Low-Power Side-Channel Attack-Resistant Asynchronous S-Box Design for AES Cryptosystems

Jun Wu
Department of Electrical & Computer Engineering
Missouri Univ of Science & Technology
Rolla, MO, USA
jw616@mst.edu

Yong-Bin Kim
Department of Electrical & Computer Engineering
Northeastern University
Boston, MA, USA
ybk@ece.neu.edu

Minsu Choi
Department of Electrical & Computer Engineering
Missouri Univ of Science & Technology
Rolla, MO, USA
choim@mst.edu

1. INTRODUCTION

Advanced Encryption Standard (AES) [1] is a symmetric encryption algorithm based on a design principle known as a substitution-permutation network. The AES cipher is a series of transformations that convert the plaintext to ciphertext by using secret keys. Each round consists of AddRoundKey, ShiftRows, MixColumns steps which are linear operations and SubBytes step to be non-linear. The AES algorithm’s operations are performed on a two-dimensional array of bytes called the State, which consists of four columns and four rows of bytes.

SubBytes step is the first step of AES round. Each byte in the array is updated by a 8-bit substitution box (S-Box), derived from the multiplicative inverse over $GF(2^8)$. AES S-Box is constructed by combining the inverse function with an invertible affine transformation in order to avoid attacks based on mathematics. A block diagram of AES S-Box is shown in Fig.1(a). In the consequent MixColumns step, an linear transformation operates on each column of the state. The last step, AddRoundkeys, it add a round key to the state by doing the bitwise XOR operation in an AES round.

Since AES has become a FIPS standard in November 2001, various attempts of attack against the AES have been made. By exhaustive search, with 256-bit keys, $2^{256}$ possibilities must be checked, which lead apparent impossibility of attacks under such method. However, side-channel attacks has been proved to successfully attack the AES. Published side-channel attacks include simple power analysis (SPA) attack and differential power analysis (DPA) attack [2], which attack the cryptosystem that inadvertently leak information about the operations they process. DPA attacks are proven to be substantially effective to either directly reveal the hidden private key or significantly reduce key search space for faster and feasible exhaustive search.

During these years, various countermeasures of resisting side-channel analysis attacks have been proposed, including software-based and hardware-based methods. The hardware implementation of the AES essentially has higher reliability than software since it is difficult to be read or modified by the attackers and less prone to reverse engineering. The goal of countermeasures against DPA attacks is to reduce or balance the power consumption. For example, one can insert addition noise to interference the power [4], insert the random delays [6], static complementary CMOS logic [3], or the masked logic. However, these methods cannot prevent DPA attacks completely because of the power leakage of CMOS
circuit [7]. Dual-rail method is the most promising logic style among many countermeasures. Sense Amplifier Based Logic (SABL) [3], Wave dynamic differential logic (WDDL) [4] and Masked Dual-rail pre-Charged Logic (MDPL) [5] are all based on dual-rail logic. The benefit of dual-rail logic is that the constant power consumption can be achieved since the signals are implemented by two complementary wires. The downside is dual-rail method generally increase the area and time delay [12]. Another good countermeasure is using asynchronous logic, [8] presents that the power dissipated is independent of the input data in asynchronous logic. In this article, we propose an asynchronous AES S-Box based on a Null Convention Logic (NCL) [9], which matches the two important properties mentioned above; dual-rail encoding and clock-free operation. It is intended to achieve low-power consumption for mobile applications and considerable resistance against side-channel attacks such as DPA.

Figure 1: (a) Combinational S-Box architecture with encryption and decryption datapaths; (b) Block diagram of multiplicative inversion over \(GF(2^8)\) component where \(MM\) is modular multiplication and \(XOR\) is Exclusive-OR operation. [14].

2. SCA VULNERABILITY OF AES S-BOX AND EXISTING COUNTERMEASURES

Differential power analysis (DPA) is a type of side-channel attacks. DPA attack can extract secret keys through statistically analyzing power consumption measurements from a cryptosystem [2]. To do the DPA attack, normally attackers would do the following steps: (1) collect the power consumption measurements from the encrypted device with random inputs; (2) classify the collected results by using decision function; (3) redo (1) with a hypothetical key; (4) sort the results to the existing sets; (5) do the average power calculation in each sets; (6) compare different results until find the correct key. If the hypothetical key is the real key, it can be identified by a obviously spikes in the differential traces. Otherwise, the key is incorrect.

