Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 10, October : 2023

SECURITY ANALYSIS OF AES ALOGORITHM FOR LIGHT WEIGHT DEVICES

Polipalli. Satheesh Kumar Research scholar, Lendi Institute of Engineering & Technology, Kumarsateesh363@gmail.com

S S Kiran Assistant Professor, Lendi Institute of Engineering & Technology, sskiranece@gmail.com

ABSTRACT

In order to communicate and collect data via unreliable wireless channels, smart devices equipped with sensors, software, electronics, and network access are linked together in the Internet of Things (IoT). IoT devices have recently taken over the world thanks to their extensive functionality and real-time data exchange. IoT devices have a wide range of capabilities, but they also have relatively poor battery life, are compact and sophisticated, and face many difficulties because of unsecured communication channels. Despite numerous obstacles, the energy problem is increasingly taking front stage. Most algorithms attempt to restrict or maximise hardware area in the name of security, while the optimisation of energy efficiency has received little attention. IoT devices have changed the focus from security to energy efficiency. We offer AES, a reduced version of AES, to meet the need.

Key words: Internet of Things, AES algorithm, VLSI architecture and Verilog HDL.

1. INTRODUCTION

Data security is the process of protecting data against all forms of unauthorised access and data corruption over the course of its full life [1]. Data security is deteriorating due to technology's constant advancement. Hackers occasionally attempt to access someone's data. Therefore, people's top concern is the protection of their data. Data security can be achieved through hardware or software methods. The use of hardware to protect data is becoming more popular these days. This is due to the fact that using hardware makes data protection more dependable, adaptable, and simple. Furthermore, the implementation of a hardware solution results in reduced latency and enhanced efficacy in safeguarding data. There exist two distinct categories of security algorithms designed to protect data against unauthorised access.. Another category of security approaches is asymmetric security methods, which encompass ECC. FPGA chips are employed as a hardware-based approach to ensure data security. The prevalence of FPGA devices, their enhanced adaptability, and superior performance in terms of speed and throughput contribute to this phenomenon.

The IoT, the next internet revolution, will change our lives. The Internet of Things (IoT) connects almost everything on the earth. This covers actual, tangible things, ranging from home furnishings to advanced engineering. Consequently, whether or not human involvement is involved, these entities that are interconnected with the Internet will possess the capability to perform actions or render choices Utilising a multitude of sensors, these devices additionally transmit real-time data to the Internet. Sensor nodes, RFID tags, and other components having resource constraints are used in IoT. These parts are susceptible to physical capture, have poor compute power, little memory, and insufficient energy supplies. Additionally, they convey real-time information through the risky wireless medium and communicate over an unsecure wireless channel. Data integrity, data freshness, secrecy, and authentication may all be crucial factors in some applications. Data encryption is therefore zMd. Author corresponding is Abdul Hamid. becoming a significant issue. Nevertheless, the requirement for short-term security can be fulfilled due to the inherent limitations of the components in terms of available resources. Resource constraints include things like limited battery life and slow processing speeds. Standardised cryptographic algorithms may use more energy to encrypt data, which significantly shortens the component lifetime. Two fundamental approaches are employed to develop and deploy security primitives equipped with highly restrictive mechanisms. The initial phase involves the development of a novel and efficient cryptographic system with reduced computational complexity. There are several recently proposed lightweight encryption algorithms that come to mind. Furthermore, a proposed alteration to the existing conventional cryptosystem that reduces its overall weight. The issue of secret key distribution continues to be considered a significant challenge, even though it shares similarities with other symmetric encryption techniques. Once more, a substantial quantity of computational tasks must be executed. The excessive consumption of power has the potential to result in the degradation or failure of Internet of Things (IoT) components due to their inherent limitations in resource availability. Based on empirical investigations conducted in the field, it has been determined that the Substitution Layer component within the round-based Advanced Encryption Standard (AES) architecture exhibits the highest energy consumption.

