Implementation of Advance Encryption Standard algorithm on FPGA for the protection of Remote Sensing Satellite

Muhammad Irshan khan
Assistant Manager SRDC-K1
SUPARCO
Karachi, Pakistan
E-mail: irshan_@hotmail.com

Syed Musaddiq Ali Shah
Director, SRDC-K1
SUPARCO
Karachi, Pakistan
E-mail: pm.prss@suparco.gov.pk

***Abstract*— Advanced Encryption Standard (AES) and state of art technology FPGAs (Field Programmable Gate Arrays) can be used together to mitigate the potential threats of interception of Satellite data and unauthorized access to the Satellite System. This paper discusses the implementation and verification of AES algorithm on Virtex 4 FPGA and its usage in the protection of Remote Sensing Satellite Data. The performance of the designed core has been verified by the Timing Simulation, on Chip Debugging and through Synthesized report. The analysis of designed core shows that this core can give throughput of 1579 Mbps and take less than 6 % resources of the FPGA, which make it suitable to be use in Remote Sensing Satellite.**

Keywords Remote Sensing Satellite; FPGA; Throughput; Modelsim; ChipScope Pro.

I. INTRODUCTION

Remote Sensing Satellite missions are benefiting humanity by their various applications. During their life span they face several threats. Among these Adversary environment of space and interception of satellite data and unauthorized access to Satellite System two of the biggest threats. Space environment affects the hardware of Satellite. Proper precaution on selection of material and equipments of Satellite reduces the impact of these threats. The interception of data and unauthorized access do more damage to satellite, they degrade the performance of Satellites and sometimes introduction of malfunctions commands by unauthorized access result in complete loss of mission. Proper use of effective cryptography such as Advance Encryption Algorithm and their implementation on state of art technology FPGAs (Field Programmable Gate Arrays) can reduce these threats effectively by encrypting the satellite data at higher data rate.

A Remote Sensing Satellite missions, consists of Ground station, Space link and a Remote Sensing Satellite. Space link consists of one forward Physical link for sending commands from the Ground System to Space Craft and one or multiple return Physical link for sending telemetry from space craft to ground station. Forward link is called Tele-command link (TCU), the commands carry out by this link are used in performing the tasks of tracking, monitoring and controlling of the Satellite. Return link is called Telemetry (TM) link, it is

Used to transfer telemetry data. Operation of the Remote Sensing Satellite can be understood by the “Fig 1”.

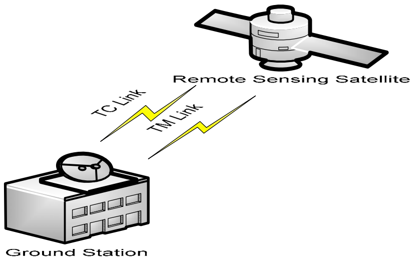


Figure 1. Remote sensing Satellite Mission

Both forward and return links are operationally sensitive. Unauthorized access to these links can result in Satellite malfunctions, which leads to degradation of Satellite operation or sometimes complete loss of space mission.

Encryption of Tele-command and Telemetry data by AES algorithm at higher data rate can significantly reduce the potential threats related to unauthorized access. This algorithm provides protection to Satellite by providing authenticity, confidentiality and Integrity to the electronic information. It converts sensitive electronic data to less sensitive form so that the data is not extracted or modified by the unauthorized users easily. The critical requirement of encrypting Remote Sensing Satellite data at higher rate can be meet by implementing the AES algorithm on FPGAs. The wide range of verification and debugging tools of FPGAs, which is used to trace out the error and verification process of design reduce the design time cycle of projects, flexibility of changing their firmware remotely, and availability of space grade FPGA in market make FPGAs suitable to be used for the implementation of AES.

