Volume 3, No. 4, April 2012
Journal of Global Research in Computer Science
ISSN-2229-371X
REVIEW ARTICLE
Available Online at www.jgrcs.info

LOW POWER ENCRYPTED MIPS PROCESSOR BASED ON AES ALGORITHM

Kirat Pal Singh[*1], Shivani Parmar[2]
VLSI Design Department
Academic & Consultancy Services Division
Centre for Development of Advanced Computing (C-DAC)
Mohali-160071, Punjab, India
kirat_addiwal@yahoo.com[*1], shivaniparmar03@gmail.com[2]

Abstract: The paper describes the Low power 32-bit encrypted MIPS processor based on AES algorithm and MIPS pipeline architecture. The pipeline stages of MIPS processor are arranged in such a way that pipeline can be clocked at high frequency and clock gating technique is used for reducing power consumption. Encryption blocks of Advanced Encryption Standard (AES) cryptosystem and dependency among pipeline stages are explained in detail with the help of block diagram. In order to reduce the power consumption, especially for portable devices and security application switching activity is used inside pipeline stages. The design has been synthesized at 40nm process technology targeting using Xilinx Virtex-6 device. The encrypted MIPS pipeline processor can work at 210MHz and power consumption is 1.313W.

Keywords: ALU, register file, pipeline, memory, T-DES, throughput

INTRODUCTION

Today's digital world, Cryptography is the art and science that deals with the principles and methods for keeping message secure. Encryption is emerging as a disintegrable part of all communication networks and information processing systems, involving transmission of data. Encryption is the transformation of plain data (known as plaintext) into inintengible data (known as cipher text) through an algorithm referred to as cipher. There are two classes of Key Based Encryption Algorithm: Symmetric and Asymmetric algorithms. The most commonly used technique for producing confidentiality in data transmission is symmetric algorithm. This algorithm performs various mathematical and logical functions on the plaintext using the same key where as asymmetric algorithm use different keys for encryption and decryption process. In both algorithms, the key is essential part of encryption and decryption process which provides secure data traffic among Sender and Receiver.

The MIPS is simply known as Millions of instructions per second and is one of the best RISC (Reduced Instruction Set Computer) processor ever designed. MIPS architecture is employed in a wide range of applications. The architecture remains the same for all MIPS based processors while the implementations may differ [1]. There is a 16- bit RSA cryptography MIPS cryptosystem have been previously designed [2]. Some adjustments and minor improvements in the MIPS pipelined architecture design are made using authenticating devices [3] such as Advanced Encryption Standard [AES] to protect data transmission over insecure medium. High speed MIPS processor possesses Pipeline architecture to speed up the processing as well as increase the frequency and performance. A MIPS based RISC processor was described in [4]. It consists of basic five stages of pipelining that are Instruction Fetch, Instruction Decode, Instruction Execution, Memory Access and Write Back. These five pipeline stages generate 5 clock cycles processing delay and several Hazards during the operation [2]. These pipelining Hazard are eliminates by inserting NOP (No Operation Performed) instruction which generate some delays for the proper execution of instruction [4]. The pipelining Hazards are of three types: data, structural and control hazard. These hazards are handled in the MIPS processor by the implementation of Forwarding Unit, Pre-fetching or Hazard detection unit, Branch and Jump Prediction Unit [2].

The Forwarding unit is used for preventing data hazards which detects the dependencies and forward the required data from the running instruction to the dependent instructions [5]. Stall occurs in the pipelined architecture when the consecutive instruction uses the same operand as that of the instruction and requires more clock cycles for execution. This reduces the performance. To overcome this situation, Instruction Pre-fetching Unit is used which reduces the Stalls and improves performance. The control hazard occurs when a branch prediction is mistaken or in general, when the system has no mechanism for handling the control hazards [5]. The control hazard is handled by two mechanisms: Flush mechanism and Delayed jump mechanism. The branch and jump prediction unit uses these two mechanisms for preventing control hazards. The flush mechanism runs instruction after a branch and flushes the pipe after the misprediction [5]. Frequent flushing may increase the clock cycles and reduce performance. In the delayed jump mechanism, Specific numbers of NOP's are pipelined after the Jump instruction to handle the control hazard. The branch and jump prediction unit placement in the pipelining architecture may affect the critical or the longest path. The standard method of increasing performance of the processor is to detect the longest path and design hardware that results in minimum clock period.

