Design And Characterization of LBlock Cryptocore*

Aljazeera.K.R
Mtech VLSI Design
NCERC, University of Calicut
Thrissur, Kerala, India
aljec05@gmail.com

Nandakumar.R
Scientist/Engineer(C)
NIELIT
Calicut, Kerala, India
nanda24x7@gmail.com

Ershad.S.B
Associate Professor
NCERC, University of Calicut
Thrissur, Kerala, India
ershadsb@gmail.com

Abstract— **Today's headlines abound with the promised explosion of the Internet of things. Studies claim that there will be the existence of billions of internet connected devices in coming years. The small size and limited processing power of many connected devices could inhibit encryption and other robust security measure. Here cryptography plays a pivotal role. Also most of the devices available in the market today are resource constrained. Hence cryptographic solutions must be easy to implement and have high performance on a wide range of severely constrained devices. The relatively new field of lightweight cryptography provides significant advantages over existing algorithms when addressing security issues for highly constrained devices. In this project a detailed study of LBlock block cipher is done which is a lightweight cipher in both hardware and 8-bit platforms and an IP core is developed. The block size of LBlock is 64-bit and the key size is 80-bit and it can achieve competitive hardware and software performances when compared with other known lightweight block ciphers. The security evaluation shows that LBlock can achieve enough security margin against known attacks such as differential cryptanalysis, linear cryptanalysis, impossible differential cryptanalysis and related-key attacks etc. LBlock cryptocore is designed, build and characterized for the efficient implementation in low resource devices. It can serve as a benchmark for the hardware design engineers to model devices that utilizes lightweight characteristics. LBlock cipher is implemented on Xilinx Spartan-6 FPGA (XC6LX16-CS324) and its performance metrics were obtained.**

Keywords—Cryptography, Internet of things, Lightweight cryptography, LBlock, Performance metrics, Xilinx.

I. INTRODUCTION

Every day hundreds of thousands of people interact electronically through e-mail, e-commerce, ATM Machines, cellular phones etc. The perpetual increase of information transmitted electronically has led to an increased reliance on cryptography. Cryptography is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography is very important to the continued growth of the internet and electronic commerce. Small internet-enabled appliances expected to flood the markets as the Internet of Things (IoT) arises. The mass deployment of these devices brings serious concerns for security and privacy. It is not easy to implement sufficient cryptographic functions on constrained devices due to limitations of their resources. The traditional cryptographic algorithms may not be suitable for such devices as they have limited memory and computational power along with serious power constraints. This led to the development of new improved branch of cryptography called Lightweight Cryptography (LWC). Lightweight properties described based on target platforms. "As light as a feather, and as hard as dragon-scales". It is an appropriate description for Lightweight cryptography. On the one hand lightweight cryptography aims to yield very lightweight implementations that are virtually "light as a feather", but on the other hand without conceding the security level too much. In fact, one major aspect of lightweight cryptography is to exploit the security-efficiency trade-offs inherent in implementations of cryptographic algorithms. "Hard as dragon scales" is a good paraphrase for this aspect, because it emphasizes that there are sufficient security levels (e.g. 80 bit key size) beside a theoretical optimal one. Lightweight block ciphers are practical to use now and has small block size (32, 48, and 64). The key size is also smaller. It simplifies key schedule and performs elementary operations with larger number of rounds. It optimize resource utilization with minimum power and energy consumption, meeting security challenges.

The increasing gap between design productivity and chip complexity, and emerging systems-on-a-chip (SoC) have led to the wide utilization of reusable intellectual property (IP) cores. An Intellectual Property (IP) core in VLSI is a reusable unit of logic or functionality or a cell or a layout design that is normally developed with the idea of licensing to multiple vendor for using as building blocks in different chip designs. Ideally, an IP core should be entirely portable i.e., able to easily be inserted into any vendor technology or design methodology. The main features of IP core include ease of reusability, ease of integration, ease of scalability and debugging. In this paper the design and characterization of LBlock block cipher cryptocore is carried out aiming at constrained devices. The rest of the paper is organized as follows. Section II presents the LBlock block cipher, section III carries the Proposed LBlock cryptocore, section IV

978-1-5090-4620-1/16/$31.00 ©2016 IEEE

provides the Performance evaluation followed by Conclusion in section V.