AES algorithm is powerful algorithm. The only method of breaking AES encryption today is the method of key exhaustion; which is to utilize the all combinations of cipher key to break the AES encryption. 2128 combinations are required by key exhaustion to decrypt the code which is encrypted by 128 bit cipher key. The processing power of today’s available processor will take years to break this code. Therefore by changing the cipher key every year we can protect the Remote Sensing Satellite during its complete life span. The development of AES algorithm on FPGA has been discussed in this paper which is capable to encrypt data at 1579 Mbps, and its verification has been performed through timing simulation on ModelSim and on chip debugging of the designed core on ChipScope Pro.

978-1-4244-5539-3/10/$26.00 ©2010 IEEE

II. Advance Encryption Standard

AES encryption algorithm has been approved by US National Standard and Technology (NIST). It is also known as Rijndael algorithm. It is a symmetric block cipher that can process data blocks of 128 bits by using any of the three different lengths of the cipher key i.e., 128, 192, and 256 bits. The basic unit in the AES algorithm is byte and its value is defined as {b7, b6, b5, b4, b3, b2, b1, b0}. It is interpreted as finite field elements using a polynomial representation.

$$b_7 x^7 + b_6 x^6 + b_5 x^5 + b_4 x^4 + b_3 x^3 b_2 x^2 + b_1 x + b_0 = \sum_{i=0}^{7} b_i x^i$$

$$b_7 x^7 + b_6 x^6 + b_5 x^5 + b_4 x^4 + b_3 x^3 b_2 x^2 + b_1 x + b_0 = \sum_{i=0}^{7} b_i x^i$$

The input, output and Cipher Key bits are processed as arrays of bytes. The length of the input and output block is fixed at 128 bits while length of the key can be varied between 128,192 and 256 bits. Terms Nb, Nk and Nr is used in the algorithm for the understanding of the algorithm, Nb defines the number of 32-bit words (number of columns) in the State, Nk defines the number of 32-bit words (number of columns) in the Cipher Key state, and Nr defines the number of rounds. AES encryption is performed by repeating implementation of rounds. The value of Nb, Nk and Nr is defined in the following table according to the three different cipher key of AES algorithm.

Cipher Key Length	Block Size (Nb words)	Key Length (Nk words)	Number of rounds (Nr)
128	4	4	10
192	4	6	12
256	4	8	14

A. *AES Cipher*

AES algorithm can be defined by the following flow chart.

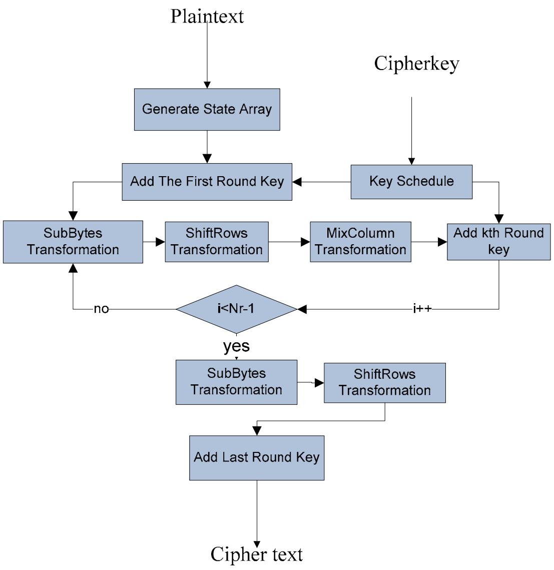


Figure 2. Data flow diagaram of AES Encryptor

At the start of the Cipher, 128 bits input is copied in the form of State array. This is converted into cipher text by the repeated application of round functions. Value of round function is dependent on cipher key length and is equal to Nr-1. The round function is composed of four transformations of substitution, shifting, mixing the column and adding of round key. The encryption process is initialized by adding the first Round Key, followed by applying the round function Nr-1 times. Last round is different from others which perform substitution, shifting and adding the round key operations while mix column operation is not performed in last round. In following section all transformations are defined.

Round Key Addition:

Exclusive OR (XOR) operation between the state array and the Round Key.

SubBytes Transformation:

SubBytes is a 16 Byte input/output nonlinear transformation that uses 1 Byte Substitution table (S-Boxes).