SYSTEM ARCHITECTURE

The single chip MIPS crypto processor are shown in Fig. 1that consists of various components like Datapath, Data I/O unit, Control Unit, Memory unit, Crypto Specific Unit, Dependency Resolver, hazard detection unit, forwarding unit, instruction fetch, decode unit and Arithmetic Logic Unit. The dedicated data processing block consist of Datapath and Crypto IP core (coprocessor) that performs the128-bit AES cipher operation.

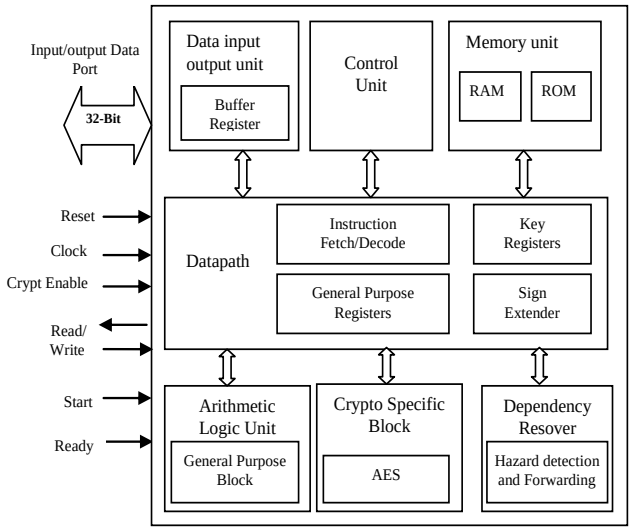


Figure. 1. MIPS crypto processor architecture

Advanced Encryption Standard (AES) algorithm operates on 128bits block size by using cipher keys with lengths 128, 192 and 256 bits for encryption process respectively. The incoming data and key are stored in a matrix called state matrix and all the operations are performed over the state matrix [6]. Datapath processing unit performs the 5 stages pipelining process inside the processor. It consists of Program Counter, 32-bit General Purpose Registers, Key Register and Sign Extender Unit. The program counter unit updates the values available at its input bus at every positive edge clock cycle and also fetches the next instruction from the instruction ROM memory. The registers are read from the General purpose register and the opcode is passed to the control unit which asserts the required control signals. Sign extension is used for calculating the effective address. The data and instruction memory have capability of storing 256 bytes and each byte is referred by the address in between 0 to 256. The address is represented by 8-bits.

The MIPS controller is the main core of the architecture which consists of control unit and ALU control signal unit. The function of controller is to controls the dedicated crypto block and performs the interface and specific operation with the external devices such as Memory, I/O bus interface controller. Single control unit controls the activities of other modules according to the instruction stored inside memory. The crypto specific block executes various other private and public key algorithms such as RSA, DSA, elliptic curve and IDEA with other application programs such as user authentication programs for securing secret information. The arithmetic logic unit (ALU) performs the NOP (no operation), addition, subtraction, OR, NOR, set less than, shift left logic operation. The data and address calculations for load and store instruction are performed by ALU. The Load and Store instructions write to and read from the RAM memory in the memory unit while the ALU results and the data read from RAM are written in to the register file by the register type and Load instruction respectively. Data I/O has two different external interfaces which stored data initially at buffer registers or move data to output. The dependency resolver block has a function to avoid stall by rearranging the instruction sequence and checking the successive instruction for their stall possibility by comparing their operands. This module handles both stalling as well as data forwarding of previous stage. In case of data dependency between two consecutive instructions the receiving instruction waits for one clock cycle. Thus dependency resolver controls the data forwarding in pipeline stages.

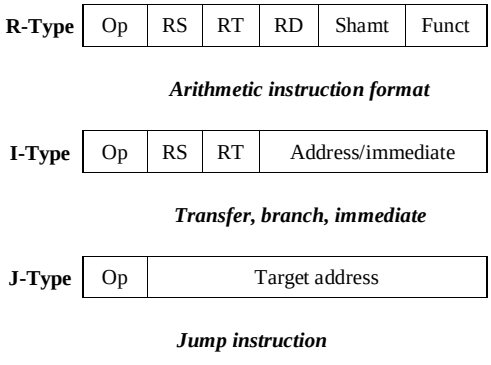
Microinstruction Set:

The MIPS instruction set is straightforward like any other RISC designs. MIPS are a load/store architecture, which means that only load and store instructions access memory. Other instructions can only operate on values in the registers [8]. Generally, the MIPS instructions can be broken into three classes: the memory-reference instructions, the arithmetic- logical instructions, and the branch instructions. Also, there are three different instructions formats (as shown in Fig.2) in MIPS architecture: R-Type instructions, I-Type instructions, and J-Type instructions. A subset of the instruction has been implemented in our design, the list of which is given in Table 2.

