FPGA Implementation of Low Power and High Speed Hummingbird Cryptographic Algorithm

Nikita Arora
Department of Computer Science
ITM University, Gurgaon, India

Yogita Gigras
Assistant Professor
Department of Computer Science
ITM University, Gurgaon, India

ABSTRACT
Hummingbird is the latest ultra-lightweight cryptographic algorithm targeted for low cost smart devices. In this paper, we design a low power and high speed lightweight cryptographic Hummingbird algorithm for hardware environment. The performance of the approach used is determined on XILINX platform using Verilog as hardware description language. We have used Verilog for designing as well as simulation purposes. To verify digital design at the software platform we used ModelSim 6.2b simulator and XILINX 9.2i ISE suite is the synthesis tool used to transform design into digital circuits. An enhanced FPGA implementation of the Hummingbird cryptographic algorithm for low power and high operating speed (with max frequency) is performed using Virtex5 family of XILINX ISE suite. Comparisons to the other reported FPGA implementations of the Hummingbird, our proposed design outperforms the previous work in terms of speed and power requirements.

Keywords: Hummingbird Cryptographic Algorithm, Lightweight Cryptography, Constrained devices, FPGA

1. INTRODUCTION
Hummingbird has a hybrid structure of both block cipher and stream cipher with a small block size. This algorithm is intended to be used in resource constrained devices, various embedded applications and low end microcontrollers. Hummingbird can also be used for authentication as it produces 64 bit authenticator**[1]** for the message which ensures confidentiality and integrity of the messages. Hence Hummingbird is also known as the Authentication Algorithm.

The main objective of this paper is to design a low power and high speed lightweight cryptographic hummingbird algorithm for hardware environment. Virtual model of hardware is designed and verified using Model Sim SE 6.2b and the design is synthesized using the synthesis tool ie. Xilinx ISE 9.2i Suite. The purpose of the tool is to generate a digital hardware design by the given logic. Vertex 5 family of Xilinx(FPGA) is selected for hardware implementation of hummingbird cryptosystem.

The post synthesis simulation is performed on the design to yield system generated .ngr and .ngc files to verify each and every interconnections at the register level. Our proposed approach uses the following design hierarchy for designing and verification of digital circuit to its FPGA implementation. To reduce the power and increase the operating speed of the Hummingbird in its hardware implementation, we have reduced the clock latency. Moreover, The number of registers used in a single path are less.

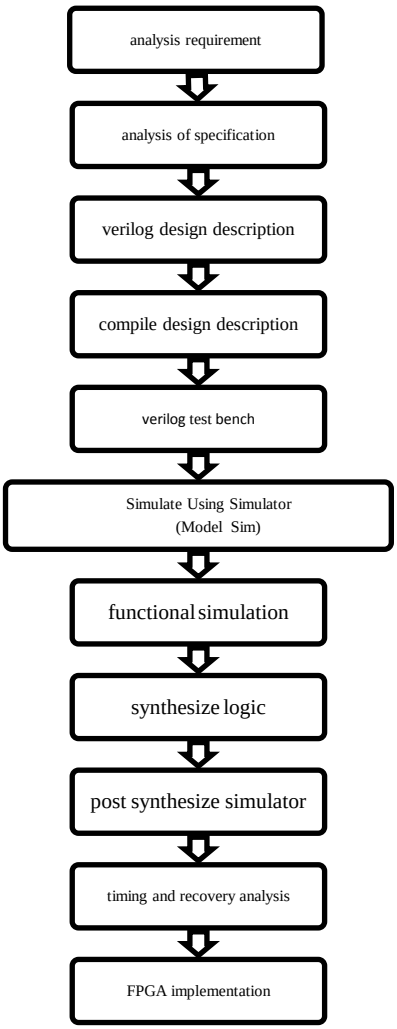

Fig 1: Design Hierarchy

2. THE HUMMINGBIRD CRYPTOGRAPHIC ALGORITHM
Hummingbird is a classic combination of block cipher and stream cipher with 16 bit block size,256 bits key size and 80 bits of internal states. Both initialization and encryption mechanism of Hummingbird Cryptosystem utilizes 4 block ciphers each of 16 bits, 4 internal state registers and a 16 bit LFSR(Linear Feedback Shift Register).Moreover the 256 bit symmetric key is divided into four 64 bit subkeys k1,k2,k3 and k4**[2]** which are used in four block ciphers respectively.