ShiftRows Transformation:

In this transformation, second, third, and fourth rows are shifted one byte, two bytes, and three bytes to the left, respectively.

MixColumns Transformation:

It operates on the State column-by-column, treating each column as a four-term polynomial , The columns are considered as polynomials over GF(2^8) and multiplied modulo x^4 + 1 with a fixed polynomial a(x), given by

a(x) = {03}x^3+{01}x^2 +{01}x+ {02}.

B. *Key Expansion*

Key Expansion module generates new round key for every round of the cipher for Adding Round Key transformation. The expansion key generates Nb(Nr-1) words. First round key is the original cipher key. Then every following word, w[i], is a result of XOR of the previous word w[i-1], and the word Nk position earlier w[i-Nk]. One additional transformation prior to the XOR is performed to words of position that are a multiple of Nk, This transformation consists RotWord, SubWord and addition of RCon(round Constant Word array).

RotWord:

It is a cyclic shift of the bytes in a word. It takes a word [a0,a1,a2,a3] as input, performs a cyclic permutation, and returns the word [a1,a2,a3,a0]

SubWord:

Substitution of all four bytes of the word with substitution table.

Addition of RCon:

XOR with a round constant: RCon[i] contains the values given by matrix [x^{i-1}, {00}, {00}, {00}], where value of x is {02} in the field GF (2^8), (note that i starts at 1). Key expansion of 128 bit cipher key is shown in the “Fig 3”.

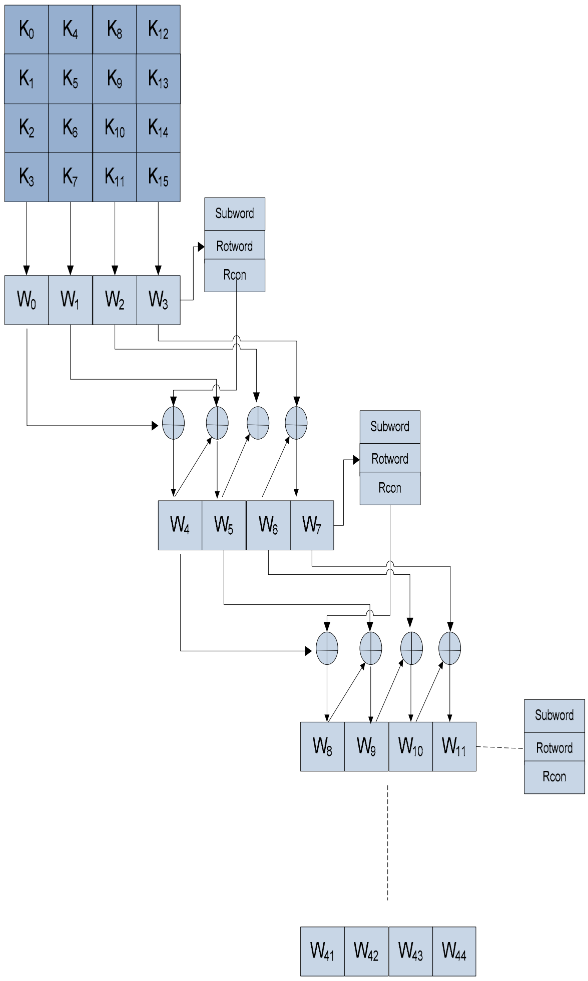


Figure 3. Data flow diagram of AES Key expansion

III. Design Development On FPGA

There are two approaches to implement any design in FPGAs i.e., top to down and bottom to up approach .In 'top to down' approach top module is designed first followed by sub modules design. However in 'bottom to top' approach lower modules are designed first which are then added to form the top module. 'Bottom to top' approach was followed in this project, the diagram below describes the design flow of the implementation of AES on FPGA.

