Introducing FOBOS: Flexible Open-source BOard for Side-channel analysis

Rajesh Velegalati and Jens-Peter Kaps

ECE Department, George Mason University, Fairfax, VA 22030, U.S.A.
{rvelegal, jkaps}@gmu.edu
http://cryptography.gmu.edu

Abstract. Side-channel analysis attacks pose a growing threat to implementations of cryptographic algorithms implemented in software as well as in hardware. Current standard side-channel evaluation boards with Field Programmable Gate Arrays (FPGAs), that allow for exploring the vulnerability of cryptographic implementations on FPGAs, are expensive and available only for a few FPGA devices. Furthermore, a complete open source software package that includes drivers that run test cases on the board, control the measurement equipment, and contain several side-channel analysis techniques is not readily available. Each user has to assemble their own setup based on software packages from multiple sources, written in multiple languages and write parts themselves. While educating students on side-channel analysis is important, hands-on laboratory exercises are not feasible do to cost and complexity. In this paper, we introduce am open-source environment, called FOBOS for conducting side-channel attacks on FPGAs. FOBOS supports multiple FPGA devices and includes all necessary software to run differential power analysis attacks, which are the most prominent kind of side-channel attacks. The components of FOBOS are build in a modular fashion so that it can easily be adapted for new FPGA boards, oscilloscopes, and attack techniques.

Keywords: Side-Channel Attack (SCA), Differential Power Analysis (DPA), SCA Board

1 Introduction and Motivation

Since its discovery by Kocher et al. in 1999 [4], power analysis has become a realistic and powerful threat to implementations of cryptographic algorithms in hardware as well as in software. Countermeasures against such attacks are going to be required for cryptographic devices (used by U.S Government) of security level 3 and above according to the National Institute of Standards and Technology [6]. Countermeasures proposed till date are also employed on most smartcards used by banks.

While the topic of side channel attacks and particular power analysis attacks entered the curriculum of cryptography courses in many universities, only very few have associated laboratory exercises.

Table 1. SASEBO boards with FPGAs as victims

Board	**Control FPGA**	**Victim**		**Wires Control–Victim**	**Host Data Communication**
		FPGA	**Techn.**		
SASEBO	Virtex-2 Pro	Virtex-2 Pro	130 nm	54	RS232
SASEBO-G	Virtex-2 Pro	Virtex-2 Pro	130 nm	53	RS232, FT245RL (USB)
SASEBO-GII	Spartan-3A	Virtex-5	65 nm	46	FT2232D (USB)
SASEBO-B	Stratix-2	Stratix-2	90 nm	53	RS232, FT245RL (USB)

A typical power analysis workstation cost around $20,000 not including commercial software licenses. The most costly device is a high-end oscilloscope. However, the power analysis evaluation boards, typically a Side-channel Attack Standard Evaluation Board (SASEBO), cost approximately $2,000. SASEBO has released 4 boards with FPGAs as victims (see table 1) covering 3 different FPGA devices. Only two of them are considered current. Many more current FPGA families exist and each family uses different building blocks and many are implemented using different CMOS technologies. Both features impact the amount of effort needed for a side-channel attack.

Only very few open source codes are available for side-channel attacks. SASEBO delivers data acquisition software which sends data to the board, receives and verifies the result and captures the power traces from the oscilloscope. For data analysis, OpenSCA (also known as DPA toolbox) was released at `http://opensca.sourceforge.net/`. Some additional Matlab and Octave scripts can be found on the webpage of [5] at `http://www.dpabook.org/`. Most of these open-source codes target software implementation on smartcards.

To our knowledge no complete software package exists that contains everything needed for evaluating the side-channel attack resistance of FPGA implementations from data acquisition to analysis.

2 Goals

FOBOS, loosely named after the Greek god Phobos ($\phi\acute{o}\beta o\varsigma$) who personifies fear and can pierce shields, is designed to be an inexpensive side channel analysis setup that includes a software package with programs for board control, data acquisition and data analysis. FOBOS uses off-the shelf FPGA boards to be used as control and victim and hence is less expensive than the traditional setup. Thus, enabling the universities to add active side channel analysis laboratory exercises to their cryptography classes. Furthermore, we are designing FOBOS in a modular fashion to allow for a multitude of victim devices while maintaining the remainder of the setup. Designers of crypto implementations and countermeasures against DPA on FPGAs can test their design techniques on FPGAs from various vendors and with different technologies. The FOBOS software package, documentation, and hardware components will be released as open-source.

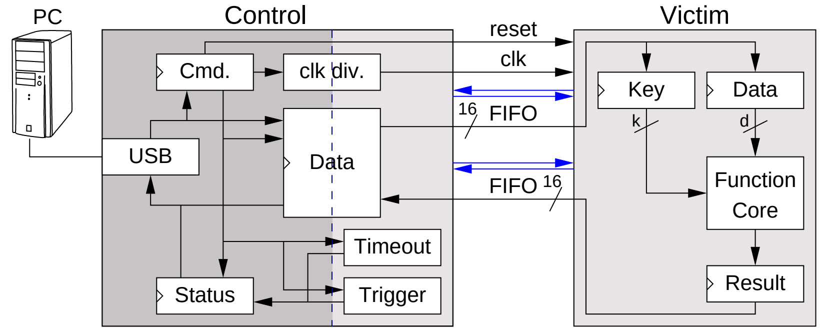

Fig. 1. FOBOS Schematic Diagram

3 Architecture of FOBOS

FOBOS has two parts, the *FOBOS Hardware* and the *FOBOS Software*. The following sections describe the functionality of various components of FOBOS.

