Differential Power Analysis Attack on FPGA Implementation of AES

Rajesh Velegalati, Panasayya S V V K Yalla

Abstract—**Cryptographic devices have found their way into a wide range of application and the topic of their security has reached great research importance. It has been proved that encryption device leaks information, which can be exploited by various attacks such as power analysis, timing analysis and electro-magnetic radiation. Differential power analysis is a powerful and efficient cryptanalytic technique which extracts information on secret keys by monitoring instantaneous power consumption of crypto processor and collecting the traces over a series of acquisitions. The focus of our project is to analyze how a crypto-processor in our case an FPGA which implements AES reveals information against Differential Power Analysis attack and also to note the number of encryptions needed to successfully extract the data and the time taken.**

Index Terms— **FPGA, Differential power analysis, AES, Power trace.**

I. Introduction

WHEN cryptographic algorithms are designed and analyzed lot effort is put into securing the algorithm against mathematical attack. But when such an algorithm is implemented on hardware it leaks some information and by analyzing such side channel information important data can be revealed. Differential power analysis or DPA uses such information that naturally leaks from the device namely power consumption.

What do we require to successfully implement a DPA attack?

Firstly we must be able to precisely measure the power consumption. Secondly we must know what algorithm is computed and third we must have either plain text or cipher text. Next the strategy will be to take as many power measurements as possible and then develop a power model (which we guess) .Now we perform statistical test on the measured power consumption and developed power model .If power model (guess) is right we can observe noticeable peaks in the statistics. We try to clarify this vague description of a DPA attack and implement it in our project.

The rest of the project report is structured as follows. Section-2 deals with introduction of Differential power analysis and different power models.Section-3 deals with the experimental set-up which includes brief description about the equipments used.Section-4 deals with the attack methodology Section-5 deals with analysis of results and finally conclusion and references.

II. Differential Power Analysis

A. Review Stage

Differential power analysis is the most popular and powerful type of power analysis attacks. It was discovered by Paul Kocher [6].The main advantage of DPA is no detailed knowledge about the cryptographic device is necessary .It mainly analyzes power consumption at particular point of time.DPA attack uses statistical methods and error correction techniques to extract information correlated to key. The number traces required for DPA depends on how well the power model used is described and noise involved. Different power models, mostly used are Hamming-Distance model and Hamming-Weight model.

Hamming Distance Model

Hamming distance model is to count number of 1-0 and 0-1 transitions that occur in the cryptographic device when it is implementing the cryptographic algorithm. This number of transitions is used to describe the power consumption of the cryptographic device at that time interval. In Hamming distance model it is assumed that the power consumption for 0-1 and 1-0 have same amount of power consumption which in most cases is not true. And also transitions 0-0 and 1-1 transitions are also assumed to contribute equally to the power consumption. So it assumes all gates contribute equally to the power consumption of the circuit and it neglects the parasitic capacitance between the transistors or wires. Since absolute values of power consumption are not needed for power analysis attacks, only relative differences between simulated power consumption values are important. It’s Ok with those two assumptions. This model is well suited for buses and registers. We have used hamming distance model in our attack.

Hamming –Weight model

This model is much simpler than the hamming distance model. This model is used when the attacker does not know the consecutive values of the data for some part of the process. Hamming weight is the number of ‘1’s in certain set of data. In this model, it is assumed that the power consumption is

Manuscript received May 08, 2008

Rajesh Velegalati is a graduate student in MSCPE @ GEORGE MASON UNIVERSITY (e-mail: rvelegal@gmu.edu).

Panasayya S V V K Yalla is a graduate student in MSCPE @ GEORGE MASON UNIVERSITY (e-mail: pyalla@gmu.edu).

proportional to the number of bits that are set in the processed data. This model in general may not be suited for power consumption of CMOS circuits.

