IP CORE DESIGN OF HIGHT LIGHTWEIGHT CIPHER AND ITS IMPLEMENTATION

Sruthi.N[1], R.Nandakumar[2] and Rajkumar.P[3]

[1]Student, VLSI DESIGN, Department of Electronics and Communication Engineering, NCERC, University of Calicut
shruthishanti@gmail.com
[2]Scientist 'C', NIELIT, Calicut
nanda@nielit.gov.in
[3]Faculty, Department of Electronics and Communication Engineering, NCERC, University of Calicut
rajkumar1073@ncerc.ac.in

ABSTRACT

In the present era of e-world where security has got a larger weightage, cryptography has its role to play. Nowadays, the devices available in the market are of resource constrained type. Hence we need lightweight ciphers for the efficient encryption of data thereby increasing the performance. In this project a detailed study of HIGHT cryptographic algorithm is done which outperforms standard algorithms. HIGHT is an ISO Standard block cipher which has 64-bit block length and 128-bit key length. HIGHT was designed to be proper for the implementation in the low resource environment such as WSN, WBN, RFID tag or tiny ubiquitous devices. It is implemented on Spartan 6 FPGA evaluation kit and performance metrics are found out. A HIGHT cryptocore is being designed, characterized and implemented which will be a reference platform for hardware design engineers to model devices which require lightweight characteristics.

KEYWORDS

HIGHT, Lightweight cryptography, low resource devices, FPGA

1. INTRODUCTION

For secret communication there is a need of hidden writing and this part of science is called cryptography With the help of cryptography we are able to achieve data integrity, data confidentiality and authentication. In such cases, certain protocols are created and analyzed and they are known as ciphers. These ciphers are the combination of mathematics, computer science and electrical science. They are mainly used in e-commerce, ATM passwords and other applications where there is a need of privacy. In today's world everyone needs privacy for communication hence cryptography has a major role to play. Ciphers are basically classified into Symmetric ciphers and Asymmetric ciphers. There is a common key for encryption and

Natarajan Meghanathan et al. (Eds) : ACITY, VLSI, AIAA, CNDC - 2016
pp. 97–105, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60909

decryption in symmetric ciphers whereas in asymmetric ciphers there is a public key to encrypt and private key to decrypt. Hence data manipulation is done. Symmetric ciphers are further classified as block ciphers and stream ciphers. In block ciphers, data is being divided into blocks of particular size and whereas in stream ciphers bit by bit manipulation of the data is being done. The block ciphers can be transformed into stream ciphers by operating in OFR and CTR modes. In stream ciphers, hidden internal state changes as the cipher operates. Block ciphers are better analyzed and has got broader range of applications. The basic 2 properties of the ciphers are diffusion and confusion. Diffusion dissipates statistical structure of plaintext over ciphertext (redundancies are dissipated) whereas confusion property gives the relationship between cipher text and key as complex as possible. The basic design elements of a cipher include block size, key size, number of rounds, subkey generation algorithm, round function, fast software en/decryption and ease of analysis. Block ciphers are iterated ones i.e they transform fixed size blocks of plaintext into identical size ciphertext through the repeated application of an invertible transformation known as round function. Round functions take different round keys k as second input which are derived from the original key. The design criteria for ciphers are efficiency. In block ciphers usage of Sbox leads to larger hardware footprint. Memory expense is the major constraint of designing a block cipher. Based on the structure of algorithm, the block ciphers are classified into SP networks and Feistel networks. The main advantages of using fiestel network are that en/decryption operations are very similar i.e only reversal of key schedule is required. The cryptographic algorithms developed before 1990s was mainly focused to work on standard devices which consume larger area and power.[2] But gradually the devices were made to work in the resource constrained environment. For securing such devices, lightweight ciphers were invented. These ciphers are developed bit away from industry demands. The design criteria of lightweight ciphers are efficiency, simplicity and security. The block size can be 32,48 or 64 bits and key size can be 80 or 128 bits. The power, area consumption of lightweight ciphers is minimum.