In the SubBytes operation, S-Box is the most critical component, as it determines the power consumption and throughput of not only the SubBytes operation but also the AES hardware implementation. Therefore, in this work, our research is focus on the S-Box design. The distinct peaks of the DPA trace occur to be closely related to first AddRound-Key operation and SubBytes operation. The experiment notes that such operations result in major data-dependent power leakages. More details of DPA attack on S-Box experiment can be found in section 5.3.

3. ASYNCHRONOUS AES S-BOX DESIGN

Asynchronous clockless circuits require less power, generate less noise and produce less electro-magnetic interference compared to their synchronous counterparts. Null Convention Logic (NCL) is a delay-insensitive logic which belongs to the asynchronous circuits categories. NCL circuit utilizes dual-rail and quad-rail logic to achieve this delay insensitivity [9]. A dual-rail signal can represent one of available three states, DATA0, DATA1 and NULL, which corresponds to boolean 0 (i.e., DATA0), boolean logic 1 (i.e., DATA1) and control signal NULL for asynchronous handshaking, respectively. In order to achieve clock-free operation, two delay-insensitive registers on both sides of the combinational NCL circuit with local handshaking signals are needed. In this research, dual-rail signals substitutes for corresponding conventional Binary signals in the NCL AES S-Box and the implementation platform is as shown in Figure 2.

Figure 2: Block diagram of the proposed NCL S-Box with I/O registers and completion detection component.

\[
q = a f f_{\text{trans}}(i) \quad q = a f f_{\text{trans}^{-1}}(i)
\]

<table>
<thead>
<tr>
<th>(q)</th>
<th>(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1_A = i_0 \oplus i_1, i_B = i_2 \oplus i_3)</td>
<td>(1_A = i_0 \oplus i_5, i_B = i_1 \oplus i_4)</td>
</tr>
<tr>
<td>(1_C = i_4 \oplus i_5, i_D = i_6 \oplus i_7)</td>
<td>(1_C = i_3 \oplus i_7, i_D = i_3 \oplus i_6)</td>
</tr>
<tr>
<td>(q_0 = i_0 \oplus i_C \oplus i_D)</td>
<td>(q_0 = i_5 \oplus i_C)</td>
</tr>
<tr>
<td>(q_1 = i_5 \oplus i_A \oplus i_D)</td>
<td>(q_1 = i_0 \oplus i_D)</td>
</tr>
<tr>
<td>(q_2 = i_2 \oplus i_A \oplus i_D)</td>
<td>(q_2 = i_7 \oplus i_D)</td>
</tr>
<tr>
<td>(q_3 = i_7 \oplus i_A \oplus i_B)</td>
<td>(q_3 = i_2 \oplus i_A)</td>
</tr>
<tr>
<td>(q_4 = i_4 \oplus i_A \oplus i_B)</td>
<td>(q_4 = i_1 \oplus i_D)</td>
</tr>
<tr>
<td>(q_5 = i_1 \oplus i_B \oplus i_C)</td>
<td>(q_5 = i_4 \oplus i_C)</td>
</tr>
<tr>
<td>(q_6 = i_6 \oplus i_B \oplus i_C)</td>
<td>(q_6 = i_3 \oplus i_A)</td>
</tr>
<tr>
<td>(q_7 = i_3 \oplus i_C \oplus i_D)</td>
<td>(q_7 = i_0 \oplus i_B)</td>
</tr>
</tbody>
</table>

Table 1: Boolean equations for Affine transformation and inverse Affine transformation components.

The AES S-Box algorithm adapted in this research follows
the combinational logic circuit architecture presented in [14].
The affine transformation and inverse affine transformation components follow a series of Boolean equations given in Table 1. As shown in the table, the affine transformation and inverse affine transformation components require 16 and 12 XOR gates, respectively.

The multiplicative inversion in GF(2^8) follows the procedure shown in Figure 1(b). Map, square, multiplication operations also require significant amount of XOR gates of which the sum is 95. To convert the conventional S-Box into NCL, replacing the Boolean XOR and AND operation into a dual-rail NCL gate is required.

