FPGA Based Secure System Design-an Overview

Gurjit Singh Walia, Gajraj Kuldeep, Rajiv Kapoor, A K Sharma, Navneet Gaba

Abstract- ***The implementation of cryptographic algorithm on FPGA is highly addressed in different forums due to its paramount advantages over the other platforms. Most of the secure systems are designed using SRAM based FPGAs with additional security features provided by the manufactures. In this paper, firstly, attempts are made to address different security problems of FPGA based secure systems. The difficulty levels that an attacker may face while implementing an attack are also tabulated. Finally, some constructive recommendation for tackling these security issues are proposed for designing secure systems.***

Keywords- Cryptography, FPGA, Secure system, Security, ASIC, SRAM

I INTRODUCTION

Field Programmable Gate Array (FPGA) is used to implement user defined function using interconnected reprogrammable functional blocks (embedded processors, giga-bit serial transceivers, clock mangers, digital signal processing blocks, ethernet controller etc).

The selection of the implementation platform for cryptographic application is dependent upon many critical factors such as complexity of algorithm and its application area, cost, speed, power consumption and desired security aspects (physical security, side channel leakage etc.) [1]&[19]. FPGAs are generally preferred over Application Specific Integrated Circuits (ASIC) when reprogrammability, power and price are considered as metric for decision [2], [10]&[16]. The inherent properties of FPGA, such as parallel operations and execution of customized functions make them performance competitive over the sequential microprocessor and microcontrollers. According to Wollinger et al. [4] and Wollinger and Paar [1], the potential advantages of FPGAs in cryptographic applications are algorithm agility, algorithm upload, architecture efficiency, resource efficiency, algorithm modification and throughput.

The rapid acceptance of FPGA solutions and sub-sequentially increase in these solutions in the market has created a host of new security concerns for system-level designers and system evaluators.

Manuscript received Dec 05, 2011.

Gurjit Singh Walia, Scientist 'D' , Scientific Analysis Group , DRDO , Delhi ,9873091231,(E-mail-gurjitwalia@gmal.com).

Gajraj Kuldeep, Scientist 'C' , Scientific Analysis Group , DRDO , Delhi ,9811614501,(E-mail-gajrajkuldeep@gmail..com)

Rajiv Kapoor, Professor and Head, Delhi Technological Univeristy (DCE), Delhi , ,9873091231 ,(E-mail-rajiv.kapoor@gmail.com).

A K Sharma, Scientist 'F', Scientific Analysis Group , DRDO , Delhi ,011-23817170,(E-mail-aksharma@gmail.com)

Navneet Gaba, Scientist 'F', Scientific Analysis Group , DRDO , Delhi ,011-23817170,(E-mail-navneetgaba200@gmail..com)

From the attacker point of view, not only the FPGA design but also the embedded information in the data being sent to or from a system is important. So taking this into consideration, classification of this topic was done in [7] and [6]:

a) *Intellectual property (IP) security*: Security concerns about the protection of vendor own design (IP) from being "cloned" or reverse engineered.
b) *Data security:* Security concerns about the protection of user design from being copied, corrupted, or otherwise interfered with.

When considering from cryptographic point of view, data security is the main area to be addressed. So the main work in this paper is based upon taking the data security issues as foremost important. In literature, much work has been done on development of cryptographic application on FPGA [3]&[18], but, still, some nooks are still left untouched when secure systems are considered as whole. Taking these observations into due consideration, it is hoped that this manuscript will definitely give a seminal learning to academia, industry and defense establishments.

In this paper, in Section 2, the security problems for secure systems are elaborated. In section 3, difficulty levels an attacker will face while implementing an attack is tabulated. Finally, in Section 4, some of design considerations that should be incorporated in secure system design have been suggested.

II SECURITY PROBLEMS FOR FPGA BASED SECURE SYSTEM

The motivation behind any attacker who is considered to be an adversary is to disturb the system functionality. The attackers' power is faithfully elaborated by IBM [8], Class I (clever outsiders) attackers do not have sufficient knowledge of the system but are often very intelligent; Class II (knowledgeable insiders) attackers have experience and specialized technical education and have expertise with sophisticated tools to analyze parts of a systems; Class III (funded organizations) attackers are able to develop teams of specialists and use the most sophisticated and expensive analysis tools as they have no limitation of money and they are consider to do in-depth analysis of system.