After the initialization phase of the Hummingbird, the 16 bit plaintext is enciphered by passing four similar block ciphers

E_{ki} {i:1,2,3,4}which is a typical substitution-permutation network that alternates between confusion and diffusion**[3].** The confusion in the cipher is introduced through the use of S-boxes and diffusion is achieved through the permutations. Hummingbird block cipher is a 4 round process initialized using 64 bit random nonce where each round updates the internal states and LFSR. Each regular round comprises of three steps namely Key Mixing, Substitution step(S-boxes),permutation step. The algorithm uses 4 Serpent like S-boxes of size 4*4.The S-boxes used in Hummingbird is completely balanced and non-linear which ensures that the block cipher is resistant to several attacks. The permutation in the 16 bit block cipher**[4]** is represented by following transform:

L(m)=m XOR (m«6) XOR (m«10);

Hummingbird provides tradeoff between cost, size, speed and performance and hence it is targeted for hardware and micro-controller environment.

The detail portrayal for encryption mechanism of the 16 bit block cipher is depicted in the following algorithm 1.

Input: A 16-bit data block $m = (m_0, m_1, \cdots, m_{15})$ and a 64-bit subkey k_i such that
subkey $k_i = K_1^{(i)} \| K_2^{(i)} \| K_3^{(i)} \| K_4^{(i)}$
Output: A 16-bit date block $m' = (m'_0, m'_1, \cdots, m'_{15})$

1: **for** $j = 1$ to 4 **do**
2: $m \leftarrow m \oplus K_j^{(i)}$ [key mixing step]
3: $A = m_0 \| m_1 \| m_2 \| m_3, \quad B = m_4 \| m_5 \| m_6 \| m_7$
$C = m_8 \| m_9 \| m_{10} \| m_{11}, D = m_{12} \| m_{13} \| m_{14} \| m_{15}$
4: $m \leftarrow S_1(A) \| S_2(B) \| S_3(C) \| S_4(D)$
[substitution layer]
5: $m \leftarrow m \oplus (m \lll 6) \oplus (m \lll 10)$
[permutation layer]
6: **end for**
7: $m \leftarrow m \oplus K_1^{(i)} \oplus K_3^{(i)}$
8: $A = m_0 \| m_1 \| m_2 \| m_3, \quad B = m_4 \| m_5 \| m_6 \| m_7$
$C = m_8 \| m_9 \| m_{10} \| m_{11}, D = m_{12} \| m_{13} \| m_{14} \| m_{15}$
9: $m \leftarrow S_1(A) \| S_2(B) \| S_3(C) \| S_4(D)$
10: $m' \leftarrow m \oplus K_2^{(i)} \oplus K_4^{(i)}$
11: **return** $m' = (m'_0, m'_1, \cdots, m'_{15})$

Algorithm 1: 16 bit block cipher Encryption Algorithm**[4]**

2.1Hummingbird Encryption Block Diagram

The given figure 2 below presents the hardware architectural block diagram of Hummingbird Encryption Module after the synthesis. The block diagram shows the input and the output port. Reset signal is used to reset the states back to the initial states. In input port, the data input, four 16 bit plaintext and 64 bit random nonces are predefined in the algorithm which yields the corresponding ciphertext and the updated internal state registers when we trigger the clock and reset signal is 1.