2. LITERATURE SURVEY

The literature review concentrates on using AES, especially for low power consumption, high security, higher performance, and increased efficiency. In-depth research and analysis are also done on the viability of implementation in a VLSI context. NIST (2001) provided computer security. Two Federal Information Processing Standards papers presented the operational methods for two block cypher algorithms. The block cypher algorithm in this mode uses two inverse functions on a key.In 2004, Francois-Xavier Standaert, Gael Rouvroy, Jean-Jacques Quisquater, and Jean-Didier Legat conducted a presentation that delved into the investigation of the effective utilisation of reconfigurable hardware for the implementation of Rijndael encryption. It discussed numerous strategies for effective Advanced Encryption Standard algorithm implementations on FPGA. Many different approaches can result in efficient block cypher designs. The definition of an effective technique took into account the inherent limits of FPGAs. The authors proposed algorithmic enhancements for the substitution box utilised in these structures, along with effective combinations of the diffusion layer and the key addition. The numerical value provided by the user is 25. Farhadian and Aref (2009) introduced a methodology based on power functions to derive and approximate the s-boxes in an efficient manner. The utilisation of power functions over finite fields and the incorporation of specialised inversion functions are integral components in the design structure of S-boxes for cryptographic algorithms, as outlined in the research article. A novel and efficient method is introduced for the cryptanalysis of S-boxes. The aforementioned methodology is characterised by its simplicity, since it does not include any heuristic endeavours. Moreover, it can serve as a convenient tool for promptly identifying fundamental approximations. This unique technique can be utilised to derive approximations for advanced encryption standards (AES) such as S-boxes, as well as for other encryption algorithms like Camellia and Shark.

3. ADVANCED ENCRYPTION ALGORITHM (AES)

The symmetric block cypher AES is capable of handling 128-bit block with 128, 192, or 256-bit keys. An AES input or output is a series of 128-bit binary digits 0 or 1. Bit capacity limits blocks or sequences. Figure 1 illustrates how the algorithm known as AES processes the state, an array with two dimensions of bytes.. Four rows of bytes constitute the block length and contain Nb bytes apiece. In this context, the variable Nb is assigned a value of 4, which denotes the number of columns in the state, measured in 32-bit words.

