Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ECC IMAGE ENCRYPTION USING SYSTEM GENERATOR

SARA CHILLALI[1], LAHCEN OUGHDIR[2]

[1]Sidi Mohamed Ben Abdellah University, FP, LSI, Taza, Morocco

[2]Sidi Mohamed Ben Abdellah University, ENSAF, Fez, Morocco

E-mail: [1]sara.chillali@usmba.ac.ma, [2]Lahcen.oughdir@usmba.ac.ma

ABSTRACT

This work includes research findings on encryption, decryption, and various image processing operations using programmable operations, on Matlab, Simulink, and the Xilinx system generator. The main question is how to ensure the security of images stored in real time on embedded systems which depends on some precise operations on these systems. To answer this question, we will build new and original methods using cryptography on an efficient elliptical curve. The concept of software and hardware co-simulation for our encryption method, presents an efficient architecture of various imaging, encryption, decryption and key generation algorithms using such an elliptical curve, thus using Simulink Xilinx, we build Simulink Xilinx models for implementing multiple hardware operations on various Xilinx FPGAs, for all kinds of color and grayscale images with the minimum number of generator blocks possible.

Keywords: *Xilinx System Generator, FPGA, ECC, Encryption, Decryption.*

1. INTRODUCTION

The obligation to store images of different sizes using embedded systems and to encrypt them to secure them, leads us to reflect on the construction of new cryptographic methods that resist spy attacks. Many algorithms have been proposed to encrypt the image, such as DES, Triple-DES, AES and RSA which cannot be used to encrypt the image, due to its size, so these algorithms do not take into account the statistical properties of the image such as the strong correlation between the pixels, also the Hill cipher, the Vigenère cipher and the affine cipher, these cryptosystems are vulnerable to attacks; such as chosen or known clear text attacks, and the difficulty of passing the secret key due to its size.

We start from an image M and an image S of the same size which represent the secret. [8, 9] The image S, which we want to encrypt and send to a correspondent, the main thing being that it is exactly the same size as M.

This M image is the only one we see if we don't have a computer to run an XOR. [6, 7]

We present in detail our method used for encryption of different types and sizes, its implementation using certain elliptic curves, the efficiency of this method is ensured by the difficulty of solving the problem of the discrete logarithm on elliptic curves (DLP). [4]

First, we build an encryption key K, on an elliptical curve well chosen by the Diffie-Hellman method, [1] then a matrix key A, will be built from the key K.

The digital image is represented by a matrix of dots called pixels. [2]

We can represent an image by a matrix $M(m_{i,j})$; $m_{i,j} \in \{0,1,2,\dots,255\}$.

Secondly, we give a new method of image encryption, based on this problem, using the Diffie-Hellman protocol on such a curve, the security of this type of encryption is proven because the non resolution of this problem on these curves.

In our work entitled: "ECC Image Encryption Using Matlab Simulink Blockset" [3], we proposed a software simulation method, providing models (set of Simulink blocks) and presenting the concept of software simulation using Matleb Simulink for processing and encryption of images. In this work, we propose an encryption method by processing the image from the three parts; FPGA, MATLAB Simulink and Xilinx System Generator. Simulink is an environment integrated into Matlab.

2. SECRET KEY BY DIFFIE HELLMAN METHOD

We choose an elliptic curve on a finite field $\mathbb{F}_P$, whose DLP is difficult to solve and this to

have a sure security, using the Diffie-Hellman method on such a curve we can build a secret key which allows us to generate a secret matrix to encrypt and decrypt our images. Recalling that an elliptic curve on a finite field $\mathbb{F}_P$, where p is a prime number greater than or equal to 5, is a curve defined by an equation of the form: [5]

$$y^2 = x^3 + ax + b, \quad (1)$$

where a and b are in $\mathbb{F}_P$ and $\Delta = -16(4a^3 + 27b^2)$ is not equal to zero in $\mathbb{F}_P$.

We denote this elliptic curve by:

$$E_{a,b}(p) = \{(x,y) \in \mathbb{F}_p^2 / y^2 = x^3 + ax + b\} \cup \{[0:1:0]\} \quad (2)$$

We define the law + on $E_{a,b}(p)$ by:

Let $P(x_1, y_1)$ and $Q(x_2, y_2)$ are two points in $E_{a,b}(p)$, then we can uniquely describe a point, $P + Q = R$, in the following way.