III. EXPERIMENTAL SETUP

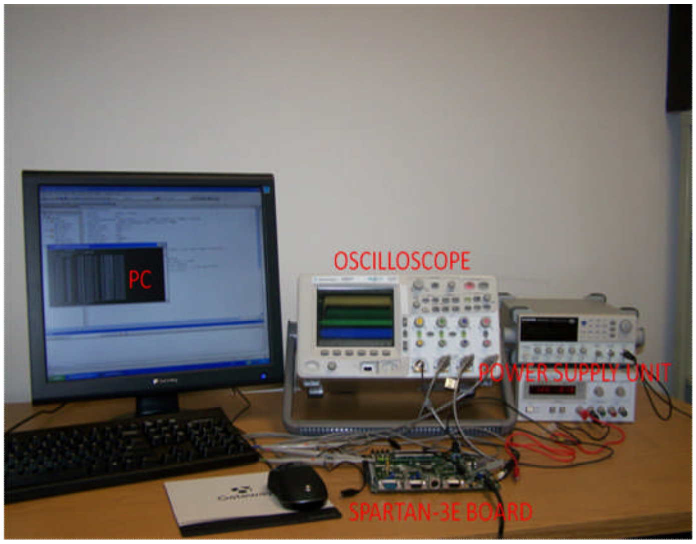


Figure1: experimental setup

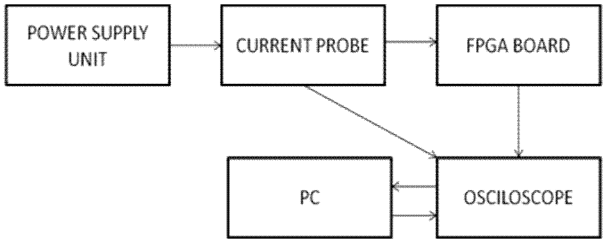


Figure 2: Basic block diagram of the experimental setup

Table provides a summary of Equipment used for measurement .Detailed description of the Equipment used is given below.

Equipment Type	**Model**	**Characteristics**
Power Supply	Agilent E3620A Multiple output DC power supply	Linear power supply
Oscilloscope	Agilent DSO6054A	500MHz B.W 4GSa/s sample rate
Current Probe	Tektronics CT_2	Freq.resp 1.2KHz to 2MHz
Test Board	Spartan 3E starter kit	Capacitors De-soldered
P.C	Intel[R] core[TM]2 6600 @ 2.4GHz	Has a 3.24 GB of RAM and a external Hard disk space of 320 GB

TABLE 1: Test Equipment Summary

Power supply:

The on-board IC power supply is by passed to obtain minimum power line noise. For this purpose we used Agilent E3620A multiple output DC power supplies to supply power to internal logic of the FPGA.

Oscilloscope:

Agilent DSO6054A series Oscilloscope used in this project has a high band-width of 500MHz and samples at a rate of 4GSa/sec and has a Standard 8Million points Mega Zoom Deep memory. However this particular Oscilloscope does not have the functionality of conducting multiple triggers and taking out multiple traces. So we had to improvise and take the entire power trace after the first trigger and then modify our Mat lab code accordingly. Deep memory present for this type of Oscilloscope helped our cause greatly.

Current Probe:

A 1mV/mA current probe “Tektronics CT-2” is used to connect in series with the power supply to sample the current consumption into voltage variation which can be measured by the oscilloscope

Test Board:

AES is implemented on Xilinx Spartan -3E starter kit which contains Xilinx XC3S500E Spartan-3E FPGA. It has up to 232 user I/O pins and is 320-pin FPGA package which has over 4500 clb slices .On board clock runs at 50MHz using a crystal oscillator is brought down to 5.0MHz using DCM (digital clock manager). So that enough sample points per cycle can be taken. This board consists of two jumpers (JP6 and JP7) for current sensing which is set according to connect the current probe. We connect the current probe at JP7.

Most Xilinx FPGA devices need three power supplies: One for IO blocks for the peripherals of FPGA, One for the Auxiliary components like DCM and one for internal FPGA functions like logic and routing resources. The power consumption should be measured at internal logic supply. In order to maximize the success of detecting variations in power consumptions we removed all the decoupling capacitors present on the internal logic supply input. A description of which decoupling capacitors are removed is shown in the Appendix.

Personal Computer:

In order to communicate between the FPGA and the oscilloscope and to perform the statistical analysis a PC is used. The model of the PC used in this project is described in the table 1.

