A Low Area AES Encryption Core with Silicon Demonstration in 180nm CMOS Process

Anh-Thai Nguyen, Van-Lan Dao, Van-Phuc Hoang*

Le Quy Don Technical University, 236 Hoang Quoc Viet Str., Hanoi, Vietnam

Received: August 01, 2017; Accepted: November 26, 2018

Abstract

This paper presents a low area, low power AES encryption core with the combination of several optimized components in the AES core and some modifications in the core architecture for emerging wireless networks and IoT systems. The detail results of area-speed-power trade-offs in the proposed AES core design are also presented and discussed. The implementation and chip measurement results in 180nm CMOS technology show that the proposed AES encryption core can reduce the area and power consumption significantly. The power consumption of the proposed AES encryption core is only 7.1 μW/MHz and the area is 2.3 kgates which are much lower than other AES cores presented in literature.

Keywords: AES, ASIC, low area, low power, CMOS

1. Introduction

Currently, wireless networks are highly employed for many applications such as personal area connection, broadband internet connection, smart home, smart environment monitoring, etc. Due to the employment of the wireless channel, secure connectivity is becoming a more and more essential issue for these networks [1]. Moreover, emerging Internet of Things (IoT) applications need the hardware security assurance [2]. Advanced Encryption Standard (AES) is a highly recommended security standard of data encryption for emerging wireless networks and IoT applications [3]. Although AES encryption/decryption algorithms have been standardized, the efficient hardware architecture and implementation methods are the topics which many researchers are focusing on. However, with the fast development of many portable, wearable applications and devices, especially in IoT systems, the low area, low power and secure hardware implementations are highly required. Therefore, the higher energy efficiency VLSI implementations are highly expected. In the era of IoT, low power and high security requirements can be promisingly fulfilled by hardware cryptography implementation.

The objective of this paper is to design a low area, low power AES core which includes both encryption and decryption functions for such area and power constrained wireless networks and applications. Our main contribution is that a low area, low power AES encryption core implementation with the silicon demonstration is proposed by combining several optimized components in the AES core and some modifications in the core architecture for the high hardware resource efficiency in the ASIC platform.

In this paper, Section 2 describes the compact AES core architectures. Section 3 and section 4 present the optimized S-box and improved key-expansion unit which are two essential components in the proposed AES encryption core. Then, section 5 presents the implementation results and section 6 concludes the paper.

2. Low area AES encryption core architecture

AES encryption core processes data in 128-bit blocks with the key lengths of 128, 192 or 256 bits. In this paper, for a low area implementation, the key length of 128-bit is chosen. Figure 1 shows the 128-bit AES encryption/decryption algorithms. The left hand side is the encryption flow and the right hand side is the decryption one. In this paper, to reduce the AES encryption core area, we employ 8-bit architecture with compact S-boxes so that the AES core encrypts one 8-bit data block in each clock cycle. Authors in [4]-[5] also focused on the optimizing AES encryption core for the low area implementation. However, they used an LUT-based (non-optimized) S-box that may result in a high area ASIC implementation. Hence, some papers such as [6]-[9] proposed the optimized S-box designs for low area AES implementations. However, more efficient AES encryption cores are highly required.

* Corresponding author: Tel.: (+84) 982712371
Email: nguyenanhthai77@gmail.com

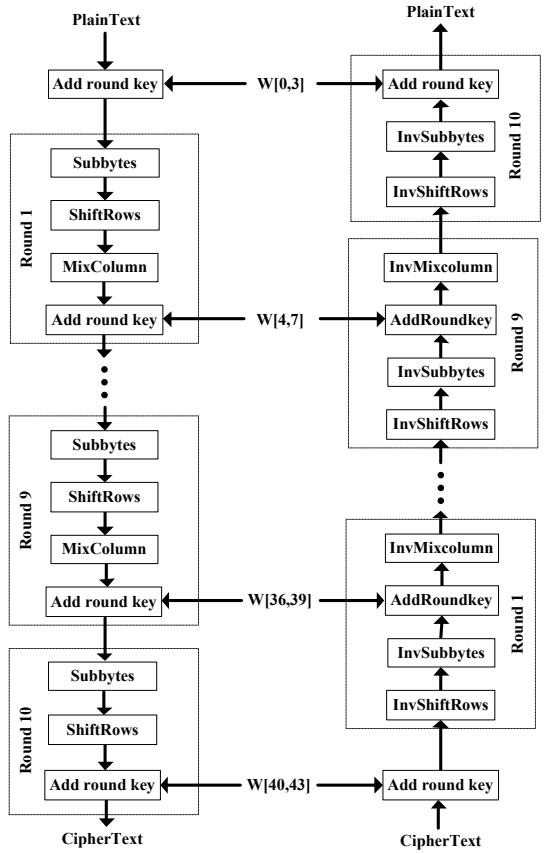


Fig. 1. Standardized AES encryption and decryption algorithms.

