DEU FMD 27(79) (2025) 152-159

Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering
Elektronik/Online ISSN: 2547-958X

RESEARCH ARTICLE / ARAŞTIRMA MAKALESI

An Energy-efficient Parallel ASIC Implementation of Advanced Encryption Standard (AES) Algorithm Robust against Side-channel Attacks

Gelişmiş Şifreleme Standardı (AES) Algoritmasının Yan-Kanal Saldırılarına Dayanıklı ve Enerji Verimliliği Yüksek Paralel ASIC Uygulaması

Serdar Ünal [1], Faik Başkaya [2*]

[1] Fraunhofer Institute for Integrated Circuits IIS, Fraunhofer-Gesellschaft, Erlangen, GERMANY
[2] Department of Electrical and Electronics Engineering, Boğaziçi University, İstanbul, TÜRKİYE
Corresponding Author / Sorumlu Yazar *: faik.baskaya@bogazici.edu.tr

Abstract

Encryption becomes more crucial than ever in an increasingly interconnected world. Advanced Encryption Standard (AES) is still considered secure after more than 20 years thanks to its mathematical properties. However, side-channel attacks (SCA) threaten improper AES implementations. In this paper, different AES implementations are introduced, and their resistances against power SCA, namely Correlation Power Analysis (CPA) attack, are shown. For energy efficiency, the increase in power consumption due to the extras added for countering SCA was minimized by register-level organizations and process-related optimizations. Different AES implementations were constructed and processed through Cadence ASIC flow (TSMC 65 nm LP technology). SCA resistance was evaluated using the ChipWhisperer platform operating on realistic power consumption values obtained after RTL-to-GDSII flow. The results demonstrate that pipelining and unrolling the AES rounds increase the SCA resistance at the expense of a minimal reduction in energy efficiency. The proposed implementations are suitable for use with different side-channel attack countermeasures.

Keywords: *ASIC Implementation of Advanced Encryption Standard (AES), Hardware Security, Digital CMOS Design, Side-Channel Attacks, Correlation Power Analysis (CPA), ChipWhisperer*

Öz

Şifreleme, giderek birbirine bağlanan bir dünyada her zamankinden daha önemli hale gelmektedir. Gelişmiş Şifreleme Standardı (AES), matematiksel özellikleri sayesinde 20 yıldan fazla bir süre sonra hala güvenli kabul edilmektedir. Ancak yan kanal saldırıları (SCA), uygunsuz AES uygulamalarını tehdit etmektedir. Bu çalışmada farklı AES uygulamaları tanıtılmakta ve bunların güç Yan-Kanal Saldırısı'na (SCA), spesifik olarak Korelasyon Güç Analizi (CPA) saldırısı, karşı dirençleri gösterildi. Enerji verimliliği açısından, yan-kanal saldırısına karşı yapılan eklemeler nedeniyle güç tüketiminde meydana gelen artış, yazmaç düzeyindeki organizasyonlar ve çip akışı bazlı optimizasyonlar ile minimuma indirildi. Farklı AES uygulamaları oluşturuldu ve Cadence ASIC akışı (TSMC 65 nm LP teknolojisi) aracılığıyla işlendi. Yan-Kanal Saldırısı direnci, RTL'den GDSII'ye çip akışından sonra elde edilen gerçekçi güç tüketimi değerleri üzerinde çalışan ChipWhisperer platformu kullanılarak değerlendirildi. Sonuçlar, AES turlarının boru hattına yerleştirilmesinin ve açılmasının (unroll), enerji verimliliğinde minimum azalma karşılığında Yan-Kanal Saldırısı direncini arttırdığını göstermektedir. Önerilen uygulamalar farklı Yan-Kanal Saldırısı savunma önlemleriyle kullanılmaya uygundur.

Anahtar Kelimeler: *Gelişmiş Şifreleme Standardı (AES) ASIC Uygulaması, Donanım Güvenliği, Dijital CMOS Tasarımı, Yan-Kanal Saldırıları, Korelasyon Güç Analizi (CPA), ChipWhisperer*

1. Introduction

As the world turns into a global village, communication between people around the world is increasing. A tremendous amount of information flows through many channels at every time instant. Although communication security has been an important issue since very old times, security concerns are growing more than ever with increasing communication volume. Encryption is a method that is used to eliminate or reduce security concerns. In encryption, the plain text messages are converted to cipher text messages using a cipher key so that an adversary who intercepts the unintelligible cipher text message from the communication channel cannot understand the real message. There are two types of encryptions, namely symmetric and asymmetric encryption. In symmetric encryption, the same key is used for both encryption and decryption, whereas in asymmetric encryption a different key is used for each of the encryption and decryption actions. The focus of this work is the power side-channel attack on AES algorithm, which is a symmetric encryption algorithm. AES algorithm was developed after a competition organized by the US National Institute of Science and Technology (NIST) in 2000. The winner of the competition was the Rijndael algorithm [1], which has been called AES since then, and it was published as Federal Information Processing Standard (FIPS) in 2001 [2]. Since its declaration as Type-1 Suite-B Encryption Algorithm, it has been accepted as suitable for securing classified and unclassified information worldwide [3]. AES algorithm contains four fundamental operations: SubBytes, ShiftRows, MixColumns, and AddRoundKey. These operations are combined to form a round