Besides a series of XOR gates with AND gates, two NCL multiplexers are needed for switching between encryption and decryption process. Unlike boolean logic, NCL has 27 fundamental threshold gates [9] to realize arbitrary logic. In order to achieve the input-completeness and observability, it is important to choose appropriate threshold gates. For example, in the design of a 2:1 multiplexer, according to the Karnaugh map in Figure 3(a), the sum-of-product (SOP) functions can be simplified as follows:

\[ Z^0 = A^0 S^0 + S^1 B^0; \]  
(1)

\[ Z^1 = A^1 S^0 + S^1 B^1; \]  
(2)

After modifying both functions for input-completeness, new SOP functions are obtained as follows:

\[ Z^0 = A^0 S^0 (A^0 + A^1) (B^0 + B^1) + S^1 B^0 (A^0 + A^1) (B^0 + B^1); \]  
(3)

\[ Z^1 = A^1 S^0 (A^0 + A^1) (B^0 + B^1) + S^1 B^1 (A^0 + A^1) (B^0 + B^1); \]  
(4)

and both of them can be mapped to a NCL circuit with a TH24comp gate, a THAnd0 gate and a TH22 gate. The finalized NCL MUX logic diagram is shown in Figure 3(b).

Likewise, XOR function and AND function can also be implemented by threshold gates. An input-complete XOR logic is mapped to two TH24comp gates. An input-complete AND logic is mapped to a THAnd0 and a TH22 gate. The finalized NCL XOR and AND logic diagrams are as shown in Figure 4.

3.1 NCL registers and completion detection

In NCL, each NCL combinational logic block should be bracketed by input and output registrations to alternate a NULL wavefront and DATA wavefront to achieve delay-insensitivity. Since two consecutive DATA wavefronts are separated by a NULL wavefront, a reference clocking signal is not needed. Each NCL register has a single bit \( K_o \) (i.e., output acknowledgement signal) and \( K_i \) (i.e., input acknowledgement signal) signals which alternate between 0 and 1, defined as request for null (i.e., rfn) and request for data (i.e., rfd), respectively. Timing is locally handled by this delay-insensitive hand-shaking protocol.

In the completion detection component, the \( K_o \) signals are gathered and they are operated through an cascade of AND gates where output is set to \( K_i \) of the previous register, determining the state of current operation. Notably, the proposed NCL S-Box design shown in Figure 2 is free from glitches. Two possible transitions, NULL-to-DATA and DATA-to-NULL are monotonic and glitch-free since only 0 \( \rightarrow \) 1 wire transitions are possible for NULL-to-DATA cycle and 1 \( \rightarrow \) 0 wire transitions for DATA-to-NULL cycle, respectively. Therefore, the proposed NCL S-Box is completely immune to side-channel attacks based on glitch power/noise measurements.

4. FUNCTIONAL VERIFICATION OF THE PROPOSED NCL S-BOX DESIGN

The proposed NCL S-box has been implemented in VHDL and simulated with ModelSim by Mentor Graphics. In addition, a separate timing simulation has been done with Quartus II by Altera. By referring the waveform shown on Figure 5, the initial value of the input and output is NULL and DATA0, respectively, as previously input register is reset to NULL and output register is reset to DATA. As soon as reset falls down to 0, \( K_o \) from the output register becomes 1 and \( K_i \) for the input register connected to \( K_o \) becomes 1. As \( K_i \) rises, the input is changed to the waiting input signal, 0101010101010110 in dual-rail signaling which means 00000001 in Binary. Due to the propagation delay, the output arrives in approximately 5ns to be 0110101010100101 in NCL which means 01111100 in Binary. As every bit of the output signal changes to either DATA0 or DATA1 from NULL, \( K_o \) falls to 0 which means that output register has received the proper output DATA wave.

Table 3 shows the encryption and decryption simulation results for 10 arbitrary sample inputs, 5 for encryption and 5 for decryption, respectively. On the NCL S-Box output column, the results are shown as 16 bits, which are the extended
Figure 5: Sample Mentor Graphics ModelSim waveform for the proposed NCL S-Box.

dual-rail signals. For example, for input 158, the NCL S-Box output is 01 01 01 01 10 01 10 10, and this dual-rail encoded data word is equivalent to 00001011 in Binary which matches to the output of the conventional synchronous S-Box.