The aim of hardware attacker is to get secret information and to make the system non functional. The various attacks are reported in[1], [4], [13], [12], [9], [11], [14] , [15]&[23] and these are further elaborated in present work taking into account of present technology and tools available and intensive research work carried out at our laboratory. These attacks are categorized as follow with necessary hardware and software tools required for their implementation and also way out for prevention of these attacks.

A. Black Box Attack

In this attack, the attacker inputs all possible combinations, while saving the corresponding outputs.

After the log of different combination, the algorithm could be deduced with arduous efforts. For implementation of this attack a lot of processor power is required. However, this attack will be less feasible as the number of logic elements and complexity of the FPGA increases. The cost of the attack rises with the usage of state machines, LFSRs, and if pins can be used for input and output. However, advance mathematical techniques such as SAT Solvers could aid the attacker for launching black box attack.

B. Read-back Attack

The idea of this attack is to read the configuration of the FPGA through the JTAG or programming interface in order to obtain secret information (e.g. keys, algorithm). As reported [20], scan chain in FPGA can be exploited to decipher the cryptogram and this can be avoided by tree based pattern with self checking compactor. Advance tools for easy debugging such as Xilinx **JBits** are required for read back attack implementation. However**,** this attack will be less feasible if the programming port is disabled and after detection of interference the whole configuration is deleted or the FPGA is destroyed.

C. Cloning of FPGAs

In this attack, the attacker targets the configuration file of the system. As shown in fig:1 , the bit stream is stored in the flash memory for SRAM based FPGA and this need to be transferred during power on from flash to SRAM FPGA.

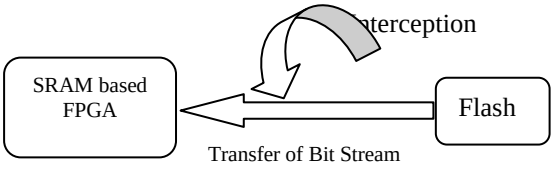

Fig 1 : Configuration of SRAM FPGA

This file could be intercepted during transfer by the third party to make a clone of the same. Advanced memory programmer, logic analyzer, data loggers etc. are required for launching this attack .The use of flash based FPGA could avoid this attack. But using flash based FPGA has inherent limitation for usage in secure system. The encryption of the configuration file is the most effective and practical counter measure against the cloning of SRAM FPGA.

D. Physical Attack

The aim of attack is to investigate the chip design in order to get information by probing inside the chip. This attack can be achieved through visual inspections and by tools such as optical microscopes, mechanical probes, Focused Ion Beams, Electron –beam tester. However, the manufacture could take precautionary measure in order to avoid this attack.

E. Side channel attacks (Power analysis, timing behavior, electromagnetic radiation)

Any physical implementation of a cryptographic system may provide a side channel that leaks information which can be exploited by the attacker to launch this attack.

By simple power analysis, differential power analysis, simple electro-magnetic analysis (SEMA), differential electro-magnetic analysis (DEMA) side channel information could be exploited. As shown in Fig: 2,

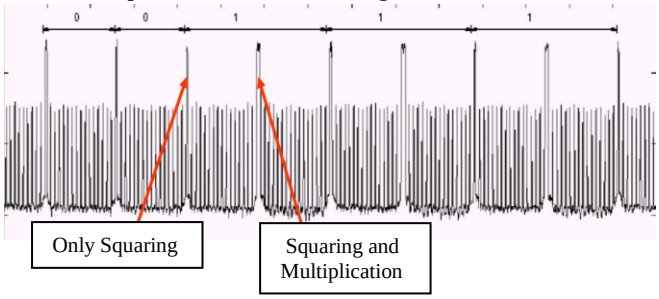

Fig 2: Power analysis of RSA

The RSA algorithm is implemented with main usage of squaring and multiplication functions. As reported in the literature, CMOS gates while switching draw current spikes which are exploited using the highly sensitive E & H probes and high end oscilloscope for launching this power analysis attack. However, preventive measures for side channel power analysis of RSA system for key generation could be taken either at designer or manufacture level.