II. LBLOCK BLOCK CIPHER

LBlock (LuBan lock) is a lightweight block cipher proposed by Wenling Wu and Lei Zhang in ACNS2011. It ciphers blocks of size 64 bits under keys of size 80 bits using 32 rounds of modified Fiestel network. In the ciphers that uses fiestel structure the block to be encrypted is split into two equal-sized halves. Round function is applied to one half using a subkey and the output is xored with other half. The two halves are then swapped to get the final ciphertext. The specification of LBlock consists of three parts, encryption, decryption and key scheduling. The basic process is shown below in *"Fig.1"*

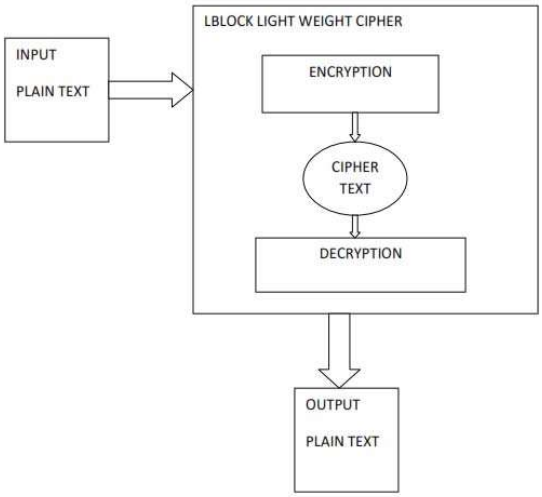


Fig.1. Simple block diagram

The LBlock cipher consists of ten s-boxes in parallel. The round function uses eight 4 X 4 S-boxes from $s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7$. The other two s-boxes s_8, s_9 are used for key scheduling. The contents of s-boxes used in LBlock algorithm is given below in table I.

TABLE I. CONTENTS OF S-BOXES

s0	14	9	15	0	13	4	10	11	1	2	8	3	7	6	12	5
s1	4	11	14	9	15	13	0	10	7	12	5	6	2	8	1	3
s2	1	14	7	12	15	13	0	6	11	5	9	3	2	4	8	10
s3	7	6	8	11	0	15	3	14	9	10	12	13	5	2	4	1
s4	14	5	15	0	7	2	12	13	1	8	4	9	11	10	6	3
s5	2	13	11	12	15	14	0	9	7	10	6	3	1	8	4	5
s6	11	9	4	14	0	15	10	13	6	12	5	7	3	8	1	2
s7	13	10	15	0	14	4	9	11	2	1	8	3	7	5	12	6
s8	8	7	14	5	15	13	0	6	11	12	9	10	2	4	1	3
s9	11	5	15	0	7	2	9	13	4	8	1	12	14	10	3	6

A. Encryption

The given 64-bit plaintext M is divided into two separate sequences of equal length; X1 is the left 32-bit half and X0 is the right 32-bit half to encrypt using LBlock block cipher. This is denoted as:

$$M\ (64\text{-bit}) = X1\ (32\text{-bit}) \parallel X0\ (32\text{-bit}) \quad (1)$$

And then the data processing procedure can be expressed as follows

1. For i = 2,3,…….33,do

$$X_i = F(X_{i-1}, K_{i-1}) \oplus (X_{i-2} <<< 8) \quad (2)$$

2. Output $C = X32 \parallel X33$ as the 64-bit ciphertext (3)

The encryption procedure for LBlock block cipher is shown below in *"Fig.2"*.

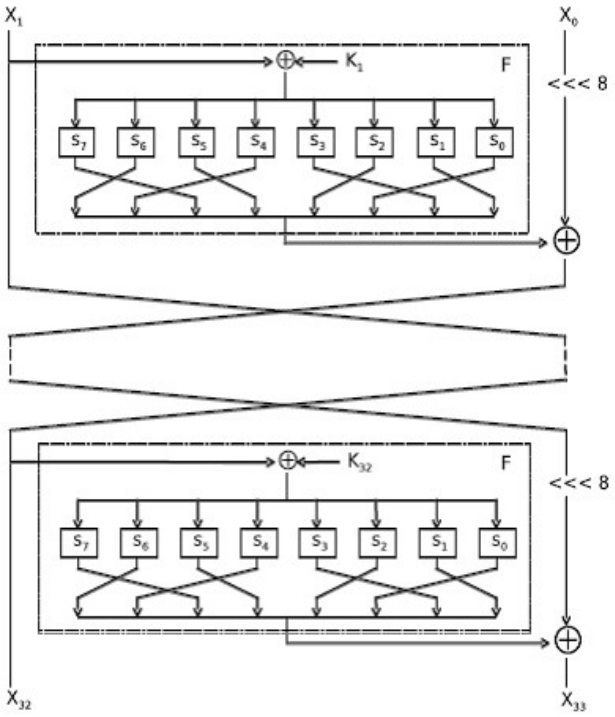


Fig.2. LBlock encryption

The encryption procedure involves round function, shifting and xor-ing operations. The components used in each round are: Round function-F which is constructed from two other functions, Confusion function S and Diffusion function P. It is defined as

$$U = F(X, Ki) = P\ (S\ (X \oplus Ki)) \quad (4)$$

Where U is the output of round function F and K_i denotes the sub key of each round.