Simulation Results

<table>
<thead>
<tr>
<th>Mode</th>
<th>Input</th>
<th>S-Box</th>
<th>NCL S-Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption</td>
<td>9</td>
<td>00000001</td>
<td>0101010101010110</td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>10100010</td>
<td>1001100101011001</td>
</tr>
<tr>
<td></td>
<td>106</td>
<td>01000010</td>
<td>0101010101010101</td>
</tr>
<tr>
<td></td>
<td>122</td>
<td>11011010</td>
<td>1001101010101001</td>
</tr>
<tr>
<td></td>
<td>158</td>
<td>00001011</td>
<td>0101010110011010</td>
</tr>
<tr>
<td>Decryption</td>
<td>32</td>
<td>01101000</td>
<td>0110011001101001</td>
</tr>
<tr>
<td></td>
<td>51</td>
<td>01100110</td>
<td>0110100101101001</td>
</tr>
<tr>
<td></td>
<td>156</td>
<td>00011100</td>
<td>0101011010101001</td>
</tr>
<tr>
<td></td>
<td>185</td>
<td>11011011</td>
<td>1010011010101001</td>
</tr>
<tr>
<td></td>
<td>203</td>
<td>01101001</td>
<td>0110011010101010</td>
</tr>
</tbody>
</table>

Table 2: Simulation results for 20 arbitrary samples from conventional synchronous S-Box and the proposed NCL S-Box. Note NCL S-Box outputs are dual-rail encoded.

In order to further verify the functionality of the proposed NCL S-Box, a timing simulation has been done on a FPGA platform. In this experiment, the critical component is the top VHDL architecture, which deals with the I/O handshaking signal and forcing random inputs automatically. Figure 6(a) shows the block diagram of this top architecture, consisting of a NCL S-Box and a I/O controller. There are 21 inputs of NCL S-Box: RESET, 8-bit dual-rail input data (Din), 2-bit dual-rail select (Sel) signals. Since the input data and select are dual-rail signals, they take up 16 wires and 4 wires, respectively. Dout is the dual-rail 8-bit output data, which is the ciphertext output. As shown in Figure 6(a), only RESET is controlled externally, other input signals are provided by the I/O controller. The main controller circuit is shown in Figure 6(b).

In the controller unit, a look-up-table (LUT) is used to generate random inputs. a 10-bit counter provides \(2^{10}\) bit address into the LUT, which means 1024 random numbers can be generated per simulation round. The generated binary number is to be converted to the dual-rail signals by the "Single-rail to Dual-rail Converter". Ko is the output acknowledgement signal coming from S-Box. It acts like clock signal for the other models in the controller. LUT, counter, converter and multiplexer (MUX) are controlled by Ko. When Ko is ‘1’, it means NCL S-Box is ready for NULL wavefront, then MUX will send all 0s to "Sel" and "Din" to nullify the NCL S-Box. Otherwise, MUX will select the DATA. Timing simulation has been done using Quartus II by Altera. By referring the waveform shown on Figure 7, signal "q_tuoren" is the generated binary random number input from the LUT. Also, dout[7:0].RAIL0/1 represent dual-rail output signal’s rail values. As shown in the given waveform, when the input is 00000000, the output is 0110101010100101 from dout[7] to dout[0], which is 01100011 in binary. After comparing with the substitution table in [1], this output has been verified as correct. The other values have been also verified by following the same simulation method.

5. POWER CONSUMPTION SIMULATION AND COMPARISON

After the functional verification, the VHDL code has been synthesized to a verilog netlist using Leonardo tool. Design Architect-IC imports the verilog netlist to generate the
Figure 7: Timing Simulation waveform of NCL S-Box in Quartus II.

schematics. Finally, Accusim and AdvanceMS from Mentor Graphics are used for experimental analog and mixed signal simulations. The same procedure has been done in both NCL AES S-Box and the legacy synchronous combinational AES S-Box for comparing the results.

5.1 Total Power Consumption

Analog netlists for the two different S-Box designs have been generated by Accusim. Then we add the input to force signals to the netlists. Finally, Eldo has been used to obtain the total power dissipation of each design to perform one substitution round.