In this paper, HIGHT cryptographic algorithm is implemented in both software and hardware platform. The results and the resource utilized by the design is also given.

2. HIGHT

The block cipher HIGHT was developed in Korea. HIGHT is the shortform of HIGh security and lightweight. HIGHT is a ISO/IEC 18033-3:2010 which has 64 bit input /output data block with no Sbox, 32 round with XOR, modular addition and circular shift operations. The HIGHT algorithm is defined below,

The entire plain text is divided into 8 subtexts, each 8 bit each. From the 128 master keys are being divided into 16 keys, 8 bit each. 8 whitening keys are generated from the master keys and the 128 subkeys from the constant generation algorithm. Out of these 8 whitening keys, first 4 are used in the initial transformation of the plain text and last 4 are used in the final transformation. Constant generation algorithm is based on a 7-bit LFSR. The 7 bits i.e initial state of the LFSR is ‘0101101’ and from this basic constant, by doing the XOR operation of last 2 bits next constant is being generated and the process is continued to generate further 127 constants.[1] These 128 constants along with the master keys are used to generate 128 subkeys. In the 32 rounds of HIGHT, each round uses 4 subkeys for the operations.

The plain text P

(1) P=P7||P6||P5||P4||P3||P2||P1||P0

Master Key K

K=K15||K14||K13||K12||K11||K10||K9||K8||K7||K6||K5||K4||K3||K2||K1||K0

Whitening and Subkey generation

a) The generation of whitening keys is defined as follows

for i = 0,1,2,3:
WKi=K(i+12)
For i=4,5,6,7:
WKi=K(i-4)

b) The 128 subkeys are used for encryption and decryption, 4 subkeys per round . The generation of sub keys is defined as follows.

(1) s0 = 0 , s1 = 1, s2 = 0, s3 = 1, s4 = 1, s5 = 0, s6 = 1
d0 = s6 || s5 || s4 || s3 || s2 || s1|| s0
(2) for i =1 to 127 ;
s (i+6) = s(i+2) [^] s(i-1)
di = s(i+6) || s(i+5) || s(i+4) || s(i+3) || s(i+2) || s(i+1) || s(i)
(3) for i = 0 to 7 ;
for j = 0 to 7 ;
SK (16 * i + j) = K (j - i mod 8) [+] d (16 * i + j)
for j= 0 to 7 ;
SK (16 * i + j + 8) = K ((j – i mod 8) + 8) [+] d(16 * i + j + 8)

Initial transformation

(2) X0,0 = P0 [+] WK0
X0,2 = P2 [^] WK1
X0,4 = P4 [+] WK2
X0,6 = P6 [+] WK3
X0,1 = P1
X0,3 = P3
X0,5 = P5
X0,7 = P7

32 rounds

(3) For i = 0 to 30:

X(i+1),0 = Xi,7 [^] (F0(Xi,6) [+] SK(4*i + 3))
X(i+1),2 = Xi,1 [+] (F1(Xi,0) [^] SK(4*i))

X(i+1),4 = Xi,3 [^] (F0(Xi,2) [+] SK(4*i + 1))
X(i+1),6 = Xi,5 [+] (F1(Xi,4) [^] SK(4*i + 2))
X(i+1),1 = Xi,0
X(i+1),3 = Xi,2
X(i+1),5 = Xi,4
X(i+1),7 = Xi,6

For i=31:

X(i+1),1 = Xi,1 [+] (F0(Xi,0) [^] SK124)
X(i+1),3 = Xi,3 [^] (F1(Xi,2) [+] SK125)
X(i+1),5 = Xi,5 [+] (F0(Xi,4) [^] SK126)
X(i+1),7 = Xi,7 [^] (F1(Xi,6) [+] SK127)
X(i+1),0 = Xi,0
X(i+1),2 = Xi,2
X(i+1),4 = Xi,4
X(i+1),6 = Xi,6