Note that a resistor is not used for power measurement as it creates a negative feed-back loop. If resistor is included, it would be in series to the logic power supply .If the current flowing through the resister increases, than more voltage would be dropped across the resistor which will result in decreases the drop across FPGA. This negative feedback loop requires us to choose the value of resistor carefully. Bigger value of resistor will cause bigger variation and smaller value of resistor won’t cause enough voltage drops to be registered across oscilloscope. Thus a resistor is not used in our test setup.

Software used:

Synthesis and Implementation of AES

For synthesizing and implantation of AES we used Synplicity pro and Xilinx ISE 9.1 .Once the code is synthesized and implemented Xilinx iMPACT software is used to load the AES into the FPGA. By using iMPACT we can directly program the FPGA or load the program into a PROM so that there is no need to program the FPGA again and again. The Spartan 3E starter kit has several provisions in order to communicate to PC in our case we used a USB cable.

Communication between PC and Oscilloscope

Agilent IO libraries suite 14.2 was installed on to the PC which installs the drivers required for the Agilent oscilloscope to communicate with PC. Oscilloscope is connected to the PC by using a General Purpose Interface Bus (GPIB) USB cable which the windows PNP manager will automatically detect. C++ program is used in order to communicate between the PC and the Oscilloscope.

Processing of obtained Data from the Oscilloscope

The data obtained from the Oscilloscope is processed and the DPA statistical analysis is performed in Mat lab. As we mentioned before, the Oscilloscope used in this project supports only one trigger so once the first trigger gets high, the entire power trace is sampled. The required samples are sampled from the sampled data using Mat lab.

IV. Attack Methodology

In order to attack the AES presented in the above section Consider the following figure

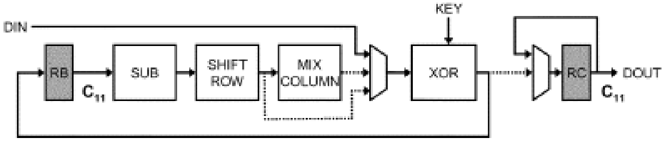


Figure 3: Arichtecture of AES implemented on FPGA [9]

The AES takes up 128-bit plain text and 128-bit key to produce 128-bit cipher text. Each round has a round key say K_0 to K_{11} .computed from the original key. We attack the last round key K_{11}.Due to the reversible nature of the round key computation algorithm, the original key can be computed from the round key.

The attack can be placed byte by byte on the last round key K_{11} (K_{11} [0] to K_{11} [15]) can be cracked separately. To perform the attack we operate the AES in OFB mode where the output from the previous encryption is given as input for next encryption. The estimation of power consumption for the last round is compared to a measurement of the power consumption in the 11th clock cycle (last round). Let the cipher text output after the last round be C11 and data input to this round is D11.In the last round

$$C_{11}= K_{11}\ \boldsymbol{Xor}\ \textbf{(}\text{shift row (sub byte }(D_{11})))$$

K11 and D_{11} are unknown. Only known value is C_{11}. D_{11} is found for different key guess values. Initially the attack is done on first 8 bits of the key. So for different key guesses from 0x00 to 0xFF (256 possible values), D11's is found using the function below. Using the obtained $D_{11,}$ the hamming distance for $D_{11}$and C_{11} is found and say it P_{guess}

$$D_{11}=\text{subbyte}^{-1}(\text{shiftrow}^{-1}(K_{11\ (guess)}\ \text{xor}\ C_{11}))$$
$$P_{guess}=\text{Hamming Distance }(D_{11,}\ C_{11}).$$

This part is done using AES C code. The hamming distance of the first 8 bits is computed. The following table is created which consist of hamming weights

KEY GUESS [K.G]	**CIPHER TEXT [C.T] 1**	**[C.T] 2**	**[CT] 3**	**[C.T] 4**	**----**	**[C.T] 255**
K.G 0	4	3	7	5	----	4
K.G 1	2	4	1	0	----	5
K.G 2	3	7	2	4	----	3
-----	4	1	0	5	----	2
-----	3	4	3	7	----	1
K.G 254	7	2	4	6	----	6
K.G 255	4	3	7	5	----	4

Table 2: Key guess table generated using AES C code.