DOI: https://doi.org/10.21205/deufmd.2025277920

Geliş Tarihi / Received: 13.05.2024
Kabul Tarihi / Accepted: 28.07.2024

that is executed multiple times using round keys, which are obtained using KeyExpansion routine from the initial cipher key. For more than 20 years, AES has successfully resisted attacks directed against its mathematical structure [4]. However, side-channel attacks that utilize unintended leakages from the implementation such as power consumption, electromagnetic radiation, execution time, sound, etc., are also posing a threat to AES implementations. These leakages are used together with knowledge about the system to disclose secret information [5]. Numerous papers have been published about side-channel attacks on AES [6-8]. In this paper, an ASIC simulation environment that enables the evaluation of power side-channel resistance of an IC design before fabrication is introduced. In addition, the effects of architectural decisions such as pipelining the rounds or unrolling the rounds on the power side-channel attack resistance are also examined. The main objective of this work is to design a power side-channel attack resistant AES block. This paper is organized as follows. Section 2 provides the underlying theoretical principles of obtaining a side-channel attack resistant and energy-efficient design and the methods to achieve these objectives. Section 3 describes the implemented AES versions. Section 4 presents the design environments and evaluation of the implemented AES versions according to the design objectives. Section 5 concludes the paper.

2. Methodology

The applied methods for improving side-channel resistance and energy efficiency are explained in Section 2.1 and Section 2.2, respectively.

2.1. Applied methods to increase side-channel attack resistance

A power side-channel attack is the main focus of this paper. More specifically, Correlation Power Analysis (CPA), initially developed by Brier [9], was executed on the power traces collected from the simulation of the AES blocks. In general, CPA uses the Pearson correlation function to calculate the correlation values of different power traces [9,10]. The Pearson correlation function can be formulated as

$$C(T,P) = \frac{\mu(\mathrm{TP}) - \mu(\mathrm{T})\mu(\mathrm{P})}{\sqrt{\sigma^2(T)\sigma^2(P)}} \quad (1)$$

where T is the set of power traces, P is the set of estimated power values from the power model, μ is the population mean, and σ is the standard deviation. A set of power traces was obtained using the Cadence environment while running encryption on 500,000 random inputs. The estimated power consumption values according to the power model corresponding to each random input were calculated via ChipWhisperer's functions. Depending on the attack point, either Hamming Weight (HW) or Hamming Distance (HD) functions are used to model the corresponding power consumption for a given register state or register transition, respectively. HW is directly obtained by adding all the bit values in a register; therefore, having more 1's in the register bits means a higher power consumption value. On the other hand, HD is calculated as the number of bits that have flipped between two consecutive states of a register [11]. Since the bit values and bit changes give information about the processed information, the pipelining method was employed to mix power consumption values of different inputs with each other by taking advantage of processing different inputs simultaneously. Unrolling the AES rounds, i.e. making them a feedforward combinational path that spans from the first round to the last, was also employed in order to make state transitions less visible as all the rounds execute at the same clock period instead of waiting for the next clock edge.

2.2. Applied techniques for improving energy efficiency

Lowering the power consumption is the key to achieve energy efficiency. In this work, several techniques were applied to reduce power consumption. The consumed power is proportional to (α), which is the activity factor as shown in

$$P_{static} = I_{static} V_{DD} \quad (2)$$

$$P_{dynamic} = \alpha C V_{DD}^2 f + V_{DD} I_{sc} \quad (3)$$

$$P_{total} = P_{static} + P_{dynamic} \quad (4)$$

where P=power, I=current, V_{DD}=supply voltage, α =activity factor, C=load capacitance, f=switching frequency, I_{sc}=short circuit current [12]. As visible in Eq. (3), the activity should be decreased to reduce the dynamic power consumption. In order to accomplish this, RTL was coded in such a way that the switching activity of the modules was reduced by the enable signals and fixing of the module inputs. In addition, clock gating was enabled in ASIC flow to decrease the unnecessary switching activity. TSMC 65 nm via Europractice offers two types of processes: General Purpose (GP), and Low Power (LP) [13]. LP process was preferred in this work to further reduce the overall power consumption. Another approach for power reduction is the utilization of different threshold voltage standard cells. There are three standard cell types in the LP process according to the threshold voltage of transistors they are built from: low threshold voltage cells (LVT), standard threshold voltage cells (SVT), and high threshold voltage cells (HVT). Power consumption is proportional to the MOSFET current, which has a relation with the threshold voltage as explained in the following formula

$$i_D = \frac{1}{2} k_n' \left(\frac{W}{L}\right)(V_{GS} - V_t)^2 \quad (5)$$

where i_D is the MOSFET current in saturation, k_n' is the process transconductance parameter, (W/L) is the transistor aspect ratio, V_{GS} is the gate-to-source voltage, and V_t is the threshold voltage [14]. Decreasing the threshold voltage (V_t) increases i_D which means the transistor switches faster but at the cost of a higher current. The design tools were adjusted to favor the use of HVT cells over LVT and SVT cells whenever possible.

3. Constructed AES Versions

In this work, four different AES versions were constructed to observe their power side-channel attack resistance performances. RTL codes for all designs were simulated using xsim from Vivado Design Suite (version 2020.2). Vivado Design Suite was used only as an RTL simulator as the focus of this work is ASIC design instead of FPGA design. Test vectors from NIST "The Advanced Encryption Standard Algorithm Validation Suite" (AESAVS) [15] were used to validate the designs.

3.1. Rolled version

The first version is called the rolled version. This is the base version of AES, and it is used as a reference point in this work. The block diagram of the rolled version can be seen in Figure 1. The different blocks contained in each round are SubBytes, ShiftRows, MixColumns, and AddRoundkey, which are shown in Figure 2 in their processing order. In the rolled version, only one hardware round block is used for nine rounds, where each round takes one clock cycle. To complete the encryption, one more round that is missing the MixColumns block from the original round is required to process the 128-bit block. The rolled version

is compact; however, more cycles are required to complete each encryption. Besides, it does not contain any specific side-channel attack countermeasure.