3.1 FOBOS Hardware

A schematic diagram of the FOBOS hardware is shown in Fig. 1. It consists of two boards *Victim Board* & *Control Board* connected together by the so called bridge connector. The cryptographic algorithms whose security needs to be evaluated are implemented on the FPGA of the Victim board. Data i.e. plaintext and/or key is sent from the PC via USB to the control FPGA, which then forwards the data to the Victim FPGA. After processing the Victim FPGA send the results back to the Control FPGA which in turn forwards the results to the PC for verification.

Control Board: The control board used by FOBOS is either a Nexys2 or a Nexys3. Table 2 shows details of the both boards. Currently our FOBOS prototype only supports the Nexys2. The control board contains several modules (see Fig. 1) and two clock domains. It uses the on-board 50MHz oscillator as base clock for the USB communication. The second clock is generated through a clock divider circuit which uses the Digital Clock Managers (DCMs) to generate a clock in the range of 31.25KHz ∼ 50MHz from the 50MHz oscillator depending upon the user's choice and the oscilloscope specification. This clock is used for communication with the victim FPGA and also provided to the victim FPGA board.

Table 2. FOBOS FPGA Control Boards

Board	FPGA	Technology	Connector	PC-Control	Cost
Nexys 2	Spartan-3E	90 nm	Hirose FX2 (43)	USB2	$149
Nexys 3	Spartan 6	45 nm	VHDC (40)	USB2	$199

The control board receives commands from the PC and returns a status. This is facilitated through the 8-bit Command and Status registers. We use them to implement a simple protocol between PC and Control FPGA which is explained in Sect. 3.2.

The Trigger module generates a reference point from which the oscilloscope should start measuring the power consumption of the victim FPGA. Depending upon the user's requirement, this reference point can be set through a command to the beginning of the cryptographic operation or to specific clock cycle during the computation. This reference point is later used to perform signal alignment of several power traces.

A Timeout module makes sure that PC receives a status (of TIMEOUT) if an exception occurs during the communication with the victim or if the victim does not respond within a given time. This timeout value can be specified through a command. The timeout counter is automatically reset, each time the victim returns data.

Victim Board: We are investigating several FPGA boards available in the market, which can be used as Victim boards for FOBOS. Table 3 shows some potential Victim boards. The column "V_{Core} Jumper" indicates whether the board contains a jumper on the core power line which allows for by-passing the on board core power supply and inserting a current sensor (resistor or current probe) to measure the power consumption of the victim FPGA. So far, we have successfully used the Spartan 3E Starter Kit, Spartan 3E-1600 Developer Board, and the Altera DE1 board as FOBOS Victim boards. As the Altera DE1 does not have V_{Core} Jumper, we had to de-solder the voltage regulator for core voltage. On all board we also removed several capacitors. The other boards in the table are still to be investigated and tested by us.

Table 3. FOBOS FPGA victims

Board	**FPGA**	**Technology**	**V_{Core} Jumper**	**Connector**	**Cost**
Spartan 3E Starter	Spartan-3E	90 nm	yes	Hirose FX2 (43)	$189
Spartan 3E-1600 Dvlp.	Spartan-3E	90 nm	yes	Hirose FX2 (43)	$225
Altys Board	Spartan-6	45 nm	no	VHDCI (40)	$349
Genesys Board	Virtex-5	65 nm	no	2xVHDCI (2x40)	$995
Altera DE1	Cyclone-2	90 nm	no	2xIDE (2x36)	$150
Altera DE2-115	Cyclone-4	60 nm	no	IDE (36)+7, HSMC (82)	$595
Cyclone III Starter	Cyclone III	65 nm	yes	HSMC(80)	$199

Bridge Connector: The bride connector is a PCB with connectors that securely connect the victim board to the control board. The bridge connector also has an SMA port which can be used to supply a clock from an external source to both control and victim boards.

