International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

FPGA Implementation of Cryptographic Hummingbird Algorithm with Improved Security: A Review

Nikita Samrit[1], Shubhada Thakare[2]

[1]P.G. Student [ESC], Department of Electronics Engineering, Government, College of Engineering, Amravati, India

[2]Assistant professor, Department of Electronics Engineering, Government, College of Engineering, Amravati, India

Abstract: ***Cryptographic algorithms are used to provide security, confidentiality and integrity of information.Hummingbird is a Lightweight Authenticated Cryptographic Encryption Algorithm. This light weight cryptographic algorithm is suitable for resource constrained devices like Smart cards, RFID tags and wireless sensors. Hummingbird is a combination of both block cipher and stream cipher along with a rotor machine equipped. It is based on novel rotor stepping rules. Hummingbird can provide the designed security with small block size.Therefore it can meet the stringent response time and power consumption requirements. This paper gives the review work done on hummingbird cryptographic algorithm on different platform like microcontroller, sparton-3FPGA etc.***

Keywords: Cryptography, Ciphertext, Hash function, Novel rotor, Plaintext, Private key, Public key

1. Introduction

Cryptography is used to hide the information content using a key. Some of the application of cryptography include the security of ATM cards, computer passwords and electronic commerce. Cryptography is derived from the greek words kryptos which means hidden and graphos means written respectively. There are two types of Cryptography i.e. encryption and decryption. Encryption converts the plain text to cipher text and decryption converts cipher text to plain text. Depending on key, cryptography is classified as private key and public key cryptography. In private key cryptography for both encryption and decryption same key is used which is shared between them only. Examples are AES, DES. In public key the encryption key is public, while another key called private key is used for decryption. So two keys used are different in asymmetric key cryptography. It is more secure than private key cryptography. Examples are RSA and ECC.

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.

The various cryptographic methods like AES, DES have been failed to meet the requirements of constrained devices. This innovate ultra-light weight cryptography. Hummingbird cryptography is one of such method. Hummingbird is a combination of both block cipher and stream cipher along with a rotor machine equipped with novel rotor stepping rules. Hummingbird cryptography starts with initialization process which is used to initialize internal state registers. After that encryption process is performed iteratively. The initialization process is used to get the initial value of LFSR. The 256 bit wide key is divided into four 64 bit wide sub-keys which are used by 4 block ciphers. These sub-keys are again divided into four 16 bit wide round keys for each round.

The design of Hummingbird has a 256 bit wide key and 16 bit wide block data and performs the encryption in stream wise. By using this algorithm lower area, lower power, low cost and less processing time can be achieved. This model is considered as a hybrid model. The hybrid model 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.

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 tag key will be converted into hashes. Size of both the hashes should be in equal. Now tag will send the 64 bit hashed key to the reader. After receiving both the hashed key will be compared on reader side. If both hash matches, it will accept the tag otherwise will go 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 to initialize the LFSR after receiving the initialization vectors and the key. After initialization, key is divided as per requirement and encryption is done using hashed key. Decryption is performed where the encrypted tag data taken as the input. In the throughput oriented design, we unroll the round operation in block cipher to process the 16 bit data in one clock cycle. The hash function is designed by designer. Here, only the designer knows what mathematical functions are used in hash algorithm. Hashing is irreversible 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.

Table: Comparison of cryptographic algorithms

	DES	AES	Hummingbird
Year	1977	2000	2010
Key Length	56 bits	128, 192 or 256 bits	256 bits
Cipher Type	Symmetric Block Cipher	Symmetric Block Cipher	Symmetric Stream and Block Cipher
Block Size	64 bits	128 bits	16 bits
Security	Proven Inadequate	Considered Secure	Highly Secured

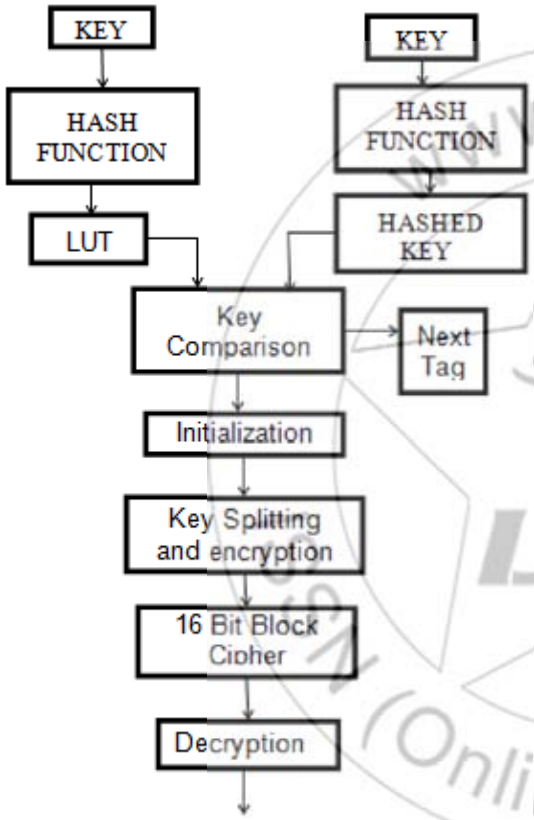

Fig.1, Flowchart of Secured Hummingbird Mutual Authentication Protocol using Hash Functions

2. Related Work

F. Xinxin et.al. explained the efficient software implementation of an ultra-lightweight cryptographic algorithm Hummingbird on a zero-power 4-bit MARC4 microcontroller from Atmel. Also compared the performance of Hummingbird and the other ultra-lightweight block cipher PRESENT on the same platform. 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 msec under a typical low power configuration of a 4-bit microcontroller such as 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]

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. [2]

G. Guang describes 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. [3]

The paper [4] 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 determinethe 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. [5]

Paper [6] 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.

In the paper [7], a lightweight remedial scheme in response to the Saarinen's attack is presented. The scheme is quite efficient both in software and hardware since only two cyclic shifts involved. This also maintains compact design of hummingbird.

M.Biaoet.al.give 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. [8]

In all of the above papers some of them try to improve area, some of them to reduce power and some of them to increase the overall performance. In some paper they try to reduce the trade-off between area, power, cost requirements and check the hummingbird cryptography security performance.

3. Conclusions

The design of Hummingbird Cryptographic Algorithm is based on an elegant combination of a block cipher and stream cipher with 16- bit block size, 256-bit key size, and 80-bit internal state. The size of the key and the internal state of Hummingbird provides a security level which is suitable for resource constrained devices. The use of Hash algorithm make system more secure.

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 of these, 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 .In shows that further work is needed to further enhance the above parameter of hummingbird cryptography & its security.

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] 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.

[3] 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.

[4] 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.

[5] 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.

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

[7] Xinxin Fan and Guang Gong, Honggang Hu " Remedying the Hummingbird Cryptographic Algorithm " *International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11* 2011.

[8] 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.

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