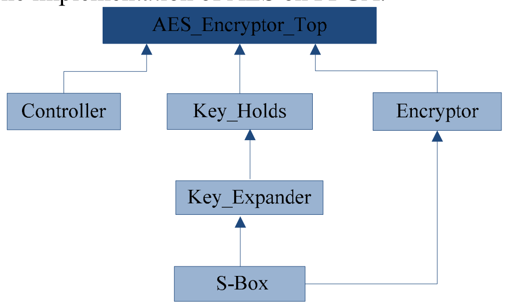


Figure 4. Implementation of AES on FPGA

AES algorithm has been implemented with the cipher key length of 128 bit. Proper functioning of core is very essential for Remote Sensing Satellite therefore this core is not particularly implemented for the optimization of area or throughput, mixed implementation approach has been used, in which both pipelining and iterative approach has been utilized. We could increase the throughput of the design by unrolling of loops but that would increase the area in device proportionally. This design implementation gives throughput of 1579 Mbps. Functionality of sub modules of AES is given as under.

Functions of Sub modules

S-box:

The pre-computed S-box implemented as the 8x256 (2^8) RAM block of FPGA, which works as a look-up table and perform the SubByte transformation.

Key Expander:

This module generates key schedule, this module expands the cipher key and generates 10 round keys, this module works whenever a new key is given to the module.

Key Hold:

This module stores the round keys generated by the key expansion module, and work as a look-up table, it is operated in co-ordination of AES encryptor and provide proper round key to the encryption module at proper time.

Encryption Module:

This module implements the AES algorithm which is explained above.

Controller:

It controls the operation of complete encryption process. It consists of a state machine which ensures the proper state of the algorithm in given time.

AES Encryptor Top:

This module communicates with other module of the Satellite payload it accepts input and give encrypted output, this module also integrate all sub modules of the AES algorithm.

IV. Implementation Details

Information of utilization of resource by the designed core and Timing summary of the design is most crucial part of the implementation of any project on FPGA. Our system took reasonably less hardware resources and can work on with sufficient throughput. The throughput provided by this core is sufficient for the application of remote sensing satellite. Resource utilization summary and timing summary of designed AES core is given as under.

A. Resource Utilization

This project was implemented on Virtex-4 XC4VFX60 of package FF1152, Resources utilized by this project on this FPGA is given as under. The IOs of the algorithm is reduced when the wrapper of the core is designed to use core with integration of other modules used in Remote Sensing Satellite.

TABLE I. Device Utilization Summary

Virtex-4 XC4VFX60 **Package FF1152**	**Resource Utilization**		
	Used	***Available***	***Percent utilization***
Number of Slices	2110	25280	8%
Number of Slice Flip Flops	2291	50560	4%
Number of 4 input LUTs	2580	50560	5%
Number of IOs	391	576	67%
Number of FIFO16/RAMB16s	10	232	4%

B. *THROUGHPUT CALCULATION*

Throughput of the designed AES core can be calculated by the timing summary of the synthesis report, which is given in “Table II”.

TABLE II. TIMMING SUMMARY

Timing Characteristics	***Delays***
Minimum period	3.857ns (Max: Frequency 259.269MHz)
Minimum input arrival time before clock	5.038ns
Maximum output required time after clock:	6.404ns
Maximum combinational path delay:	7.723ns

Vertex 4 Xc4vfx60 is used as design hardware

From the report we can see that the synthesized design can run at a maximum clock frequency of 259.269MHz.By considering this frequency we can calculate the throughput of the design as under.

$$f = 259 \quad MHz$$

$$T = \frac{1}{f} = \frac{1}{259 \times 10^{6}} = 3.86 \quad ns$$

Our designed core takes 21 cycles to encrypt one block of data using 128-bit cipher key which can be seen in “Fig 5”, therefore time required to encrypt the block of 128 bit is. $t_{128} = 21 \times T \quad (s)$

Throughput is defined as the rate at which output data is provided. So this is given by,

$$TP_{128} = \frac{1}{t_{128}} \quad (blocks / s)$$

One block of AES is of size 128-bits. So the above throughput can be converted into bits per second by multiplying with the block size.

$$TP_{128} = \frac{1}{t_{128}} \times 128 = 1579 \quad Mbps$$

Thus the maximum throughput achievable with this core is 1579 Megabits/second.