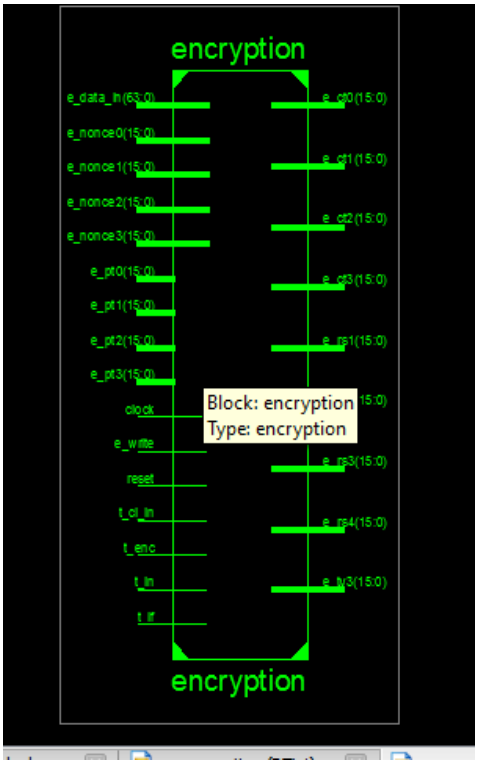

Fig 2: Block Diagram of Hummingbird Encryption

3. FPGA IMPLEMENTATION OF HUMMINGBIRD

Since we are emphasizing on the hardware implementation of the Hummingbird algorithm, so the FPGA(Field Programmable Gate Arrays) is the hardware platform selected depending on the application needs and constraints. FPGA configuration is specified using a hardware description language (HDL). Verilog is the hardware description language used for designing as well as simulation purposes. FPGAs comprises of logic blocks(flip flops, gates, memory elements) that are used to implement any logic functionality. We have described the FPGA implementation of Hummingbird Cryptographic algorithm using Virtex-5 xc5v1x20t-2-ff-323 of Xilinx(FPGAs) for the hardware implementation of the cryptosystem. The implemented design consumes low power of 262.57 mW for 2.5 V with the operating speed or frequency of 152.905 MHz.

4. IMPLEMENTATION RESULTS AND COMPARISONS

The hardware design of Hummingbird on FPGA is presented using Hardware description language as Verilog via Xilinx Plan Ahead simultaneously. Virtual model of hardware is verified and simulated via. Model-Sim simulator and synthesized using Xilinx ISE suite. The design layout is presented by integrating all the components ie. Initialization module,cipher, S-box, LFSR, Encryption/Decryption and top module of the algorithm.

After simulation and synthesis, the next step is place and route which provides the hardware design for the proposed Hummingbird Cryptographic Algorithm. All experimental results were extracted after place and route with the Xilinx ISE 9.2 I Design Suite and the target device is xc5v1x20t-2-ff-323 with speed grade -2.

4.1 Hardware Platform Design Implementation

The Xilinx Plan Ahead tool is embedded within the Xilinx ISE suite; it provides the post synthesis analysis and generates the hardware platform . After performing the syntax checking and RTL or technology design from the Xilinx ISE suite, the Xilinx plan ahead fetch the code directly. The hardware design of each of Hummingbird Encryption technique is implemented below in the figures: 3, 4.

After the functional simulation, the Xilinx synthesis technology synthesizes VHDL or Verilog code to create Xilinx specific .ngr and .ngc files.

Register-transfer level (RTL) is a simplified delegation of the pre-optimized design optimized at the register level in terms of adders , multipliers, counters, AND/OR Gates. The RTL schematic represents the intermediate block which are monitored for the speed optimization.

Technology schematic

This is the schematical model of logic elements optimized to the Xilinx target device or "technology" in terms of LUT's, I/O buffers, carry logic and other technology specific components. These .ngc files contains logical data along with the constraints.