F. Reverse –Engineering of the Bit Streams

The aim behind this attack is to get the design of proprietary crypto algorithm or the secret keys by reverse engineering of Bit streams. As shown in Fig 3 , a typical design flow of FPGA consist of various stages which are shown below.

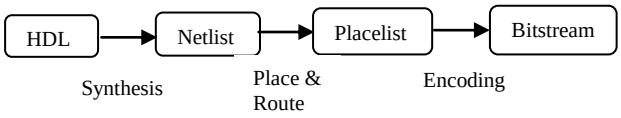

Fig 3: Software flow for FPGA implementation

Attacker may deduce HDL code from bitstream file. Advance tools are available for reverse engineering of Bit stream such as **Debit** which gives information about look up tables (LUT). Hiding keys in the look up table and RAMs can partially avoid this attack.

G. Tampering in Tools

In this attack, an adversary could add additional functionality to expose sensitive information, or provide unauthorized access. Layout-versus-schematic (LVS) tools could be used for detecting tampering in tools which is shown in the fig 4 below:

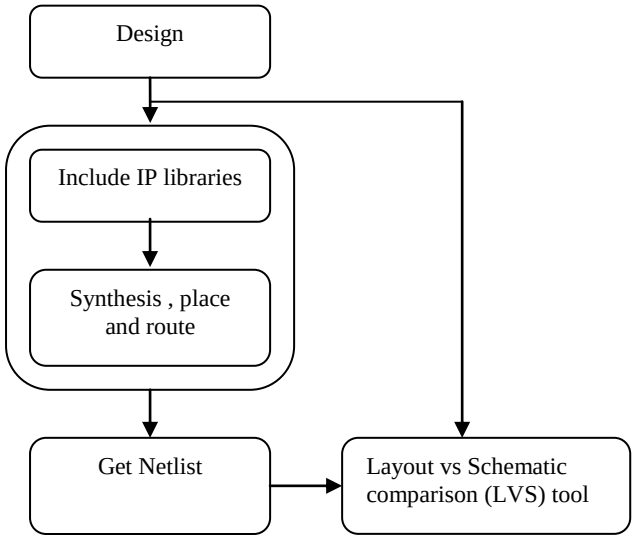

Fig 4: Design validation

However, Tampering could be avoided by comparing the implemented design and the original design.

H. Fault Attacks

Fault attacks exit where some hardware fault (an unexpected condition or defect) leads to a processing mistake that could be beneficial to the attacker. Advance timing comparison tools, logic analyzer etc. are imperative for implementation of this attack. However, this attack will be less feasible by avoiding supply of noisy power, incorrect voltage, excessive temperature, radiation or high energy beams such as UV, laser, etc.

I. Tempering in Hardware

Attackers make use of system hardware to get the secure information from the secure system. Attacker could exploit ports such as JTAG and redundant hardware present in the secure system. Tools such as Quartas , ISE , Visual DSP++ , emulator , memory readers etc. are essential for this attack. However, Designer could prevent this attack by taking care of temper resistance, detection, response and evidence while designing the secure hardware.

J. Trojan

Trojan is intentional malicious code written into the system design or it may be the malicious modification in the hardware circuitry, usually to the attacker benefit. Access to secure system and knowledge of software used is needed for implementation of Trojan. Xiaoxiao Wang et al. [22], classified Trojan into the different category which are shown below in fig 5.

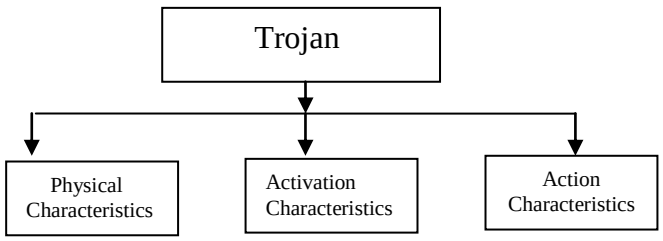

Fig 5: Classification of Trojan

The physical characteristics such as design of system can be exploited for inserting hardware Trojan in secure system. Activation can be subdivided into externally activated and internally activated Trojan. When doing the statistical analysis, activation of Trojan is considered to be a very rare event. Further, the action characteristic is subdivided into three different divisions: changing of functionality, alteration in specification and transmitting information through any medium which may be wired or wireless.