- $R = [0:1:0]$, for $x_1 = x_2, y_2 = -y_1$; (P and Q are opposites of each other)
- $R(x_3, y_3)$;

$x_3 = t^2 - x_1 - x_2$ and $y_3 = -tx_3 - s$, with:

$$t = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, & \text{if } P \neq Q \\ \frac{3x_1^2 + a}{2y_1}, & \text{if } P = Q \end{cases}$$

and

$$s = \begin{cases} \frac{x_2 y_1 - x_1 y_2}{x_2 - x_1}, & \text{if } P \neq Q \\ y_1 - \frac{ax_1 + 3x_1^3}{2y_1}, & \text{if } P = Q \end{cases}$$

A series of k consecutive additions of point P is called multiple point of order k;

$$kP = \underbrace{P + P + \cdots + P}_{k \text{ times}} \quad (3)$$

$(E_{a,b}(p), +)$ is an commutative group, [0 :1 :0] is the neutral element.

Let Q also be an element of a cyclic subgroup $G =< P >$.

An integer k that solves the equation $kP = Q$ is termed a discrete logarithm of Q to the base P. One writes $k = \log_P Q$.

Given P and Q, find k such that $kP = Q$, is called discrete logarithm problem (DLP).

3. SECRET MATRIX

Let K be the secret key built by the Diffie-Hellman method on a well-chosen elliptic curve $E_{a,b}(p)$, of which P the public point of this curve of known order r, we construct the secret encryption matrix T as follows:

1. The construction of the secret matrix $A = (a_{i,j})_{i=1,\ldots,f;\ j=1,2,3,4}$.

For $1 \leq i \leq f$, if $iK = (x_i, y_i)$, then $a_{i,1} = i$, $a_{i,2} = x_i$, $a_{i,3} = y_i$ and $a_{i,4} = x_i + y_i$ i.e :

$$A = \begin{pmatrix} 1 & x_1 & y_1 & x_1 + y_1 \\ \vdots & \vdots & \vdots & \vdots \\ f & x_f & y_f & x_f + y_f \end{pmatrix} \text{ modulo f.}$$

2. The construction of the secret matrix T that is the same size as the image.

One of the algorithms at this work is to find the key matrix T from matrix A.

Let $[n, m, p]$ the size of the image to be encrypted, then T.

➢ **Algorithm :**

```
INPUT:
1- INPUT n, m, p
2- T ← zeros(n,m,p)
OUTPUT:
1- for i from 1 to n do
      for j from 1 to m do
       for k from 1 to mod((i+j),f)do
          T(j,i,1) ←mod(A(k,2),f);
       end do
    end do
  end do
2- for i from 1 to n do
      for j from 1 to m do
       for k from 1 to mod((i+j),f)do
          T(j,i,2) ←mod(A(k,3),f);
       end do
    end do
  end do
3- for i from 1 to n do
      for j from 1 to m do
       for k from 1 to mod((i+j),f)do
          T(j,i,3) ←mod(A(k,4),f);
       end do
    end do
  end do
   return T;
```

4. SIMILUNK XILINIX DESIGN

To perform the cryptography task, Simulink provides the Xilinx system generator which needs to be mapped to MATLAB, whose Matlab environment, Simulink is directly used to build algorithms, see (Simulink User Guide, 2010, Matlab hdl coder and Xilinx System Generator) using tools.

In the Matlab Simulink environment, in an appropriate simulation time, we simulated these algorithms and then obtained the results by configuring System Generator for the appropriate FPGA board, Virtex5 xc5vlx110t-3ff1136.

After compilation, the programming in VHDL is created in a file, it can remain open from Xilinx ISE.

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

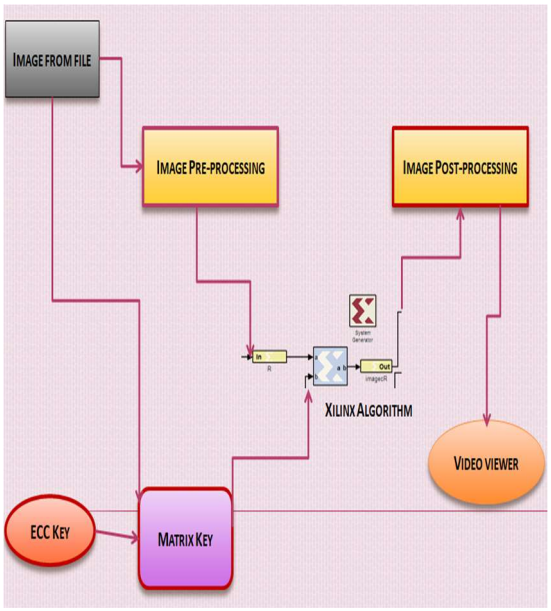

Figure 1: Design model of the system generator encryption.

To check the encryption and decryption module that has been implemented on the FPGA, by the Xilinx generator system which generates a file (UCF) to do such a test.

We represent our proposed approach in Figure 1, which illustrates the system generator encryption design model.