Final transformation

(4) C0 = X32,0 [+] WK4
C2 = X32,2 [^] WK5
C4 = X32,4 [+] WK6
C6 = X32,6 [^] WK7
C1 = X32,1
C3 = X32,3
C5 = X32,5
C7 = X32,7

Final Cipher Text

5) C=C7||C6||C5||C4||C3||C2||C1||C0

The F0 and F1 round functions are:

F0(x)=(x<<<1) [^] (x<<<2) [^] (x<<<7)
F1(x)=(x<<<3) [^] (x<<<4) [^] (x<<<6)

The decryption operation is identical in operation to encryption apart from the following two modifications

(1)All [+] operations are replaced by [-] operations except for the [+] operations connecting SKi and outputs of F0

(2)The order in which the keys WKi and SKi are applied is reversed.

The toplevel block diagram of HIGHT is shown below

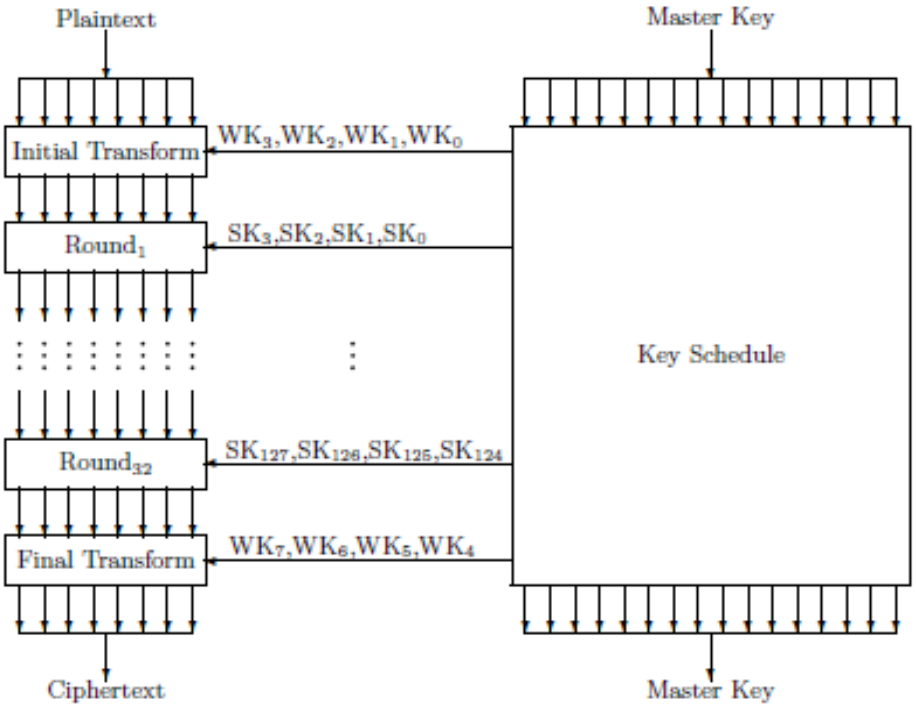

Figure 1. Toplevel Diagram

3. SOFTWARE PLATFORM IMPLEMENTATION

All the cryptographic algorithms are implemented on a software platform so that their behavior in such an environment is found out. The software platform implementation mainly aims at optimization of speed, memory size, power or energy.