Firstly, the proposed hardware architecture for the AES encryption core is shown in Fig. 2 with the parameters in Table I. In this table, *w* is the datapath width and *n* is the bit-width of the mixcolumn block. The AES core encrypts a *w*-bit data block in each clock cycle. The AES encryption core includes a key expansion unit, a mixcolumn unit, a shift-row unit, a shift register and a byte permutation unit using S-box. In the shift register as depicted Fig. 3, the control signals (E1, E2) are generated from the controller. As shown in Fig. 4, the proposed AES core employs a simple counter-based controller. The control signal is generated from a counter, comparators and a simple logic circuit. The upper half (with higher significant bits) of the counter output (CNT) is fed to key expansion block and the lower half is used to select the operations in each AES encryption round. To provide more detail implementation results showing the area-speed-power trade-offs, the proposed AES encryption core was designed with different datapath width values ranging from 8-bit to 64-bit. However, in the silicon demonstration, due to the limited chip area allocated for the core, an 8-bit architecture *(w=8)* with the optimized S-box is chosen to reduce the AES core area. Two S-box blocks are used in byte permutation and key expansion units [5]. The 8-bit architecture was also employed in [9]. However, the non-optimized S-box leads to more optimizations required. As shown in Fig. 2, the AES encryption core includes a key expansion unit, a mix-column unit, a parallel to serial converter and a byte permutation unit. Table II lists the function of each signal in the proposed AES core. S-box 1 and S-box 2 are two sub-blocks in the byte permutation unit as described in [9]. The detail implementation of this byte permutation unit will be presented in the next section. In the decryption core as depicted in Fig. 3, an additional inverse S-box is used. The 8-bit AES core requires 160 clock cycles for each encryption operation.

Table I. Datapath width and mixcolumn bit-width values

w	8	16	32	64
n	32	32	32	64

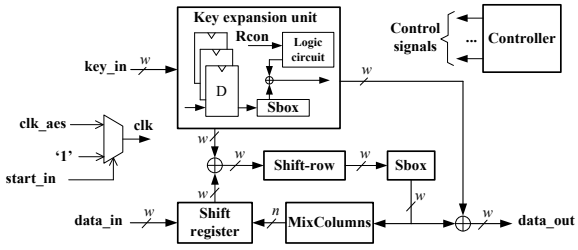


Fig. 2. The proposed AES encryption core architecture. The 8-bit architecture corresponds to the case of $w = 8$.

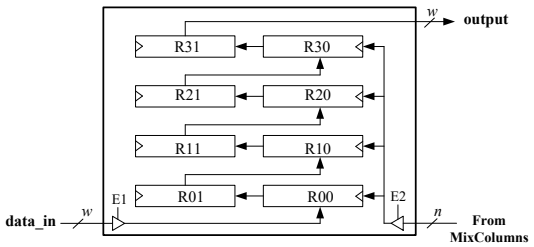


Fig. 3. The shift register block in proposed AES encryption cores.

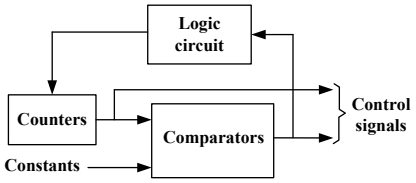


Fig. 4. The counter-based controller in the proposed AES encryption core.

