IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DYNAMIC & SECURE SUBSTITUTION BOX FOR EFFICIENT SPEECH ENCRYPTION ENGINE

B. Ravibabu (MTech) - G. Pushya - ,B. Rohith Sai – V. Reshma - P. Priyanka – E. Rohith – B.Ravibabu

Siddharth Institute Of Engineering & Technology , Puttur, Andhra Pradesh

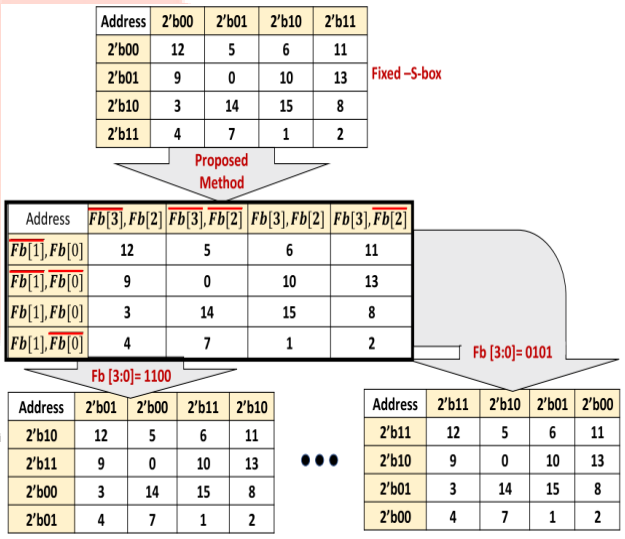
***Abstract*— The Dynamic & Secure substitution box is mainly used in Internet of things which is used globally and devices having resource limitations for the purpose of better encryption. The main advantage of this DS2B is it's elementary structure and better encryption performance. By slightly changing DS2B variables we can generate multiple S-boxes which are robust and less prone to attacks. The main aim of the above mentioned encryption scheme is to attain high non-linearity & low uniformity of differential one. Utilizing FGPA an efficient speech encryption engine can be designed. DS2B makes Speech signals which may be analog or digital very difficult to detect. This proposed encryption method makes signal to achieve high throughput than the older one which is used. As Internet of things are used widely these days using efficient speech encryption engine is necessary to avoid manipulation of speech signals.**

Keywords*— *Internet of things, FGPA, throughput, non-linearity, Look up table

I. INTRODUCTION

Privacy has always been a concern in modern world. We cannot Hide anything whether it may be images, video, audio, or any sort of multimedia. Therefore, the main aim is to reduce the impact of the breach which is occurred frequently. Therefore, one of the method is dynamic speech encryption which is used to prevent the speech manipulation. Creating confusion in the audio samples which may be analog or digital hides the data and secures it. It multiplies the strength & strong trait of the audio or speech signal. Due to low budget, small memory & limited computation capability it is a challenging task to implement encryption algorithms. S-boxes occupy up to 62 percent of the total cryptographic area, and it utilize up to 35 percent of the total budget. Substitution box is a non-linear device encryption which is used for the purpose of encryption algos. S-box is primarily used as a non-linear element in block encryption algorithms in DES & AES. Without confusion in the encryption algorithm, it becomes endangered to various attacks including changing speech signals by varying different parameters. By creating more confusion in the output safety level of a speech signal is increased. All cryptographic attacks target s-box which should be strong & secure. Hence, more randomness is needed in the S-box to secure the block encryption pattern. S-box is very important for the purpose of encryption standards because of their randomness generation & key & text securing

Here the analog signal encryption algorithm determines how secure & strong the speech signal is.



(a) FIGURE 1. DS2B lookup table design model, the address is generated based on changing Feedback bits. The addresses are 2-bits (row) and 2-bits (column). Each data which is selected is of 4 bits.

To conceal the link between key & the text which is cipher we use S-box which used to generate randomness in the modern algorithms (Cryptographic algorithms) Because of Nonlinearity functionality the input data which is generally an audio signal output which is encrypted data is resulted by S-box and the attacker cannot gain any information related to the speech signal from encrypted output data due to its robust design. The growth in standards of a Substitution box should be simple and more efficient than all the methods proposed previously.