Figure 2, shows the co-simulation of the encryption of certain selected images, from a database which constitutes a graphical user interface which groups Matlab, Simulink and Xilinx.

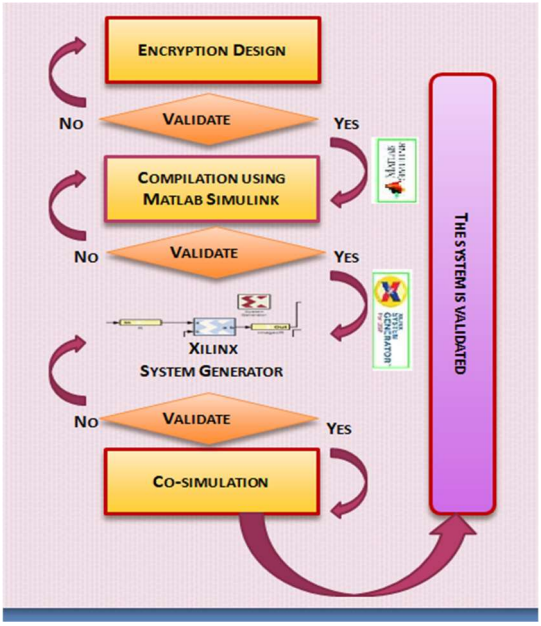

Figure 2: Proposed encryption flowchart.

4.1 Image Pre-processing

In Figure 3, we have shown the image pre-processing blocks which are used to transform the type of image entry point,

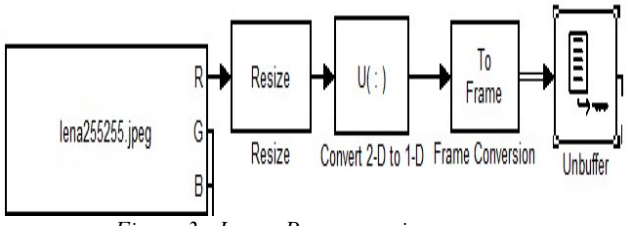

Figure 3: Image Pre-processing.

If the block generates a binary or intensity image, then this is a matrix which has n rows and m columns, i.e. the image is a matrix of size [n, m], the same for a color image generated by the block, it is a matrix of size [n, m, p], where n and m are the numbers of rows and columns in each color plane, p. Then the transpose block transposes the matrix, converting the 2-D block to 1-D converts any input data (2-D) to 1-D (serial) format and the frame conversion block defines the d mode output sampling based on the frame or sample. Unbuffer block defuses an n-by-m input into a 1-by-m output. That is, the entries are unbuffered per line.

4.2 Image Post-processing

Figure 4, shows the image post-processing blocks which are used to convert the type of image output point,

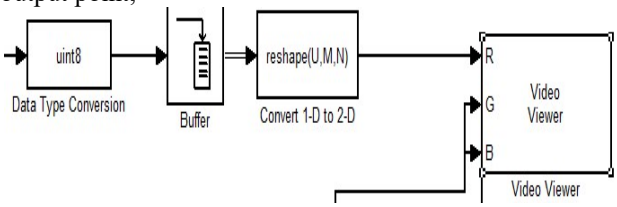

Figure 4: Image Post-processing.

The input is a real or complex value signal, the input and the output are of the same type, so the input signal is converted to the data type required for the output data type parameter. The buffer block performs frame-based processing.

The frame size of the output produced by the block is different from the data in each column of the input. The output frame rate is slower than that of the input, if the buffering of a signal to an image size is larger. Converting a 1-D to 2-D block to convert the input data (1-D) to 2-D (frame) format. A contiguous sub matrix is extracted from the n by m input matrix by the sub matrix block.

4.3 Encryption and Decryption System Generator Block

The architecture explained above only concerns the level of software simulation. For this purpose, the main image encryption module is

converted the generator block of the system illustrated in Figure 6, allows us to make the material co-simulation, with a key generated from an elliptical curve, the block was configured from the target platform. Figure 5, is an illustration of this entire architecture with the hardware and software co-simulation design.

