International Journal of Advanced Technology in Engineering and Science
Vol. No.5, Issue No. 03, March 2017
www.ijates.com
ijates
ISSN 2348 - 7550

FPGA IMPLEMENTATION OF HUMMINGBIRD CRYPTOGRAPHIC ALGORITHM WITH IMPROVED SECURITY AND THROUGHPUT: A REVIEW

Nikita Samrit[1], Shubhada Thakare[2]

[1]P.G. Student, [2]Assistant Professor, Department of Electronics Engineering,
Government College of Engineering Amravati (India)

ABSTRACT

Light weight cryptographic algorithms are essential to provide security in many resource constraint devices like RFID cards, smart cards and wireless sensors. The importance of RFID tags is increasing day by day. As they are resource constraint devices, they support only light weight cryptographic algorithm. Hummingbird is one of the Lightweight Authenticated Cryptographic Encryption Algorithm. Hummingbird is a combination of both block cipher and stream cipher along with a rotor machine equipped. This cryptographic algorithm is designed to deal with the trade off among security, cost and performance. The security and throughput of Hummingbird Cryptographic algorithm is improved by adding Hash functions. This paper gives the review work of hummingbird cryptographic algorithm done on different platform like microcontroller, sparton-3FPGA etc.

Keywords: Cryptography, Ciphertext, Hash function, Novel rotor, Plaintext, Private Key, Public Key, RFID tags.

I. INTRODUCTION

Information security and confidentiality has become important in many applications such as smart cards, Radio Frequency Identification tags (RFID). In these applications, user needs the access of private information. In electronic transactions, while making online payments customers provide credit or debit card private information. In these applications, unauthorized person can easily access the private information, if security is not provided. To protect privacy, the cryptographic algorithm embedded into these devices. RFID cards, smart cards are highly resource constraint devices, which supports only lightweight cryptographic algorithms. Cryptographic algorithms are of two types: symmetric key and asymmetric key. In symmetric key algorithms, same key is used for both encryption and decryption. In asymmetric, public key is used for encryption and private key for decryption. Symmetric

key algorithms are again divided into block cipher and stream cipher. Block cipher encrypts block by block data, whereas stream cipher encrypts bit by bit data. Block encryption process takes block of data called plaintext and

convert it into ciphertext of same size. Both encryption and decryption uses same secret key which may not have same size as block data. Original data can be recovered by decrypting the ciphertext.

The various cryptographic methods like Advanced Encryption Standard (AES), Data Encryption Standard (DES) have been failed to meet the requirements of constrained devices. DES has a block size of 64 bits and a key size of 56 bits. 64bit blocks became common in block cipher designs after DES. Key length depended on several factors, including government regulation. The 56bit key length used for DES was too short. Due to the large area required for AES, it cannot be used for highly constrained devices such as RFID tags, smart cards etc. AES has a fixed block size of 128 bits and a key size of 128, 192, or 256 bits. Key used is usually larger one and it is measured in bits. As the length of key increases, the security of cryptographic algorithm also increases. To overcome the drawbacks of AES and DES cryptographic algorithms, an ultra-light weight cryptographic algorithm called hummingbird algorithm has been proposed for resource constrained devices. Hummingbird is a combination of block cipher and stream cipher. It is based on a rotor machine which is equipped with novel rotor stepping rules. The Hummingbird algorithm uses a 256 bit of secret key. It takes 16 bit block data and performs the encryption in stream wise.

In block cipher, Shannon was the first who introduced the concept of confusion and diffusion for security. These two properties play important role for security purpose. In a block cipher confusion is performed at a substitution layer and diffusion is done at a permutation. User cannot easily identify and separate the components after confusion and diffusion techniques. The most commonly used confusion method uses s-boxes.

Hummingbird algorithm provides lower area, lower power consumption requirements. Also low cost and less processing time can be achieved. This model is considered as a hybrid model, as it is a combination of both block and stream cipher. It can provide the designed security with small block size. Therefore it can meet the stringent response time and power consumption requirements for the light weight resource constrained devices like RFID tags, Smart cards. It is developed with both lightweight software and lightweight hardware for resource constrained devices. Hummingbird is resistant to the most common attacks to block ciphers and stream ciphers like cube attacks birthday attack, structure attacks, algebraic attacks, differential and linear cryptanalysis.