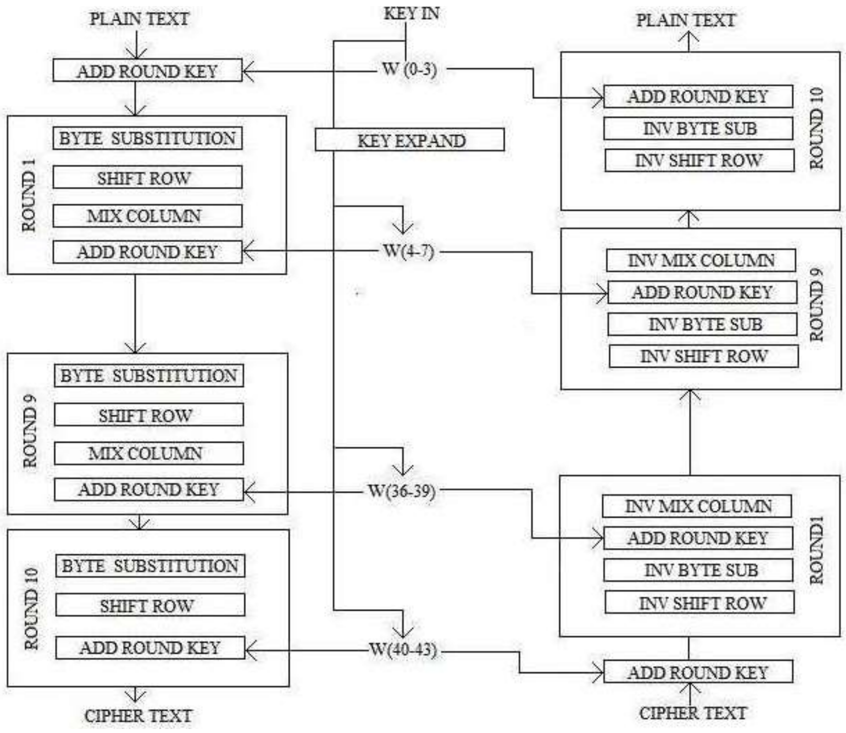


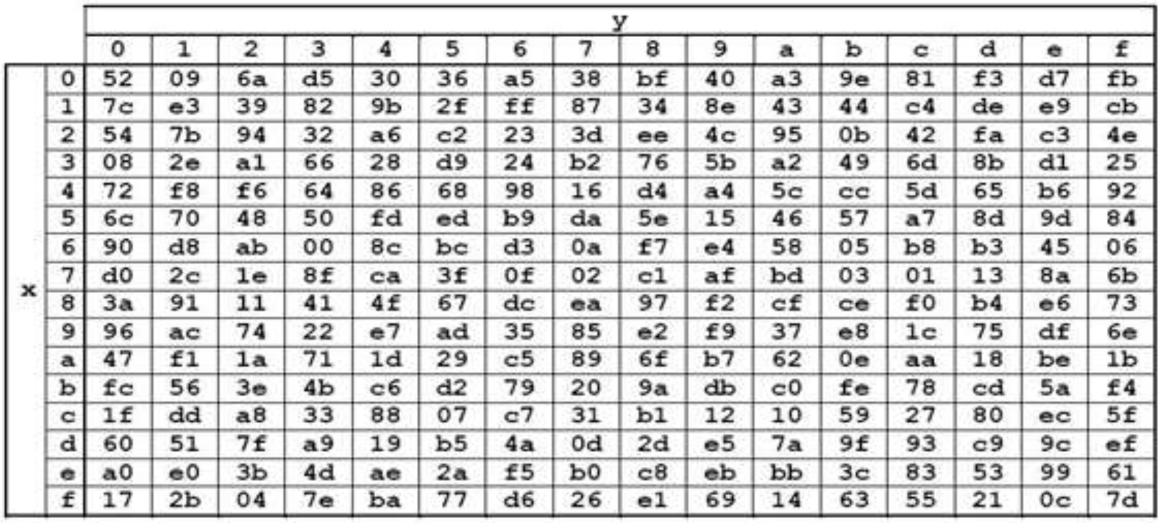
Fig.1 AES Encryption and Decryption

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

Fig.2 State Array

BYTE SUBSTITUTION

The substitute byte transformation (byte sub) is a simple table lookup. Figure.6 shows the process. This table shows the permutation of all 256 8-b values in the AES definition of a 16X16 byte matrix called an S-Box.1. Each state byte is transformed into a new byte: Row and column values of the byte value are left and right 4-b, respectively. The S-Box selects an 8-b output value using these row and column values as indexes. In the S-Box, row 9, column 5, which contains "ad," is referred by "95." As a result, the value 95 gets converted into the value ad. The byte substitution procedure is the same for decryption, but inverted S-Box is used instead, as given in Table 2, and it is applied to each byte of the state [2].



