philipp.jantscher, samuel.weiser}@student.tugraz.at

Introduction

We have developed one of the first hardware implementations of the PRIMATEs HANUMAN-120 [1] algorithm. This algorithm has been proposed for the CAESAR competition [2].

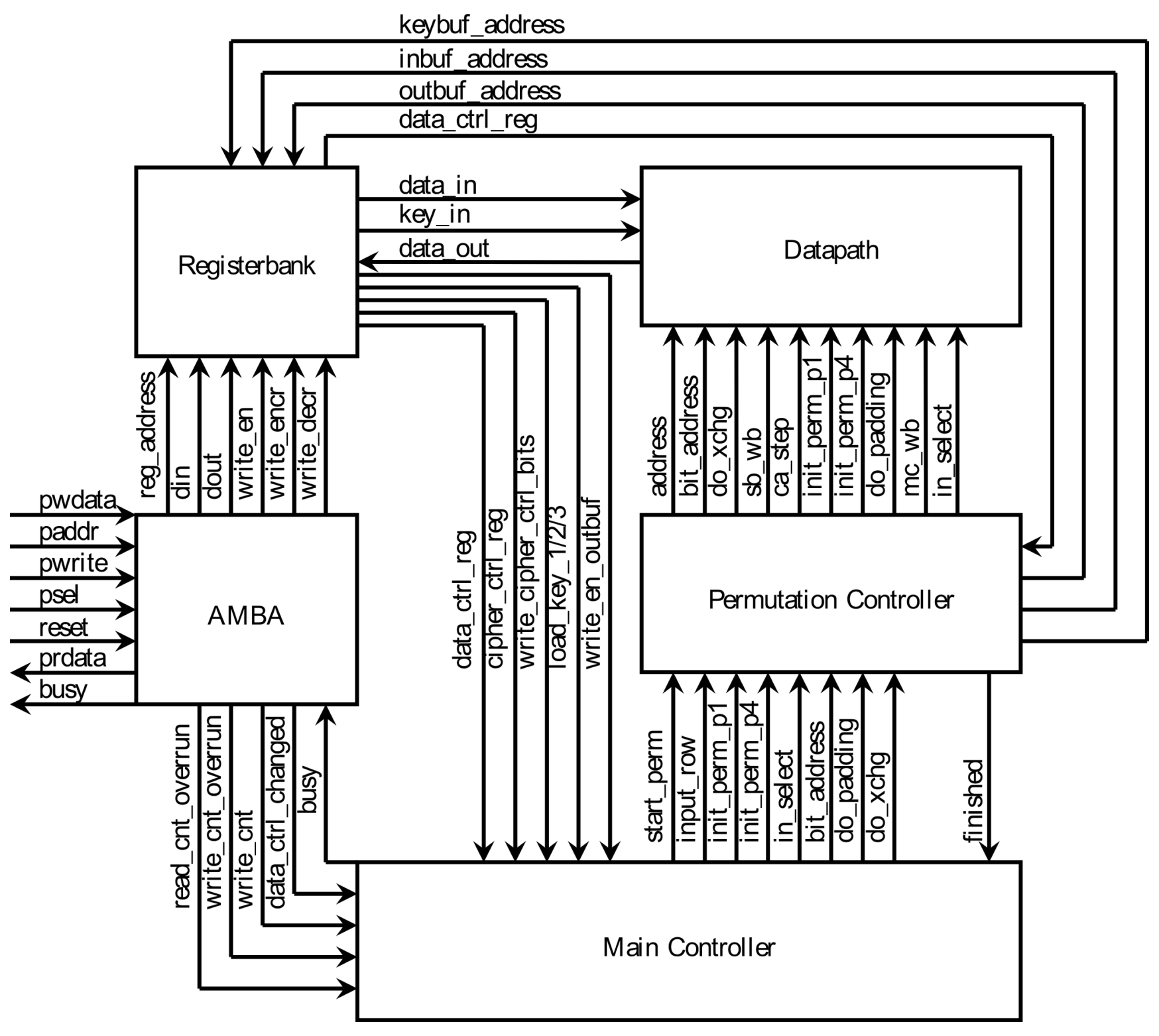
The CAESAR competition was initiated in 2013. It aims at finding the new authenticated encryption standard, which not only provides confidentiality but also data integrity and authenticity.

Our prototype implementation focuses on low chip area. It demonstrates the resource-saving design of modern authenticated encryption algorithms.