V. VERFICATION AND VALIDATION OF DESIGN AES CORE

Verification and validation process of designed algorithm has been done through ChipScope pro tool, ModelSim and real time verification of code with the help of MATLAB. The availability of variety of tools for the verification of the designed code on FPGA provide flexibility to verify the designed code helps in meeting the Time to Market requirement of various system.

A. *Timing Simulation*

Timing simulation is a recommended step in large projects, it provides help in reduction of design time cycle. It helps us in analyzing the timing behavior and calculating throughput and latency. Timing diagram results are equivalent to the actual timing behavior of the design. Timing simulation utilizes the Libraries built-in hardware recourses (Gate, Ram blocks, Carry logic, DSP block etc) of target FPGA, which incorporate timing and performance information of these primitives. ModelSim is famous tool for carrying out this simulation. Timing simulation of our design tells that, it takes 21 clock cycles to complete the Key expansion process, and takes 22 clocks cycles to complete encryption process. Timing simulation of encryption module is given in “Fig 5”.

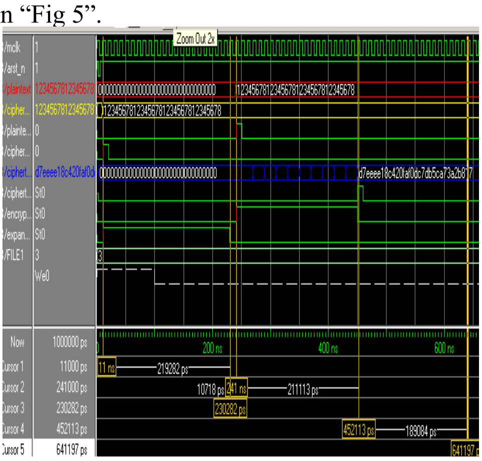


Figure 5. Timing Simulation of AES Design

B. *Real time testing of AES design*

MATLAB is a popular choice for the testing of design algorithm. Real Time testing of the core has been done by setting up a link between FPGA and PC using RS 232 (serial communication protocol). The implementation of AES algorithm along with RS 232 protocol and buffer on FPGA has been shown in the “Fig 6”.In real time testing cipher key and plaintext 128 bits inputs are provided to the FPGA by serial cable. FPGA performs the necessary steps of key scheduling and encryption according to the data provided and returns the encrypted data to PC through serial cable. The transmitted encrypted data is received serially in MATLAB where AES decryptor has been made which decrypt the data according to AES standard.

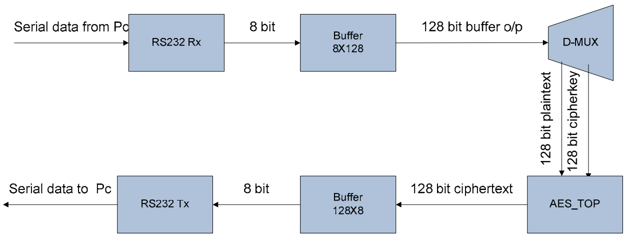
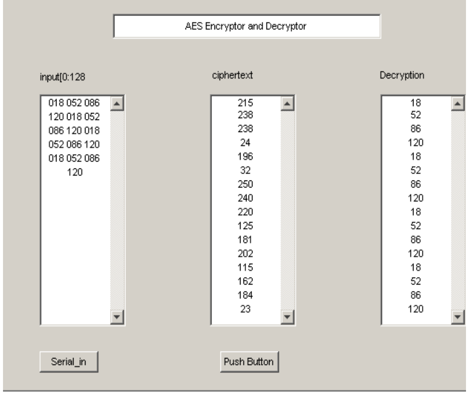


Figure 6. Implementation of Serial communication protocol modules and buffer module for real time testing of AES core

The blocks of RS 232 Rx, RS 232 Tx shown in the block diagram represent the RTL core of these modules in FPGA for the data reception and data transmission between FPGA and PC, two buffer modules 8 x 128 buffer and 128 x8 buffer is used to make AES compatible with the operation of RS 232 protocol. D-Mux is used for the de-multiplexing of cipher key and plaintext data. The result of this verification method can be seen in the "Fig 7".