Advance MS (ADMS) is an extension of Mentor Graphics ELDO simulator, it can be used to calculate the power consumption of mixed-signal designs. This method is quite useful for NCL S-Box design, since all TH gates can be represented by transistor-level cells and high-level design description languages such as VHDL can be used to define the connectivity of those TH gates used. Also, a testbench can be written in one of the available high-level design languages such as VHDL to supply inputs and collect outputs for both functional verification and faster mixed-signal simulations. This method works well to generate power traces while applying varying inputs, which is a necessity for DPA-resistance analysis. Power simulation results from both Accusim and AdvanceMS for the proposed NCL S-Box and conventional synchronous S-Box are shown in Table 3.

<table>
<thead>
<tr>
<th>Temperature: 27°C</th>
<th>VDD: 1.8V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synch S-Box</td>
<td>NCL S-Box</td>
</tr>
<tr>
<td>Total Power Dissipation (Watts) - Accusim+Eldo</td>
<td>2.474E-08</td>
</tr>
<tr>
<td>Total Power Dissipation (Watts) - AdvanceMS</td>
<td>2.686E-08</td>
</tr>
</tbody>
</table>

Table 3: Power simulation results for synchronous AES S-Box and NCL AES S-Box using Accusim and AdvanceMS.

5.2 Power Trace Generation Method

As described in the previous subsection, we have used Mentor Graphics’ AdvanceMS tool to generate power traces from both synchronous S-Box and NCL S-Box to compare their abilities to resist DPA attack. In this experiment, a VHDL testbench has been written and used as a top architecture which includes the analog entity (i.e., output netlist generated by Accusim). Therefore, it is a mixed-signal simulation where AdvanceMS is the appropriate simulation engine. Figures 8 and 9 show sample power traces obtained from the synchronous S-Box and the proposed NCL S-Box, respectively, using AdvanceMS. Power traces from the synchronous S-Box are highly dependant upon changes in inputs and that is the major reason why DPA works on it. However, power traces from the proposed NCL S-Box show very high regularity and independence from changes in input.

Figure 8: Sample power trace from the synchronous combinational logic AES S-Box.

Figure 9: Sample power trace from the proposed clock-free NCL combinational logic AES S-Box.

5.3 Analysis of DPA-Resistance

According to the results of total power dissipation of both design, we find that the total power of the synchronous S-Box always change along with the different inputs. On the contrary, the total power of the proposed NCL S-Box does not change with the input changes. To compare DPA-resistance of the two designs, we have written a Python script to force one thousand different inputs to the regular AES, and extract the total power value respectively. Sorting them in two groups in order to conduct DPA and guess the correct key.

Firstly, we have arbitrarily selected a secret key of 0. Then we have run one thousand random inputs with this secret key, measuring power and sorting them in two groups. The decision function depends on the certain bit of the output value. For example, when the 7-bit of outputs is ‘1’, then related power belongs to the group 1, otherwise, once this bit is ‘0’, then sorting the power to group 0. This first case is to represent the situation when an attacker make the correct key hypothesis. Secondly, we have tried different keys (i.e., 2, 9, 50, 77, 78 and 128) and have followed the same procedure as described above. Then the average power
of each group has been calculated. The absolute value of difference of each group shows the DPA peaks. As Figure 10 shows, if the key is incorrect, the difference is smaller compared with the difference which is calculated by correct key. In above experiment, the correct key is 0. Therefore, Figure 10 shows the "key0" has the highest DPA peak than other key hypothesis.

![DPA Peak vs Key Hypothesis](image)

**Figure 10:** Different DPA peaks for the synchronous S-Box. The correct key hypothesis of 0 has the highest peak and DPA works.

We have also conducted a similar DPA-resistance analysis on the proposed NCL AES S-Box, but the DPA attack was not successful due to much higher regularity in power traces from the proposed NCL S-Box design.

### 6. CONCLUSION

A new asynchronous combinational S-Box design for AES cryptosystems has been proposed and validated in this work. The proposed S-Box design is based on a delay-insensitive logic paradigm known as Null Convention Logic (NCL) and achieves improved low-power operation and DPA-resistance over its clocked counterpart. The proposed NCL AES S-Box has been implemented in VHDL and simulated with Mentor Graphics EDA tool set. Various tools including ModelSim, Accusim, Design Architect-IC, Eldo and AdvanceMS have been used to perform functional verification, low-power operation and DPA-resistance. The proposed design has been compared with the existing synchronous combinational logic AES S-Box design and both reduced power consumption and improved DPA-resistance has been verified.

### 7. REFERENCES