Table II. Signals in the proposed aes core

Signal	Direction	Description
load_in	Input	Control signal to load data and key
unload_in	Input	Control signal to unload data and key
start_in	Input	Control signal to start the encryption
key_in	Input	Key input
data_in	Input	Data input
data_out	Output	Data output
busy_out	Output	To indicate that the output is ready to read
comp	Output	To indicate that the output is ready to read and the new input data can be fed

3. S-box design

S-box is an important block in the AES core so that some papers on S-box optimization for the specific requirements have been published such as in [6-9]. It can be optimized for speed or area depending on the application requiring the core. When using the LUT-based architecture, a 256-byte memory block is required so that the area may be high. Therefore, to reduce the complexity, we try to propose an alternative S-box architecture for the compact AES implementation.

Actually, S-box is an 8×8 matrix for the two following transformations. The first one is the byte inversion in which each byte is substituted by its inverted version (by the multiplication in GF(2^8)) and the second transformation the affine transformation in GF(2^8) according to (1).

$$y_i = x_i \oplus x_{(i+4) \bmod 8} \oplus x_{(i+5) \bmod 8} \oplus x_{(i+6) \bmod 8} \oplus x_{(i+7) \bmod 8} \oplus c_i \quad (1)$$

in which, $0 \le i < 8$ and $x = "x_0x_1x_2x_3x_4x_5x_6x_7"$ is the result of byte inverting, and $y = "y_0y_1y_2y_3y_4y_5y_6y_7"$ is the result of affine transformation. Byte c is the constant of {63} or {01100011}. The matrix form of this transformation is shown in (2).

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 10001111 \\ 11000111 \\ 11100011 \\ 11110001 \\ 11111000 \\ 01111100 \\ 00011111 \end{bmatrix} . \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad (2)$$

As we can see, each bit of one byte in GF(2^8) can be considered as a coefficient for an exponent in the polynomial of GF(2^8). As stated in [9], every component in GF(2^8) can be presented as a linear polynomial with the coefficients in GF(2^4). The linear polynomial can be written in the form of *(bx+c)*, via a second order polynomial of (x^2+Ax+B). Then, the inverting of any polynomial in the form of *(bx + c)* can be shown in (3).

$$(bx+c)^{-1} = b(b^2B + bcA + c^2)^{-1}x + (c+bA)(b^2B + bcA + c^2)^{-1} \quad (3)$$

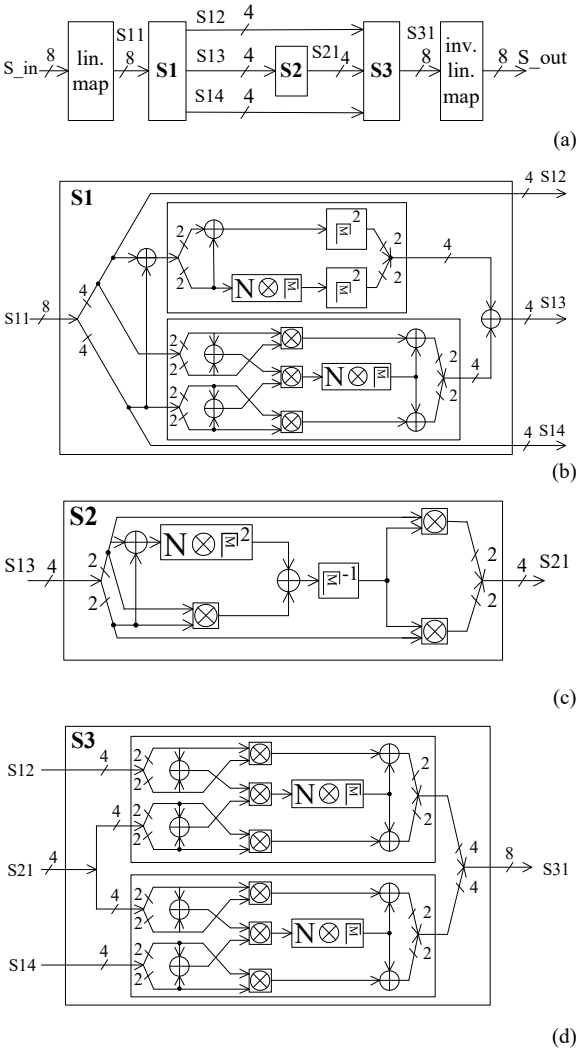


Fig. 5. Compact S-box architecture.