input[0:128

018 052 086
120 018 052
086 120 018
052 086 120
018 052 086
120

ciphertext

215
238
238
24
196
32
250
240
220
125
181
202
115
162
184
23

Decryption

18
52
86
120
18
52
86
120
18
52
86
120
18
52
86
120

Figure 7. MATLAB GUI used for the real time verification of AES module on FPGA

C. *Verfication of Design using Chipscope*

ChipScope Pro is another tool which helps in reducing the development time of FPGA project. By using this tool we can check the internal signals of the FPGAs. It is very helpful in debugging the design core. It runs at the system clock rate and reduces verification times by as much as 50%. It has helped in identifying many errors during development of the core and it has also used as a verification tool in this project. Result of debugging of communication of serial data between FPGA and computer is shown in "Fig 8". In this diagram dout_byte represents the serial data send from PC i.e., plaintext or cipher key, and uart_tx_data represents the data encrypted by AES algorithm.

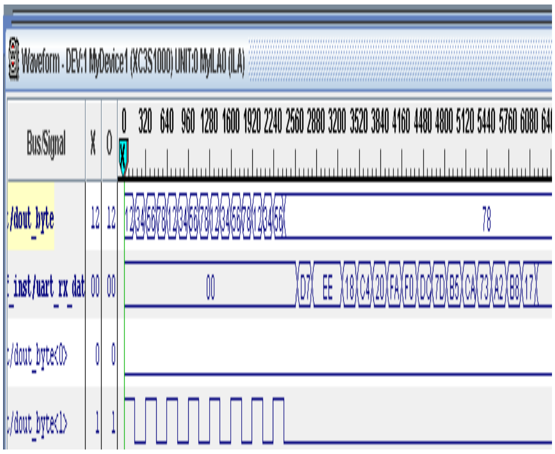


Figure 8. On chip debugging and verification of AES core on Chip Scope Pro tool

VI. Conculsion

The potential threats of interception and unauthorized access to Remote Sensing Satellite system can be reduced by encrypted remote sensing satellite data with AES encryption. Implementation of AES algorithm on FPGA provide sufficient throughput so that AES can be used with the operation of Remote sensing satellite. The implementation of the AES core in this paper provides throughput of 1579 Mbps and take 6 % resources of the Virtex 4 FPGA, which is sufficient for the application of Remote Sensing Satellite. The availability of various verification tools i.e., ModelSim and ChipScope Pro of FPGAs reduces the design time cycle of projects, which is desired for the development Satellite. Moreover the flexibility of FPGA to change firmware on board can be used to change the cipher key every year and hence the protection of Satellite System during its complete life span.

References

[1] National Institute of Standards and Technology, ""Specification for the Advanced Encryption Standard (AES)," Federal Information Processing Standard 197 , November 26, 2001;

[2] Implementation of High Speed AES algorithm on FPGA B.E Project Report Batch 2003-04 by Aamir Ahmed Khan

[3] Encryption Algorithm Trade Survey, Informational Report, Green Book, CCSDS 350.2-G-1 ,March 2008.

[4] Tanya Vladimirova, Roohi Banu and Martin N. Sweeting "On-Board Security Services in Small Satellites" Surrey Space Centre School of Electronics and Physical Sciences University of Surrey, Guildford, UK, GU2 7XH.

[5] M.R.M. Rizk, Senior Member, IEEE and M. Morsy" Optimized Area and Optimized Speed Hardware Implementations of AES on FPGA". Mehran Mozaffari-Kermani and Arash Reyhani-Masoleh, A

[6] Structure independent Approach for Fault Detection Hardware implementation of the AES. Security Threat Against Space Mission INFORMATIONAL REPORT CCSDS 350.1-G-1 GREEN BOOK October 2006.

[7] THE APPLICATION OF CCSDS PROTOCOLS TO SECURE SYSTEMS Informational Report CCSDS 350.0-G-2 Green Book January 2006.