However, ATPG based Trojan detection technique and side channel analysis of RF emitted could be used for detection of Trojan. Also, Trojan could be avoided if Code walk through by independent authority.

III ATTACKER DIFFICULTY LEVELS

As per IBM systems Journal, Abraham et al. [8] has marked security levels for secure systems. While considering different technological development in the field of FPGA since 1980 and taking IBM security levels as reference for classification, we have marked different level of difficulties that an attacker may face while launching the attack in table 1 below. The few of listed difficulty levels are measured by actual performing the experiments but for some of attacks efforts are made to implement them, but either lack of time, prohibitively high difficulty of implementation or potentially destructive nature put impediments for actual measurements. Further, the listed difficulty level classification may vary with the attacker who is going to perform attack.

Table 1 : Attacks in Secure Hardware

Sr No.	Security Shortcomings of FPGA	*Attacker Difficulty Level
1.	Black Box Attack	5
2.	Readback Attack	2
3.	Cloning of FPGAs	
	a. Indigenous Flash/Antifuse FPGA	5
	b. Flash FPGA	4
	c. Antifuse FPGA	4
	d. SRAM FPGA with bitstream encryption.	3
	e. SRAM FPGA	1
4.	Reverse –Engineering of the Bit streams	4
5.	Physical Attack	5
6.	Side channel attacks	4
7.	Tampering in tools	3
8.	Fault attack	4
9.	Tempering in hardware	5
10.	Trojan	1

**Note: Level represented in scale of 1-5 with 5 as the maximum level of difficulty to launch the attack.*

IV RECOMMENDATIONS FOR FPGA BASED SECURE SYSTEMS

Although, there is no upper limit on the level of security that can be achieved, here some of key features which should be incorporated during design of FPGA based secure systems have been suggested.

A. In order to avoid reverse engineering and cloning of FPGA design, PROM should store only the encrypted bitstream file and also there should be feature of on-chip bit stream decryption. Now days, vendors are providing these features with 3- DES implementation for bit stream encryption [17].

B. Cryptographic operations in secure system are extensively used which causes long term retention effects in SRAM memory cell. Designer should include dummy cycles while doing cryptographic operation.

C. In order to avoid the tampering in original design by the design tools, the final implemented design and original design should be checked for their equivalence. It may be part and parcel of validation stage of software development cycle.

D. For secure system, FPGAs which have security level 3 or above as shown in table 1 above should be used to avoid reverse engineering of bitstream. Also, Designer should include the feature such as deletion of bitstream when tempering is detected.

E. The designers should be cleaver enough so that they can design their systems with due consideration of all the attacks based on malicious logic in the base array of FPGA. It is suggested that designer implements all critical parts of the design with minimum three level of modular redundancy (MR).

F. In case of physical compromise of the system, the designer should incorporate the feature of self destruction of the system. For example, with the detection of unauthorized interference, system design should be such that it erases the firmware & secret key information from the system.

G. The designers should give due consideration to make all unused I/O pins/ports as tri state so that unauthorized access could be avoided.

H. The design of FPGA should be such that provision of maintaining keys by the user must be avoided in order to cater for future changes in the design of secure systems.

I. The design should have provision of software countermeasure for all the side channel attacks. For example, design should mask secret key with the random values.

J. The designer should use TRNG for initialization of algorithm and also strength of crypto algorithm should be high in FPGA based secure systems to avoid Black Box attack.

K. The designer should use customized FPGA based board for cryptographic applications so that minimum resources with actual use should be included in the hardware.

L. Physical/logical separation should be incorporated for plain and cipher data at design level.

V CONCLUSION

FPGAs in cryptographic applications are generally preferred over ASIC due to their potential advantages such as algorithm agility, algorithm upload, architecture efficiency, resource efficiency, algorithm modification and throughput. However, when security aspects are taken into consideration there are numerous nooks for improvement which have been described in this paper. The tabulated difficulty levels may vary with the class of attacker and development of technological tools. Efforts are made to address all the possible vulnerabilities but sill this area is open for research due to latest developments of sophisticated tools and hardware.