In this experiment the maximum value of the current measured is taken. Let the measured power be $P_{measured}$

$$P_{measured} = \max\ (I_{supply})$$

Since attack is done on first 8 bits of the key, effect of all others is treated as noise. We find the correlation between $P_{guess\ and}$ $P_{measured.}$ The correlation coefficient is explained in the next section. The maximum value of the correlation is the right key of the last round.

$$\text{Max } f_{cor}(K_{11})=\text{correlation coefficient}(P_{guess},\ P_{measured})$$

Since there is lot of noise in the measured current, the correlation function may not give the right value. So in order to remove the effect of noise, thousands of combinations of plaintext-cipher text for the same key are measured. This method is repeated until you find the complete round key. After round key is obtained, the actual key can be obtained from it.

Correlation function

There are several methods to perform statistical analysis on the obtained data and the calculated data. One way is to calculate the Cross-Correlation between the P_{guess} and $P_{measured}$ the other is to find out the Correlation coefficient between the P_{guess} and $P_{measured}$.

Cross correlation also termed as cross-covariance is a measure of relation between two random vectors or two signals. By cross correlating between an unknown signal and a known signal we can find out the features of the unknown one. The cross correlation between two signals

$$(F * G)[m] = \sum_{j=0}^{n-m-1} F^*(j) * G(m+j)$$

Where F[n] and G[n] are two n-bit sequences .Cross correlation between two n-bit sequences will produce a (2*n) - 1 –bit sequence.

Correlation coefficient ‘R’ gives us the measure of the linear relationship between two variables. It is also called Pearson’s product moment correlation coefficient .It is given by

R(x, y) = covariance (x, y) / (STD(x) * STD(y))

Where STD= Standard Deviation.

Its value will always be between -1 to 1. (-1) indicate that x and y have a perfect negative linear relationship, (0) indicates that there is no linear relationship between x and y and (1) indicates that x and y have perfect positive linear relationship. So similar variables or sequences will have a higher Coefficient compared with others and hence when we plot such a correlation function we will get a “spike” in the graph. The following graph shows us a correlation between measured power trace (noise added to key guess) and Key guess. If the key at row 80 which is ‘’50’ in hex is the correct one, we would observe a spike because the correlation coefficients between the two sequences will be high.

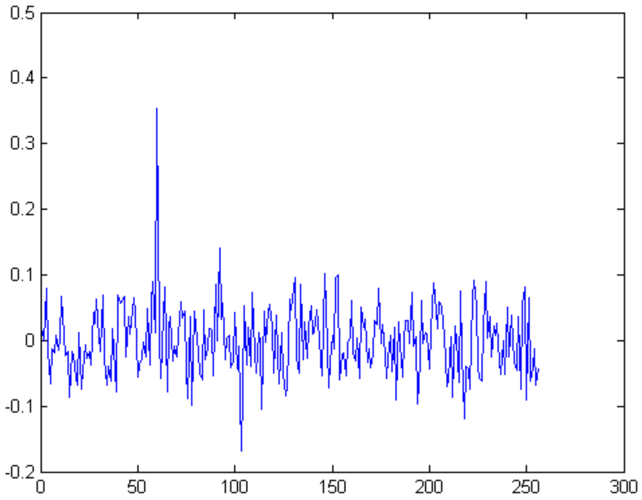


Figure 4: correlation between two sequences

V. Analysis of Results

As mentioned, AES is implemented on FPGA and the following power traces are obtained. You can observe the 10+1 rounds in the below traces (figure 5 and figure 6). In the trace the top one is the actual power traces measured and bottom one is the trigger signal which will trigger at every 11th clock cycle. The second figure is zoomed view of the actual traces.

Problems faced

Since the oscilloscope doesn’t have provision for segmented memory to record the traces for multiple triggers, we have taken the reading for the complete encryption and removed the unwanted sample using the Mat lab code. We also have to reject the first 200 encryption samples due to the capacitance effect of the start button.

