AES Engine with Current Equalizer and Random Clock Generation

EECS 627 Final Report

Yuchen Fan, Ruohan Luo, Yixin Ma, Yuxiang Mu, Ziyou Su

Abstract - **This report presents an AES encryption engine against power side-channel attacks. The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data accepted worldwide. Hardware implementation of it gives low latency and high throughput but leaks important side channel information that can be employed to extract the secured encryption key. We propose incorporation of a switch-capacitor current equalizer, which isolates the power supply of the cryptographic unit from external voltage sources, as well as random clock generation, which adds random jitters to the clocks. The scheme increases the non-determinism of power traces and minimizes the power leakage, thereby enhancing the ability of the physical device to resist against attack.**

I. Introduction

Security is a significant design objective in modern technology, and becomes especially crucial in recent years with the rapid growth of mobile devices and wireless communication. Dedicated low power ASIC encryption engines are integrated in varieties of devices, ranging from smartcards to mobile and desktop processors. Despite the high throughput and low latency it can provide, the physical implementation of the AES engine has a non-negligible weakness. The AES core leaks physical information called side-channel information including power consumption, timing information, electromagnetic emission and even acoustics, which can be exploited to perform side channel attacks to recover the secret key. The differential power analysis (DPA), which employs statistical analysis of the measured power traces, is a major technique used to perform side channel attack.

Hardware countermeasures have been developed since the disclosure of power side channels. The basic principle is to conceal the real power consumption properties of the cryptographic units. The earliest protection plans employ masked logic gates [1] in order to break the correlation between the secured key values stored in hardware and the intermediate processed values. The gate-level masking method, however, can result in drastic increase in gate count and leads to increase in area and total device power consumption, which is not desired by modern AES cores. Later countermeasures employ dual-rail precharge (DRP) [2] logic that makes the output switching activities independent of inputs to achieve constant power consumption. However, dual-rail logic introduces higher than 3x overheads in area, power and performance, and its effectiveness significantly relies on manufacture process variations. Another commonly used approach against DPA is noise generation. In the time domain, the alignment of the power traces can be broken by randomly inserting dummy instructions; in the amplitude domain, using long wires with many buffers with controlled switching activities [3] can generate coupling noise and thus increase the randomness of the power traces. The price to pay for noise generation using these two methods are degradation in throughput and increase in power consumption.

In this project, we propose an ASIC AES core against DPA by incorporating a switch capacitor current equalizer [4] and random clock generation [5]. The switch capacitor current equalizer uses an embedded capacitor to supply current to the critical logic, hence isolates the power line from the cryptographic unit; and the randomly generated clocks further introduces noise to the cipher blocks. By combining these two countermeasures, we aim to increase the ability of the AES engine to resist differential power attacks without much degradation in area and performance. An unprotected AES core is constructed to reveal its vulnerability against DPA attacks. The power traces as well as the correlation patterns of the protected AES core are further compared to those of the unprotected AES core to prove the functionality and improvements of the new design.

The rest of the paper is as follows. In Section II, we present the overall architecture of the secured AES core. In Section III, we provide detailed descriptions of the switch capacitor current equalizer as well as the random clock generator. In Section IV, experiment results of DPA attack on both the unprotected and the secured AES cores are analyzed. In section V, we discuss the functionality as well as physical design constraints verification process. Lastly, in Section VI, we conclude the work.

II. Architecture

The overall architecture of the project is shown in Fig. 1, which consists of three main components: the baseline unprotected AES core, the improved switched capacitor current equalizer and a random clock generator.

The baseline AES core realizes the encryption process. It takes one 8-bits input for plaintext and one 8-bits input for cipher key. It takes 16 cycles to read in the entire 128-bit

plaintext and 128-bit cipher key through a SIPO unit. There will be one set-up round as well as ten rounds for the encryption process where each round contains four procedures, including *SubBytes*, *ShiftRows*, *MixColumns* and *AddRoundKey* except for the last round that does not need step *MixColumns*. After last round, there will be another 16 rounds to output 128-bit ciphertext through a PISO unit, with 8-bit output each cycle. In this mechanism, since we process the current encryption and read in the plaintext and key for next encryption at the same time, therefore the average time for an complete encryption process is 16 cycles.

In order to prevent the leakage of the side channel information, we added the on-chip current equalizer. The ring oscillator produces a high frequency clock for current equalizer. *Vdd_critical* produced by the current equalizer will be supplied to the initial, first round and the last round blocks to prevent power leakage, while the blocks in other rounds are still supplied by the supply voltage VDD.

We added a random clock generator to increase the uncertainty of the signal arrival times at sequential elements in the circuit, which is also a good protection against the side-channel attack. The ring oscillator generates a primary clock for normal round. The clock generator generates a series of clocks with different delay. A random select signal is obtained from PRNG for MUX array to select one of the produced clock for each path and provide them to initial setup, first round and last round.

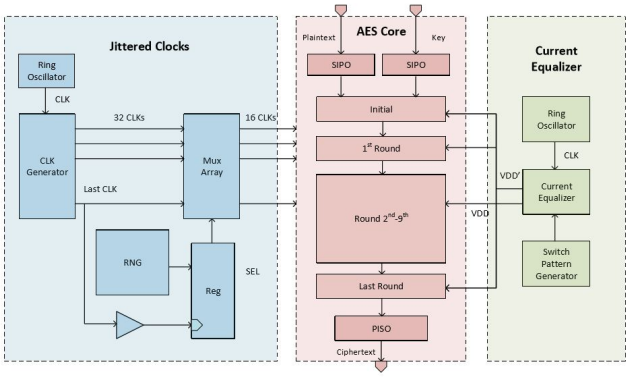

Fig. 1. Overall architecture of AES engine with features

III. Design Description

A. Improved Current Equalizer

The current equalizer (Fig. 2.) introduced in Tokunaga paper [4] has three switches (phases): supply, logic and shunt, and three modules, one in each phase (i.e. the corresponding switch is on) in rotation. Using this block, we can isolate the power supply because AES logic directly gets power from the internal storage cap. It also equalizes the charging current drawn from power supply by pre-discharging the internal cap to ground during the shunt phase, which can increase difficulty of side-channel power attack, and thus make the encryption process more secure.

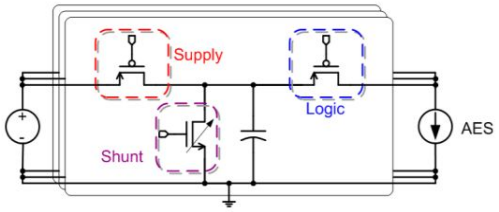

Fig. 2. Current equalizer from previous work [4]

Since it's not possible for us to determine the ability of our AES against DPA before top level APR is done, we had to set up experiments (Fig. 3.) to analyze security of the block during implementation. People usually conduct attacks at power supply, so we gave a current pulse at *Vdd_critical* end with other variables controlled and observe the corresponding current changes at Vdd_supply end. The current leaked to Vdd_supply end somewhat reflects the time and amplitude of the input current, so it might leak information about calculations in AES logic.

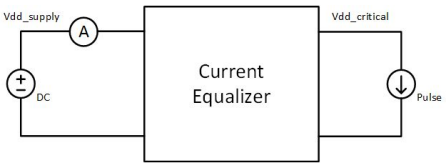

Fig. 3. Experiments setup

From our experiments, we found that leakage worsens as the internal voltage decreases at the storage capacitor and such voltage drop will also slow down our AES logic. Hence, we must maintain *Vdd_critical* voltage level around true Vdd during the logic phase. AES will consume a current at milliamperes from power supply according to FastSpice result, which means the storage cap must be large. Tokunaga [4] used 100pF cap, we found it also applies to our typical current consumption. Then, driving strength is important because of the large cap and fast clock. Low Vth switches work best because it provides largest ON current. We considered to use stacking to reduce sub-threshold leakage like [7]. However, since speed is key to the function of this block, we had to size up the stacking transistors which made it not effective in leakage saving. Besides subthreshold leakage, body leakage is also large. To reduce the large body leakage, we tried to isolate the N-wells.

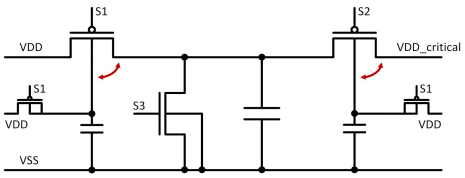

Fig. 4. 3-switch topology with isolated body

Fig. 4. is our final topology of a single module. The two arrows show where the large body leakages happen. We tied the two N-wells to smaller caps charged by smaller transistors. The body voltages can be kept around true Vdd because the smaller caps are large enough to handle the body leakage current. In this way, body leakage current will not flow to true Vdd directly.

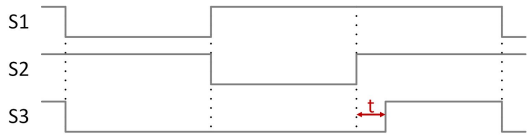

Fig. 5. Switch pattern

The control signals (Fig. 5.) of the five switches are produced by pattern generator. When EN is set, the current equalizer works. Here, we produced a delay at S3, the shunt phase, to prevent current flowing from another working module to ground through S2 and S3 in this module during switching. While precharging before AES logic working, and in the 2nd to 9th rounds of the encryption which is less likely to be attacked, EN is kept zero, then all signals are low to provide true Vdd to AES logic through the two PMOS transistors.

Fig. 6. shows a simulation result of the final topology. Pink curve is the current input at *Vdd_critical* end. Here, we gave two typical pulses from FastSpice results of our AES logic. Red curve is the current at Vdd_supply end. We can see the periodic charging current, which is not strongly affected by input pulses. Green curve is the voltage at *Vdd_critical*. The max is about 1.2V, and at input pulses, it will temporarily drop below 1.2V.

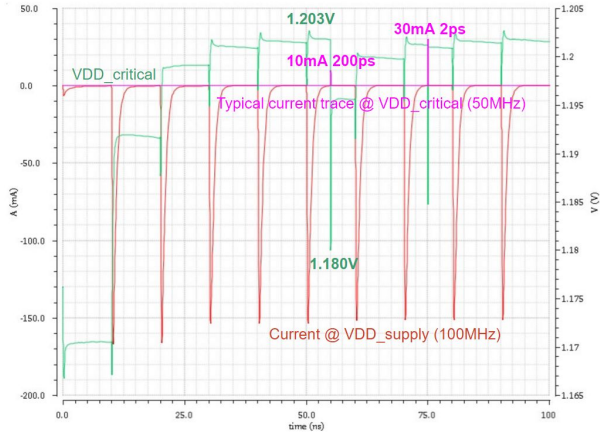

Fig. 6. Current Equalizer Simulation Results

B. Reduced Isolated Blocks

Tokunaga paper pointed out that, according to the analysis of encryption procedures [4], *MixColumns* is a late procedure in the encryption cycle, which occurs after the switching activity has been desynchronized. In order to directly attack the *MixColumns* step, the whole encryption cycle has to be analyzed by attackers, which is difficult because the information in middle rounds is invisible to attackers. Therefore, to save power and area, *Vdd_critical* is not supplied to *MixColumns* procedure in the baseline design.

In our design, *MixColumns* is not treated differently because we write the verilog of the AES logic as a whole. Instead, we adopted another idea to save power. According to the research of DPA approaches [6], the first and last round of encryption process are often attacked through DPA process because of their relatively weak data complicacy. Therefore, based on the baseline design, we will only supply *Vdd_critical* to the initial *AddRoundKey* step, the first and last (10th) encryption round to reduce the power consumption and performance penalty. This improvement is also illustrated in Fig. 1.

C. Randomly-Generated Jittered Clocks

The previous sections described the improvements adopted to our baseline current equalizer, but we think there remains space for us to do further enhancement. According to the research about hardware countermeasures to power side-channel attack [6], the effectiveness of current equalizer depends on the quality of switches, which has process variations at semiconductor level and their sub-threshold current might still be utilized by attackers. Misalignment in the ring oscillator due to temperature or clock skew might leak information as well.

It's straightforward for us to think that combining two countermeasures should be more secure. However, according to our literature research, some combinations may weaken either design. For example, some error detection algorithms will make the calculation signature in power traces more obvious for DPA. Therefore, we choose to apply randomly-generated jittered clocks instead of architecture level countermeasures.

Initially, we should generate 32 phase-shifted clocks. Then we should apply random clocks to 3 sequential elements: the initial *AddRoundKey* step, the first round and the last round. This is because these 3 steps are more likely to be attacked by the attackers. For each of these sequential elements, every 8 bits of the step share the same random clock. Since our AES engine is a 128-bit design, hence we need to use 16 32-to-1 multiplexers to choose 16 random clocks for each step. Therefore, we can increase the uncertainty of signal arrival time and thus randomize the power leakage information. This random clock selection process can be illustrated in the following Fig. 7.

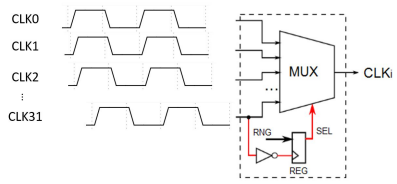

Fig. 7. Random clocks multiplexer [5]

As shown in the following Fig. 8, we firstly used a ring oscillator to generate a reference clock with frequency of 100MHz. Then we added a buffer between every two consecutive clocks to further generate 32 phase-shifted clocks.

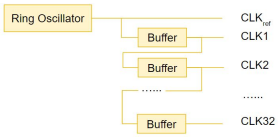

Fig. 8. Generation of 32 phase-shifted clocks [5]

As mentioned before, our clock frequency is 100 MHz, which means the clock period (T) is 10 ns. During the synthesis tests, we found that the longest combinational logic delay (t_d) is 5.40 ns, the clock skew (t_{skew}) is 0.06 ns and the setup time (t_{su}) is 1.62 ns. In order to maintain correct functionality, the longest clock delay between CLK32 and reference clock should not be larger than ($T-t_d-t_{skew}-t_{su}$) which means the maximum delay of each buffer ($t_{BufDelay}$) is:

$$t_{BufDelay} <= (T-t_d-t_{skew}-t_{su})/32 = 0.09 \text{ ns}$$

In order to effectively add buffer delays in our simulation, we need to manually draw the layout of clocktree, which means we must set a fixed $t_{BufDelay}$ value. Although there exist other $t_{BufDelay}$ values that can also make our random clock design work correctly, analyzing the different of $t_{BufDelay}$ values would be unnecessary because it means we need to manually draw every clocktree layout for each different $t_{BufDelay}$ value, which is infeasible. Therefore, we set our $t_{BufDelay}$ value to 0.06 ns, which is less than the upper limit 0.09 ns. Using the 0.06 ns buffer delay, our corner case tests (ff, fs, sf, ss) can also get the correct encryption results in Spice simulation.

D. *Random Number Generator*

As mentioned in the previous section, we need to use 16 32-to-1 multiplexers to choose 16 random clocks for the initial *AddRoundKey* step, the first round and the last round, which means we need have 80 selection bits per encryption. These selection bits are generated by the random number generators. Since the average time for an encryption is 16 cycles and the analyzed clock frequency is 100 MHz, hence the required random number generation rate is 500 Mbps.

In design review I, we presented the solution as using sequential hybrid random number generator consisting of one TRNG and M PRNGs to provide M different random SEL signals required by the mux array. However, as pointed out by our instructor, it might be over-designed for our AES engine because it could provide way more generation bits. Therefore, we decided to use PRNG as our random number generator. Compared with the TRNG design, though PRNG is deterministic, it could already provide enough randomness to our jitter-clock selections and it would cost less area.

E. *Full Chip Floorplan & Layout*

The full-chip floorplan are shown in Fig. 9, we use one encryption round repeatedly for 10 times for each plaintext. We use the counter to record the round number in order to determine the corresponding voltage and clock inputs for the certain round. Compared with pipelined design, it is area-efficient. However, since the next input will not be able to send until the former one finishes 10 rounds, the throughput will decrease. We give the area and power priority over the throughput in this design.

Two ring oscillators are used to provide the regular clock frequency at 100 MHz, and the backup clock frequency at 18 MHz, which will be used in case that the engine cannot work at the regular clock frequency in some extreme cases.

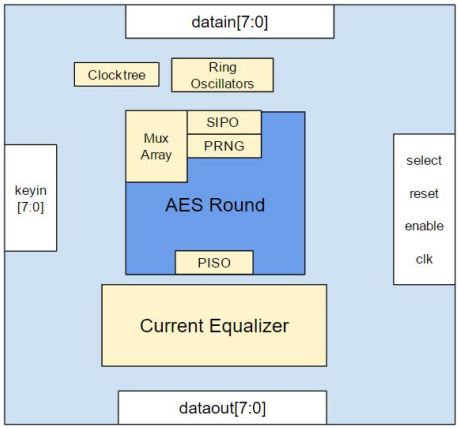

Fig. 9. Full Chip Floorplan

The full-chip layout are shown in Fig. 10, The total area including pads is 1.6mm * 1.6mm. Without pads, our core layout area is 0.5mm * 0.8mm In the layout design, we try to reduce the routing by put the pins near the corresponding pins on pads.

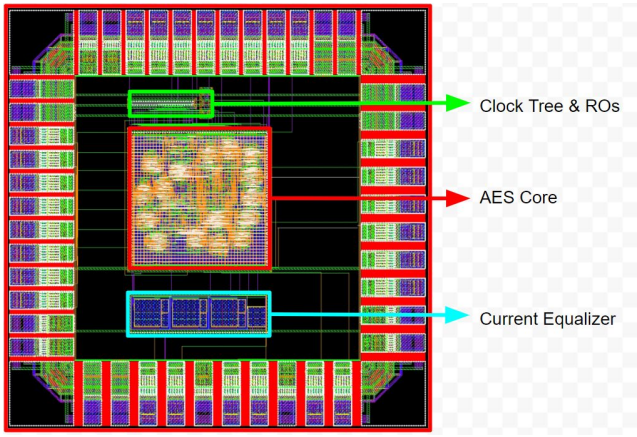

Fig. 10. Full Chip Layout

IV. Experiment Results

A. DPA on the Unprotected AES Core

We performed correlation power analysis, an efficient DPA method, to the first S-Box output of the unprotected AES core. Power traces simulated from CustomSim FastSpice simulator are used to expose the fragility of the unprotected AES core against DPA attacks. Compared to power traces obtained using oscilloscopes, results measured from simulators are more ideal since no disturbances or measurement inaccuracies are introduced but comes at a price of longer simulation time. A power trace of ten encryption rounds is shown in Fig. 11. The result of the DPA attack on the first block of the secret key is plotted in the bar graph in Fig. 12. The plot shows the correlation coefficient of 256 guessed keys for the block at the S-Box output timing point after applying 8000 encryption runs. The correct key is highlighted in red and the incorrect keys are black. The higher correlation of the correct key shows that the DPA attack performed succeeded. Fig.13. plots the correlation of all 256 guessed keys at the entire encryption run. The highest correlation of the correct key occurs around sample point 2200, which should be the time when S-Box output is calculated.