The method of getting dynamic & secure substitution box is used mainly in encryption applications which may be audio, video or any other data. Feedback signal gives LUT design, it is used as an address. The design for preparing dynamic & secure substitution box is in the below table as it shows what the bits are resulted in:

High non-linearity substitution box with 4-bits should be created. The address of the substitution box is being replaced with four bit Feedback signal ; from previous O/P Fb signal is obtained

The S-box Look up table address is obtained as follows: Fb [1] is inverted twice then repeated twice. Feeback signal 1[0] is inverted for each row of the table. The same for feedback [2] and feedback[3].

On the basis of previous things of inverting fb, all 4 bit feedback values are obtained & lookup table is generated.

The possibility of s-boxes which can be generated in inverting & non -inverting process is 24 in which the s-boxes which are 16 in number are being generated from a four bit input. For each Feedback value, a Substitution box is obtained.

- If Feedback = 4'b1100, a new Substitution box address is created. In which, the input ''0'' will be replaced with ''14''.
- If Feedback = 4'b0101, a new Substitution box address is created. In which, the input ''0'' will be replaced with ''2''.

Both inverting and the non-inverting output is changed to get other sixteen substitution boxes. The proposed method for generating the 4-bit S-box can be generalized to obtain an 8-bit S-box as shown in Fig. 2. As can be seen, the addresses are 4-bits (row) and 4-bits (column). Each selected data is of eight bits.

	$Fb[7],Fb[6],Fb[5],Fb[4]$	$Fb[7],Fb[6],Fb[5],\overline{Fb[4]}$	$Fb[7],Fb[6],\overline{Fb[5]},Fb[4]$	$Fb[7],Fb[6],\overline{Fb[5]},\overline{Fb[4]}$	$Fb[7],\overline{Fb[6]},Fb[5],Fb[4]$	$Fb[7],\overline{Fb[6]},Fb[5],\overline{Fb[4]}$	$Fb[7],\overline{Fb[6]},\overline{Fb[5]},Fb[4]$	$Fb[7],\overline{Fb[6]},\overline{Fb[5]},\overline{Fb[4]}$	$\overline{Fb[7]},Fb[6],Fb[5],Fb[4]$	$\overline{Fb[7]},Fb[6],Fb[5],\overline{Fb[4]}$	$\overline{Fb[7]},Fb[6],\overline{Fb[5]},Fb[4]$	$\overline{Fb[7]},Fb[6],\overline{Fb[5]},\overline{Fb[4]}$	$\overline{Fb[7]},\overline{Fb[6]},Fb[5],Fb[4]$	$\overline{Fb[7]},\overline{Fb[6]},Fb[5],\overline{Fb[4]}$	$\overline{Fb[7]},\overline{Fb[6]},\overline{Fb[5]},Fb[4]$	$\overline{Fb[7]},\overline{Fb[6]},\overline{Fb[5]},\overline{Fb[4]}$
$Fb[3],Fb[2],Fb[1],Fb[0]$	99	124	119	123	242	107	111	197	48	1	103	43	254	215	171	118
$Fb[3],Fb[2],Fb[1],\overline{Fb[0]}$	202	130	201	125	250	89	71	240	173	212	162	175	156	164	114	192
$Fb[3],Fb[2],\overline{Fb[1]},Fb[0]$	183	253	147	38	54	63	247	204	52	165	229	241	113	216	49	21
$Fb[3],Fb[2],\overline{Fb[1]},\overline{Fb[0]}$	4	199	35	195	24	150	5	154	7	18	128	226	235	39	178	117
$Fb[3],\overline{Fb[2]},Fb[1],Fb[0]$	9	131	44	26	27	110	90	160	82	59	214	179	41	227	47	132
$Fb[3],\overline{Fb[2]},Fb[1],\overline{Fb[0]}$	83	209	0	237	32	252	177	91	106	203	190	57	74	76	88	207
$Fb[3],\overline{Fb[2]},\overline{Fb[1]},Fb[0]$	208	239	170	251	67	77	51	133	69	249	2	127	80	60	159	168
$Fb[3],\overline{Fb[2]},\overline{Fb[1]},\overline{Fb[0]}$	81	163	64	143	146	157	56	245	188	182	218	33	16	255	243	210
$\overline{Fb[3]},Fb[2],Fb[1],Fb[0]$	205	12	19	236	95	151	68	23	196	167	126	61	100	93	25	115
$\overline{Fb[3]},Fb[2],Fb[1],\overline{Fb[0]}$	96	129	79	220	34	42	144	136	70	238	184	20	222	94	11	219
$\overline{Fb[3]},Fb[2],\overline{Fb[1]},Fb[0]$	224	50	58	10	73	6	36	92	194	211	172	98	145	149	228	121
$\overline{Fb[3]},Fb[2],\overline{Fb[1]},\overline{Fb[0]}$	231	200	55	109	141	213	78	169	108	86	244	234	101	122	174	8
$\overline{Fb[3]},\overline{Fb[2]},Fb[1],Fb[0]$	186	120	37	46	28	166	180	198	232	221	116	31	75	189	139	138
$\overline{Fb[3]},\overline{Fb[2]},Fb[1],\overline{Fb[0]}$	112	62	181	102	72	3	246	14	97	53	87	185	134	193	29	158
$\overline{Fb[3]},\overline{Fb[2]},\overline{Fb[1]},Fb[0]$	225	248	152	17	105	217	142	148	155	30	135	233	206	85	40	223
$\overline{Fb[3]},\overline{Fb[2]},\overline{Fb[1]},\overline{Fb[0]}$	140	161	137	13	191	230	66	104	65	153	45	15	176	84	187	22