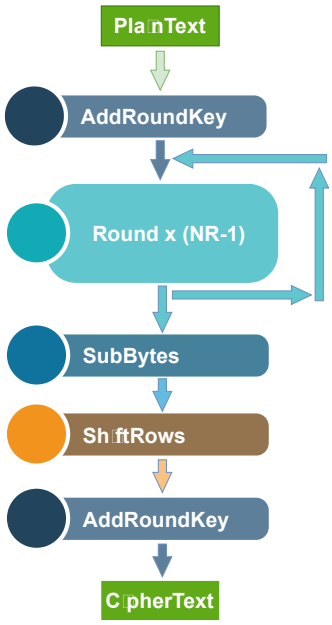

Figure 1. AES block diagram.

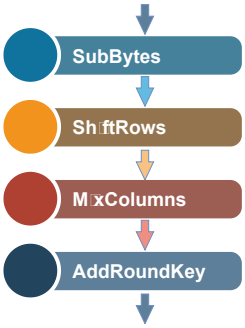

Figure 2. Block diagram of each AES encryption round.

3.2. Pipelined versions

Two different pipelined versions were implemented in this work. The pipelined versions are named 32-bit pipelined and 64-bit pipelined according to the bus widths of the input/output interfaces of the implementation, while both versions process 128-bit plain texts internally. Figure 3 depicts the block diagram of the 32-bit pipelined version. In this implementation, the nine rounds of the rolled version are executed in three identical blocks, which also serve as stages of the pipeline, and the remaining processing is completed combinationally with AddRoundkey, SubBytes, ShiftRows, and AddRoundkey blocks. Figure 4 describes a detailed view of these blocks, which are called "main round" in this work.

Main round block of the pipelined version consists of the same blocks as the round block of the rolled version; however, AddRoundkey block has the first order here as opposed to the last order in the rolled version round block. Main round block also has two inputs and two outputs as opposed to the single-input single-output round block of the rolled version. Every clock cycle, one of the two inputs is routed into the 128-bit input register of the main round block, selected by the 2-to-1 multiplexer. Using this multiplexer, main round block either accepts a new input from the previous stage of the pipeline or uses its own output as its new input, effectively processing the same input for another round. Multiplexer select input is received from a simple shift register called S_{count}, which is four bits for the 32-bit pipelined version and two bits for the 64-bit pipelined version; thus, receiving a new input from the previous pipeline stage once every four cycles or once every two cycles, respectively. This means that every main round block processes a 128-bit input for four cycles if it is a 32-bit pipelined implementation and for two cycles if it is a 64-bit pipelined implementation. This decision for the number of cycles per stage was made based on the bus width; receiving the 128-bit input plain text takes four clock cycles and two clock cycles with a 32-bit bus and 64-bit bus, respectively.

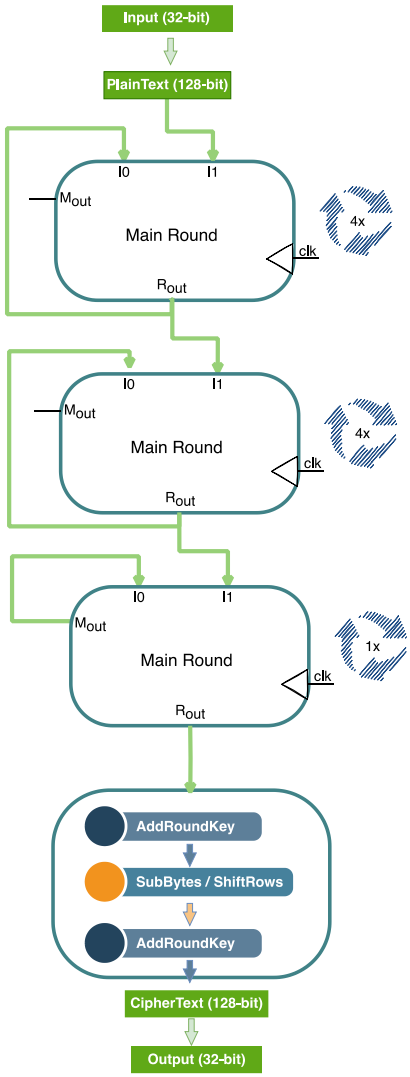

Figure 3. 32-bit pipelined AES implementation.

Since each main round block processes a given input for four cycles or two cycles before forwarding the output to the next stage, three main round blocks are required for the 32-bit pipelined implementation and five main round blocks are required for the 64-bit pipelined implementation. The last main round block in both implementations process only one round of the 9 total rounds. This irregularity is handled by bypassing the processing blocks in the last main round blocks for the cycles a new input is not being received using the M_{out} output port instead of R_{out}. Thanks to M_{out} & R_{out} combination, design simplicity was obtained by using the same main round block despite the mentioned irregularity for the last stage.

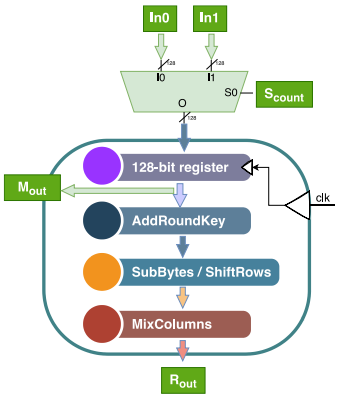

Figure 4. Block diagram of 32-bit pipelined AES implementation main round.