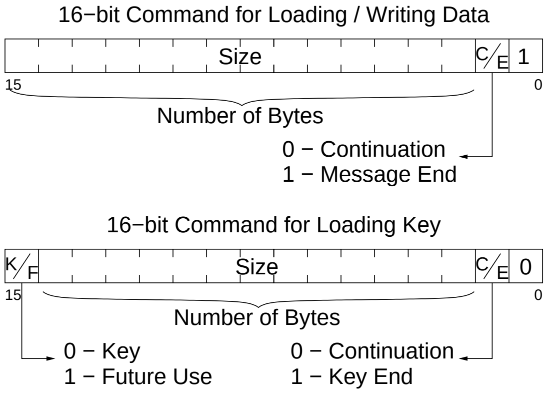

Fig. 2. FOBOS Protocol

FOBOS Control-Victim Protocol The FOBOS Control-Victim Protocol uses a simple FIFO interface to transfer data to and from the control and victim FPGAs. The functionality of the input and output ports of the protocol is described in [1], [2]. All data and key to and from the FPGA is broken into segments. The first 2 bytes (16-bit) of each segment is a command word, which decides the nature of the segment and the number of bytes being sent. The format of the 16-bit command words is shown in Fig 2. A '0' value in the LSB and a '0' value in the MSB of the command word indicates that a key is being sent. Similarly a '1' value in the LSB indicates that data is send. The bit in position 1 indicates with a '0' that more segments are following the current one, a '1' indicates that the current segment is the last. This protocol does not require the control board to know what the block size of the cryptographic function is. The widths of the databuses 'k' and 'd' indicated in Fig 1 can be defined by the user according to the requirement of the cryptographic implementation. The MSB bit value '1' for a 16-bit command for loading the key is left explicitly for future use.

3.2 FOBOS Software

FOBOS PC- Control Communication Protocol: FOBOS uses a set of scripts written in perl, to automate the process of data transmission and collection between the PC and the control board. The scripts are controlled by a configuration file and command line options. The command line options override the options in the configuration file. Here is an example command line.

```
> FOBOS -ifmt text -d data.txt -k key.txt -to 290 -tr 10 -c 10MHz
  -ofmt text -o cipher.txt
```

This command sends data from data.txt and a key from key.txt, which are both in the format of ASCII coded Hexadecimal values, to the victim. It sets the timeout to 290 clock cycles and the trigger to 10 clock cycles after processing starts. The command will return upon receiving a status (other than busy) from the control board. The victim clock is set to run at 10MHz and the result will be stored in ASCII coded Hexadecimal values in the file cipher.txt.

FOBOS uses the command register (shown in Fig. 1) to pass the option values to the modules inside the control FPGA. The command register is also used to signal the control board that PC is ready to transmit the data. The status register (shown in Fig. 1) on the other hand, is used for signaling the PC that the control FPGA is ready to transmit the data obtained from victim FPGA or to report errors.

FOBOS Data Acquisition Module: The data acquisition module configures the oscilloscope and retrieves its data. Its behavior is determined by a configuration file which uses a generic, oscilloscope band independent description. A special, oscilloscope dependent sub-module translates the configuration file to commands which are oscilloscope specific. The sub-module of our prototype uses the Virtual Instrument Software Architecture (VISA) library which is a standard for configuring and programming instruments using a variety of interfaces. Presently, FOBOS prototype supports communication for oscilloscopes from Agilent Technologies. In future we plan to provide support for oscilloscopes from other brands.

FOBOS Data Analysis Module: The Data Analysis module consists of 3 sub-modules: A raw data processing module, a leakage detection module and a DPA attack module. The raw data processing module transforms the raw data obtained from the oscilloscope to the actual voltage values using the “preamble” information returned by the oscilloscope. The “preamble” contains the digitized waveform settings, for example X-Origin, Y-Origin, number of data points transferred etc.

The leakage detection module, detects any significant leakage of information from the crypto implementations on the victim FPGA but not the nature of the information leakage or the secret key. This module uses Welch’s t-test [3], a non-parametric method of hypothesis testing procedure, to detect information leakage at different confidence intervals depending upon user’s requirement.

The DPA attack module contains a library of the state-of-the art side-channel distinguishers. The user hast to generate a hypothetical power model and can choose to test his/her own power model with one or all distinguishers to (try) obtain the secret information. Presently, the FOBOS prototype only supports CPA and Mutual Information analysis as side-channel distinguishers.

4 Conclusion

Currently FOBOS is a prototype under development. We hope that our choice of making the complete design open-source, giving the user the option of using Matlab or Octave, and by enabling the use of Xilinx and Altera university program boards, will make a hands-on side-channel attack experience possible for a wider audience.

References

1. Cryptographic Engineering Research Group, George Mason University: Hardware Interface of a Secure Hash Algorithm (SHA), v. 1.4 edn. (Jan 2010)
2. Gaj, K., Homsirikamol, E., Rogawski, M.: Fair and comprehensive methodology for comparing hardware performance of fourteen round two SHA-3 candidates using FPGA. In: Mangard, S., Standaert, F.X. (eds.) Cryptographic Hardware and Embedded Systems, CHES 2010. LNCS, vol. 6225, pp. 264–278. Springer Berlin / Heidelberg (2010)
3. Goodwill, G., Jun, B., Jaffe, J., Rohatgi, P.: A testing methodology for sidechannel resistance validation. The Non-Invasive Attack Testing Workshop (NIAT 2011) (Sept 2011)
4. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Advances in Cryptology - CRYPTO'99. Lecture Notes in Computer Science (LNCS), vol. 1666, pp. 388–397. Springer Verlag, Berlin (Aug 1999)
5. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks- Revealing the Secrets of Smart Cards. Springer (2007)
6. National Institute of Standards and Technology NIST, FIPS Publication 140-3: Security Requirements for Cryptographic Modules (Sep 2009), dRAFT