Instruction Type	*Instruction*
R-Type	AND, OR, NOR, ADD, SUB, SLT
I-Type	ADDI, SUBI, NORI, ANDI, SLTI,SLL, SRL
	LW,SW, LKLW, LKUW
	BEQ, BNE
J-Type	J, JR, JAL, CRYPT

Figure.2 Implemented MIPS Instruction Types

Table 1. MIPS Instruction Format



Field	Description
Op[31-26]	6-bit operation code
RS[25-21]	5-bit source register
RT[20-16]	5-bit target register
Immediate[15-0]	16-bit immediate address
Target[25-0]	26-bit jump target address
RD[15-11]	5-bit destination register
Shamt[10-6]	5-bit shift amount
Funct[5-0]	6-bit function field

ADVANCED ENCRYPTION STANDARD

There are numerous encryption algorithms that are now commonly used in computation, but U.S government has adopted the Advanced Encryption Standard (AES) to be used by Federal departments, and agencies for protecting sensitive information. The AES algorithm (shown in Fig. 3)

is a symmetric cipher and used a single secret key for both the encryption and decryption. In addition, the AES algorithm is a block cipher as it operates on fixed-length groups of bits (blocks), whereas in stream ciphers, the plaintext bits are encrypted one at a time, and the set of transformation applied to successive bits may very during the encryption process. The AES algorithm operates on block length [N_b] of 128-bits, by using cipher keys with key length [N_k] of 128, 192 or 256 bits or the encryption process.

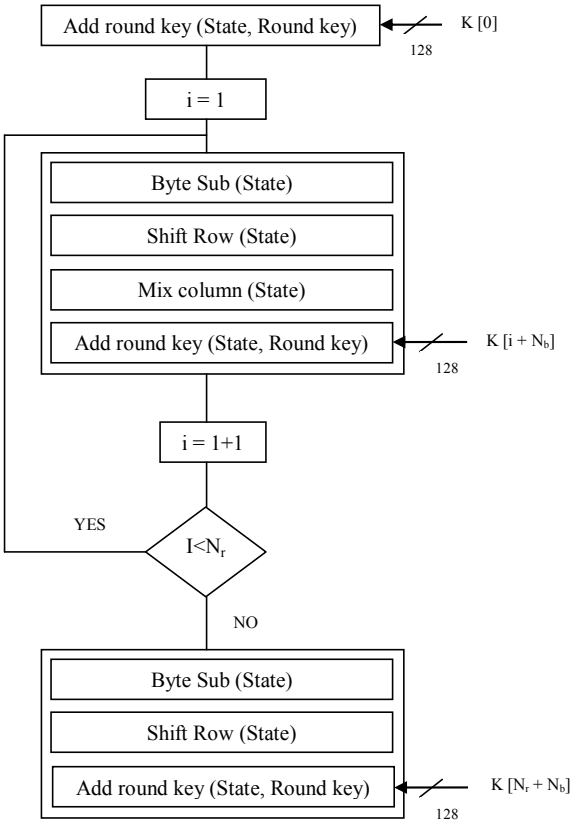


Figure.3. AES Block Diagram for Key Length of 128bits and the number of Iterations required are 10(Nr = 10)

The encryption and decryption process of AES block consists of number of different transformation applied consecutively over the data block bits, considered as a 4x4 array of 8 bit bytes (also called "state" in the algorithm). The state undergoes four different transformations in each round having fixed number of iterations. These transformations are *"Sub Byte", "Shift Row", "Mix Column",* and *"Add Round Key"* transformations. *"Sub Byte"* can be implemented by non–linear substitution of bytes that operates independently on each byte of the state using a substitution LUT (S-box). In this S-box; each byte in the state matrix is an element of a Galois Field GF (2^8), with irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x+1$. In simple terms, GF ($2^n$) is a set of 2^n elements each represented by an n-bit string of 0's and 1's and affine transformation is applied (over GF (2)).