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

Table.1 AES S-Box

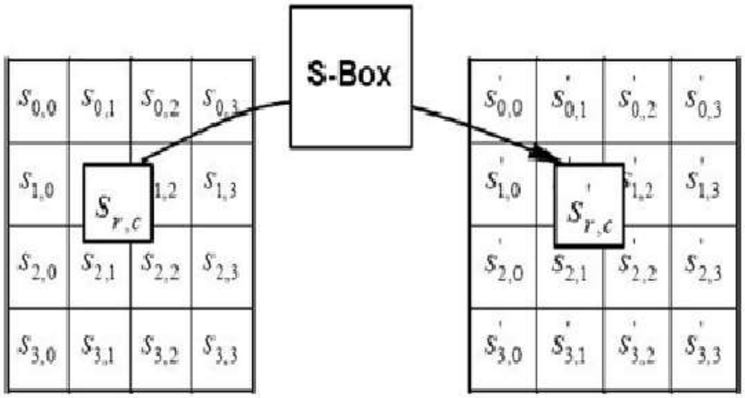
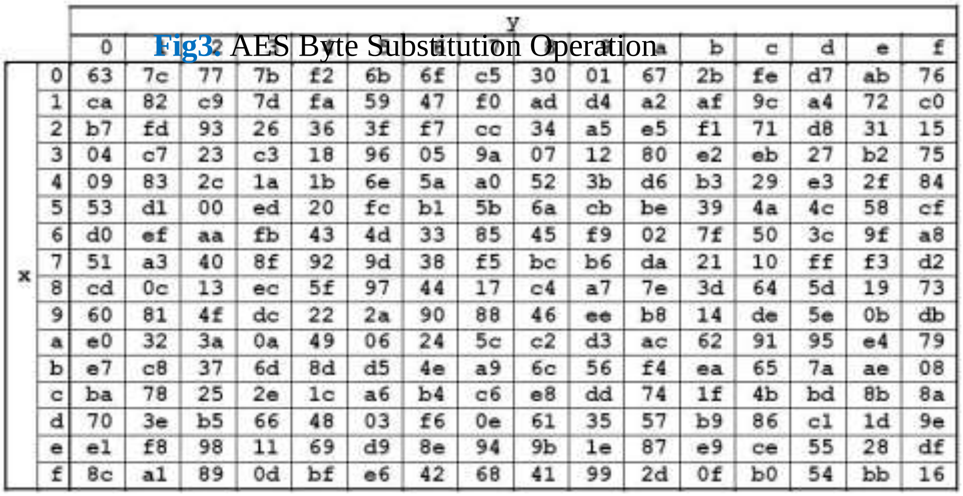


Fig 4. AES Shift row

Fig3: AES Byte Substitution Operation

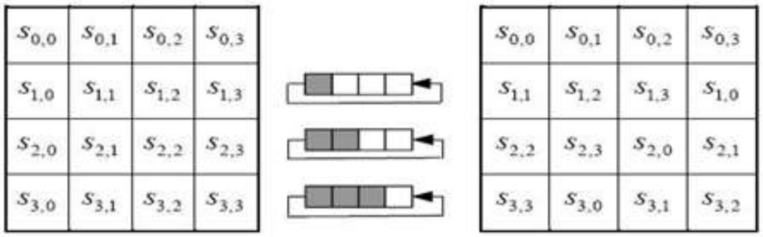


		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	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

Table 2. AES Inverse S-Box

SHIFT ROW

The first row remains unchanged while the next three rows are left circularly shifted by 1, 2, and 3 bytes [4]. This is conceptually depicted in Figure 4.



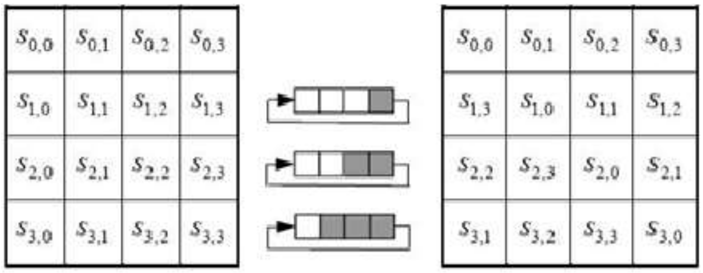

$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,3}$ $s_{1,0}$ $s_{1,1}$ $s_{1,2}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,1}$ $s_{3,2}$ $s_{3,3}$ $s_{3,0}$

Fig 5. AES Inverse Shift Row operation

The first row remains unchanged during the decryption transformation, whereas rows 2, 3, and 4 shift by 1, 2, and 3 bytes, respectively. Figure.8 shows this method.

MIX COLUMN

Mix columns change the state column-by-column into a four-term polynomial. Multiple the column by a fixed polynomial a(x) to get polynomials over GF (28).

Let s`(x) = a(x) x s(x):

The four bytes in a column are changed as a result of this multiplication to the following: And for the encryption procedure, examples of Mix column change are provided in Figure 6.

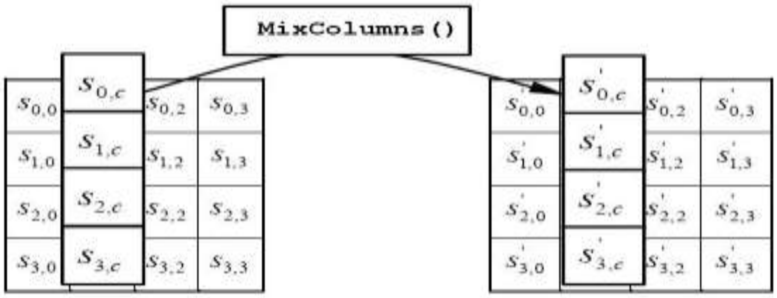

MixColumns ()
$s_{0,0}$ $s_{0,c}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,c}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,c}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,c}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,c}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,c}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,c}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,c}$ $s'_{3,2}$ $s'_{3,3}$

Fig 6. Mix column

The inverse of the decryption mix column transformation multiplied by the fixed polynomial a-1(x) is inv mix column.

$a^{-1}(x) = \{0b\}x^{3}+\{0d\}x_{2}+\{09\}x+\{0e\}$

Let s`(x) = $a^{-1}(x)$ x s(x):

This multiplication replaces four column bytes.

ADD ROUND KEY

The Add round key transformation bitwise XORs state 128-b and round key. Figure.7 depicts the columnwise and byte-level operation between four state column bytes and one round key word. Adding a round key to decrypt is XOR's inverse [2], [4].

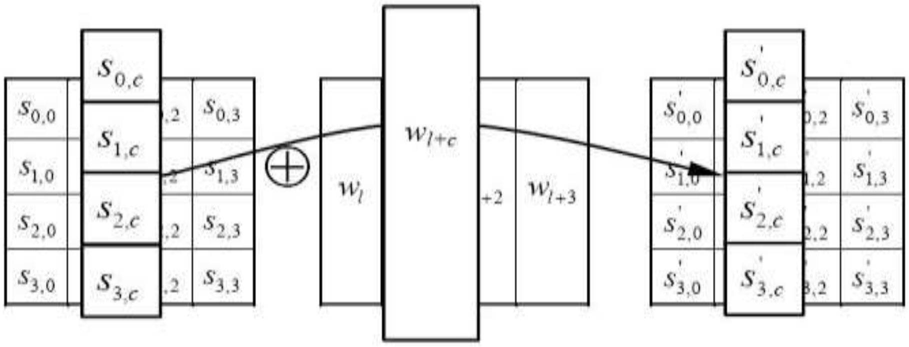
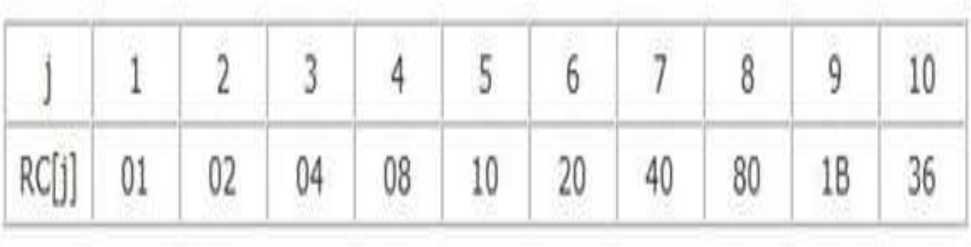


Fig 7. Add round key

KEY EXPANSION

The 4-word (16-byte) key is expanded into a 44-word linear array using AES. This gives a 4-word round key for the initial Add Round Key stage and all 10 cypher rounds. The key is copied in the larger key's first four words. Four words at a time are added to the larger key. Previous and subsequent words, w[i-1] and w[i-4], will influence each new word. Three out of four use a simple XOR. A more sophisticated function is used for words in the w array that are multiples of 4. Figure.11 uses the symbol g to symbolise this intricate function and shows how the first eight words of the enlarged key are generated.