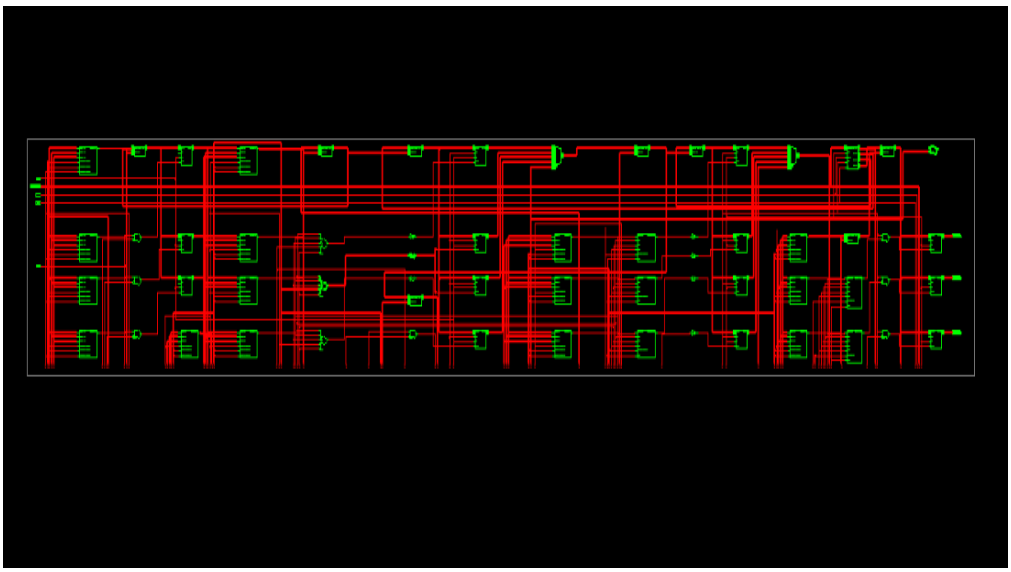

Fig 3 :RTL on system schematic (.ngr file)

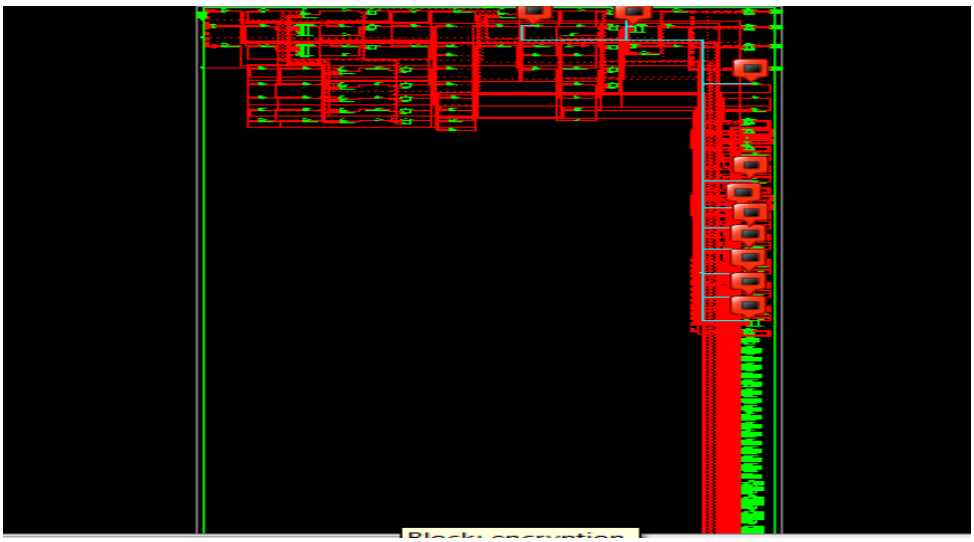

Fig 4: Technology Schematic (.ngc file)