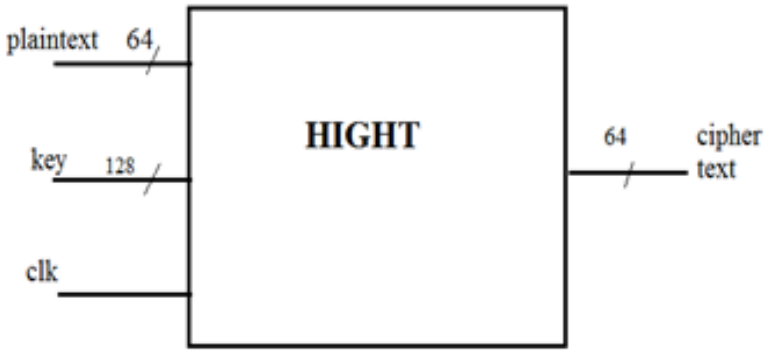

Figure 2:Input Output Diagram

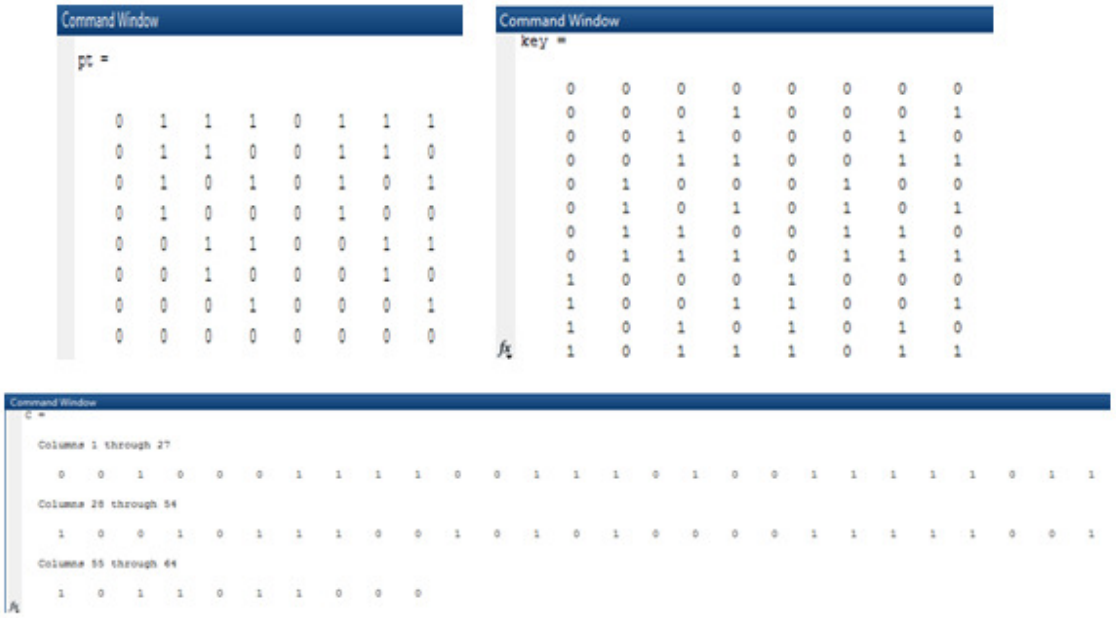

Figure 3: MATLAB Result

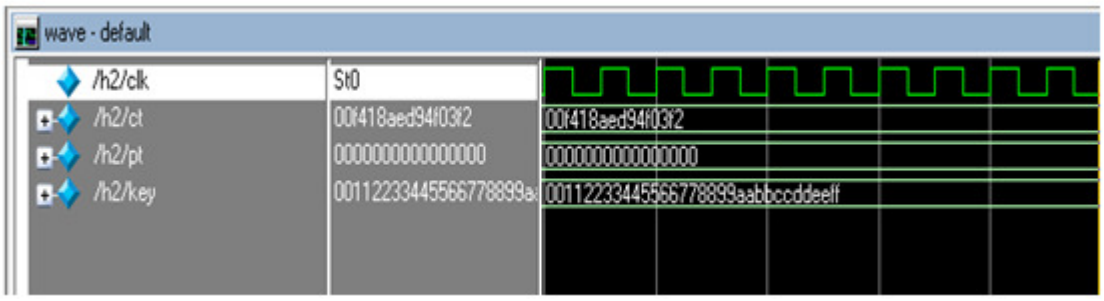

Figure 4: ModelSim Result

The HIGHT algorithm was implemented in MATLAB and MODELSIM 6.2c. Based on the input- output diagram of HIGHT, the Verilog code was created and was implemented in the software environment. The MATLAB calculator for the HIGHT was created and it was verified.To simulate the Verilog code, the code was run on the ModelSim and the results were found out. The results obtained are shown.