Also as shown in the second power trace there are some harmonics present when trigger is active. We decreased the clock to 5MHz but the circuit operates at a higher clock frequency. So the power wave above the trigger when it is active ,it should have a spike at the starting of the trigger and then a flat line which as shown is not .Due to this harmonics and also delay between actual clock rise and trigger rise we were not able to get the correct data.

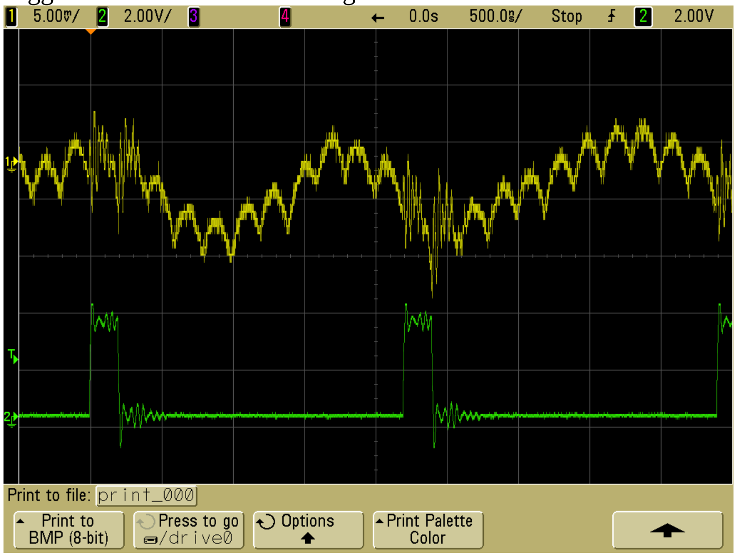


Figure5: Power trace measured on oscilloscope. Top signal measured current and bottom one is the trigger signal

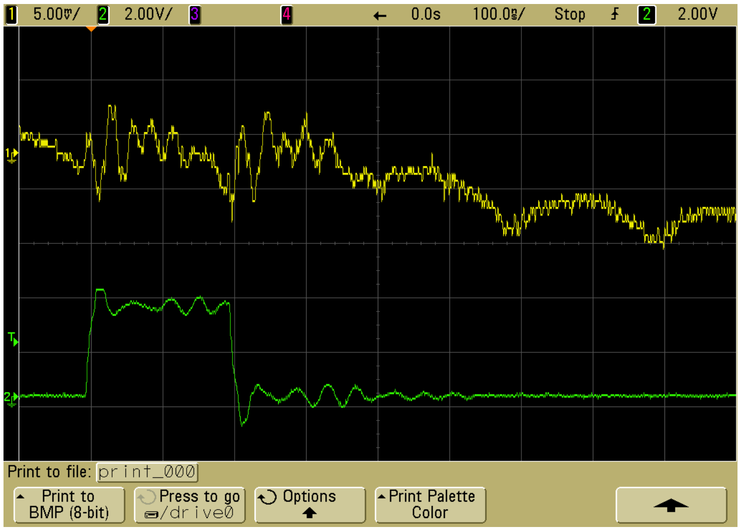
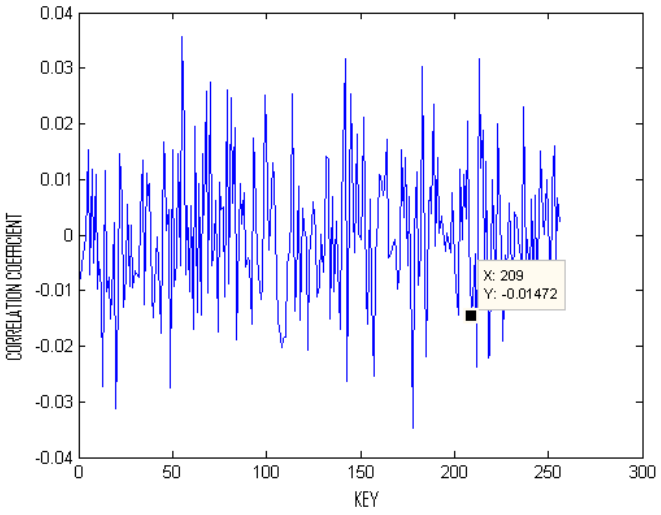


Figure 6: zoomed view of the trace above.