This hardware implementation shall serve as a reference point for future projects.

Compared with several hardware implementations of the sponge-based hash function KECCAK (winner of the NIST SHA-3 competition in 2012), this prototype implementation is able to keep up in terms of chip size.

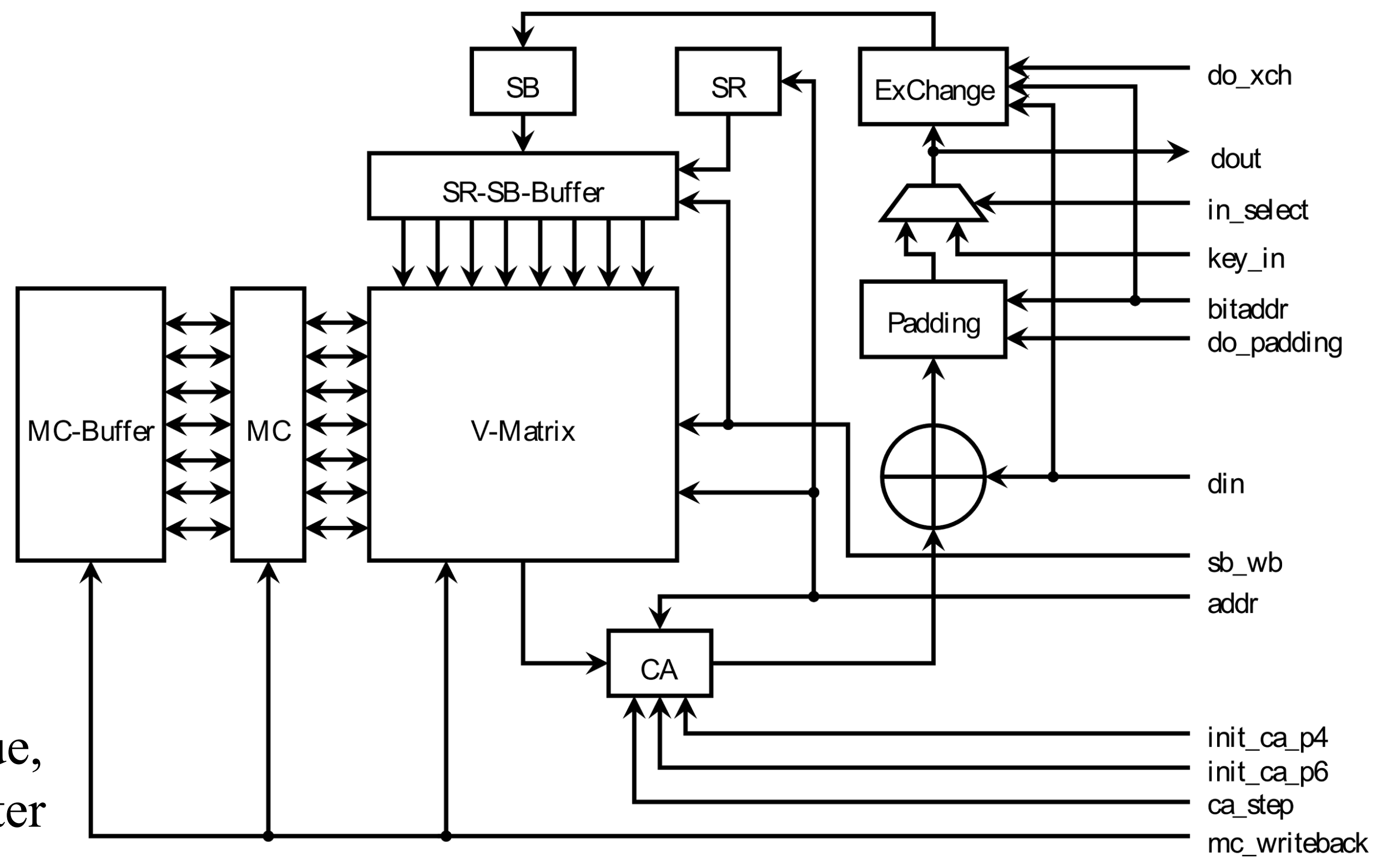
Architecture



- **AMBA Interface**: Handles chip I/O
- **Main Controller**: Handles encryption and decryption and starts the permutation controller
- **Permutation Controller**: Controls the Datapath in order to do permutations
- **Registerbank**: Contains registers for input and output buffering, configuration, control and the key