The pipelined configuration allows concurrent processing of three or five inputs in the 32-bit or 64-bit versions, respectively. This increases the throughput and the side-channel attack resistance because of the power consumption interference of individual inputs with each other at the expense of additional hardware and power consumption. A new cipher text can be obtained once every four or two cycles using the 32-bit or 64-bit pipelined configurations, respectively.

3.3. Unrolled version

In the unrolled version, each round of the rolled version is implemented as a separate block with no hardware reuse, and there are no registers between the blocks of different rounds, which effectively turns the entire AES into a purely combinational circuit that can encrypt the input plain text in a single clock cycle. Since the encryption operation fits into one clock cycle, the unrolled version requires a longer clock period. In addition, it occupies the largest area among all versions because of the high number of hardware blocks used. The unrolled version uses a 32-bit interface that is similar to the interfaces of the rolled and 32-bit pipelined versions.

4. Design Environments and Evaluation

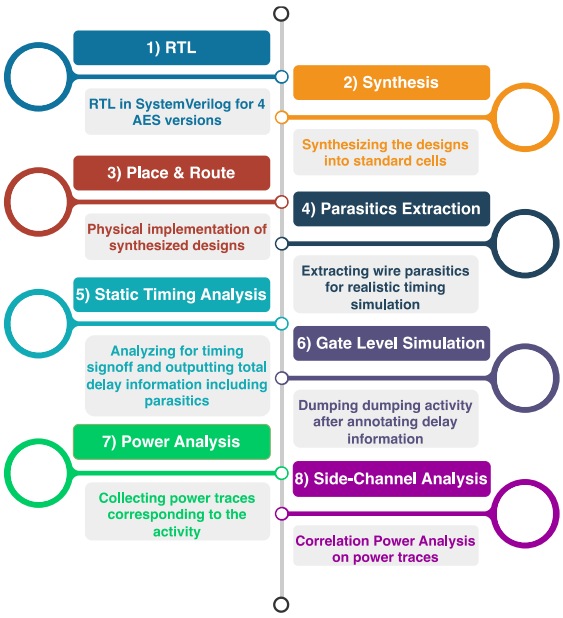

Figure 5. Experiment flow.

All four AES versions explained in Section 3 were processed through the ASIC flow using Cadence tools. ChipWhisperer from NewAE Technology, which is an open-source toolchain containing target hardware, capture hardware, firmware, and software, was used for side-channel attack resistance evaluation. Since power results were obtained by simulation outputs, only the analyzer [16] module of ChipWhisperer was used in this work. The experiment flow summarized in Figure 5 is explained in detail in Section 4.1, Section 4.2, and Section 4.3.

4.1. ASIC environment

TSMC 65 nm LP technology was used for the RTL-to-GDSII flow of different AES implementations. The versions and purposes of the used IC tools are as follows: Genus Synthesis Solution (v19.11) for synthesis, Innovus Implementation System (v19.11) for Place & Route (P & R), Quantus Extraction Solution (v19.1.3) for RC parasitics extraction, Tempus Timing Signoff Solution (v19.11) for Static Timing Analysis (STA), Xcelium Logic Simulation (v19.03) for gate-level simulation, Voltus IC Power Integrity Solution (v19.11) for power analysis.

There is a single functional mode and five different Process-Voltage-Temperature (PVT) corners used in this work, which can be found in Table 1. The corners are listed as Best Case (BC), Low Temperature (LT), Typical Case (TC), Worst Case (WC), and Worst Case Low Temperature (WCL). These PVT corners were combined with the RC extraction corners to obtain the delay corners. The operating frequencies of the designs were chosen to satisfy timing constraints for all the mentioned delay corners after Multi-Mode Multi-Corner (MMMC) analysis.

Table 1. Process, Voltage & Temperature (PVT) corners.

Corners	Process	Voltage (Volts)	Temperature (°C)
BC	FF	1.32	0
LT	FF	1.32	-40
TC	TT	1.2	25
WC	SS	1.08	125
WCL	SS	1.08	-40

For each version, the P & R tool is configured to use 60% standard cell density; therefore, the resulting area is (10/6) times the combined area of the standard cells obtained from the synthesis stage. Six out of nine available metal layers were used in this work as the blocks were designed to be IP's suitable to be integrated into top-level System-on-Chip (SoC).

Design statistics for all versions after P & R can be inspected in Table 2, where logical instances values increase from left to right implying increased hardware usage from the rolled version to the pipelined versions, and to the unrolled version. The unrolled version requires a higher clock period because of the combinational nature of the design completing the entire encryption in a single clock cycle. Positive Worst Negative Slack (WNS) and Worst Hold Slack (WHS) ensure that the signals do not arrive too late or too early so that the flip-flops can receive the correct values. The tool was configured to use HVT cells as much as possible since they consume less power, as explained in Section 2.2. Yet, some LVT cells were still required in the timing-critical paths. The HVT proportion is greater than 50% for all four versions, in line with the energy-efficient target. The layout of the 32-bit pipelined version at the end of the P & R can be seen in Figure 6, which was partitioned to show the approximate locations of the major components of the design. Different colors in the layout correspond to different metal layers. The seemingly regular pattern of nets shows power and ground lines. The yellow

arrows at the perimeter of the block correspond to inputs and outputs at the interface of the block.

Table 2. Design statistics after P & R.

	Rolled	Pipe. 32	Pipe. 64	Unrolled
Logical Inst.	20,808	35,986	50,809	80,062
Period (ns)	8	8	8	40
WNS (ns)	0.297	0.086	0.264	0.124
WHS (ns)	0.094	0.090	0.080	0.056
LVT (%)	4.3	11.5	10.0	20.1
SVT (%)	2.2	10.2	7.9	21.4
HVT (%)	93.4	78.3	82.1	58.4

4.2. Trace collection

A test bench containing encryption operation of 500,000 random inputs with the cipher key 128'h2b7e151628aed2a6abf7158809cf4f3c was used for the simulation. The gate-level simulation was made after annotating the gate-level netlist received from Innovus with cell delays and interconnect delays using information from Quantus and Tempus tools so that the parasitic effects are taken into account and the simulation is more realistic. Activity information resulting from the test bench was dumped in multiple Value Change Dump (VCD) files from Xcelium tool, which store the switching activity of nets by recording the times each net switch at. It is important to annotate switching information during power analysis to obtain realistic power consumption values. The power analysis was done at the typical corner. The current drawn by the AES blocks while encrypting 500,000 random inputs was dumped with 50 ps time steps during power analysis in Voltus tool, which provides enough precision for the side-channel attack analysis of both 8 ns (rolled and pipelined) and 40 ns (unrolled) period cases. The current values were interpreted as power values assuming constant supply voltage. The output text files were collected to be used in the power side-channel attack resistance evaluation. To give an insight into the time required to collect power traces, the average Voltus runtime for power analysis of 100,000 input encryption is 5.84 days for the rolled version in a server.

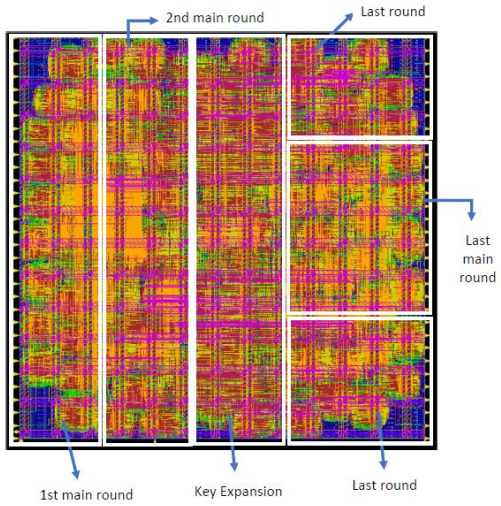

Figure 6. 32-bit pipelined AES implementation layout.

4.3. Side-channel attack resistance evaluation

The outputs from the Voltus tool were prepared to be used by the ChipWhisperer tool, which was used to evaluate side-channel attack resistance from the generated traces. 500,000 input values, 500,000 key values (all the same 128-bit value), and 500,000 trace sets are stored as MATLAB arrays. Each trace set corresponds to the time interval where their respective input is encrypted. These arrays are converted into NumPy arrays and transferred to the ChipWhisperer program. The program uses Correlation Power Analysis (CPA) attack by utilizing the Pearson correlation function [17], as explained in Section 2.1 The software model of AES is already available in the program. The power consumption is modeled by calculating Hamming Weight or Hamming Distance according to the bits of the intermediate values that are generated by the functions of the AES software model. For a particular leakage model (e.g. AddRoundKey output), the power consumption is calculated targeting that point, and the result is compared with the actual traces for every possible key byte combination. Different byte values are tested and ranked according to the correlation value coming from Eq. (1). If the model and the actual power traces are consistent with each other, the correlation value increases. There are many different leakage models targeting different parts of the AES algorithm. The attack point in time is also detected using correlation. As the other time instants give a low correlation with the guessed key byte combination while comparing traces belonging to the different outputs, the correct attack time becomes pinpointed with a high correlation value [18]. To give an insight about the time required to do side-channel analysis, the ChipWhisperer runs for approximately 31.5 hours to analyze 400,000 traces of rolled version for the round_1_2_state_diff_sbox leakage model in a desktop computer with 32 GB RAM and 12 CPU threads. ChipWhisperer results for the rolled version with 500,000 random inputs and round_1_2_state_diff_sbox leakage model can be seen in Table 3, where five most likely guesses for every byte of the cipher key are shown. The guesses that matched the original cipher key were highlighted with a yellow color.

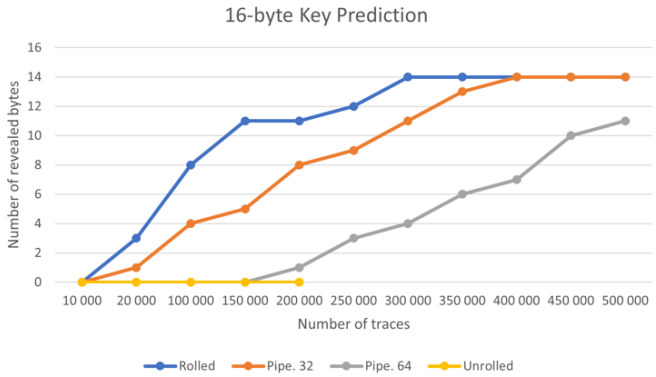

Figure 7. Results of side-channel attacks on all AES versions.