Hummingbird mutual authentication protocol is used to achieve the trust relationship between RFID tags and readers. By using this protocol, the reader can identify the correct key that is communicating with a tag without exposing the identity of tags. In this protocol, the reader first sends a QUERY signal with a 16 bit SESSION ID as input to the tag. After receiving the QUERY, the tag will generate four 16 bit random vectors. These 4 random vectors will be used for initializing the four status registers. After initialization, it perform three times encryption of RS1#RS3 message data as input and generate three cipher texts CTO,CTl,CT2 as tag indicators. After that the tag will transmit these three cipher texts together with the initialized vectors to the reader. With the key present, reader perform encryption and generate three cipher texts. Three cipher text of both tag and reader are then compared. If it matches, then the tag will accept otherwise move for the next tag.

This protocol takes much area to store these indicators for further comparison. To reduce the area consumption, hash algorithm is used for secured mutual authentication. It will convert the key into hashes and compare with the hashed keys present in the look up tables for security purpose.

The hash function is designed by designer. Only the designer knows what mathematical functions are used inside hash algorithm. Hashing is not reversible process. It is impossible to have two different inputs which have the same hash value. This is the main security property of hash called strong collision resistance. Only the designer knows the internal structure of a hash function. That is why the hash function is not easily broken by an un-authorized person.

The hash algorithm will be performed first on the reader side to convert the 64 bit reader key into hashes and save it in the LUT for future comparison. Similarly in the tag side the 64 bit tag key also converted into hashes. Size of both the hash keys should be equal. Now tag will send the 64 bit hashed key to the reader. Then both the hashed key will be compared on reader side. If both matches, it will accept the tag otherwise will move for the next tag. If key matches, the tag will generate four random initialization vectors with the encrypted data to the reader. The reader will do the initialization process by generating 16 bit cipher text. After initialization, key is divided as per requirement and encryption is done using hashed key. Decryption is performed using same key where the encrypted tag data taken as the input.