(b) FIGURE 2. DS2B Look up table design, in which the address is generated based on changing the Feedback bits. Each selected data is of eight bits.

I-1 Differential and Linear Cryptanalyses:

The 2 important methods in analysis of cryptographic methods of a block cipher are linear & differential cryptanalyses. In the below equation linear & differential analyses on the substitution boxes are shown. Linear indicator (LS) is used to calculate & measure the non-linearity of the Substitution box. It is denoted by the following equation:

$$L(S) \triangleq \max_{\substack{a\in\mathbb{F}_2^n* \\ b\in\mathbb{F}_2^m*}} \left| \left|\{x \in \mathbb{F}_2^n \mid a\cdot x = b\cdot S(x)\}\right| - 2^{n-1} \right|$$

Where ''L(S)'' is the linear indicator, ''x'' is the input x = [x3, x2, x1, x0] and ''S(x)'' represents the output S(x) = [S(x3), S(x2), S(x1), S(x0)]. The range of a, b and x is [0 : 2 n − 1]. Here variables a and b are input & output and a · x is represented as a · x = a3 · x3 ⊕ a2 · x2 ⊕ a1 · x1 ⊕ a0 · x0 and b · S(x) = b3 · S(x3) ⊕ b2 · S(x2) ⊕ b1 · S(x1) ⊕ b0 · S(x0).This operation represents AND and all these are binary operations.

$$\ell_{max}(S) = 2L(S)$$

In the above equation max(S) is also known as linearity of the substitution box S(x) and the nonlinearity is shown as:

$$NL(S) = 2^{n-1} - L(S)$$

From the above equation it can be seen that Linear Indicator is directly proportional to linearity & nonlinearity parameters which are the main parameters used to differentiate between secure & non secure elements. DU should be checked for any substitution method mentioned here

$$DU \triangleq \max_{\substack{a\in\mathbb{F}_2^n* \\ b\in\mathbb{F}_2^m*}} \left|\{x \in \mathbb{F}_2^n \mid S(x)+S(x+a)=b\}\right|$$

In The above equation a, b and x variables have range from [0: 2 n − 1]. L and du values has to be low to withstand different types of cryptographic attacks. Substitution boxes which are 16 in number are generated which is obtained from the feedback signal on the basis of algorithm which is a 4-bit.