The *"Shift Row"* can be implemented using a cyclically shift the rows of the state over different offsets. *"Mix Column"* are considered as most complicated operation in the algorithm and need GF (2^8) fields and multiply by modulo x^4+1 with a fixed polynomial $a(x)=\{03\}x^3 + \{01\}x^2 + \{02\}x$. *"Add Round Key"* is added to the state by a logical XOR operation. Each round key consists of N_b words from the key expansion. These N_b words are added into the state columns. Each round key is a 4-word (128bit) array generated as a product of previous round key, and a sense of substitution LUT for each 32-bit word of the key. The key expansion generated a total of $N_b(N_r+1)$.

DESIGN AND IMPLEMENTATION METHODOLOGY

Current applications demand high speed processor for large amount of data transmission in real time. As compared to software alternatives, hardware implementation provides highly secure algorithms and fast solutions approaches for high performance applications. Software approaches could be a good choice but it has some limitations like low performance and speed. Main advantages of software are low cost and short time to market. But they are unacceptable in terms of high speed and performance specification. So that, Hardware alternatives could be selected for implementing MIPS crypto processor architecture.

Hardware implementation supports both Field Programmable Gate Arrays (FPGAs) and Application Specific Integrated Circuits (ASIC) at high data rates. Such design has high performance but more time consuming and expensive as compared to software alternatives. The detailed comparison of hardware v/s software solutions for implementing the MIPS crypto processor architecture is shown in Table 2. Based on the comparison, hardware solution is a better choice in most of the cases because they have high performance. The main advantage of FPGA in hardware alternative, FPGA are low density and low area consumption. Logic integration, size and density are the major drawbacks in ASIC but have higher performance than FPGA.

Table 2. Hardware v/s software alternatives for crypto processor

Parameters	Software	Hardware	
		FPGA	ASIC
Performance	Low	Medium-High	Very High
Power consumption	Depends	Very high	Low
Logic integration	Low	Low	High
Tool cost	Low	Low	Low
Test development complexity	Very low	Very low	High
Density	High	Very low	High
Design efforts	Low-Medium	Low-Medium	High
Time consumed	Short	Short	High
Size	Small-medium	Small	Large
Memory	Fine	Fine	Fine
Flexibility	High	High	-
Time to market	Short	Short	High
Run time configuration	-	high	-

Implementation of Cryptographic Engine:

The global architecture of encrypted and decrypted MIPS pipeline processor is modified in a way that it executes encrypted instruction. Fig. 4 shows the block diagram of encrypted MIPS processor. To modify MIPS processor for encryption, we insert the cryptography module Advanced Encryption Standard (AES) to the pipeline stage. Only single cryptographic module is used in same hardware implementation. The instruction fetch unit of encrypted MIPS contains Program Counter (PC), Instruction Memory, Decryption core and MUX. The Instruction memory reads address from the PC and stores instruction value at the particular address that is pointed by the PC. Instruction

Memory sends encrypted instruction to MUX and decryption core.

The decryption core gives decrypted instructions which are further sent to the MUX. The output of MUX is fed to the IF register. The MUX control signal comes from control unit. The instruction decode unit contains Register file and Key register. Key register stores the key data of encryption/decryption core. Key address and Key data comes from write back stage. The key data to be stored into the register file and remains same for all program instruction execution. The control unit provides various control signals to other stages. This acts as select line for two multiplexers. When the control unit detects a store/branch/jump it asserts the control signal high and keep it asserted till a load instruction is detected.

During that period, the write back stage gets the forwarded data and the memory stage gets a constant zero value thus preventing only further transitions. When the control signal is de-asserted, then the data pass through the standard pipeline structure. The execute unit executes the register file output data and performs the particular operation determined by the ALU. The ALU output data is sent to EXE register which temporarily store address value. The Memory Access Unit contains Encryption core, Decryption core, Data Memory, MUX and DEMUX. The second register data from register file is fed to the encryption core and the MUX. Here the crypt signal enable/disable encryption operation when occurs. The read/write signal of data memory describes whether reading/writing operation is done. Output of data memory pass through DEMUX whose one output goes to decryption core and other to MEM register. Here the unencrypted memory data and decrypted data are temporarily stored to the MEM register. The MEM output is fed to the write back data MUX and according to the control signal, the output of MUX goes to register file.