Table 1:Function Table

PLAINTEXT	MASTERKEY	CIPHERTEXT
0011223344556677	ffeeddccbbaa99887766554433221100	23ce9f72e543e6d8
0000000000000000	00112233445566778899aabbccddeeff	00f418aed94f03f2
0123456789abcdef	00112233445566778899aabbccddeeff	73aa299327a22684
0123456789abcdef	ffeeddccbbaa99887766554433221100	8181e2a70f8346f7
0000000000000000	ffeeddccbbaa99887766554433221100	3181ff9102b64cca

4. HARDWARE PLATFORM IMPLEMENTATION

Hardware implementations are mainly done on FPGA and ASIC technology. In ASIC, main aim is to reduce the design time. Comparing to ASIC implementation, FPGA is more advantageous because it provides flexibility, agility of algorithms and modifications are made easier. The Verilog code was run on Xilinx 14.3 and the synthesis results were obtained. The code was implemented on a Spartan -6 evaluation kit XC6SLX45T-3FGG484.

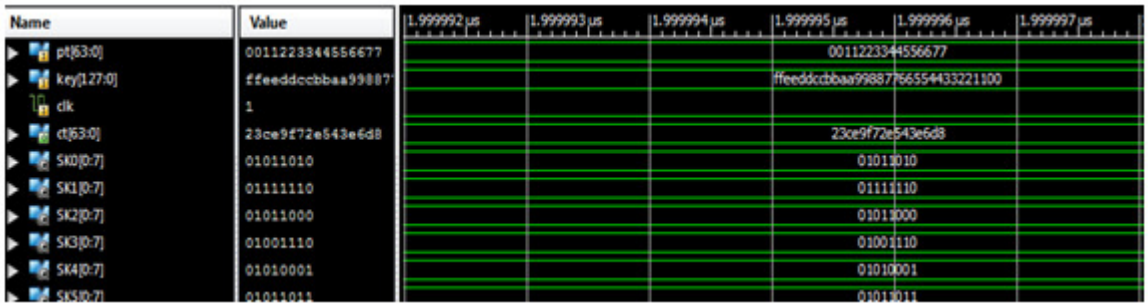

Figure 5: Xilinx Result

5. ONCHIP DEBUGGING AND PROTOTYPING RESULTS

Finally design is being analyzed using the ChipScope Pro Analyzer and on chip results were obtained. These results were used to compare with the simulation and synthesis results .The results obtained are shown below

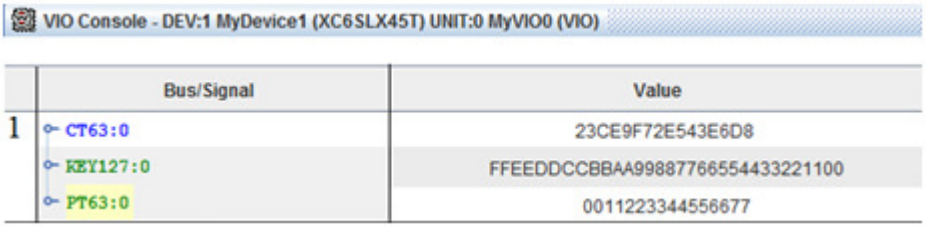

Figure 6 : On Chip Debugging Results

Power Analysis report is obtained on XPA tool on the Spartan-6 kit and from the report the power consumed by the design is equal to 0.037 W. After implementing on the FPGA kit, the design was implemented on ASIC platform and the area, power and timing details were obtained. The area consumed was found to be 0.22µm2 and power consumed was found to be 0.06 nW .And the maximum frequency of operation is found to be 119.847 MHz . These results obtained from the ASIC implementation are used to calculate the performance metrics of the HIGHT cryptographic algorithm. Calculated throughput is 767.018Mbps.