16 substitution boxes are generated in which NL(s)=4, L=4 and De-max = 4 .This indicates that all the S-boxes which are generated has same nonlinearity factor and it is of important one. Based on the above equation which we analyzed NL=2n-1-L(s), nonlinearity is obtained & computed in which nonlinearity is directly proportional to 'n' i.e. as 'n' increases nonlinearity increases. Nonlinearity of 112 is obtained for an eight bit DS2B. 256 substitution boxes are generated based on the proposed algorithm from the feedback signal. Same as DS2B that is a 4 bit one, all the 8-bit DS2B's which are 256 in number have the similar non-linearity. All the equations describe the approximations and distribution tables of the DS2B. are given below. In the linear equation a · x = b · S(x) where a is the input and b is the output . The number of matches is then added to eight which is negative. At last we obtain the concluded output.

Linear approximation probability (LP) is as:

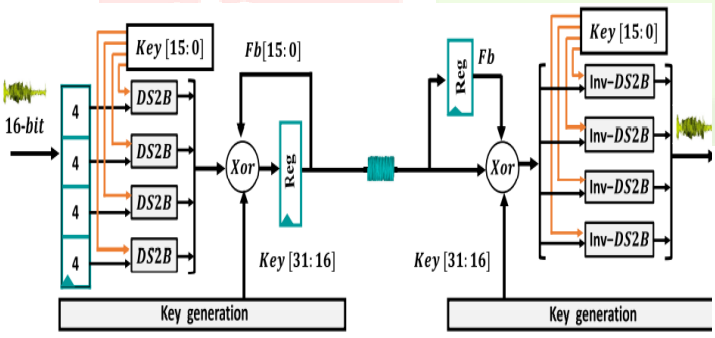
$$LP = max_{a,b\neq 0} \left| \frac{\{x/a \cdot x = b \cdot S(x)\} - 2^{n-1}}{2^n} \right|$$

A and b represents input & output masks. By generalizing the above equation we get to know that the substitution box with smallest LP value will have good nonlinearity.

By using linear & differential cryptanalyses for the substitution box gives L=8, NL =0 an DU = 16. Linear analyses results shows that the substitution box achieved non-linearities which are unnecessary. so static S-box has many disadvantages compared to dynamic one.

I-2 Speech Encryption Scheme:

This new speech encryption method is fully different from previous one's which has many drawbacks. The figure below shows the design of the of the proposed encryption/decryption scheme without the key generation procedure (previously used method). The 16 bit input speech / Audio signal is given as input here. To drive the four DS2B blocks the given 16 bit input should be divided into four parts. The four blocks of DS2B inputs are taken from feedback which is in the ratio of 15 and 0 and from the input audio signal which consists of 16 bits. The outputs of the four DS2Bs are added to obtain the result then ''XORed''(XOR operation is If the two bits are same, the result is 0. If the bits are different, the result is 1.) with the Feedback signal to produce the encrypted signal. As at last we obtain the encrypted signal from the previous encrypted output or from the signal generated by the feedback.