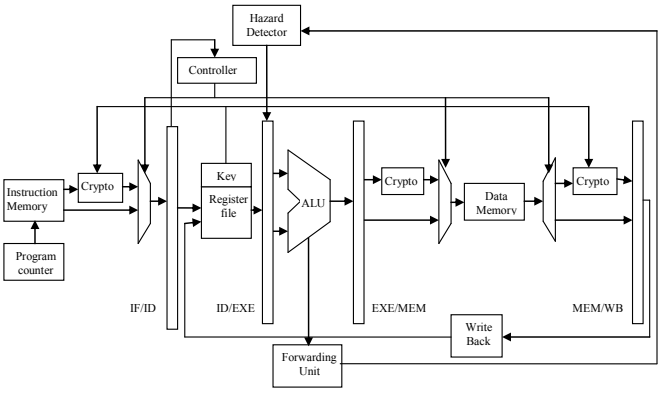


Figure. 4. Detailed MIPS crypto processor architecture

Power Reduction Technique:

One of the key concerns in any microprocessor based system is power consumption. Power dissipation is either static or dynamic. Static power dissipation is caused due to the leakage and short circuit current while dynamic power dissipation is due to switching activity of the various transistors in the circuit. Dynamic power forms the major chunk of power dissipation in CMOS circuits and have require a lot of attention. In our design power reduction is achieved through bypassing pipeline stages that cause unnecessary switching activity. One of the influencing factors of dynamic power dissipation is switching activity and dynamic power is given by the equation,

$$P = 0.5\ C\ (V_{dd})^2\ E\ (sw)\ F_{clk}$$

In the above equation C represents capacitance and V_{dd} represents the drain voltage across CMOS circuit and F_{clk} is the clock frequency. There by decreasing switching activity (E (sw)) results in reduced dynamic power consumption. The pipeline stages for different type of instructions are shown in Fig. 5.There are NOP inserted inside the pipeline stage which act as delay of one clock cycle.

IF	ID	EXE	MEM	WB

Store Instruction

IF	ID	EXE	MEM	WB

Load Instruction

IF	ID	EXE	NOP	WB

R-Type Instructions and Arithmetic I-Type

IF	ID	EXE	MEM	NOP

Branch Instruction

IF	ID	NOP	NOP	NOP

Jump Instruction

Figure. 5. MIPS Instruction Format

It can be seen that the data memory stage of the pipeline is not used by any of the arithmetic instructions. Transition during this unused state causes extra power dissipation. To avoid this wastage, the pipeline is reconfigured to bypass this stage for these set of instructions. Hence data obtained from execution stage is forwarded directly to the write back stage. During this time, the EXE/MEM pipeline registers are maintained at zero value thus ensuring that no transition take place and power dissipation is reduced. Arithmetic instruction has a 'NOP' stage in the MEM stage while there is a 'NOP' during the write back stage for the Store/Branch/Jump instructions. Hence the write back stage of the arithmetic could be moved to the MEM stage without causing any resource conflicts. A load instruction used all five stages of the pipeline and hence a resource conflict will arises. So, data has to pass through the regular pipeline structure till it encounters a store/branch/jump instruction after which the reconfigured pipeline can be again brought in.

IMPLEMENTATION RESULTS

The complete pipeline processor stages are modeled in VHDL. The syntax of the RTL design is checked using Xilinx tool. For functional verification of the design the MIPS processor is modeled in Hardware Descriptive Language. The design is verified both at the block level and top level. The complete design along with all timing constraints, area utilization and optimization options are described using Synthesis Report. The design has been synthesized targeting 40nm triple oxide process technology using Xilinx FPGA Virtex-6 (xc6vlx240t-3ff1156) device. The Virtex family is the latest and fastest FPGA which aims to provide up to 15% lower dynamic and static power and 15% improved performance than the previous generation. It is obvious that there is a trade-off between maximum clock frequency and area utilization (number of slices LUT's) because the basic programmable part of FPGA is the slice that contains four LUTs (look up table) and eight Flip flops. Some of the slice can use their LUT's as distributed RAM.