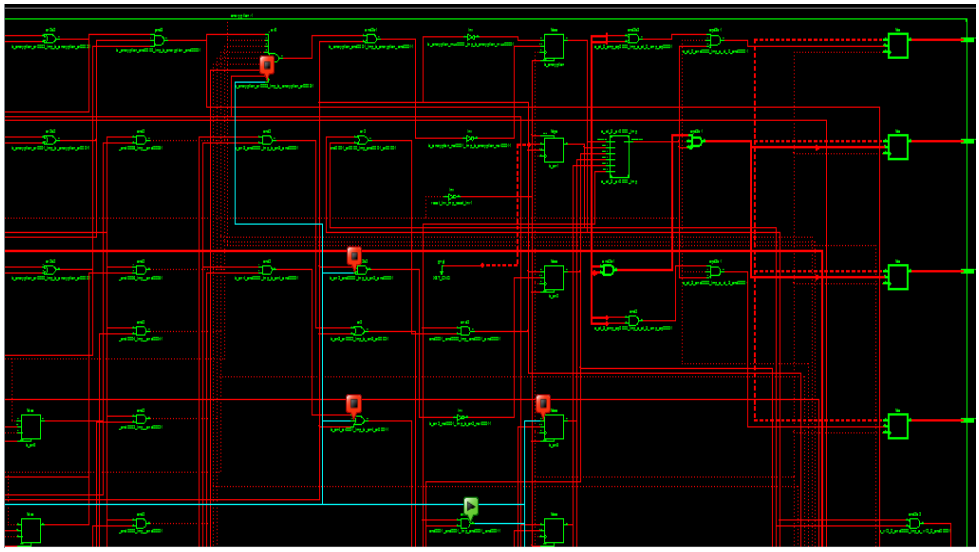

Fig 5: Technology view (.ngc file)

4.2 Post Synthesis Simulation:

After synthesis, we run post synthesis simulation for verification of our designed functionality. The following table **1** depicts, the usage of operations in implementing the hardware architecture of hummingbird algorithm. In the proposed architecture design, the number of slice LUT's (look up table) utilization is 28%. It shows advantage in terms of input/output bonds (IOB) which is 161%. IOB provides multiple usage for performance enhancement, through single input. In this paper, the FPGA implementation of hummingbird is shown, as it is applicable for low cost resource constrained devices like RFID tags, smart cards, credit cards and wireless sensor nodes.

Table 2 presents the comparison of existing FPGA implementations of block ciphers with our proposed Hummingbird implementation. The FPGA implementations of XTEA, ICEBERG, SEA and AES is compared with the Hummingbird cipher. Our proposed approach requires low power consumption of 262.57mW. As seen from the table, our work ie. Our designed Hummingbird cryptosystem has the highest operating speed with the maximum frequency of 152.90 MHz. This table is given in order to observe where our implemented Hummingbird cipher stands among the FPGA implementations of other lightweight cryptosystems. We have depicted this comparisons among the algorithms in terms of FPGA target device, packages, speed grade, synthesis and tools used for implementation. Hummingbird outperforms the above lightweight implementations of the algorithms in terms of the operating speed.The implementations ICEBERG and SEA cipher is only attainable on Virtex-2 FPGAs family.

Table 1: Hummingbird Device Utilization Summary Sheet

Device utilization summary sheet			
Logic utilization	**used**	**available**	**utilization**
Number of slice register	4242	12480	33%
Number of slice LUT's	3504	12480	28%
number of bonded IOB's	278	172	161%
Number of fully used LUT-FF pair	1907	5839	32%
Number of BUFG/ BUFCTRLS	8	32	25%

Table 2: Performance comparison of FPGA implementation of cryptographic algorithm

Cipher	Key size	Block size	FPGA device	Total occupied slices	Max. freq. (MHz)
Hummingbird[2]	256	16	Spartan-3 XC3S200-5	273	40.1
XTEA**[5]**	128	64	Spartan-3 XC3S50-5	254	62.6
SEA**[6]**	126	126	Virtex-2 XC2V4000	424	145
ICEBERG**[7]**	128	64	VIRTEX-2	631	-
AES**[8]**	128	128	Spartan-2 XC2S30-6	522	60
AES**[9]**			Spartan-3XC3S2000-5 Spartan-2 XC2S15-6	17425 264	196.167
AES**[10]**			Spartan-2 XC2V40-5	1214	123
AES**[11]**			Spartan-3	1800	150
This work	256	16	Virtex-5 XC5V1X20T-2-FF-323	4242	152.905

5. CONCLUSION

An FPGA implementation of the Hummingbird Cryptographic algorithm based on Virtex-5 xc5v1x20t-2-ff-323 of Xilinx Design Suite is presented in this paper. The simulation results show that our design consumes low power,262.57 mW for 2.5 V at 152.905 Mhz which is the maximum frequency of the cryptosystem.