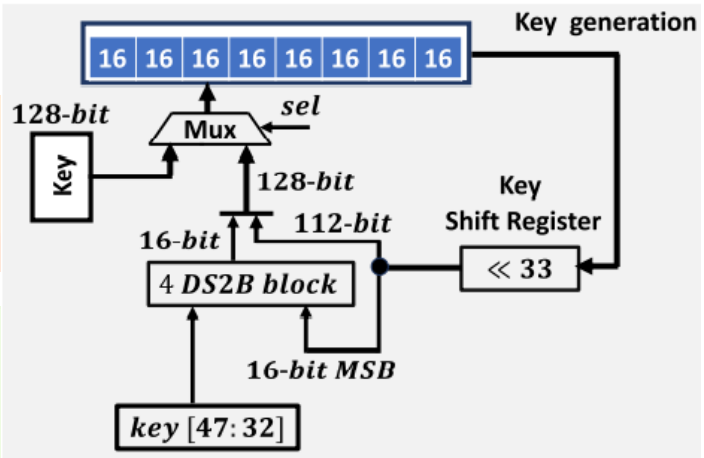


© Fig- 3: Dynamic encryption hardware implementation along with key generation

The decryption process can be done by inverting the operation as we see in the above figure. In previous method there is no generation of key block but in dynamic S-box we use key generation block as shown in the above figure. Here the input encryption key is 128 bits. The encryption key drives the key generation block, and a multiplexer loads the input key when the select ''sel'' signal is set to ''1'' or to select the previous encryption key the ''sel'' signal is set to ''0''. The key register is shifted left (33-bit) as shown below. The important 16 bits here which are present in the register which is a shift register it accelerates the four DS2B blocks present. The shift key register LSB and the outputs of 4 DS2B blocks is combined to create an encryption key. Key [15:0] bits are used to create the 4 DS2Bs. Lastly, the outputs of the DS2B's are XORed with the key of 31 and 16 which are in the ratio to generate encrypted signal. Decryption is reversing the encryption output and a larger bit DS2B can be further reduced to smaller bit DS2B. but doing this will lead to security reduction and increasing in attacks, so it is better to avoid that operation which will lead to the safety concern.

I-3 Encryption Performance Analysis:

Three sample files which are generally audio are used to check the level of performance of the proposed encryption scheme. Here in the first file sample the voice sample was sampled at 44KHz and similarly Files 2 and 3 are obtained from the database where speech signals are located and sampled at 50 KHz. Second file is an empty signal used to test the proposed encryption method against zero input speech. The time waveforms of three test files show various amplitude of the input speech(voice) signal. Graph shows the distribution of frequencies from which we cans see the spectral density of all the speech signals. Here speech signal varies with time. The histogram also shows the distribution of values for the overall speech file. To generate a secure encrypted speech signal, the encrypted speech signal should have no proposed correlation with the input voice signal.



Input of 128 bits is given to the element. One 4-bit DS2B is Used. The waveforms of the encrypted speech signal obtained from the above encryption scheme which are in time domain is approximately equal to the noise and it is also unique compared to original one. The amplitudes of the obtained signals are scattered uniformly and this shows perfection in the histograms when seen. The proposed 4-bit DS2B result shows that the randomness in speech file 2 is more compared to speeches 1 and 3. Speech file 2 is bad in terms of result. To solve the problem of the silence periods present key generation block is added to the previous scheme. The various graphs showing the encrypted speech files are shown for the proposed encryption method & are obtained for speech file 2 and 3 as well. Along with the key generation. As can be seen, all these results are more uniform and different than the original speech file and have no similarity to the input speech signal.

II VERILOG (HDL)

Verilog is a HDL as it is one of the hardware languages primarily used in electronics to communicate with the hardware devices similar to the programming languages. It is an automated design system which is an integrated company. It officially started in the

year 1985. It's features are partially taken from previous HDL called HiLo and also from the most popular programming language called 'C'. there were many updates for this language from initial time as it has many drawbacks. As it was not standard at that time it was considered slow but by improving many aspects it came up with a revised edition in the year 1990.In the year 1985 the simulator was first used and promoted. Verilog-XL has many features which has XL algorithm which was efficient method to perform simulation at gate level.

```
module KEY_GENERATION (clk,rst,SEL,KEY_IN,KEY_OUT);
  input   clk;
  input   rst;
  input   SEL;
  input    [127:0] KEY_IN;
  output   [31:0] KEY_OUT;



  wire [127:0] CON_KEY;
  wire [15:0]DE2B_OUT;
  wire [127:0]KEY_SHIFT_OUT;
```

est_bench.v | CORE_TOP.v | DS_4IN_LUT_ENGINE.v | DS_8IN_LUT_ENGINE.v | KEY_GENERATION.v

Popularly used approach used in Verilog is top down methodology which was used by Synopsys previously. In 1990 Verilog gone many changes if it a closed language and it's standards are different from previous one's then VHDL will be more industry specific. An event organized by the esteemed organization called cadence which is Verilog open in the year 1991. It released documentation for the HDL language. This was the event which "opened" the language. Verilog language's code is easy to write because of its some 'C' language features & methods.