Confusion function-S

The non-linear layer of the Round Function, F that consists of eight 4 X 4 S-boxes namely S_0, S_1, S_2, S_3, S_4, S_5, S_6, S_7. The confusion function S defined as

$$Y=Y_7 \parallel Y_6 \parallel Y_5 \parallel Y_4 \parallel Y_3 \parallel Y_2 \parallel Y_1 \parallel Y_0 \rightarrow Z \quad (5)$$

Where Y is the input to s box and Z is the output. The output of the s box is defined as Z and it is found as

$$Z7= s7\,(Y7),\ Z6= s6\,(Y6),\ Z5= s5\,Y5),\ Z_4= s4\,(Y4),$$
$$Z3= s3\,(Y3),\ Z2= s2\,(Y2),\ Z_1= s_1\,(Y_1),\ Z_0=s_0\,(Y_0). \quad (6)$$

$$Z = Z_7 \parallel Z_6 \parallel Z_5 \parallel Z_4 \parallel Z_3 \parallel Z_2 \parallel Z_1 \parallel Z_0. \quad (7)$$

Diffusion function-P

Diffusion function P is defined as a permutation of eight 4-bit words, and it can be expressed as the following equations.

$$Z = Z_7 \parallel Z_6 \parallel Z_5 \parallel Z_4 \parallel Z_3 \parallel Z_2 \parallel Z_1 \parallel Z_0 \rightarrow U \quad (8)$$

The output of round function is U and is obtained as

$$U_7=Z_6, U_6=Z_4, U_5=Z_7, U_4=Z_5,$$
$$U_3=Z_2, U_2=Z_0, U_1=Z_3, U_0=Z_1. \quad (9)$$

$$U= U_7 \parallel U_6 \parallel U_5 \parallel U_4 \parallel U_3 \parallel U_2 \parallel U_1 \parallel U_0. \quad (10)$$

The round function complete operation is shown below in the *"Fig.3"*.

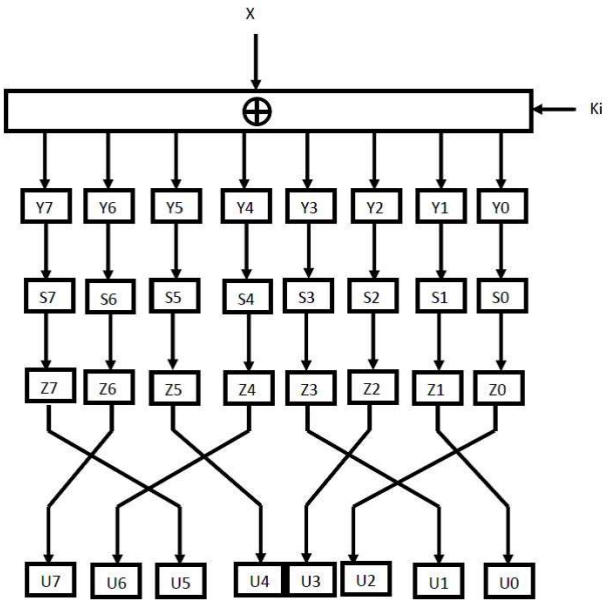


Fig 3. Round function operation

B. Decryption

The decryption algorithm of LBlock is the inverse of encryption procedure and it consists of a 32-round variant Fiestel structure too. Let $C = X_{32} \| X_{33}$ denotes a 64-bit ciphertext, and then the decryption procedure can be expressed as follows

1. For j=31, 30 ….0, do

$$X_J = (F\,(Xj+1,\ Kj+1) \oplus Xj+2) >>> 8 \quad (11)$$

2. Output M=X1‖X0 as the 64-bit plaintext. (12)

C. Key scheduling

The Key Schedule of LBlock block cipher accepts an 80-bit master key, $K = k_{79}, k_{78}, k_{77},\ldots, k_1, k_0$ which is stored in a key register. After 32 rounds of execution, this algorithm will produce 32 round sub keys denoted as K_i.

Step 1: Round sub key, K_1 is the leftmost 32 bits of the master key, K.

Step 2: For $1 \le i \le 31$, the key register is updated as follows:-

a) K<<<29

b) $[k_{79}, k_{78}, k_{77}, k_{76}] = s_9\,[k_{79}, k_{78}, k_{77}, k_{76}]$
$[k_{75}, k_{74}, k_{73}, k_{72}] = s_8\,[k_{75}, k_{74}, k_{73}, k_{72}]$

c) $[k_{50}, k_{49}, k_{48}, k_{47}, k_{46}]$ xor $[i]_2$.

d) Round sub key, K_{i+1} is the leftmost 32 bits of the current key register.