Hence when we found correlation coefficients between the measured and calculated power wave. It’s point to the wrong key as shown in the plot below. The correct key should be ‘D0’ which is 208 in decimal and since Mat lab offsets it by ‘1’ the correct peak should be at 209.



VI. Conclusion

As shown in the report we can identify several aspects of the cipher which is being executed on a crypto processor (in our case an FPGA) just by observing the power traces.DPA is one of the powerful analytical process which can reveal important information like Key very fast taking with very little amount of time. Our future work would be to get the Key from the power traces and if the key is obtained then attack on other implementations of the ciphers. But until a suitable masking technique is invented, cryptographers will always be concerned with DPA attack.

Appendix

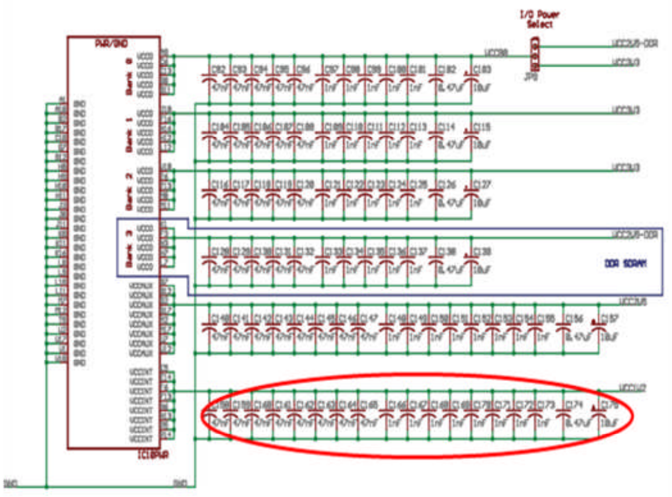


Figure 7: removed capacitances on the FPGA board

Acknowledgment

We would like to thank our professor Dr Kris Gaj for giving us an opportunity to do this project. We would also like to thank Dr Hwang and Dr Kaps for all their help throughout the project and also Shanuk Shah and Nick Ton for helping us at various stages of our project.

References

[1] Paul Kocher, Joshua Jaffe and Benjamin Jun, “Differential Power Analysis”, Advances in Cryptology CRYPTO’99, Lecture Notes in Computer Science (LNCS), vol.1666, Springer Verlag, Berlin, pp. 388-397, Aug, 1999.

[2] Pengyuan Yu,” Implementation of DPA-resistant circuit for FPGA”, Master’s Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2007

[3] Power Analysis Attacks: Revealing the secrets of smart cards by Stefan Mangard, Elisabeth Oswald and Thomas Popp,ISBN-13:9780387308579

[4] Thomas S. Messerges, Ezzy A. Dabbish, and Robert H loan. ”Investigations of power analysis attacks on smart cards”, USENIX Workshop on Smartcard Technology, pp.151-161, 1999.

[5] Jean-Sebastien Coron Christophe Clavier,” Differential power analysis in the presence of hardware counter measures”, CHES 2000, vol 1965, pp252-263.

[6] Joshua Jaffe Paul Kocher,”Introduction to differential power analysis and related attacks”, Cryptography Research, pp 1-5, 1998.

[7] Larry T. McDaniel III,” an investigation of differential power analysis attacks on FPGA based encryption systems”, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, May, 2003.

[8] Quisquater Jean Jaques, Joye Marc” Power analysis of an FPGA implementation of Rijindael: is pipelining a DPA countermeasure? ”, CHES, Lecture Notes in Computer Science (LNCS), vol 3156, Springer, July, 2004.

[9] K.Tiri, D Hwang, A. Hodjat, B.Lai, S Yang , P. Schaumont, and I. Verbauwhede, ”A side-channel leakage free co-processor IC in 0.18μm CMOS for embedded AES-based cryptographic and biometric processing ,42nd Design Automation Conference, pp 222-227,2005

[10] FIPS 197-Advanced Encryption Standard (http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf)