

FPGA Implementation of Short Duration Video Encryption & Decryption

K Jyothi¹, S.V.R. Manimala²
MVSR Engineering College, Nadergul, Hyderabad
jyothikarnkota@gmail.com¹, svrm.73@gmail.com²

Abstract:

The Cryptographic Algorithm is most widely used throughout the world for protecting information. Cryptography is the art of secret writing, followed by the guarantee to authenticate data and important messages and protect the systems from valid attacks. This paper aims at improving the level of security and secrecy provided by the Video encryption. The image encryption and decryption algorithm is designed and implemented to provide confidentiality and security in transmission of the image based data as well as in storage. This encryption was carried on FPGA. FPGA offer a more speed than existing implementations. This research proposes the AES algorithm with regard to 256 bits key length. In Spartan3 EDK we implemented the AES algorithm through softcore Processor

Keywords: Advanced Encryption Standard, FPGA, VHDL.

I. Introduction

Since the rise of the Internet one of the most important factors of information technology and communication has been the security of information. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. Steganography is the art of secret communication. Its purpose is to hide the very presence of communication as opposed to cryptography whose goal is to make communication unintelligible to those who do not possess the right keys. Digital images, videos, sound files, and other computer files that contain perceptually irrelevant or redundant information can be used as “covers” or carriers to hide secret messages. After embedding a secret message into the cover-image, a so-called stego-image is obtained. It is important that the stego-image does not contain any easily detectable artifacts due to message embedding. A third party could use such artifacts as an indication that a secret message is present. Once this message detection can be reliably achieved, the steganographic tool becomes useless. Obviously, the less information is embedded into the cover-image, the smaller the probability of introducing detectable artifacts by the embedding process. Another important factor is the choice of the cover-image. The selection is at the discretion of the person who sends the message. The sender should avoid using cover-images that would be easy to analyze for presence of secret messages. Although computer-generated fractal images may seem as good covers⁶ because of their complexity and irregularity, they are generated by strict deterministic rules that may be easily violated by message embedding

II. CRYPTOGRAPHY

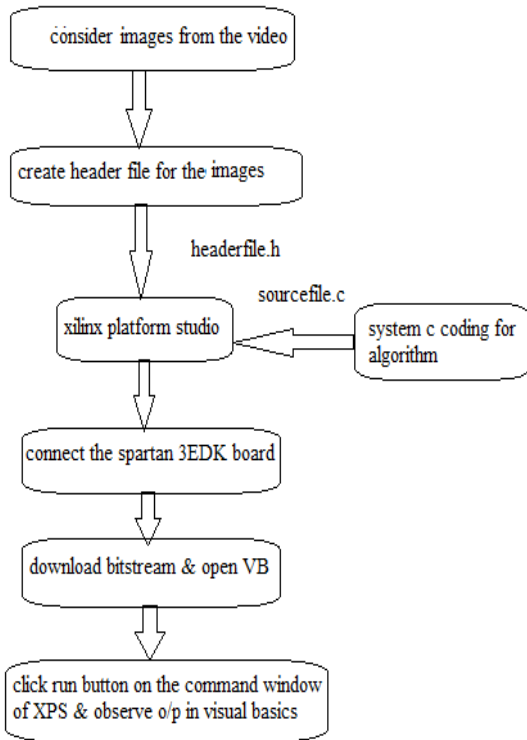
Cryptography is the intelligent art to make a data as secret. It refers sometimes study or analysis of data in secret type. Cryptography is the best tool for protected communication of image, text, video etc, [1]. Cryptography is exploiting the study of hidden information and makes information as secret. It classifies three different kinds (i) Secret key cryptography (ii) Public key cryptography (iii) Hash functions [4]. In Secret key cryptography, both the dispatcher and beneficiary are used same password or key through their transmission [10]. Secret key cryptography is sometimes known as Symmetric key system. In Public key or asymmetric cryptography, the sender and receiver use a pair of keys for unlock the content. Keys work in pairs matched public and private keys [11]. Hash functions not depend on the secret key. Encryption and decryption are the two methods to guarantee data as secret in cryptography. Encryption is the process to convert original image data in to some other anonymous structure using a key which is not identified by anyone.

Decryption defines the recovery of original data from the encrypted thing. Some of image encryption techniques are used traditionally is Data encryption standard, triple DES, blowfish, advanced encryption standard, scan based, and chaos based etc. [6][12]. An encryption system generally needs a key generation algorithm to arbitrarily produce keys

I. Proposed Block Diagram:

Video Encryption is the majorly in order to secure the data from the hackers in the channel and at receiver side retrieves the original video. In order to encrypt the pixel values here we uses symmetric encryption technique i.e

AES algorithm. First the input video was converted to the frames into Images.



Then the Images are Converted to the header File since we are implement this on the XPS(Xilinx Plat Form Studio) . Then the header file of the image and Source C code file of AES files are given XPS then the bit stream was dumped into the FPGA.

Source Code Block Diagram:

The proposed algorithm was based on Image encryption based on double aes technique the cipher that was generated after the aes encryption was given as a message the aes encryption algorithm again so the the key length may increases from 256 bit to 512 bits so that robustness increases

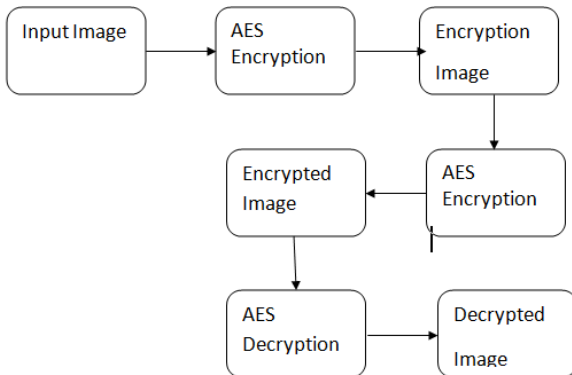


Fig 1: AES Image Encryption and Decryption

III. AES Algorithm

AES is a interchangeable block cipher with a block size of 128 bits. Key lengths can be 128 bits, 192 bits, or 256 bits;8 called AES-128, AES-192, and AES-256, respectively. AES- 128 uses 10 rounds, AES-192 uses 12 rounds, and AES-256 uses 14 rounds.

The main loop of AES9 performs the subsequent functions:

- SubBytes ()
- ShiftRows ()
- MixColumns ()
- AddRound Key()

The first 3 functions of AN AES spherical square measure designed to thwart cryptology via the strategies of “confusion” and “diffusion.” The fourth perform truly encrypts the info. engineer represented the ideas of confusion and diffusion in his seminal 1949 paper, “Communication Theory of Secrecy Systems:”

“Two strategies recommend themselves for frustrating a applied mathematics analysis. These we tend to could decision the strategies of diffusion and confusion.”¹⁰

Diffusion suggests that patterns within the plaintext square measure spread within the cipher text. Confusion suggests that the connection between the plaintext and therefore the cipher text is obscured.

A simpler way to view the AES function order is:

1. Scramble each byte (SubBytes).
2. Scramble each row (Shift Rows).
3. Scramble each column (Mix Columns).
4. Encrypt (Add Round Key).

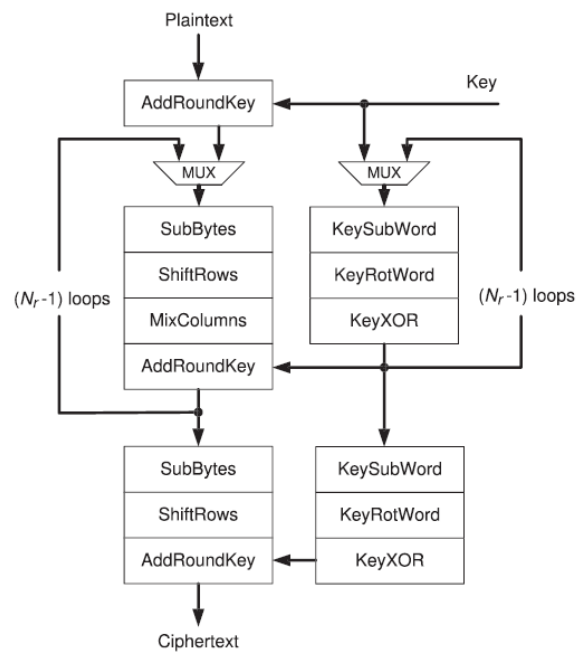


Fig 2 Basic Block Of AES

A. SubBytes ()

The **SubBytes()** transformation is a non-linear byte substitution that operates independently on each byte of the State using a substitution table (S-box). This S-box Fig2 which is invertible, is constructed by composing two transformations:

1. Take the multiplicative inverse in the finite field $GF(2^8)$, described in Sec. 2.2.4.2; the element {00} is mapped to itself.
2. Apply the following affine transformation (over $GF(2)$):

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Fig 3: S-Box

B. Shift Rows ()

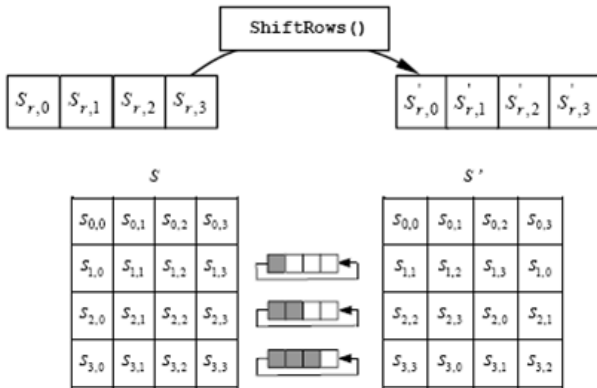


Fig 4: Shift Rows

Shift Rows() provides diffusion by mixing data within rows. Row zero of the State is not shifted, row 1 is shifted 1 byte, row 2 is shifted 2 bytes, and row 3 is shifted 3 bytes, as shown in the *FIPS* illustration that follows:

C. Mix Columns ()

Mix Columns ()also provides diffusion by admixture knowledge at intervals columns. The four bytes of each column within the State square measure treated as a 4-byte range and reworked to a different 4-byte range via finite field arithmetic, as shown within the *FIPS* illustration that

follows:

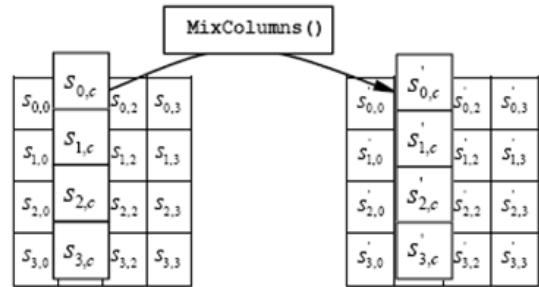


Fig 5: Mix Columns

D. AddRoundKey ()

The actual ‘encryption’ is performed in the AddRoundKey () function, when each byte in the State is XORed with the subkey. The subkey is derived from the key according to a key expansion schedule, as shown in the *FIPS* illustration that follows:

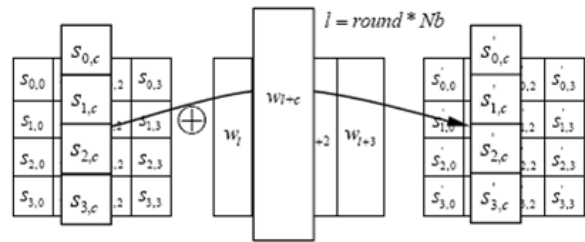


Fig 6: Add Round Key

IV. AES Decryption

Decryption occurs through the function AddRound Key(), plus the inverse AES functions InvShiftRows(), InvSubBytes(), and InvMixColumns(). AddRoundKey() does not require an inverse function, as it simply XORs the state with the sub key (XOR encrypts when applied once, and decrypts when applied again).

VI Implementation

The Field Programmable Gate Array is majorly used for generation ASIC IC’s to the computations. They offer more speed in execution process. SO, for generation ASIC IC’s FPGA’s are majorly used.

TABLE1
Spartan3EDK Configuration

Property Name	Value
Family	Spartan 3
Device	XC3S200
Package	TQG144
Speed Grade	-4

A. Xilinx Platform Studio

The Xilinx Platform Studio (XPS) is that the development atmosphere or user interface used for planning the hardware portion of your embedded processor system. B. Embedded Development Kit Xilinx Embedded Development Kit (EDK) is associate integrated software system tool suite for developing embedded systems with Xilinx MicroBlaze and PowerPC CPUs. EDK includes a spread of tools associated applications to help the designer to develop associate embedded system right from the hardware creation to final implementation of the system on an FPGA. System style consists of the creation of the hardware and software system parts of the embedded processor system and also the creation of a verification element is elective. A typical embedded system style project involves: hardware platform creation, hardware platform verification (simulation), software system platform creation, software system application creation, and software system verification. Base System Builder is that the wizard that's wont to mechanically generate a hardware platform in keeping with the user specifications that's defined by the MHS (Microprocessor Hardware Specification) file. The MHS file defines the system design, peripherals and embedded processors]. The Platform Generation tool creates the hardware platform mistreatment the MHS file as input. The software system platform is defined by MSS (Microprocessor software system Specification) file that defines driver and library customization parameters for peripherals, processor customization parameters, custom anyone hundred ten devices, interrupt handler routines, and different software system connected routines. The MSS file is associate input to the Library Generator tool for personalization of drivers, libraries and interrupts handlers.

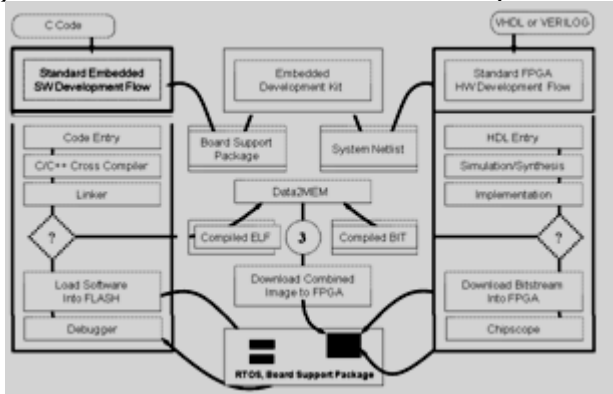


Figure 7: Embedded Development Kit Design Flow

The creation of the verification platform is facultative and is predicated on the hardware platform. The MHS file is taken as Associate in Nursing input by the Siegen tool to make simulation files for a particular machine. 3 varieties of simulation models will be generated by the Siegen tool: behavioral, structural and temporal arrangement models.

Another helpful tool on the market in EDK is Platform Studio that provides the GUI for making the MHS and MSS files. Produce / Import IP Wizard that permits the creation of the designer's own peripheral and imports them into EDK come. Platform Generator customizes and generates the processor system within the sort of hardware net lists. Library Generator tool configures libraries, device drivers, file systems and interrupt handlers for embedded processor system. Bit stream Initialize tool initializes the instruction memory of processors on the FPGA shown in figure2. Antelope Compiler tools are used for collection and linking application executable for every processor within the system. There are 2 choices on the market for debugging the appliance created victimization EDK namely: Xilinx microchip correct (XMD) for debugging the appliance package employing a microchip correct Module (MDM) within the embedded processor system, and package programmer that invokes the package programmer appreciate the compiler getting used for the processor. C. package Development Kit Xilinx Platform Studio package Development Kit (SDK) is Associate in Nursing integrated development atmosphere, complimentary to XPS, that's used for C/C++ embedded package application creation and verification. SDK is made on the Eclipse open source framework. Soft Development Kit (SDK) may be a suite of tools that allows you to style a package application for elite Soft IP Cores within the Xilinx Embedded Development Kit (EDK). The package application will be written during a "C or C++" then the entire embedded processor system for user application are completed, else correct download the bit file into FPGA. Then FPGA behaves like processor implemented on it in a Xilinx Field Programmable Gate Array (FPGA) device.

Hardware implementation was through system C coding and its results are as follows