The key scheduling process is shown below in *"Fig.4"* and *"Fig.5"*.

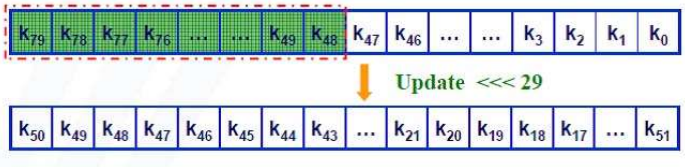


Fig.4. Key scheduling part1

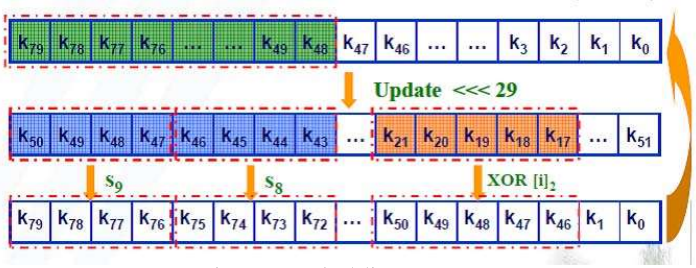


Fig.5. Key scheduling part 2

III. PROPOSED LBLOCK CRYPTOCORE

Nobody wants to invent the bicycle again. Modules, which are widely used, are available for different technologies, not always for free. One such module can have many parameters and a special software to set them properly. Each module should come with a reference design and a testbench in VHDL/Verilog. The source code is typically not available, or is just a question of price. In general the usage of IP cores saves time and money. Here an IP core for LBlock lightweight block cipher is designed. The top level architecture of the designed LBlock cryptocore is given below in the *"Fig.6"*.

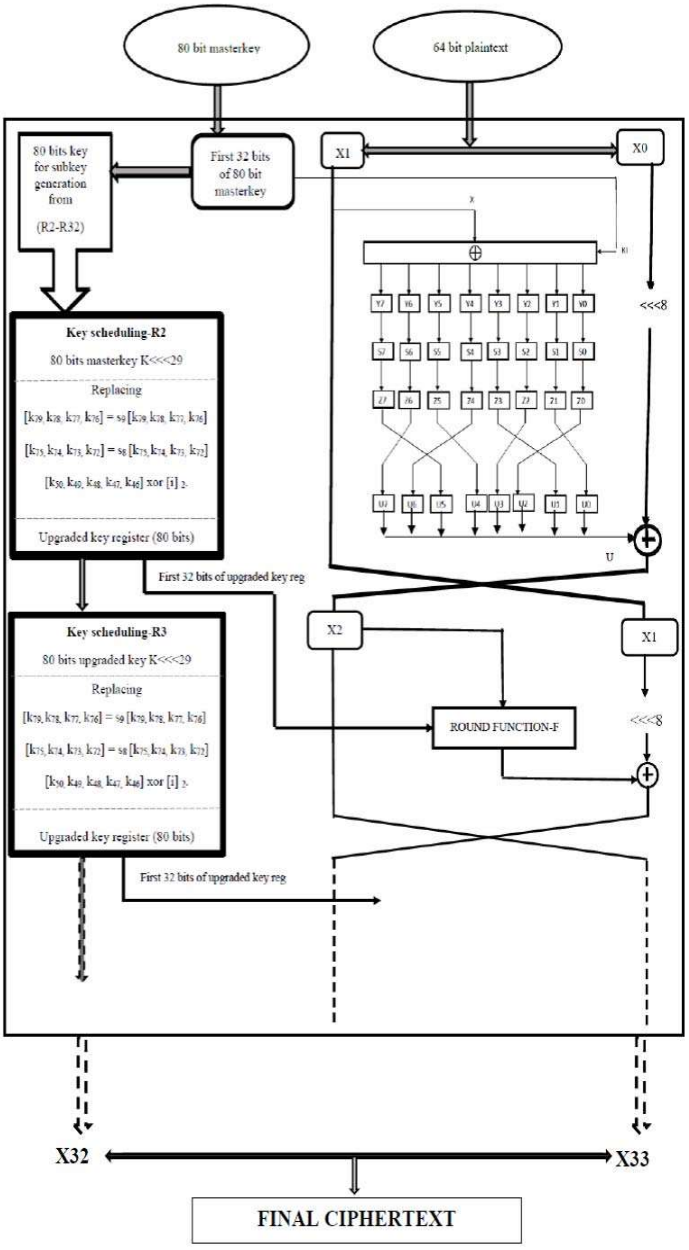


Fig.6. Top level architecture of LBlock cryptocore

In this proposed work the most direct way of implementation of the LBlock encryption is done. This is to reduce the errors that can occur during designing. The output of one round serves as the input to the next round, in both plaintext mixing and key scheduling. The round function operation of each round depends on the subkey. Hence both the intermediate values of ciphertext and subkey generation are interdependent. The design is organized in such a way that only after the completion of one round another begins. It is executed sequentially. Finally the two sub ciphertext or the intermediate values of round 31 and 32 are concatenated to produce the ciphertext. The structure of the design is iterated.