ChipWhisperer results with eleven different input spaces (10,000 - 50,000, - 100,000 - 150,000 - 200,000 - 250,000 - 300,000 - 350,000 - 400,000 - 450,000 - 500,000) can be observed in Figure 7. The revealed bytes are the correct bytes that are visible in the first row of their tables similar to the presented values in Table 3. That is, if the correct byte is the most-correlated guess for a particular byte of the cipher key, it is counted into the number of revealed bytes. Figure 7 compares different AES versions according to the round_1_2_state_diff_sbox leakage model, which appears to be the most successful attack for the collected traces. Different leakage models from ChipWhisperer were also tried but after a small number of inputs round_1_2_state_diff_sbox leakage

model starts to dominate and becomes the most effective leakage model for different versions. Therefore, side-channel attack resistances of all four versions were compared using the round_1_2_state_diff_sbox leakage model, which focuses on the Hamming distance between the first and the second round SubBytes outputs. It should be noted that the trace number in the analysis of the unrolled version was limited to 200,000 in these tests, since its larger area & higher standard cell count makes trace collection & side-channel attack testing prohibitively resource-intensive. The worst performing version is the rolled version as expected since it does not contain any specific side-channel attack countermeasure. In general, more bytes are revealed as the input size is increased. However, the number of revealed bytes may not change even after a large number of traces, since some bytes are revealed easily, and some bytes need many more traces to be disclosed as can be seen in results where the number of revealed bytes appears as saturated indicating remaining bytes need much more traces. The pipelined versions perform better with a smaller number of revealed bytes compared to the rolled version. The 64-bit pipelined version is more resistant than the 32-bit pipelined version, as expected. The best-performing version is the unrolled version when the comparison is made at 200,000 traces. The purely combinational encryption approach of the unrolled version leaves little space for attackers, which comes at the cost of a significant increase in area.

Table 3. Byte prediction results after 500,000 traces collected from the rolled AES version.

f	*e*	*d*	*c*	*b*	a	9	8	7	6	5	4	3	2	1	0
2b	7e	15	16	28	9e	c2	a6	ab	f7	15	88	09	cf	4f	3c
31	9a	c1	db	22	ae	62	3b	4e	43	f1	cf	3e	5c	57	02
8f	db	17	3d	53	f3	c5	30	05	2d	c9	a6	74	ab	42	7c
05	7c	aa	3c	71	4d	d2	64	84	7b	3a	3d	e7	48	1f	a7
9b	14	6d	a4	d0	3f	ff	b6	3b	28	b1	48	9b	3c	2b	78

The unrolled and pipelined designs are compared against six designs from the literature in Table 4. The designs from the literature contain different countermeasures for the AES against the power SCA. The effectiveness of our unrolled and pipelined implementations was evaluated comparatively with these countermeasures using Power-Performance-Area (PPA) and SCA resistance metrics. Four of the designs were fabricated and side-channel attacks were executed on the hardware [19-21, 23]. On the other hand, the design Ref. [22] used simulation to evaluate side-channel attack resistance similar to our work. The design Ref. [24] was fabricated but the results are from simulation. Measurements-to-Disclosure (MTD) in Table 4 is the number of traces required for differentiating the correct secret key from all wrong key guesses [25]. It can be reported as the number of traces needed to disclose either one byte from the cipher key or all bytes of the cipher key. In this work, the all-bytes approach was chosen, i.e., "≥ 500K" means, 500,000 traces were collected, analyzed, and still not all of the bytes were disclosed after 500,000 traces. Trace numbers were restricted to 500,000 as collecting traces in simulation with numbers comparable to attacks executed directly on hardware is impractical. Besides, fewer traces in a noise-free simulation environment gives enough insight into the design compared to the noisy hardware setup where the noise requires more traces to be collected for revealing the key. It is reported in Ref. [26] that 160 times more measurements were necessary for a real hardware attack compared to the simulated attack. Therefore, MTD values found in simulations should be scaled by such a large number for a fair comparison of simulation and hardware results unless advanced techniques are employed for reducing noise in hardware setups. In addition, the design Ref. [22] in Table 4, which use simulation to evaluate side-channel resistance, do not report MTD value at all. There is also Ref. [27] that again uses simulation to evaluate SCA resistance against a countermeasure. It has high throughput, but it was not added to the table as it lacks area, power, and MTD information. Ref. [28] uses the same ChipWhisperer Analyzer module but in addition to being done on the FPGA, it also lacks area and power information.

4.4. Energy efficiency evaluation

Energy efficiencies of different designs are compared via a Figure of Merit (FoM) metric, which has a unit of (Gbps/mW). The metric was developed to enable a fair comparison between different designs in different technologies as smaller technology nodes enable higher frequency designs which also result in higher dynamic power consumption. The FoM is calculated as follows:

$$FOM = \frac{throughput(Gpbs)}{power\ consumption(mW)} \tag{7}$$

According to the results reported in Table 4, The 64-bit pipelined version has a throughput greater than Refs. [19], [21], [22], [23], and [24]. Ref. [20] has a throughput higher than the 64-bit pipelined version, but its power consumption is more than three times higher. In terms of area, pipelined versions are the smallest except Ref. [22], [24] which are in the advanced nodes where transistor sizes are smaller. The number of gates in Ref. [22] is also fewer indicating a compact design; however, with less throughput and higher power consumption compared to the 64-bit pipelined version. Operating frequencies in this work are suitable for applications in different domains. Since the frequencies are not very high, the power consumption is smaller and the design does not require special very high-frequency signal handling issues when integrated into the top-level designs while still providing comparable throughput. As far as energy efficiency is concerned, pipelined designs have a higher FoM compared to the unrolled version as well as the designs from the literature except Ref. [24]. However, the design in Ref. [24] has much lower MTD value compared to the pipelined versions. The rolled version has a slightly higher FoM than the pipelined versions; however, it does not contain any specific power side-channel attack countermeasures and it tends to reveal more bytes than the pipelined and unrolled versions as demonstrated in Figure 7.

Table 4. Comparison of different AES implementations.