Datapath

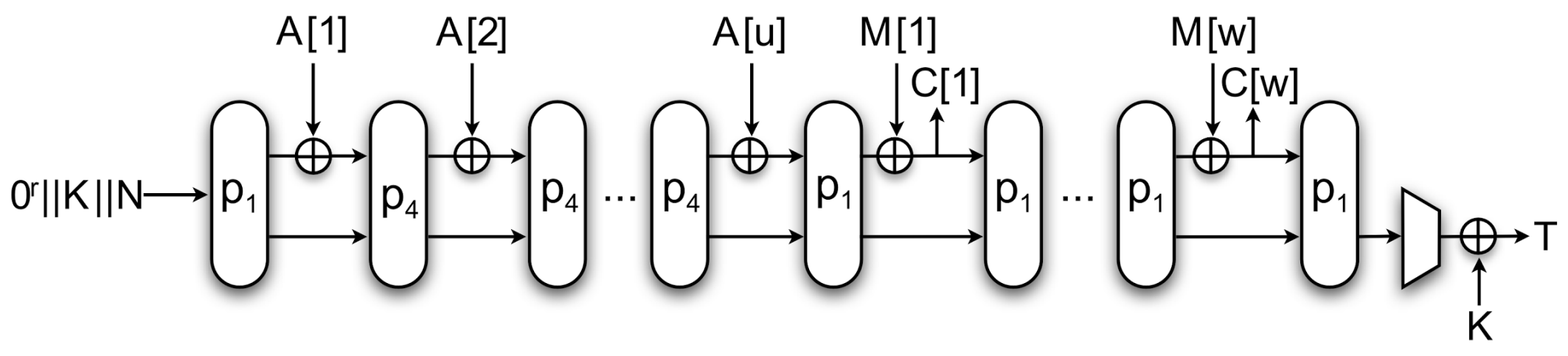
- **V-Matrix**: The current cipher state (280 bit)
- **SubBytes** (SB): Element substitution using a Lookup-Table
- **ShiftRows** (SR): Roundshift the column address depending on the row address
- **MixColumns** (MC): Column-wise Galois multiplication using Lookup-Tables
- **ConstantAddition** (CA): XOR a constant value, generated using a Linear Feedback Shift Register
- **Padding** & **ExChange**: Handle padding on bit level



Algorithm Description

The PRIMATEs family [1] consists of the algorithms GIBBON, HANUMAN and APE. We implemented HANUMAN-120.

PRIMATEs use a Sponge-based design for doing authenticated encryption. The construction allows reuse of the same permutation for encryption and decryption. The algorithm operates on 5-bit elements.