IV. PERFORMANCE EVALUATION

A. *Software model using MATLAB*

Firstly the LBlock block cipher encryption algorithm is modelled using MATLAB R2013a and calculator is created for the same. Then function table is built for various texts and the output encrypted texts.64-bit plaintext is given as input along with 80-bit master key and 64-bit ciphertext is obtained after encryption. The results are shown below:

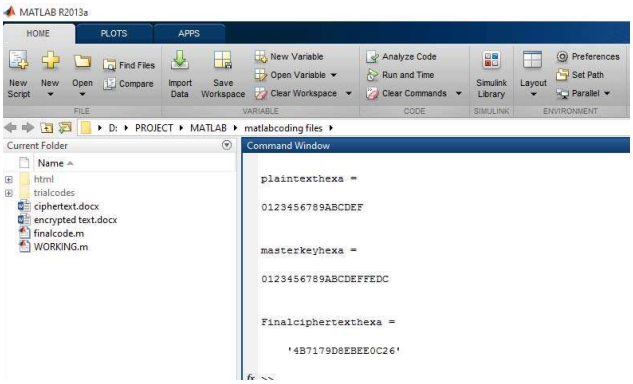


Fig.7. MATLAB Result

The intermediate values are obtained and verified. Here the modeling is carried out in the most direct form. The inputs and outputs are expressed in hexadecimal. The function table created for various input values is given below.

TABLE II. FUNCTION TABLE

PLAINTEXT	MASTERKEY	CIPHERTEXT
0000000000000000	00000000000000000000	C218185308E75BCD
0123456789ABCDEF	0123456789ABCDEFFEDC	4B7179D8EBEE0C26
ABADCBE098765432	12345678987654321042	2A076AAB07A8FE49
012A345B678C9DEF	13C3B58789ABCDEFFEDC	1EF60498ACA69A46
1000200030004000	10012002300340045005	8D17319ECAB1C084

B. *Software Platform Implementations*

For devices of constrained environments, software platform implementations have its own significance. Software implementations method will provide structured approach systematically to integrate a component into the workflow of an organizational structure or an individual end-user. This entry focuses on the process modeling (Process Modeling) side of the implementation. It makes it much easy to build application as it provides all necessary building blocks to implement specific functionalities. Since platform provides readymade functionalities and library, time consumed for software development life cycle is expected to reduce as both development and testing time consumption can be reduced.

The LBlock encryption core is simulated on ModelSim SE 6.2c using Verilog hardware description language and the results are compared and verified with the previous ones obtained from MATLAB. The results are found to be in order. Simulation is performed using the graphical user interface

(GUI), or automatically using scripts. The results are shown below.

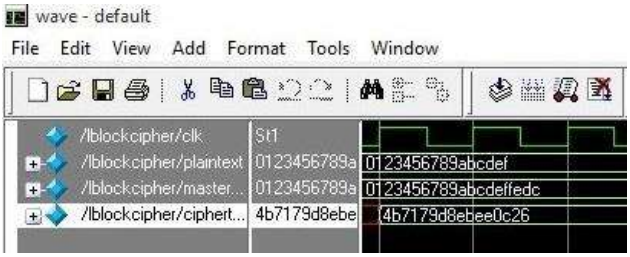


Fig.7. ModelSim Result

C. *Hardware platform implementation and metrics*

In hardware designs area, timing, energy, power and efficiency metrics are the basic performance metrics. Based on the type and completeness of the implementation, design area utilized will be different. Hardware designs are typically implemented using full-custom design, ASIC or FPGA technology. ASIC designs are based on automated design flows to lessen the design time. In case of ASIC implementation, the area is expressed in μm^2, is given by physical design tools [2]. For pre-layout design area is expressed in the number of gate equivalent (GE).The throughput is a function of design frequency [2]. Power and energy are essential performance metrics for ciphers targeted for energy constrained low-resource devices. Power is also dependent on clock frequency similar to throughput. The average power dissipation indicates the rate of energy consumption.

The input-output diagram gives the black box representation of a design. The overall idea of the design can be easily obtained from it. It should be compact and easy to understand. The design has three inputs: plaintext, masterkey and clock. The plaintext is 64 bit wide and masterkey is 80 bit wide. The output is the ciphertext which is 64 bit wide. The input-output diagram is given below.