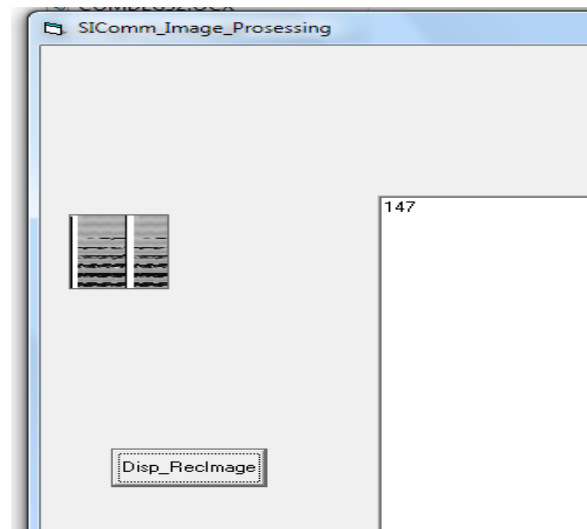


Fig 8: Encrypted Image

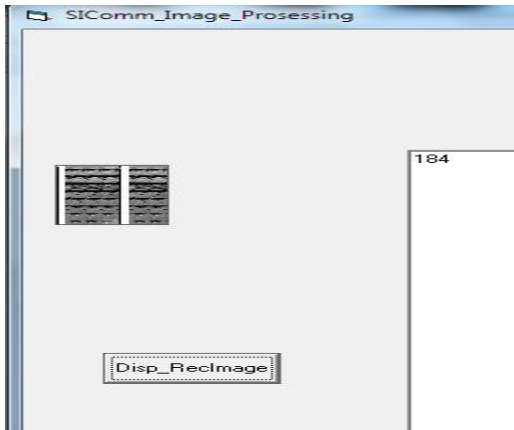


Fig 9: Double Encrypted Image

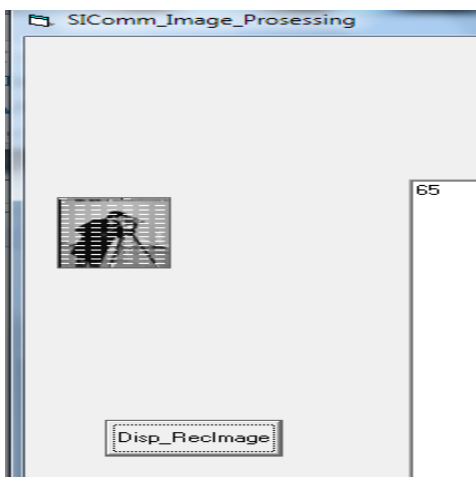


Fig 10: Decrypted Image

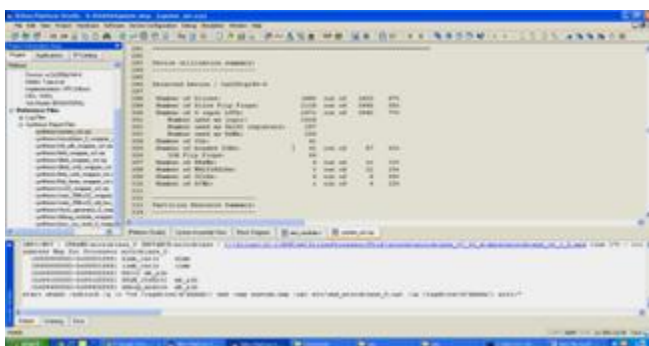


Fig 11: XPS Synthesis report

VII. CONCLUSION

In this work Advanced Encryption algorithm was implemented using FPGA. This system works on Micro Blaze architecture of Spartan3 EDK. On the opposite hand, synthesis results show that area consumption is low, using simply 100 percent of logic circuits of FPGA for AES, permitting the implementation of this method over inexpensive FPGAs.

REFERENCES

- [1] "Supplemental Streaming SIMD Extensions 3," <http://en.wikipedia.org/wiki/SSSE3>, 2012.
- [2] Mr. Atul M. Borkar, Dr. R. V. Kshirsagar and Mrs. M. V. Vyawahare, "FPGA Implementation of AES Algorithm", International Conference on Electronics Computer Technology (ICECT), pp. 401-405, 2011 3rd.
- [3] Ahmad, N.; Hasan, R.; Jubadi, W.M; "Design of AES S-Box using combinational logic optimization", IEEE Symposium on Industrial Electronics & Applications (ISIEA), pp. 696-699, 2010.
- [4] J. Granado-Criado, M. Vega-Rodriguez, J. Sanchez-Perez, and J. Gomez-Pulido, "A New Methodology to Implement the AES Algorithm Using Partial and Dynamic Reconfiguration," Integration, the VLSI J., vol. 43, no. 1, pp. 72-80, 2010.
- [5] Daemen J., and Rijmen V, "The Design of Rijndael: AES-the Advanced Encryption Standard", Springer-Verlag, 2009
- [6] S. Qu, G. Shou, Y. Hu, Z. Guo, and Z. Qian, "High Throughput, Pipelined Implementation of AES on FPGA," Proc. Int'l Symp. Information Eng. and Electronic Commerce, pp. 542-545, May 2009.
- [7] "Int'l Technology Roadmap for Semiconductors, Design," http://www.itrs.net/Links/2009ITRS/2009Chapters_2009Tables/2009_Design.pdf, 2009.
- [8] M. Matsui and J. Nakajima, "On the Power of Bitslice Implementation on Intel Core 2 Processor,"