ACKNOWLEDGEMENT

We are heartily grateful to Dr. P. K. Saxena, Director, outstanding Scientist 'H', for his support, benevolent guidance & motivation which inspired us and framed the state of mind that allowed this work to happen. We are indebted to Dr. S. S. Bedi, Associate Director for his invaluable support and informative suggestions.

REFERENCES

1. T. Wollinger and C. Paar. How Secure Are FPGAs in Crypto-graphic Applications?(Long Version).Report 2003/119,IACR,2003.
2. A. Elbirt, W. Yip, B. Chetwynd, and C. Paar. An FPGA-based performance Evaluation of the AES block cipher candidate algorithm finalists. IEEE Transactions on VLSI Design, 9(4):545{557, August 2001.
3. Jr. Kaliski , B. S. Koc , C. K. and C. Paar , Eds. 2002. Workshop on Cryptographic Hardware and Embedded Systems — CHES 2002. Vol. LNCS 2523. Springer-Verlag, Berlin, Germany.
4. T. Wollinger, J. Guajardo and C. Paar " Security on FPGAs: State of the Art Implementations and Attacks" ACM Special Issue Security and Embedded Systems Vol. No. March 2003.
5. [K. K Parhi, VLSI Digital Signal Processing Systems: Design and Implementation, John Wiley & Sons, Inc.1999.
6. T. Kean. Cryptographic rights management of FPGA intellectual property cores. In Field Programmable Gate Arrays Symposium, pages 113-118, New York, NY, USA, 2002. ACM Press.
7. Actel Corporation www.actel.com ,"Design Security in Nonvolatile Flash and Antifuse FPGAs", Security Backgrounder ©2002.
8. D.G Abraham, G.M. Dolan, G.P. Double and J.V. Stevens. 1991. Transaction Security System. IBM Systems Journal vol. 30 no. 2New York: International Machines Business Corporation: 206-229.
9. P. Vikram. Embedded systems challenge faq. Information Systems And Internet Security Laboratory,Department of Computer Science, NYU- Polytechnic University.
10. D.P. Wilt, R.C. Meitzler, and J.P. DeVale. Metrics for TRUST in Integrated Circuits, 2008
11. A. Ross. Security Engineering, A Gudie to Building Dependable Distributed Systems. Wiley, New York, second edition edition, 2008.
12. http://www.bltinc.com/services .fpga-reverseengineering.htm. FPGA Reverse Engineering Services.
13. E. J. Chikofsky and J. H. Cross, II, "Reverse Engineering and Design Recovery," IEEE Software, vol. 7, no. 1, pp. 13-17, January 1990.
14. P. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS and other systems," in Advances in Cryptology: Proceedings of CRYPTO'96, N. Koblitz, Ed., 1996, vol. 1109 of LNCS, pp. 104– 113,Springer- Verlag.
15. P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in Advances in Cryptology: Proceedings of CRYPTO'99, M. Wiener, Ed.1999, vol. 1666 of LNCS, pp. 388–397, Springer-Verlag

16. J. G. Proakis and D. G. Manolakis – Digital Signal Processing – Principles, Algorithms and Applications; Third Edition; Prentice Hall of India, 2003.
17. Xilinx Inc., www.xilinx.com. "Using Bitstream Encryption", in Chapter 2 of the Virtex II Platform FPGA.
18. T. Kean, "Secure Configuration of Field Programmable Gate Arrays", Proceedings of FPL 2001, Belfast, UK. Published as Springer LNCS.
19. Algotronix Ltd., "Method of Protecting Intellectual Property Cores on Field Programmable Gate Array", unpublished pending patent application.
20. D Mukhopadyay et al., " Cryptoscan: A Secured Scan Chain Architecture", IEEE @ 2005 Proceedings of the 14th Asian Test Symposium(ATS '05).
21. A Baumgarten et al "A case study of hardware Trojan design and implementation" Springer-Verlag 2010
22. Xiaoxiao Wang; Tehranipoor, M.; Plusquellic, J "Detecting malicious inclusions in secure hardware: Challenges and solutions" 2008 IEEE international workshop on hardware-oriented security and trust (HOST)