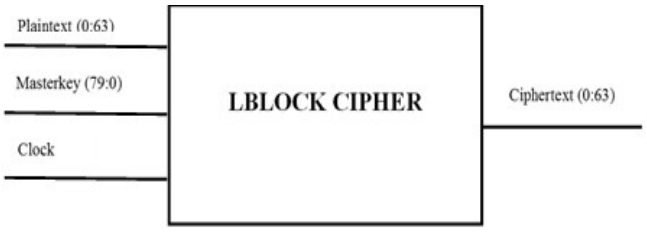


Fig 8: Input-output diagram for LBlock Cipher

Compared with ASIC designs and full-custom design, FPGA designs provide advantages such as reduced development cost, shorter time to market and flexibility. The other benefits include algorithm agility, upgrading device by new algorithm uploading and algorithm modification. The following list of results will give the hardware performance metrics for the LBlock block cipher cryptocore.

1) Synthesis result.

Xilinx ISE 14.7 by Xilinx for synthesis and analysis of HDL designs, enabling the developer to synthesize their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer is used as a software tool for software platform implementations. Verilog code developed for the algorithm and it is simulated. The following results are obtained.

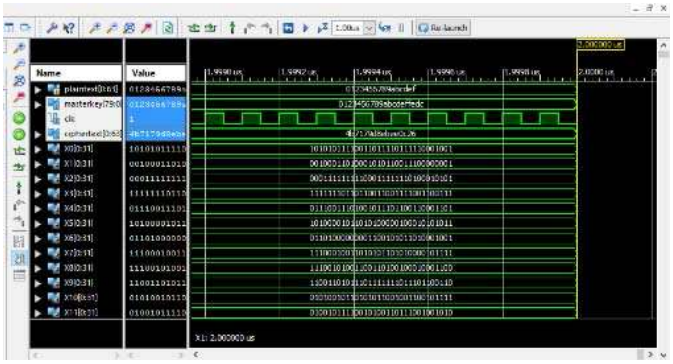


Fig.9.Xilinx Result

The LBlock encryption is implemented in Verilog and synthesized it on Xilinx Spartan-6 (XC6LX16-CS324) FPGA Kit to check for its hardware performance. Xilinx ISE 14.7 is used as FPGA development environment during the implementation process (i.e., synthesis, map and place& route).

The table below shows the overall resource utilized during implementation process.

TABLE III. RESOURCE UTILIZATION SUMMARY

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	3,188	18,224	17%
Number used as Flip Flops	3,188		
Number of Slice LUTs	3,275	9,112	35%
Number used as logic	2,807	9,112	30%
Number used as Memory	334	2,176	15%
Number used as Shift Register	334		
Number used exclusively as route-thrus	134		
Number of occupied Slices	1,400	2,278	61%
Number of MUXCYs used	16	4,556	1%
Number of bonded IOBs	1	232	1%
Number of BSCANs	1	4	25%
Average Fan-out of Non-Clock Nets	3.23		

The resource utilized by the design is obtained after synthesis. 3188 slice registers are used of the available 18,224 slices. The utilization percentage is only 17%. The cryptocore design occupies only 15% of the memory. Hence the space is reduced in the design.

2) Power Utilization Summary

FPGAs are widely used in many applications due to its advantages over others. FPGAs can provide huge benefits to the system design by reducing the power consumption. The power analysis for LBlock cipher is done using XPower Analyzer (XPA) tool on Spartan-6 FPGA. Xilinx Power tool performs power estimation and analysis for the design given. The result obtained after running XPA is 0.021 W. The results are shown below in *"Fig.10"*.