The low power and High speed FPGA implementation is very precisely achieved by the proposed algorithm due to its prominent internal structure. Hence this high performance ultra-lightweight hybrid model will meet the power consumption requirements with constricted response time for diverse embedded applications and can be widely suitable for hardware environment. The design can be implemented on every electronic system which is the part of mobile adhoc network to prevent the security breach.

REFERENCES:

[1] Markku-Juhani O. Saarinen," Cryptanalysis of Hummingbird-1", Revere Security,16 Feb 2011.

[2] Xinxin Fan; Guang Gong; Lauffenburger, Hicks, "FPGA implementations of the Hummingbird cryptographic algorithm" , 2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp.48-51, 13-14 June 2010.

[3] Nikita Arora, Yogita Gigras, " Lightweight VLSI Design of Hybrid Hummingbird Cryptographic Algorithm", In proc. Of the International Journal of VLSI and Embedded Systems-IJVES, Vol 05, Article 03261; March 2014.

[4] Revini S. Shende, Mrs. Anagha Y. Deshpande, "VLSI Design Of Secure Cryptographic Algorithm", In proc. Of the International Journal of Engineering Research and Applications, Vol 3,Issue 2,March-April 2013.pp 742-746 .

[5] J.-P. Kaps, "Chai-Tea, Cryptographic Hardware Implemenations of xTEA", The 9th International Conference on Cryptology in India -INDOCRYPT 2008, LNCS 5356, pp. 363-375, 2008.

[6] F. Mace, F.-X. Standaert, and J.-J. Quisquater, "FPGA Implemenation(s) of a Scalable Encryption Algorithm", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 16, no. 2, pp. 212-216, 2008.

[7] F.-X. Standaert, G. Piret, G. Rouvroy, and J.-J. Quisquater, "FPGA Implementations of the ICEBERG Block Cipher", Integration, the VLSI Journal, vol. 40, iss. 1, pp. 20-27, 2007.

[8] P. Chodowiec and K. Gaj, "Very Compact FPGA Implementation ofthe AES Algorithm", The 5th International Workshop on Cryptographic Hardware and Embedded Systems - CHES 2003, LNCS 2779, pp. 319-333, 2003.

[9] T. Good and M. Benaissa, "AES on FPGA from the Fastest to the Smallest", The 7th International Workshop on Cryptographic Hardware and Embedded Systems - CHES 2005, LNCS 3659, pp. 427-440, 2005.

[10] G. Rouvroy, F.-X. Standaert, J.-J. Quisquater, and J.-D. Legat, "Compact and Efficient Encryption/Decryption Module for FPGA Implementation of the AES Rijndael Very Well Suited for Small Embedded Applications", International Conference on Information Technology: Coding and Computing - ITCC 2004, pp. 583-587, 2004.

[11] P. Bulens, F.-X. Standaert, J.-J. Quisquater, and P. Pellegrin, "Implementation of the AES-128 on Virtex-5 FPGAs", Progress in Cryptology - AFRICACRYPT 2008, LNCS 5023, pp. 16-26, 2008.

Authors Biography:

Nikita Arora[1] is a M.Tech student in the department of Computer science & Engineering, ITM University, Gurgaon. She received her B.Tech degree in Computer science & Engineering from GGSIPU, India. Her research interests include Cryptography, network security and software engineering domain.

Yogita Gigras[2]**,** currently working as an Assistant Professor in the department of Computer science & Engineering, ITM University, Gurgaon, India. She is working on a PhD in Soft Computing. Her areas of interest include Analysis and Design of Algorithms, Object Oriented Programming, Operating System, Computer Networks, and Soft Computing.