Table 2 :Resource Utilization Summary

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	2,409	54,576	4%
Number used as Flip Flops	2,409		
Number of Slice LUTs	2,689	27,288	9%
Number used as logic	2,453	27,288	8%
Number used exclusively as route-thrus	236		
Number of occupied Slices	991	6,822	14%
Number of MUXCYs used	568	13,644	1%
Number of bonded IOBs	1	296	1%
Number of BSCANs	1	4	25%
Average Fan-out of Non-Clock Nets	3.85		

6. CONCLUSIONS

This paper focuses on the characterization of HIGHT algorithm and has developed an IP Core of HIGHT which will be reference one for the design engineers. A detailed study on HIGHT block cipher was done and carried out its algorithm validation. HIGHT block cipher is a lightweight block cipher of block size 64 bit and key size 128 bit targeted to provide cryptographic security for resource constrained applications e.g. RFID, sensor networks etc. The behavioural description of the design is written in Verilog HDL and simulated using XilinxISE 14.3 and ModelSim 6.2 c software platforms. Then the design is successfully implemented on Xilinx Spartan6 FPGA. The performance metrics were found out and the results are presented. A detailed analysis of HIGHT cryptographic algorithm was done. In-depth analysis of linear and differential attacks needs to be carried out.

ACKNOWLEDGEMENTS

The gratification of this Project will be incomplete without mentioning all the people who helped me to make it possible, whose gratitude and encouragement were valuable to me.I would like to thank my guides for their whole hearted support. I would also like to thank my parents and friends who encouraged me and gave me the motivation to complete the work. Above all I would like to thank God for his abundant grace.

REFERENCES

[1] Deukjo Hong, Jaechul Sung, Seokhie Hong, Jongin Lim, “HIGHT: A New Block Cipher Suitable for Low-Resource Device,” Springer 2006.

[2] Mohd Bj, et al, “A survey of lightweight block ciphers for low-resource devices-Comparative studies and open issues,” Journal of Network and Computer Application , 2015

[3] B.Han, H.Lee, H.Jeong, “The HIGHT Encryption Algorithm,” Internet Working group,June 24,2011

[4] Woo Kwon Koo, Hwa seong Lee, Yong Ho Kim, Dong Hoon Lee, "Implementation and Analysis of New lightweight cryptographic algorithm suitable for wireless sensor networks ," International Conference on Information Security and Assurance, IEEE, April 2008

[5] Panasayya Yalla and Jens-Peter Kaps, "Lightweight Cryptographys for FPGA ," International Conference on Reconfigurable Computing and FPGAs, IEEE, Dec 2009, pp. 225–230.

[6] Fernando Melo Nascimento , Fernando Messias dos Santos , Edward David Moreno, "A VHDL Implementation of the Lightweight Cryptographic Algorithm HIGHT ,"

AUTHORS

Sruthi. N did BTech from NSS College of Engineering in Electronics and Communication during 2010-2014 under University of Calicut. Doing MTech in VLSI DESIGN at Nehru College of Engineering (2014-2016) under University of Calicut.

R. Nandakumar working as Scientist 'C' at NIELIT, Calicut. ME in Communication Systems and MBA in Project Management. Area of specialisation includes VLSI DESIGN and Communication Engineering. Coordinator for PG Diploma VLSI Design, Coordinator for PG Diploma ESDM Resource Person for STTPs, Corporate & Industrial Training & Collaborative Workshops, IEEE NIELIT SB Counselor. In charge of NBA Accreditation

P.RajKumar has been working as Senior Assistant Professor at Nehru College of Engineering and Research Centre in the Electronics and Communication Engineering Department since June 2013. His educational qualifications include Master of Engineering (M.E) in the specialization Communication Systems and Bachelor of Engineering (B.E) in Electronics and Communication Engineering. His areas of research interest comprise Image Processing, Networks and VLSI Design.