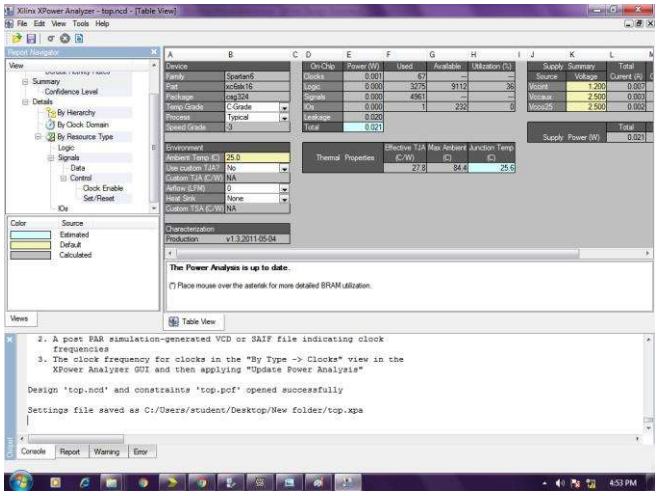
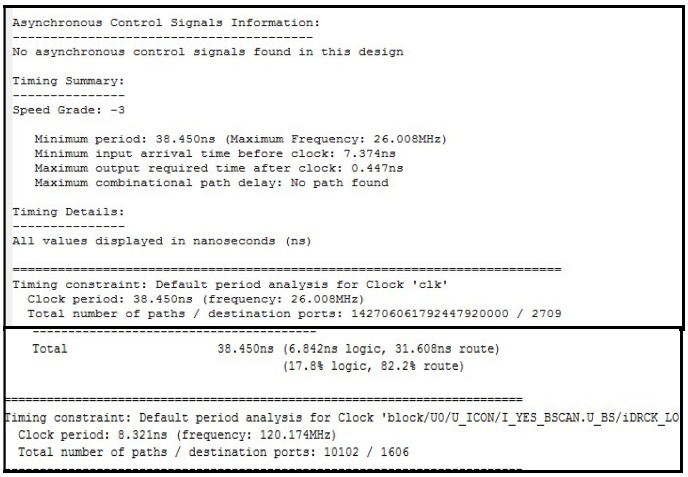


Fig.10. Power utilization summary

3) Timing Analysis and Throughput.

Timing analysis is done by considering the architecture of the FPGA device and the logic implemented. Timing analysis tool will analyze and consider the architecture of the FPGA devices ,logic implemented ,with interconnecting routing delays by default.Throughput is calculated using the timing analysis report. Throughput can be defined as the data processed by a design within a fixed amount of time. It depends on block size of the algorithm, frequency and latency of the hardware design. Throughput is the rate of production or the rate at which something can be processed. It's the amount of data processed by a design within a fixed amount of time. The importance of throughput is coming from being a number that weights:block size which is the characteristic for the algorithm used in an application ,the frequency which is characteristic for the hardware design performance and the latency which is the characteristic for the hardware design architecture. From the timing report *("Fig.11")*the clock frequency is found as 120.174 MHz with a clock period of 8.321 ns. The block size is a fixed amount of data that the algorithm will process at a time.The block size is 64 in this case. The latency is the number of cycles to encrypt a message and latency here is 8 cycles. The throughput obtained at frequency 120.174 MHz is 924.304289 Mbps. After implementing the design on a Spartan-6 FPGA (XC6LX16-CS324), the timing report is analysed. It is shown below.

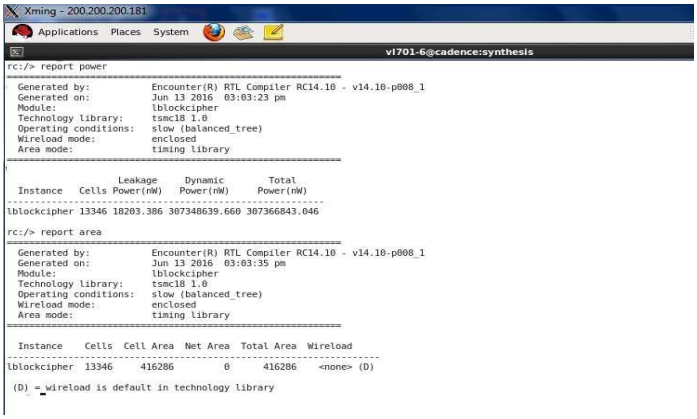


```
Asynchronous Control Signals Information:
----------------------------------------
No asynchronous control signals found in this design

Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 38.450ns (Maximum Frequency: 26.008MHz)
   Minimum input arrival time before clock: 7.374ns
   Maximum output required time after clock: 0.447ns
   Maximum combinational path delay: No path found

Timing Details:
---------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default period analysis for Clock 'clk'
  Clock period: 38.450ns (frequency: 26.008MHz)
  Total number of paths / destination ports: 142706061792447920000 / 2709
-------------------------------------------
   Total                    38.450ns (6.842ns logic, 31.608ns route)
                                     (17.8% logic, 82.2% route)

=========================================================================
Timing constraint: Default period analysis for Clock 'block/U0/U_ICON/I_YES_BSCAN.U_BS/iDRCK_LO
  Clock period: 8.321ns (frequency: 120.174MHz)
  Total number of paths / destination ports: 10102 / 1606
-------------------------------------------------------------------------
```