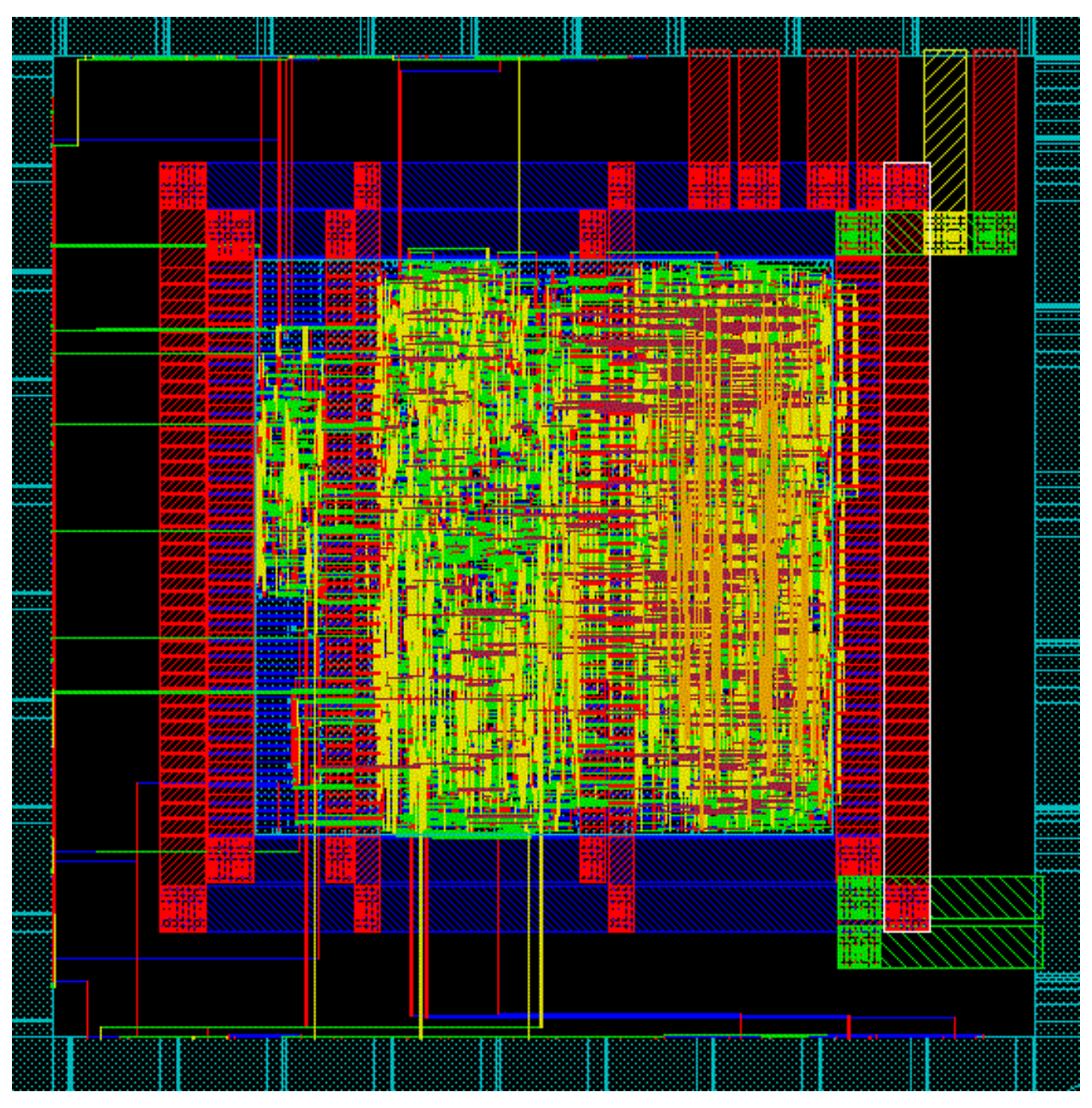


[1]

Starting with key K and nonce N, the "*Sponge*" absorbs authenticated data A, doing permutations p_i. Now the "*Sponge*" is ready for absorbing plaintext messages M_i while squeezing ciphertext C_i. Finally, tag T is generated which vouches for authenticity.

The AES-like [6] permutation p_i consists of the operations SubBytes, ShiftRows, MixColumns and ConstantAddition. MixColumns uses a reduced Galois matrix for minimizing hardware requirements.

Synthesis Results & Comparison



Synthesis Result	Value	Unit
Size	7116 36435	GE µm²
Power consumption	18.76	µW / MHz
Throughput @1MHz	2114 18.92	cycles / block kBit /s
Max. clock frequency	108	MHz

Algorithm	Area	State Size
-	GE	bits
HANUMAN-120	**7116**	**280**
KECCAK(1600) [3]	5522	1600
KECCAK(1600) [4]	9300	1600
KECCAK(1600) [5]	20790	1600
KECCAK(800) [5]	13000	800
KECCAK(400) [5]	5090	400
KECCAK(200) [5]	3170	200

HANUMAN-120 can keep up with current KECCAK implementations in terms of chip size and security, as the design of the PRIMATEs family is based on the same strong design principles as AES.

References

[1] E. Andreeva, B. Bilgrin, A. Bogdanov, A. Luykx, F. Mendel, B. Mennink, N. Mouha, Q. Wang, and K. Yasuda. Submission to the caesar competition. http://primates.ae/, 2014.
[2] CAESAR. Competition for Authenticated Encryption: Security, Applicability, and Robustness. http://competitions.cr.yp.to/caesar.html
[3] P. Pessl and M. Hutter. Pushing the limits of sha-3 hardware implementations to fit on RFID. In Springer, editor, CHES 2013, volume 8086 of LNCS, pages 126 – 141. Springer, 2013.
[4] G. Bertoni, J. Daemen, M. Peeters, G. Van Assche, and R. Van Keer. Keccak implementation overview, 2012.
[5] E. B. Kavun and T. Yalcin. A lightweight implementation of keccak hash function for radio-frequency identification applications. In Radio frequency identification: security and privacy issues, pages 258–269. Springer, 2010.
[6] NIST. Announcing the ADVANCED ENCRYPTION STANDARD (AES). In FIPS-197, 2001.