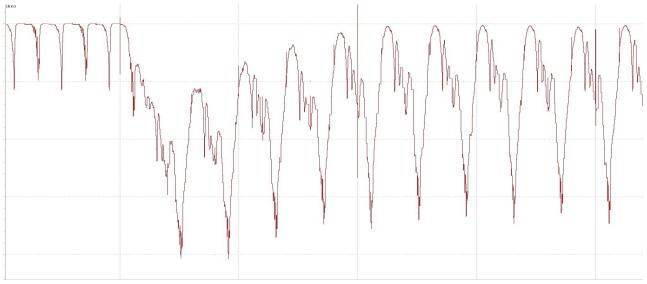

Fig. 11. Power consumption of 10 encryption rounds

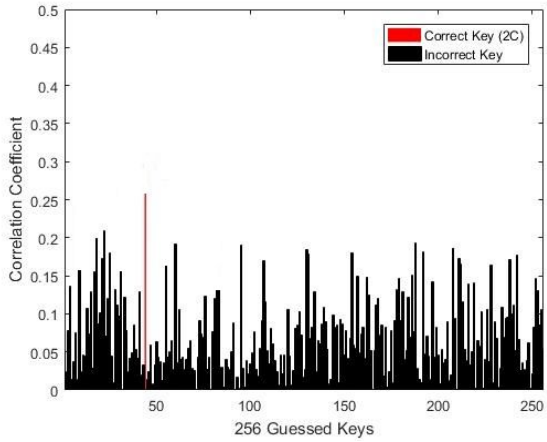

Fig. 12. Correlation coefficient of 256 guessed keys at the S-Box output timing point

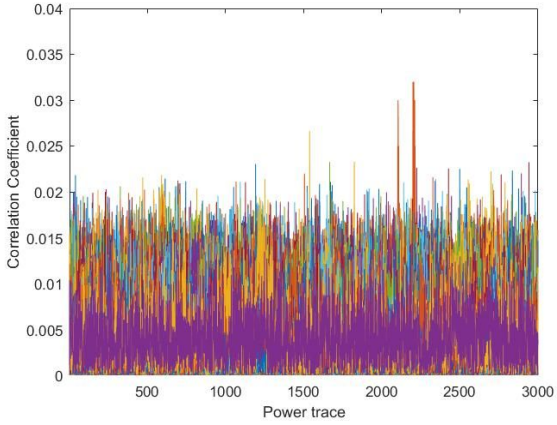

Fig. 13. Correlation coefficient of 256 guessed keys at the entire encryption round, the probed S-Box output calculation occurs at sample point 2200

B. DPA on the Secured AES Core with Random Clock

DPA attack was also performed on the AES core with random clocks. The result is plotted in the bar graph in Fig. 14. The result was obtained with 10000 encryption rounds. The correct key was not exposed and the correlation coefficient is reduced to half of that without random clocks, and this proves the efficiency of the random clocks in rejecting DPA attacks.

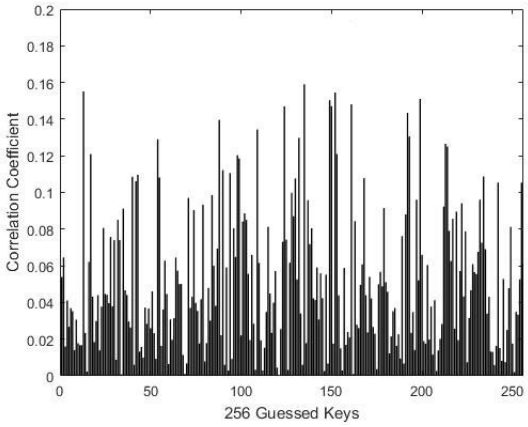

Fig. 14. Correlation coefficient of 256 guessed keys with randomly generated clocks

C. DPA on the Secured AES Core with Current Equalizer & Random Clock

The result of the DPA attack on the ultimate protected AES core with the combination of current equalizer and random clocks is plotted in the bar graph in Fig. 15. The result was obtained after 20,000 encryption runs and the secured key is not disclosed. Compared to the unprotected AES core, the correlation coefficient is more uniformly distributed and are also around 1 magnitudes lower. The correlation coefficient is also smaller than that of the AES core with only current equalizer [4] at the same number of encryption runs, which proves the efficiency of the

combination of current equalizer and randomly generated clocks.

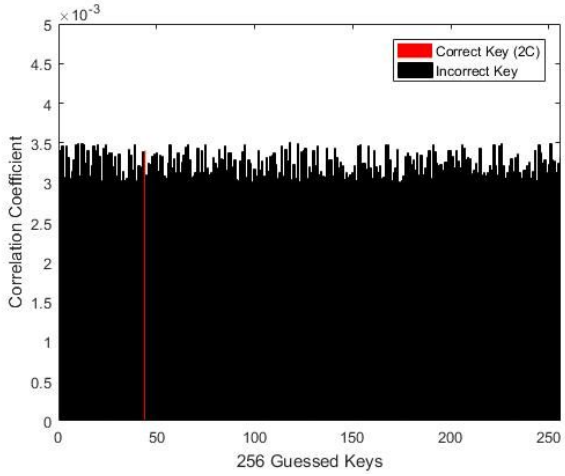

Fig. 15. Correlation coefficient of 256 guessed keys for the AES core with current equalizer and randomly generated clocks