Fig.11. Synthesis Report

4) Area and Power

The design was implemented on ASIC platform and the area, power, timing details were obtained .The area consumed was found to be 416286 cell area for instances 13346 and total power consumed, leakage and dynamic power consumed are obtained as shown below.



```
Xming - 200.200.200.181
Applications  Places  System
                                                    vl701-6@cadence:synthesis
rc:/> report power
=====================================================================
  Generated by:           Encounter(R) RTL Compiler RC14.10 - v14.10-p008_1
  Generated on:           Jun 13 2016  03:03:23 pm
  Module:                 lblockcipher
  Technology library:     tsmc18 1.0
  Operating conditions:   slow (balanced_tree)
  Wireload mode:          enclosed
  Area mode:              timing library
=====================================================================

                    Leakage     Dynamic       Total
  Instance   Cells Power(nW)   Power(nW)     Power(nW)
---------------------------------------------------------------------
lblockcipher 13346 18203.386 307348639.660 307366843.046

rc:/> report area
=====================================================================
  Generated by:           Encounter(R) RTL Compiler RC14.10 - v14.10-p008_1
  Generated on:           Jun 13 2016  03:03:35 pm
  Module:                 lblockcipher
  Technology library:     tsmc18 1.0
  Operating conditions:   slow (balanced_tree)
  Wireload mode:          enclosed
  Area mode:              timing library
=====================================================================

  Instance    Cells  Cell Area  Net Area  Total Area  Wireload
---------------------------------------------------------------------
lblockcipher  13346     416286         0      416286    <none> (D)

(D) = wireload is default in technology library
```

Fig.12. Power,area analysis report

5) On chip Debugging Result

Fig.13. On chip Debugging Result

V. CONCLUSION

A detailed study on Lblock block cipher was done and carried out its algorithm validation. LBlock block cipher is a lightweight block cipher of block size 64 bit and key size 80 bit targeted to provide cryptographic security for resource constrained applications e.g. RFID, sensor networks etc. The behavioral description of the design is written in Verilog HDL and simulated using XilinxISE 14.7 and ModelSim 6.2 c software platforms. Then the design is successfully implemented on Xilinx Spartan6 FPGA (XC6LX16-CS324). Design verification is performed with chip scope tool of Xilinx. Test result is compared with MATLAB results and found to be right. Power utilization is analyzed using Xilinx Power Analyzer (XPA) and noted that it took only 0.021W. The throughput is calculated at frequency 120.174 MHz is 924.304289 Mbps. LBlock cryptocore was designed and characterized. It can serve as reference for hardware design engineers who wants to provide cryptographic security in constrained environments. And this work enables more researches complying with RCE constraints and allowing them to conform to the trends observed, allowing researchers to fairly compare their cryptographic implementations to other works. Also the IP core designed and characterized can be reused with slight modifications depending on applications.

Acknowledgment

I take this opportunity to acknowledge those who have been great support and inspiration throughout the research work. My sincere thanks to my guides for their diligence, guidance, encouragement and help throughout the period of research, which have enabled me to complete the research work in time. I also thank them for the time that they spared for me, from their extremely busy schedule. Thanks to my friends and colleagues who have been a source of inspiration and motivation. I appreciate my family members for their motivation, love and support in my goal. I would like to express my sincere gratitude to all those who helped me to carry out this work successfully. Finally, I would like to thank God Almighty for blessing me with his grace.

References

[1] Wenling Wu, Lei Zhang "LBlock: A Lightweight Block Cipher" Springer, 2011.

[2] Bassam J.Mohd, Thaier Hayajneh, Athanasios V.Vasilakos, "A survey of lightweight block ciphers for low-resource devices-Comparative studies and open issues," Journal of Network and Computer Application ", September 2015.

[3] Mickael Cazorla, Kevin Marquet and Marine Minier, "Survey and Benchmark of Lightweight Block Ciphers for Wireless Sensor Networks," Security and Communication Networks, Volume 8, Issue 18, pages 3564–3579, December 2015.

[4] Jia Hao Kong, Li-Minn Ang , Kah Phooi Seng, "A comprehensive survey of modern symmetric cryptographic solutions for resource constrained environments", Journal of Network and Computer Applications 49(2015)15–50.

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

[6] Sufyan Salim Mahmood AlDabbagh ,Imad Fakhri Taha Al Shaikhli, "Improving the Security of LBlock Lightweight Algorithm using Bit Permutation," IEEE , Dec 2013 DOI:10.1109/ACSAT.2013.65

[7] Hideki Yoshikawa, Masahiro Kaminaga, Arimitsu Shikoda, Toshinori Suzuki, "Secret Key Reconstruction Method using Round Addition DFA on Lightweight Block Cipher LBlock," IEEE, Oct 2014, pp.493 – 496.

[8] Jinyong Shan1, Lei Hu1, and Siwei Sun, "Security of LBlock-s against Related-Key Differential Attack," IEEE, Feb. 2015, pp 1278 – 1283.

[9] Aleksandar Milenkovic, David Fatzer, "Teaching IP Core Development: An Example".