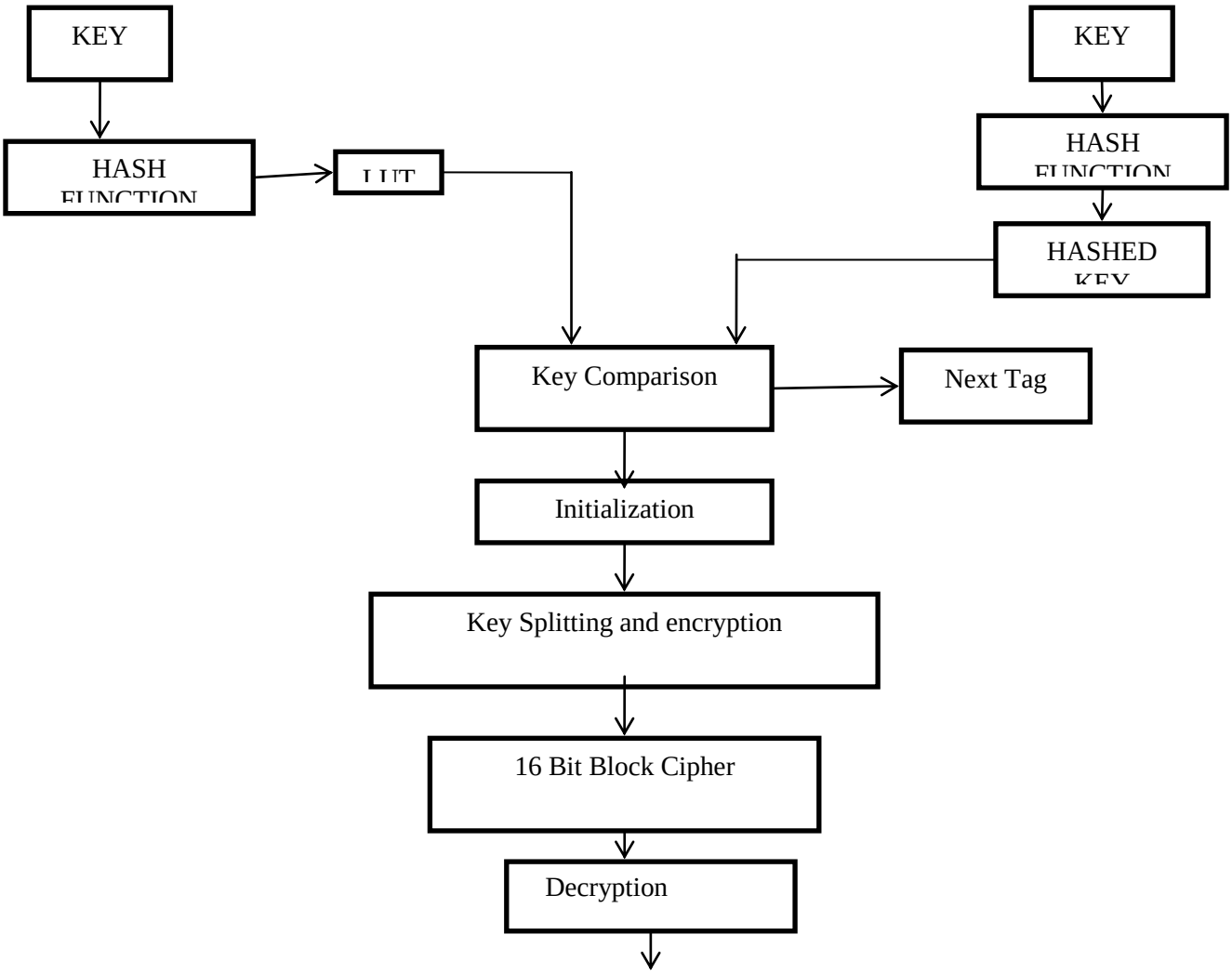

Fig. No.1, Flowchart of Secured Hummingbird Mutual Authentication Protocol using Hash Functions.

II. RELATED WORK

F. Xinxin et. al. explained the efficient software implementation of an lightweight cryptographic Hummingbird algorithm on a zero-power 4-bit MARC4 microcontroller from Atmel. Also the performance of Hummingbird and the other ultra-lightweight block cipher PRESENT on the same platform is compared. Experimental results show that after a system initialization phase 58% of faster throughput can be obtained with hummingbird than the block cipher PRESENT on the 4-bit ATAM893-D microcontroller running at 16KHz, 500KHz and 2MHz, respectively. Hummingbird can process one data block with less than 12 ms under a typical low power configuration of a 4-bit microcontroller such as an 1:8V supply voltage and a 500KHz clock frequency. Because of high data throughput and low current consumption, the ATAM893-D, a member of Atmel's MARC4 family of 4-bit single-chip microcontrollers is selected, as the target 4-bit platform. This makes it a perfect candidate for energy constrained wireless applications such as keyless entry, wireless keyboards for PC and multimedia, wireless sensors as well as other applications requiring an extremely low current consumption for extended battery life. [1]

G. Guang explains an efficient hardware implementations of a stand-alone Hummingbird component in field-programmable gate array (FPGA) devices. Hummingbird is a new ultra-lightweight cryptographic algorithm suitable for resource-constrained devices like RFID tags, smart cards, and wireless sensors. An encryption only core and an encryption/decryption core is implemented on the low-cost Xilinx FPGA series Spartan-3. By comparing results with other lightweight block cipher implementations on the same series gives better result of this method. Experimental results highlight that in the context of low-cost FPGA implementation Hummingbird has favorable efficiency and low area requirements. [2]

Hummingbird cryptographic algorithm implemented by coprocessor approach and serialized data processing principles is explained by T. San et. al. This work mainly reduces area, hence implementation on hardware can be achieved. This paper gives an enhanced hardware implementation of the Hummingbird cryptographic algorithm that is based on the memory blocks embedded within Spartan-3 FPGAs. The enhancement is from the introduction of the coprocessor approach. Also it can be obtained from the employment of serialized data processing principles. Due to compact architecture, remaining reconfigurable area in FPGAs can be used for other purposes. By comparing with the other FPGA implementation of the Hummingbird cryptographic algorithm indicate that the proposed architecture gives better efficiency and area. [3]

Paper [4] presents a novel ultra-lightweight encryption scheme, referred to as Hummingbird. This method is motivated by the design of the well-known Enigma machine. It shows that Hummingbird is resistant to the most common attacks such as linear and differential cryptanalysis. Also some properties for integrating the Hummingbird algorithm into a privacy-preserving identification and mutual authentication protocol is investigated.

The paper [5] describes a secure UHF RFID tag baseband with hummingbird cryptographic engine using SMIC 0.13um technology. Security can be enhanced by an improvement of the Gen 2 protocol based on secure engine. The implementation results show that the area of baseband is 16,986 gate equivalents and secure engine takes 23.6% of

the entire die area. The overall power consumption of baseband is 30.67uw at a clock frequency of 1.28 MHz and with 1.2V power supply, which is suitable for resource-constrained devices like RFID tags.