In this paper, the S-box is designed as presented in Fig. 5 and based on [7]-[9] to derive an efficient implementation. The S-box is transformed from GF(2^8) architecture to GF(2^8)/GF(2^4)/GF(2^2) architecture. The linear mapping block *(lin. map)* in Fig. 5 converts the basis from GF(2^8) to GF(2^8)/GF(2^4)/GF(2^2). After some processing steps,

the result from GF(2^8)/GF(2^4)/GF(2^2) is mapped to GF(2^8).

4. Rcon block optimization for key-expansion

According to [3], Rcon block takes the inputs from *r_in* signal which is the round index from 0 to 9. Moreover, in [5], Rcon is a multiplexer (MUX) circuit which uses *r_in* as the selection signal as shown in Figure 6a. In our design, Rcon block is optimized by the simple Karnaugh optimization method and the results are presented in (4) as well as in Fig. 6.

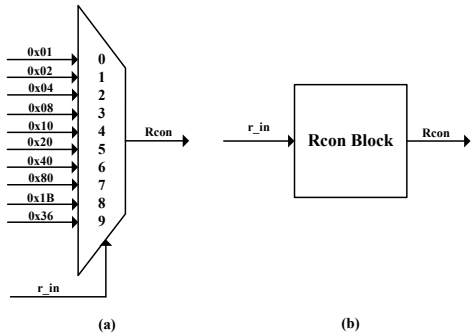


Fig. 6. Rcon block design in [7] using a MUX (a) and using Karnaugh optimization in this paper (b).

$$\begin{cases} Rcon_7 = r\_in_2.r\_in_1.r\_in_0 \\ Rcon_6 = r\_in_2.r\_in_1.\overline{r\_in_0} \\ Rcon_5 = r\_in_2.r\_in_0.\overline{r\_in_1} + r\_in_3.r\_in_0 \\ Rcon_4 = r\_in_3 + r\_in_2.\overline{r\_in_1}.\overline{r\_in_0} \\ Rcon_3 = \overline{r\_in_2}.\overline{r\_in_0}.r\_in_3 + r\_in_0.\overline{r\_in_2}.r\_in_1 \\ Rcon_2 = \overline{r\_in_2}.\overline{r\_in_0}.r\_in_1 + r\_in_3.r\_in_0 \\ Rcon_1 = r\_in_3 + \overline{r\_in_2}.\overline{r\_in_1}.r\_in_0 \\ Rcon_0 = \overline{r\_in_2}.\overline{r\_in_1}.\overline{r\_in_0} \end{cases} \quad (4)$$

5. Implementation results

To provide more detail implementation results showing the area-speed-power trade-offs, the proposed AES encryption core was implemented with different datapath width values ranging from 8-bit to 64-bit as presented in Fig. 7 and Table III. However, in the silicon demonstration, due to the limited chip area and I/O pins allocated for the core, an 8-bit architecture *(w = 8)* with the optimized Sbox is chosen. The AES encryption core was implemented with VHDL, simulation in Modelsim tool and then implemented with an 180nm CMOS standard library by Synopsys design tools. Figure 8 is the simulation model for the 8-bit AES encryption core. The input generation block generates the input vector values for AES core verification. Figure 9 and Fig. 10 present the functional simulation results in Modelsim tool and post-layout simulation results in Synopsys VCS tool, respectively. Table IV is an example of the test vector for the AES encryption core verification in the case of the encryption operation.

Table III. ASIC implementation results of proposed AES encryption core with different datpath widths in 180nm CMOS process

w	Area (kgates GE)	Speed (MHz)	Power (µW/MHz)	Cycle count
8-bit	2.3	67	7.1	160
16-bit	3.7	67	7.8	80
32-bit	4.3	67	9.5	40
64-bit	6.1	67	15.0	20

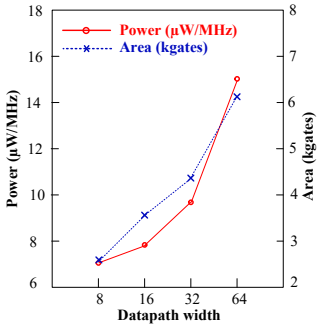


Fig. 7. ASIC implementation results of area and power consumption of the proposed AES encryption core in 180nm CMOS process with different values of datapath width (*w*)

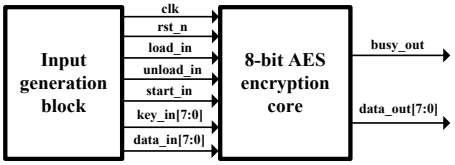


Fig. 8. The simulation model for the 8-bit AES core.