	Rolled	*Unrolled*	*Pipe. 32*	*Pipe. 64*	[19]	[20]	[21]	[22]	[23]	[24]
Technology (nm)	65	65	65	65	130	65	180	22	130	16
Area (mm^2)	0.134	0.415	0.208	0.278	1.37	0.291	0.67	0.0169	N.A	0.0012
Gates	21K	80K	36K	51K	N.A	N.A	N.A	16K	N.A	N.A
Power (mW)	5.0	21.5	16.0	32.2	44.34	98	12	41.6	11.02	0.08
Clock Cycles	10	1	4	2	11	10	11	10	10	204
Frequency (MHz)	125	25	125	125	110	1320	24	400	38.8	300
Throughput (Gbps)	1.6	3.2	4	8	1.28	16.9	0.28	5.12	0.50	0.18
MTD	≥ 500K	≥ 200K	≥ 500K	≥ 500K	≥ 10M	940K	≥ 800K	N.A	≥ 500K	15983
FoM (Gbps/mW)	0.32	0.15	0.25	0.25	0.03	0.17	0.02	0.12	0.05	2.25

5. Conclusions

Four different implementations were analyzed in this work to address the side-channel attack resistance of the AES algorithm. The rolled version is the baseline implementation where a single hardware unit is used to complete all nine rounds, one round at a clock cycle. This is used as a reference point in this work. The following two versions are 32-bit pipelined and 64-bit pipelined implementations containing additional hardware to allow more than one input to be processed simultaneously within the same clock cycle. Pipelined versions enhance resistance to side-channel attacks through increased parallelism via processing consecutive input samples simultaneously and mixing the sum of their power consumption values. Since the attacker can observe only the total power consumption, calculating correlation due to distinct inputs becomes more difficult, increasing the MTD values. The last version is the unrolled version containing purely combinational rounds with no hardware reuse. Since there is no clock transition between the different AES steps, the unrolled version benefits from a complex, non-repetitive power profile, and it becomes more difficult to differentiate the different steps from each other in the power traces. In all proposed cases; namely, the unrolled and pipelined versions, the unique power trace characteristics of each step in the standard AES implementation are obfuscated by processing multiple inputs or multiple steps simultaneously. Therefore, correlation power analysis approach effectiveness is reduced, resulting in lower correlation values.

Taking the power consumptions of the different approaches into account, pipelined versions provide a reasonably strong resistance against side-channel attacks at a reasonably low power consumption, resulting in the best FoM for 65nm or earlier technology nodes. In order to improve energy efficiency, RTL was coded to favor lower switching activity. In addition, TSMC Low Power (LP) process was chosen, and higher threshold standard cells were used as long as the timing constraints were satisfied.

All versions were designed from RTL-to-GDSII in TSMC 65 nm using Cadence tools, and the power traces obtained from Voltus tool were evaluated in ChipWhisperer program against Correlation Power Analysis (CPA). The unrolled version performed the best side-channel attack resistance at the expense of the highest area and logical standard cell count. Thanks to the efficient utilization of hardware blocks, 32-bit and 64-bit pipelined versions achieved high throughput with a lower area compared to the unrolled version. Their 16.0 mW and 32.2 mW power consumptions stand lower compared to the existing designs in the literature as well. It was demonstrated that pipelined and unrolled implementations of the AES algorithm have higher side-channel attack resistances compared to the baseline rolled implementation of AES at the expense of a minor reduction in energy efficiency.

Ethics committee approval and conflict of interest statement

This article does not require ethics committee approval. This article has no conflicts of interest with any individual or institution.

Acknowledgment

This research was conducted while the first author was an employee of TUBITAK BILGEM. We thank TUBITAK BILGEM for allowing us to use the ASIC tools required to complete this work.

Author Contribution Statement

Faik Baskaya: project administration, conceptualization, supervision, writing (review & editing). Serdar Unal: conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, visualization, writing (original draft).

References

[1] Daemen, J., Rijmen, V. 2000. The Block Cipher Rijndael. In J.-J. Quisquater, B. Schneier ed. Smart Card Research and Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 277–284. DOI: https://doi.org/10.1007/10721064_26

[2] National Institute of Standards and Technology. 2016. Cryptographic Standards and Guidelines AES Development. https://csrc.nist.gov/projects/cryptographic-standards-and-guidelines/archived-crypto-projects/aes-development (Accessed: 23.07.2022).

[3] Alghazzawi, D. M., Hasan, S. H., Trigui, M. S. 2014. Advanced Encryption Standard - Cryptanalysis research. 2014 International Conference on

Computing for Sustainable Global Development (INDIACom), pp. 660–667. DOI: 10.1109/IndiaCom.2014.6828045

[4] Socha, P., Brejník, J., Bartik, M. 2018. Attacking AES implementations using correlation power analysis on ZYBO Zynq-7000 SoC board. 2018 7th Mediterranean Conference on Embedded Computing (MECO), pp. 1–4. DOI: 10.1109/MECO.2018.8406034

[5] Zhou, Y., Feng, D. 2005. Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing. IACR Cryptol. ePrint Arch., 388. http://eprint.iacr.org/2005/388 (Accessed: 23.07.2022).

[6] Ghandali, S., Ghandali, S., Tehranipoor, S. 2021. Deep K-TSVM: A Novel Profiled Power Side-Channel Attack on AES-128. IEEE Access, Vol. 9, pp. 136448–136458. DOI: 10.1109/ACCESS.2021.3117761

[7] Mushtaq, M., Akram, A., Bhatti, M. K., Rais, R. N. B., Lapotre, V., Gogniat, G. 2018. Run-time Detection of Prime + Probe Side-Channel Attack on AES Encryption Algorithm. 2018 Global Information Infrastructure and Networking Symposium (GIIS), pp. 1–5. DOI: 10.1109/GIIS.2018.8635767