j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

Table 3. Constant values for each round

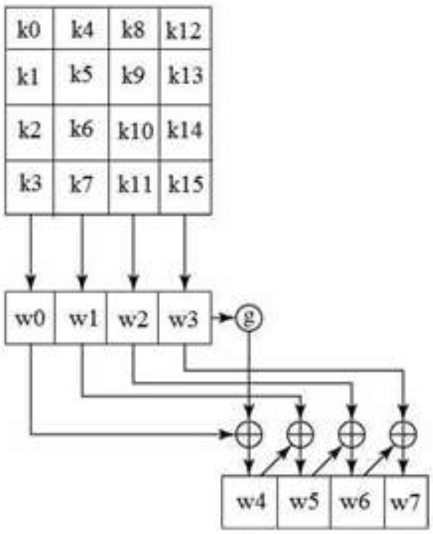


Fig 8. Key expansion

4.RESULTS

RTL SCHEMATIC: The term "Register Transfer Level" (RTL), which is commonly referred to by its acronym, represents the architectural blueprint of a system. This tool compares the current architectural design to the ideal architecture yet to be created. The HDL language, such as Verilog or VHDL, is employed to transform the architectural description or summary into an executable representation. The RTL diagram specifies the internal connection blocks to facilitate analysis.The schematic diagram of the designed architecture's Register Transfer Level (RTL) is illustrated in the image provided.

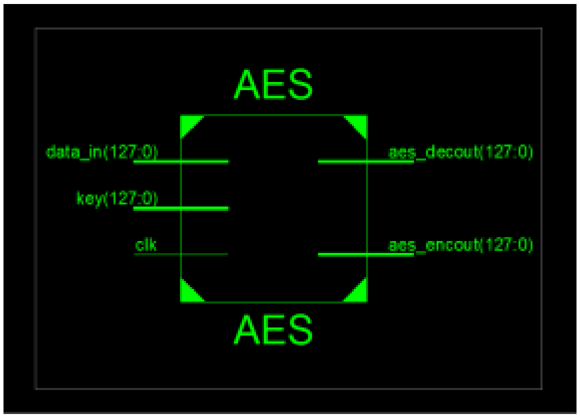


Fig 9. RTL schematic of AES

TECHNOLOGY SCHEMATIC: The technology schematic generates a visual representation of the architecture in the Look-Up Table (LUT) format, VLSI uses the LUT as an area parameter to evaluate architectural design. Lookup Tables (LUTs), square entities, allocate code memory in Field-Programmable Gate Arrays (FPGAs).

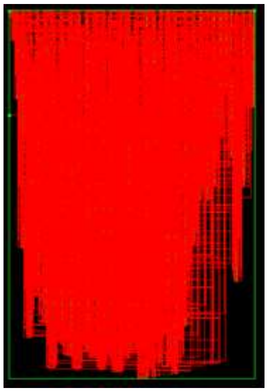


Fig 10. View technology of AES

SIMULATION: The simulation is responsible for the operational aspects, whilst the schematic is utilised for the purpose of validating the interconnections and structural components. Upon accessing the tool's home screen, the simulation window becomes active as the tool transitions from the implantation phase to the simulation phase. The output is constrained within the simulation window, taking the form of wave forms. The approach presented here has sufficient flexibility to accommodate many radix number systems.

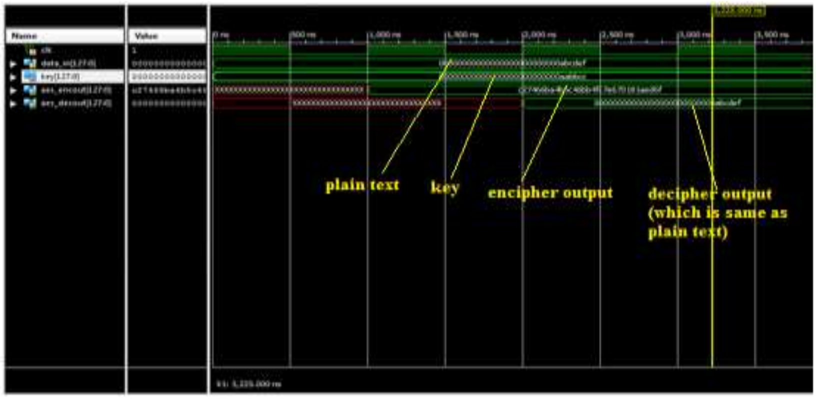
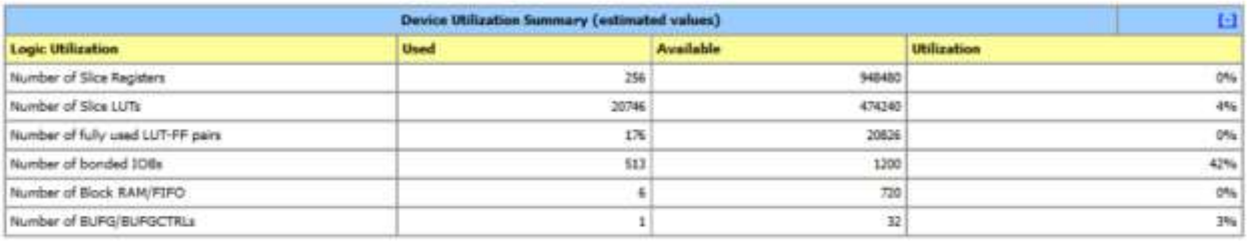