The privacy-preserving mutual authentication protocol for RFID systems using the recently proposed ultra-lightweight cryptographic algorithm hummingbird-2 is explained by F. Xinxin. The new protocol is resistant to the most common attacks of RFID systems. Also the proposed protocol is implemented on a battery less MS430-based WISP tag and determine the performance of the key search process on a laptop. Experimental results show that the hummingbird-2 mutual authentication protocol provides a highly effective and efficient security and privacy solution for low-cost passive RFID tags. [6]

M.Biao et. al. gives two different FPGA-based implementations for both throughput oriented and area oriented hummingbird cryptography. The throughput oriented design is optimized for operation speed. The area oriented design consumes smaller area resource usage. Both designs have been implemented on a Xilinx low-cost Spartan-3 XC3S200 FPGA. Experimental result shows that, the proposed design cost less FPGA slices while throughput can be obtained. It gives throughput and area oriented hummingbird design for FPGA with loop unrolled and round based structure respectively. [7]

In paper [8] an efficient hardware implementation of Hummingbird Cryptographic algorithm is implemented to get improved security and improved throughput by adding Hash functions. In this paper, encryption and decryption core is implemented on Spartan 3E and have compared the results with the other existing lightweight cryptographic algorithms. The results show that this algorithm has higher security and throughput than the existing algorithms.

In all above papers hummingbird cryptographic algorithm is implemented on different platforms like microcontroller, sparton-3FPGA, using coprocessor approach etc. In some of papers power consumption is reduced but area is increased. In some papers, area is reduced but power consumption is increased. In some paper they try to reduce the trade-off between area, power, cost requirements and check the hummingbird cryptography security performance.

II. CONCLUSIONS

Use of Hash algorithm along with hummingbird cryptographic algorithm makes system more secure. The size of the key and the internal state of Hummingbird provides a security level which is suitable for resource constrained devices. The security and throughput of Hummingbird Cryptographic algorithm is improved by adding Hash algorithms. Hummingbird is a hybrid model which is a combination of both block cipher and stream cipher. Hence it is resistant to the most common attack to both block and stream cipher.

Various papers are discussed about hummingbird cryptographic algorithm on different platform like microcontroller based on ASIC, sparton-2 FPGA, sparton-3 FPGA, etc. In all these papers, there is an enhanced research on reducing area, power requirement, & increasing speed with aim of giving better security to resource constrained devices like RFID, sensor nodes.

REFERENCES

[1] F. Xinxin, H. Honggang, G. Guang, E. M. Smith, and D. Engels, "Lightweight implementation of Hummingbird cryptographic algorithm on 4-bit microcontrollers," in internet Technology and Secured Transactions, 2009. ICITST 2009. International Conference for, 2009, pp. 1-7.

[2] F. Xinxin, G. Guang, K. Lauffenburger, and T. Hicks, "FPGA implementations of the Hummingbird cryptographic algorithm," in Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on, 2010, pp. 48-51.

[3] T. San and N. At, "Compact Hardware Architecture for Hummingbird Cryptographic Algorithm," in Field Programmable Logic and Applications (FPL), 2011 International Conference on, 2011, pp. 376-381.

[4] X. F. Daniel Engels, Guang Gong, Honggang Hu, Eric M. Smith, "Ultra-Lightweight Cryptography for Low-Cost RFlD Tags:Hummingbird Algorithm and Protocol," FC'1O Proceedings of the 14th international conference on Financial cryptograpy and data security,Springer-Verlag Berlin, Heidelberg ©201O, vol. ISBN:3-642-14991-X 978-3-642-14991-7, pp. 3-18 2010.

[5] X. Mengqin, S. Xiang, W. Junyu, and J. Crop, "Design of a UHF RFID tag baseband with the hummingbird cryptographic engine," In ASIC (ASICON), 2011 IEEE 9th international Conference on, 2011, pp. 800-803.

[6] F. Xinxin, G. Guang, D. W. Engels, and E. M. Smith, "A lightweight privacy-preserving mutual authentication protocol for RFID systems," in GLOBECOM Workshops (GC Wkshps), 2011 IEEE, 2011, pp. 1083-1087.

[7] M.Biao, R. C. C. Cheung, and H. Yan, "FPGA-based high throughput and area-efficient architectures of the Hummingbird cryptography," in IECON 2011- 37th Annual Conference on IEEE industrial Electronics Society, 2011, pp. 3998-4002.

[8] Harikrishnan T, C. Babu, "Cryptanalysis of Hummingbird algorithm with improved security and throughput," International Conference on VLSI-SATA ©2015, 978-1-4799-7926-7.