[8] Guo, S., Zhao, X., Zhang, F., Wang, T., Shi, Z. J., Standaert, F.-X., Ma, C. 2014. Exploiting the Incomplete Diffusion Feature: A Specialized Analytical Side-Channel Attack Against the AES and Its Application to Microcontroller Implementations. IEEE Transactions on Information Forensics and Security, Vol. 9(6), pp. 999–1014. DOI: 10.1109/TIFS.2014.2315534

[9] Brier, E., Clavier, C., Olivier, F. 2004. Correlation Power Analysis with a Leakage Model. In M. Joye, J.-J. Quisquater ed. Cryptographic Hardware and Embedded Systems - CHES 2004. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 16–29. DOI: https://doi.org/10.1007/978-3-540-28632-5_2

[10] Kundrata, J., Fujimoto, D., Hayashi, Y., Barić, A. 2020. Comparison of Pearson correlation coefficient and distance correlation in Correlation Power Analysis on Digital Multiplier. 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), pp. 146–151. DOI: 10.23919/MIPRO48935.2020.9245325

[11] Brown, S. D., Vranesic, Z. G. 2012. Fundamentals of Digital Logic with VHDL Design. 3rd edition. McGraw Hill Education, p. 624.

[12] Weste, N. H. E., Harris, D. 2005. CMOS VLSI Design: A Circuits and Systems Perspective. 3rd edition. Pearson Education, pp. 188-191, 196.

[13] TSMC Technologies. https://europractice-ic.com/technologies/asics/tsmc/ (Accessed: 23.07.2022).

[14] Sedra, A. S., Smith, K. C. 2011. Microelectronic Circuits. 6th edition. Oxford University Press, New York, pp. 362-366.

[15] Bassham, L. E. 2002. The Advanced Encryption Standard Algorithm Validation Suite (AESAVS). https://csrc.nist.gov/CSRC/media/Projects/Cryptographic-Algorithm-Validation-Program/documents/aes/AESAVS.pdf (Accessed: 23.07.2022).

[16] NewAE Technology. 2022. Analyzer. https://chipwhisperer.readthedocs.io/en/latest/analyzer-api.html (Accessed: 08.05.2023).

[17] NewAE Technology. 2018. Correlation Power Analysis. https://wiki.newae.com/Correlation_Power_Analysis (Accessed: 23.07.2022).

[18] O'Flynn, C. 2016. Introduction to Side-Channel Power Analysis (SCA, DPA). https://www.youtube.com/watch?v=OlX-p4AGhWs (Accessed: 11.05.2023).

[19] Tokunaga, C., Blaauw, D. 2009. Secure AES Engine with a Local Switched-Capacitor Current Equalizer. 2009 IEEE International Solid-State Circuits Conference - Digest of Technical Papers, pp. 64-65,65a. DOI: 10.1109/ISSCC.2009.4977309

[20] Lu, S., Zhang, Z., Papaefthymiou, M. 2015. 1.32GHz High-Throughput Charge-Recovery AES Core with Resistance to DPA Attacks. 2015 Symposium on VLSI Circuits (VLSI Circuits), pp. C246–C247. DOI: 10.1109/VLSIC.2015.7231274

[21] Miura, N., Fujimoto, D., Korenaga, R., Matsuda, K., Nagata, M. 2014. An Intermittent-Driven Supply-Current Equalizer for 11x and 4x Power-Overhead Savings in CPA-Resistant 128bit AES Cryptographic Processor. 2014 IEEE Asian Solid-State Circuits Conference (A-SSCC), pp. 225–228. DOI: 10.1109/ASSCC.2014.7008901

[22] Chou, Y.-H., Lu, S.-L. L. 2019. A High Performance, Low Energy, Compact Masked 128-Bit AES in 22nm CMOS Technology. 2019 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), pp. 1–4. DOI: 10.1109/VLSI-DAT.2019.8741835

[23] Kar M., Singh A., Mathew S. K., Rajan A., De V., Mukhopadhyay S. 2018. Reducing Power Side-Channel Information Leakage of AES Engines Using Fully Integrated Inductive Voltage Regulator. IEEE Journal of Solid-State Circuits, Vol. 53(8), pp. 2399-2414. DOI: 10.1109/JSSC.2018.2822691

[24] Dhanuskodi, S. N., Holcomb, D. 2019. Enabling Microarchitectural Randomization in Serialized AES Implementations to Mitigate Side Channel Susceptibility. 2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Miami, FL, USA, pp. 314-319. DOI: 10.1109/ISVLSI.2019.00064

[25] Tiri, K., Hwang, D., Hodjat, A., Lai, B.-C., Yang, S., Schaumont, P., Verbauwhede, I. 2005. Prototype IC with WDDL and Differential Routing -- DPA Resistance Assessment. In J. R. Rao, B. Sunar ed. Cryptographic Hardware and Embedded Systems -- CHES 2005. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 354–365. DOI: https://doi.org/10.1007/11545262_26

[26] Ors, S. B., Gurkaynak, F., Oswald, E., Preneel, B. 2004. Power-Analysis Attack on an ASIC AES Implementation. International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004., Vol. 2, pp. 546-552. DOI: 10.1109/ITCC.2004.1286711

[27] Peng, Y., Zhao, H., Sun, X., Sun, C. 2017. A Side-Channel Attack Resistant AES with 500Mbps, 1.92pJ/Bit PVT Variation Tolerant True Random Number Generator. 2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), pp. 249–254. DOI: 10.1109/ISVLSI.2017.51

[28] Lagasse, J., Bartoli, C., Burleson, W. 2019. Combining Clock and Voltage Noise Countermeasures Against Power Side-Channel Analysis. 2019 IEEE 30th International Conference on Application-Specific Systems, Architectures and Processors (ASAP), pp. 214–217. DOI: 10.1109/ASAP.2019.00009