The power consumption is estimated by the Xilinx XPOWER Analyser tool, using the post layout netlist of the crypto processor along with the node activity data for each algorithm. The power consumption can be further reduced

by running the processor at lower voltages than the normal voltage of 1.5v (as long as the speed and throughput requirements are satisfied). Power analysis was done for the portion between the EXE/MEM and MEM/WB stage. Clock gating technique is used to minimize energy reduction during pipeline stall stages. This technique identifies low processing requirement periods and reduces operating voltage with clock frequency (voltage-frequency scaling), resulting in reduced average operating power consumption. This may or may not occur frequently depending upon compiler efficiency. The power analysis result is carried out on the same clock frequency. In our design, a symbol is processed every clock cycle; the throughput is calculated on the basis of number of instruction execution per second. The formula for calculating throughput is:

Throughput = f * symbol width/total clock frequency

Where f is the operation frequency and symbol width is one of our parameterized values. In AES crypto processor, 43 clock cycles are used for crypto specific block to execute data, 47 clock cycles are needed to execute the R-type instruction, 48 clock cycles are needed for I-type instruction and 46 clock cycles for J-type instruction data. Table 3 shows the performance throughput; area and the estimated power consumption of AES based MIPS crypto processor. Maximum throughput of AES based MIPS Crypto processor is 560Mbits/s at 4.76ns. Moreover, it is possible to trade performance with area and power in the implementation. For example, higher performance can be obtained by running processor at higher frequency up to 300MHz for the current design (increasing power consumption) and/or using pipeline (increasing area) for more performance demanding applications.

Table 3. Throughput Estimates for the MIPS Crypto Processor

Features	***Processor***
Crypto processor	AES
Data length	128-bits
Speed Throughput	210MHz (clock rate) 560Mbits/s (Data Bandwidth)
Area	109738 Slice LUT's(look up tables)
Latency Power consumption	48 clock cycles(for encryption) 1.313W(quiescent-1.008 and dynamic-0.396)

CONCLUSION

In this paper, we have presented a power efficient hardware architecture design of 32-bit encrypted MIPS processor that executes encrypted instructions. Initially it read encrypted data from instruction memory and decrypts the same data and sent it to the next pipeline stages. The processor uses the symmetric block AES plain/cipher that can process data length of 128bits. The crypto block in the MIPS processor performs data encryption. The design has been modeled in VHDL and functional verification policies are adopted for it. Optimization and synthesis of design is carried out at latest and fastest FPGA Viretx-6 device that improves performance. Each program instructions are tested with some of vectors provided by MIPS. We conclude that the performance of MIPS crypto processor using AES is High 560Mbits/s. The power consumption of MIPS Crypto processor is 1.313W.The high performance and high flexibility of crypto processor design makes it applicable to various security applications.

REFERENCES

[1] Gautham P, Parthasarathy R, Karthi Balasubramanian.2009, "Low-power pipelined MIPS processor design", International symposium on integrated circuit (ISIC 2009), pp. 462-465.

[2] Zulkifli, Yudhanto, Soetharyo and adinono.2009, "Reduced Stall MIPS architecture using Pre-fetching accelerator", International conference on electrical engineering and informatics, IEEE, pp. 611-616, ISBN: 978-1-4244-4913-2, IEEE, Aug. 2009.

[3] Pravin B. ghewari, Mrs. Jaymala K. patil, Amit B. Chougule.2010, "Efficient hardware design and implementation of AES cryptosystem", International journal of engineering science and technology, 2010, Vol. 2(3), 2010, pp. 213-219, ISSN: 0975-5462.

[4] D. A. Patterson and J. L. Hennessy, Computer Organization and Design, The hardware/Software Interface. Morgan Kaufmann, 2005.

[5] Pejman lotfi, Ali-Asghar Salehpour, Amir-Mohammad Rahmani, Ali Afzali-kusha, and zainalabedin Navabi.2011, "dynamic power reduction of stalls in pipelined architecture processors", International journal of design, analysis and tools for circuits and systems, June 2011, Vol. 1, No. 1, pp. 9-15.

[6] Refik Sever, A. Neslin Ismailoglu, Yusuf C. tekmen and Murat Askar,"A high speed ASIC Implementation of the rijndael Algorithm", IEEE International symposium on circuits and systems, 2004.

[7] Advanced Encryption Standard (AES), Nov. 2001 Fed. Inf. Process. Syandards Pub..

[8] Rupali S. Balpande, Rashmi S. Keote.2011, "Design of FPGA based Instruction fetch & decode Module of 32-bit RISC (MIPS) processor", International Conference on communication Systems and Network Technologies, 2011, pp.409-413, ISBN: 978-0-7695-4437-3, IEEE, 2011.

[9] Saeid Taherkhani, Enver Ever and Orhan Gemikonakli.2010," Implementation of Non-pipelined and pipelined data encryption standard (DES) using Xilinx Virtex-6 technology", 10th IEEE International Conference on computer and information Technology (CIT 2010), pp. 1257-1262.