II-1 Hardware Description Language

Verilog is a simple & easy to learn and use. Its syntax is similar to C. As it's syntax is similar to c writing programs in hardware becomes easier. This shows how data flows between registers & how the data is processed. The hierarchical design of the RTL plays a very important role in this design. It contains more levels in this method flow. Hardware description language describes all the hardware components connected in an integrated circuit with the help of code at small level with flip flops & logic gates in the digital systems. HDL is also widely used in computer aided designs and also plays important role in digital circuit systems. HDL also audits the code. It contains automatic checkers to detect wrong syntax and also check for common logic coding errors.

```
1 module TestTop();
2     reg CLK;
3     reg  [31:0] cycle;
4
5     wire [7:0] ot_cnt;
6     Counter cnt(CLK, ot_cnt);
7
8     initial begin
9         CLK = 0;
10        cycle = 0;
11        cnt.cnt = 0;
12    end
13
14    always #50 CLK = ~CLK;
15
16    always @(posedge CLK) begin
17        cycle <= cycle + 1;
18        if (cycle > (`HALT_CYCLE - 10))
19            $write("%d %d\n", cycle, ot_cnt);
20        if (cycle >= `HALT_CYCLE) $finish;
21    end
22 endmodule
```

II-2 Xilinx Tool

Xilinx Tools is tool based on software which is mainly used for the purpose of circuits i.e., digital using FGPA or complex Programmable Logic Device (CPLD). The first step is entry for design, the second step is implementing the design and synthesizing it and the last step is simulation of function and testing and verification. CAD tools are used to design them using HDL -Verilog or mixed combination. With CAD tools we can design more combinational & sequential circuits like Verilog HDL designs. It is used to perform timing analysis & Model Sim Logic Simulator is used for the purpose of System level testing. Test bench programs contain simulated input signal waveforms & verify the outputs of devices under test. I simulator is a featured & fully completed simulated integrated tool. The primary step of simulation done here is flow of design with integration of I simulator in design environment. Some of its features are it supports mixed language and VHDL-93 and Verilog 2001 for MGT,PCIe etc. It doesn't require special license.

Header: module name, list of input and output ports.
Declarations: input and output ports, registers and wires.
Logic Descriptions: equations, state machines and logic functions.

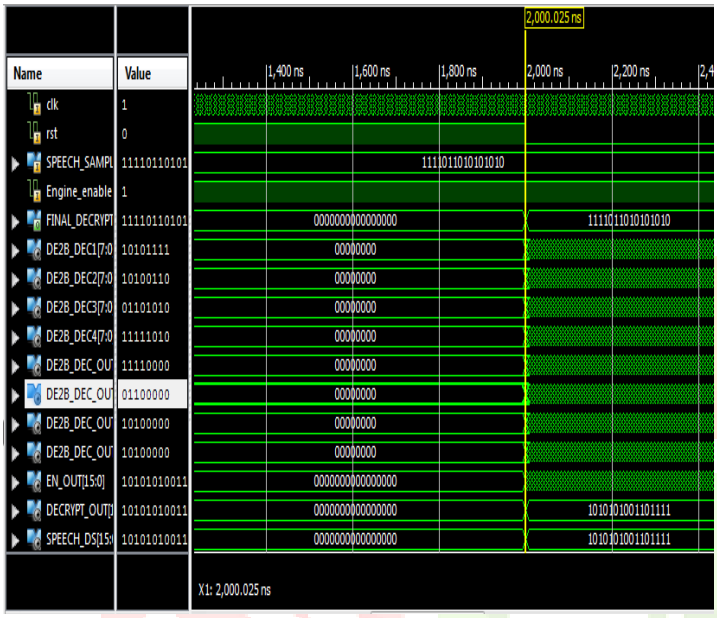
End: end module

The above format is specified for Verilog input format and the diagram which defines state doesn't exist for logic designs which are combinational.

II-3 Programmable Logic Device: FPGA

In this method designs are done with the help of Basys2 board which has a Xilinx Spartan3E –XC3S250E FPGA with CP132 package. These devices come in a variety of packages. We will be using devices that are packaged in 132 pin package with the following part number: XC3S250E-CP13. Some of the features of Xilinx tool are Power Analysis and optimization using SAIF.Memory, eeditor for viewing and debugging memory elements,

III SIMULATION RESULTS



To enhance the security of hardware oriented IOT structures crypt algorithm based S-Box are used. The security of the algorithm is increased by producing more confusion in the S-box output. Therefore, an S-box is targeted for improving the security in encryption algorithms.-based cryptography is the effective strategies that attract researchers and make them curious about it.

Chaotic systems show characteristics which are prominent to use them in encryption methods & sensitivity one of the property & unpredictability of systems make them suitable to be used in encryption. Normally the systems here are combined with the original techniques of encryption.. Tent map which is changed is used with bit permutation strategy in audio encryption application.

III CONCLUSION

The DS2B scheme passed all the tests to withstand all attacks & show it's robustness. This method proves that the obtained Substitution box doesn't compromise in terms of safety & security. Resistance shown against the attacks of cryptography. The ability is outstanding. This is mainly realized on FGPA using a hardware description language called Verilog which is used widely in all types of hardware systems. we differentiated previous static & new encryption schemes and evaluation is done on basis of minor variation in parameters and usage of hardware resources are presented. Efficiency is achieved through this Substitution box. Output of this design gives a throughput which is more than the previous ones. Due to these parameter variations & high nonlinearity & high efficiency the signals are harder to detect & manipulate. Dynamically varying substitution box has more positives than static S-box and therefore more secure & efficient.

That is the primary reason why Dynamic & secure substitution box is prominently used in all types of encryption devices mainly block encryption algorithms to make the speech signal stronger and more tolerant to all types of attacks.

IV REFERENCES

[1] W. Yu and S. Köse, ''Advanced encryption standard is implemented for IOT protections against cryptographic attacks,'' Circuits Syst. I, Reg. Papers, vol. 64, no. 11, pp. 2934–2944, Nov. 2017. [2] M. Alhawari, D. Kilani, B. Mohammad, H. Saleh, and Mohammed Ismail, ''thermal energy harvesting & power management for microwatt biochips which are wearable'' in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), May 2016, pp. 2258–2261.

[3] M. Alhawari, T. Tekeste, B. Mohammad, H. Saleh, and M. Ismail, ''Power management unit for multi-source energy harvesting in wearable electronics,'' in Proc. IEEE 59th Int. Midwest Symp. Circuits Syst. (MWSCAS), Oct. 2016, pp. 1–4.

[4] S. Mathew, S. Satpathy, V. Suresh, M. Anders, H. Kaul, A. Agarwal, S. Hsu, G. Chen, and R. Krishnamurthy, ''340 mV–1.1 V, Solid-State Circuits, vol. 50, no. 4, pp. 1048–1058, Apr. 2015.

[5] G. Bansod, N. Raval, and N. Pisharoty, ''Implementation of a new lightweight encryption design for embedded security,'' IEEE Trans. Inf. Forensics Security, vol. 10, no. 1, pp. 142–151, Jan. 2015.

[6] D.-H. Bui, ''An innovative lightweight cryptography system for Internetof-Things ULP applications,'' Ph.D. dissertation, Dept. Micro Nanotechnol./Microelectron., Hanoi Nat. Univ., Hanoi, Vietnam, Univ. Grenoble Alpes, Grenoble, France, 2019. [Online]. Available: https://tel.archivesouvertes.fr/