Fig 11. Simulation wave form of AES

PARAMETERS: VLSI metrics like area, delay, and power can be utilised to compare architectures. Verilog is the HDL language, and XILINX 14.7 has a delay-aware option.

PARAMETERS OF AES MODULE



Device Utilization Summary (estimated values)			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	256	948480	0%
Number of Slice LUTs	20746	474240	4%
Number of fully used LUT-FF pairs	176	20826	0%
Number of bonded IOBs	513	1200	42%
Number of Block RAM/FIFO	6	720	0%
Number of BUFG/BUFGCTRLs	1	32	3%

Maximum frequency: 44.714MHz, minimum period: 22.365ns
The 1.463 ns minimum input arrival time before the clock
After the clock, the maximum output is necessary for 55.093ns.

CONCLUSION

For this experiment, we used a 128-bit input and a 128-bit security key, and we watched to see how it was transmitted securely at the output. In this endeavour, the original message is not made available to the hackers. Only the sender and the recipient may see the original message. As a result, in the future, any confidential information (for military or banking purposes) can be communicated securely by employing this project, and AES is also more secure. The Advanced

Encryption Standard (AES) finds extensive utilisation in various contemporary domains, including encrypted data storage, remote access servers, cable modems, video surveillance, and online transactions. Our research aims to expand 128-bit inputs to n bits, where n is any positive integer..

REFERENCES

[1] Madakam, Somayya, R. Ramaswamy, and Siddharth Tripathi. “Internet of Things (IoT): A literature review.” Journal of Computer and Communications 3, no. 05 (2015): p.164.

[2] Wang, Yong, Garhan Attebury, and Byrav Ramamurthy. “A survey of security issues in wireless sensor networks.” IEEE Communications Surveys Tutorial (2006).

[3] Veeramallu, B., S. Sahitya, and Ch LavanyaSusanna. Veeramallu, B., S. Sahitya, and Ch LavanyaSusanna. “Confidentiality in Wireless sensor Networks.” International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013.

[4] Eisenbarth, Thomas, and Sandeep Kumar. “A survey of lightweightcryptography implementations.” IEEE Design & Test of Computers 24.6 (2007).

[5] Banik, Subhadeep, Andrey Bogdanov, and Francesco Regazzoni. “Exploring energy efficiency of lightweight block ciphers.” International Conference on Selected Areas in Cryptography. Springer, Cham, 2015.

[6] Bogdanov, Andrey, et al. “PRESENT: An ultra-lightweight block cipher.” CHES. Vol. 4727. 2007.

[7] Borghoff, Julia, et al. “PRINCEa low-latency block cipher for pervasive computing applications.” International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin Heidelberg, 2012.

[8] Beaulieu, Ray, et al. “The SIMON and SPECK lightweight block ciphers.” Design Automation Conference (DAC), 52nd ACM/EDAC/IEEE. IEEE, 2015.

[9] Suzaki, Tomoyasu, et al. “TWINE: A Lightweight Block Cipher for Multiple Platforms.” Selected Areas in Cryptography. Vol. 7707. 2012.

[10] Li, Wei, et al. “Security analysis of the LED lightweight cipher in the internet of things.” Jisuanji Xuebao(Chinese Journal of Computers) 35.3(2012): p.434-445.