The implementation results are presented in Table V in which the proposed 8-bit AES encryption core *(w = 8)* is compared with other designs. It can be seen that the proposed AES core can reduce the area and power consumption significantly compared with some other designs. The AES encryption core area can be reduced to 2.3kgates (GE: gate equivalents) and the power consumption can be reduced to 7.1µW/MHz with the supply voltage of 1.8V. Figure 11 is the chip microphotography of the proposed 8-bit AES encryption core with 180nm CMOS technology. Fig. 11a is the full chip and Fig. 11b is the AES encryption core microphotographies, respectively. The chip measurement results have confirmed the correct operation, maximum frequency and power consumption of the proposed 8-bit AES encryption core.

Table IV. A test vector for AES encryption core verification

data_in (hexa)	key_in (hexa)	data_out (hexa)
0x00,0x11,0x22, 0x33,0x44,0x55, 0x66,0x77,0x88, 0x99,0xAA,0xBB, 0xCC,0xDD, 0xEE,0xFF	0x00,0x01,0x02, 0x03,0x04,0x05, 0x06,0x07,0x08, 0x09,0x0A,0x0B, 0x0C,0x0D, 0x0E,0x0F	0x69,0xC4,0xE0, 0xD8,0x6A,0x7B ,0x04,0x30,0xD8, 0xCD,0xB7,0x80, 0x70,0xB4, 0xC5,0x5A

Table V. Implementation results of proposed 8-bit AES encryption core compared with other papers

Design	Techno.	No. of cycles	Speed (MHz)	Area (kgates GE)	Power consumption
Our work	180nm	160	67	2.3	7.1 µW/MHz (*)
[5]	130nm	160	152	3.1	37 µW/MHz
[10]	22nm	336	1133	2.0	13 mW
[11]	130nm	356	13.2	5.5	99 µW/MHz
[12]	65nm	200	11.0	0.012 mm^2	14.6 µW @ 0.5V

(*): The chip power consumption was measured with less than 10% inaccuracy.

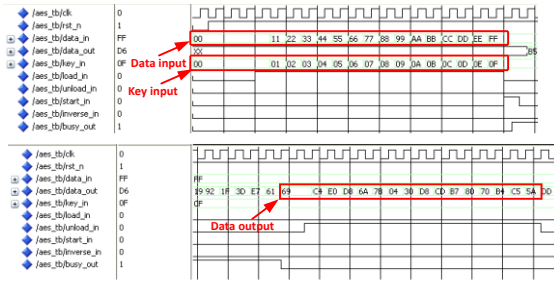


Fig. 9. Simulation results the proposed 8-bit AES encryption core in Modelsim tool.

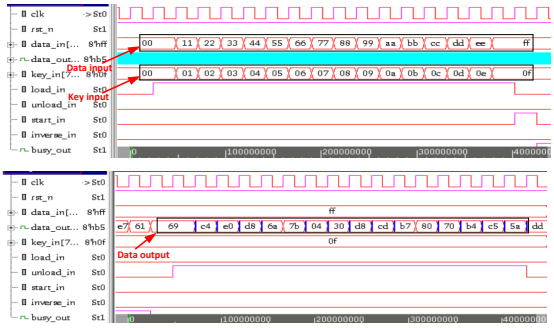
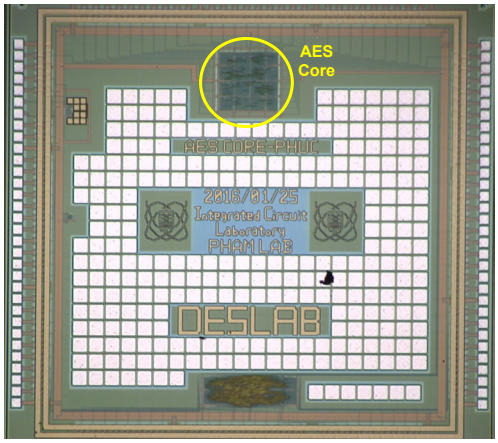
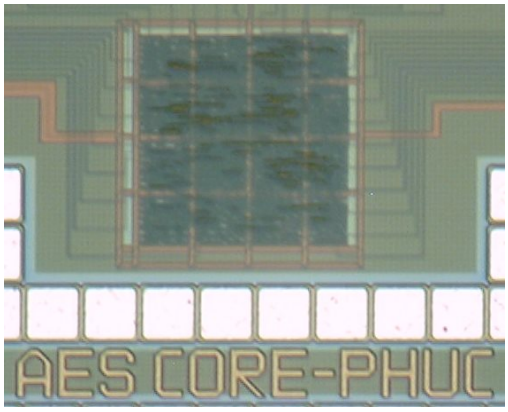


Fig. 10. Post-layout simulation results of the proposed 8-bit AES encryption core with Synopsys VCS tool.