D. Performance Summary

Table 1 below gives a summary of the performance and design specifications of both the unprotected and protected AES cores. With current equalizer and switch capacitor added, the area is increased by 33% which is reasonable. The 51% increase in power consumption of the protected AES core mainly results from the current equalizer. The functional frequency of the current equalizer has to be at least 3 times larger than that of the AES core in order to provide constant voltage level and therefore consumes considerable amount of power. The overhead in area and power consumption, however, is worthwhile provided with the enhanced ability to resist against DPA attacks. The first secured key byte of the unprotected AES was disclosed at 8000 power traces, while the key of the protected AES core was not exposed at 20,000 power traces. We were not able to obtain more power traces because of time limitation. Provided that the secured key of the AES core with only current equalizer [4] is not disclosed with 10 million power traces, with random clock generation added, our design exhibits an even smaller correlation pattern and should be more resistant against DPA attack.

	Unprotected	Protected
Countermeasure	-	power isolation & clock randomization
Max Frequency (MHz)	100	100
Max Throughput (Gb/s)	0.8	0.8
Area (mm^2)		
AES Core	0.304	0.304
Clock Tree	-	0.011
Current Equalizer	-	0.091
Total	0.304	0.406 (+34%)
Power (mW)	40.86	83.64 (+51%)
Measurements to Disclosure	8k	20k measured, not yet disclosed

Table 1. Performance and Specification Summary

V. Functionality Verification

The functionality of the design is verified via two steps. Firstly we used Synopsys VCS to simulate the unprotected AES core which was implemented using verilog, and compared the testbench output to that produced by a gold brick implemented in C language. Secondly both input and output of the testbench are exported to the CustomSim Spice simulator to simulate the protected AES core with analog blocks. Since the current equalizer and the random clocks do not affect the functionality of the AES core, the simulator should gives no comparison errors, which proves the correctness of the design.

For physical layout constraints, we performed DRC/LVS verifications and successfully passed these tests.

VI. Conclusion

In this paper, we propose and implemented a protected AES core with switch capacitor-based current equalizer and random clock generation. The current equalizer isolates the power rail from the cryptographic logic blocks to reduce power leakage, and the randomly generated clocks further introduces noise to the cipher blocks.The current equalizer are connected only to commonly vulnerable rounds of encryption to save power. The experiment results shows that the combination of the current equalizer and the random clocks greatly increased the ability of the AES core to resist against DPA attack.

Acknowledgment

We would like to thank Professor Dennis Sylvester and GSIs Qing Dong, Yiqun Zhang for their bountiful support and guidance throughout this challenging project.

References

[1] Trichina, Elena. "Combinational Logic Design for AES SubByte Transformation on Masked Data." *IACR Cryptology ePrint Archive* 2003 (2003): 236.

[2] Bucci, Marco, et al. "Three-phase dual-rail pre-charge logic." *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer Berlin Heidelberg, 2006.

[3] Le Masle, Adrien, Gary CT Chow, and Wayne Luk. "Constant power reconfigurable computing." *Field-Programmable Technology (FPT), 2011 International Conference on*. IEEE, 2011.

[4] C. Tokunaga and D. Blaauw, "Securing Encryption Systems With a Switched Capacitor Current Equalizer," in *IEEE Journal of Solid-State Circuits*, vol. 45, no. 1, pp. 23-31, Jan. 2010.

[5] Bayrak, Ali Galip, et al. "An EDA-friendly Protection Scheme against Side-Channel Attacks." *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 2013.

[6] Zhang, Lu, Luis Vega, and Michael Taylor. "Power Side Channels in Security ICs: Hardware Countermeasures." *arXiv preprint arXiv:1605.00681* (2016).

[7] Gornik, Andreas, et al. "A hardware-based countermeasure to reduce side-channel leakage: Design, implementation, and evaluation." *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.8 (2015): 1308-1319.