(a)



(b)

Fig. 11. Chip microphotography of the proposed AES encryption core using 8-bit architecture with 180nm CMOS technology, the core layout dimension is 300×300µm.

5. Conclusions

This paper has presented a low power, area efficient AES core for emerging wireless networks. The implementation results in an 180nm CMOS ASIC library show that by using an optimized S-box and an improved Rcon design, the AES encryption core area can be reduced to 2.3kgates and power consumption can be reduced to 7.1µW/MHz with the supply voltage of 1.8V. Therefore, this core is highly potential to be used in energy constrained wireless network applications such as wireless sensor networks, IoT systems for environment monitoring which requires both low power consumption and secure compact cryptography cores. In the future, we will further optimize the power consumption for the proposed AES encryption core and apply it for a real application.

Acknowledgments

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.02-2015.20.

This chip presented in this paper was fabricated in the chip fabrication program of VLSI Design and Education Center (VDEC), The University of Tokyo in collaboration with ROHM CO. LTD.

References

[1] Xiaojiang Du, Hsiao-Hwa Chen, Security in wireless sensor networks, IEEE Wireless Communications, vol.15, no.4, pp.60-66, Aug. 2008.

[2] J. Dofe, J. Frey, Q. Yu, Hardware security assurance in emerging IoT applications, IEEE Inter. Symp. Cir. and Syst. (ISCAS), pp. 2050–2053, 2016.

[3] National Institute of Standards and Technology (NIST), Advanced Encryption Standard (AES), FIPS Publication 197, Nov. 2001.

[4] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, A compact Rijndael hardware architecture with S-box optimization, Proc. ASIACRYPT 2001, pp.239-254, Dec. 2001.

[5] P. Hamalainen, T. Alho, M. Hannikainen, T.D. Hamalainen, Design and Implementation of Low-Area and Low-Power AES Encryption Hardware Core, Proc. 9th EUROMICRO Conf. Digital System Design: Architectures, Methods and Tools (DSD2006), pp.577-583, 2006.

[6] D. Canright. A very compact S-box for AES, Proc. 7th Int. Workshop on Cryptographic Hardware and Embedded Systems (CHES2005), pp.441-455, Sep. 2005.

[7] D. Canright and L. Batina, A Very Compact Perfectly Masked S-Box for AES, Proc. ACNS 2008, vol. 5037, LNCS, pp.446-459, Springer, 2008.

[8] T. Jarvinen, P. Salmela, P. Hamalainen, J. Takala, Efficient byte permutation realizations for compact AES implementations, Proc. 13th European on Signal Processing Conference, pp.1-4, Sep. 2005.

[9] K. Munusamy, C. Senthilpari, D.C.K. Kho, A low power hardware implementation of S-Box for Advanced Encryption Standard, Proc. 11th International Conference on Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp.1-6, May 2014.

[10] Mathew Sanu et al., 340 mV–1.1 V, 289 Gbps/W, 2090-gate nanoAES hardware accelerator with area-optimized encrypt/decrypt GF (2^4)^2 polynomials in 22 nm tri-gate CMOS, IEEE Journal of Solid-State Circuits 50.4, pp. 1048-1058, 2015.

[11] T. Good and M. Benaissa, 692-nW advanced encryption standard (AES) on a 0.13-µm CMOS, IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.18, no.12, pp.1753-1757, Dec. 2010.

[12] Wenfeng Zhao, Yajun Ha, Massimo Alioto, AES Architectures for Minimum-Energy Operation and Silicon Demonstration in 65nm with Lowest Energy per Encryption, 2015 IEEE International Symposium on Circuits and Systems (ISCAS), pp.1-4, May 2015.