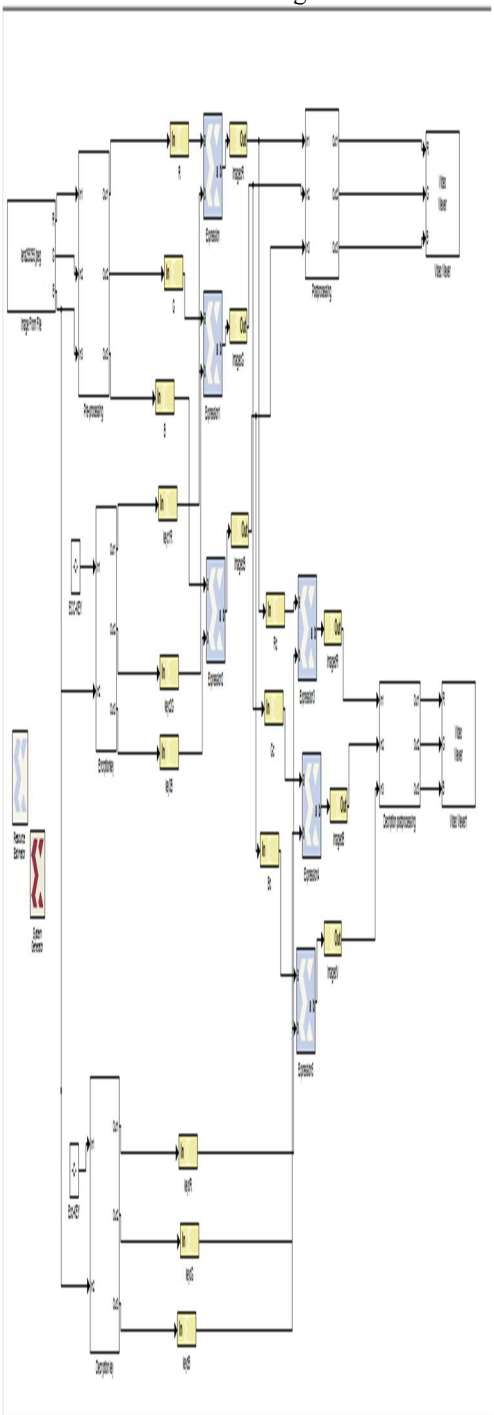

Figure 5: Encryption and Decryption System Generator Block.

5. IMPLEMENTATION AND RESULTS

Computing discrete logarithms on elliptic curves over finite fields is considered to be a very difficult problem. The best algorithms for general elliptic curves take exponential time, and do not take much advantage of properties of the curve. Solving such problems can thus be computationally infeasible for large curves. Indeed, elliptic curves are becoming increasingly appealing for applications in cryptography. Although hard instances of the elliptic curve discrete logarithm may be impossible to solve, Magma is able to efficiently solve reasonable sized instances, or instances where the large prime factor of the order of the base point is not too big.

This part describes an example of an elliptic curve defined over the finite field. The details concerning their construction, their arithmetic and the basic calculations are implemented by Magma.

Code in the Magma box:

- p:=NextPrime(2315643278321);
- k := FiniteField(p);
- A := Random(k);
- B := k!1;
- E:=EllipticCurve([k | A,B]);
- E;
- ord:=Order(E);
- S := Generators(E);
- S;
- Q := Random(E);
- Q;
- t:=Order(Q);
- ord;
- t;
- K:=34124*23467*Q;
- for i := 1 to 256 do i ,i*K;
- end for;

Elliptic Curve defined by:

y^2 = x^3 + 778992873375*x + 1 over GF(2315643278329)

(673846167572 : 885360532730 : 1)

2315644954651

330806422093

1 (1172364145294 : 987118381128 : 1)

2 (1216683893201 : 1776597408574 : 1)

3 (1353701176451 : 1870229614832 : 1)

……

256 (1137963159740 : 924589302537 : 1)

Due to its ability to generate HDL code, Xilinx System was used to implement image encryption algorithms.

We connected the Gateway In and Gateway Out blocks with the preprocessing and

post processing blocks, to implement these algorithms.

The inputs and outputs of the Xilinx part of the Simulink design are served by these blocks, we take:

a:= 778992873375;
b:=1;
p:= 2315643278329;
P:=(673846167572 , 885360532730);
The key built is:
K:=(1172364145294 , 987118381128);
we have:

$$A = \begin{pmatrix} 1 & 20 & 72 & 92 \\ \vdots & \vdots & \vdots & \vdots \\ 256 & 188 & 9 & 197 \end{pmatrix} [256]$$

To decrypt an image, we follow the following steps:

- Setp1 :

We transforms the image, 'image' to the matrix $M(m_{i,j})$; $m_{i,j} \in \{0,1,2,\dots,255\}$.

- Step2 :

We transforms M to matrix Mt and T to matrix Tm using "Encryption and Decryption System Generator Block".

- Setp3 :

We calculates the column vector, $R = \mathrm{mod}(Tm + Mt, 2)$.

- Setp4 :

We transforms R into a matrix Rt and then encrypt the image "encryptimage" by the "Encryption and Decryption System Generator Block".

5.1 Illustration

The different encryption and decryption operators implemented in this article are given below with their corresponding hardware outputs obtained.

The input image used for encryption and the outputs of various operators is illustrated in Figure 6.

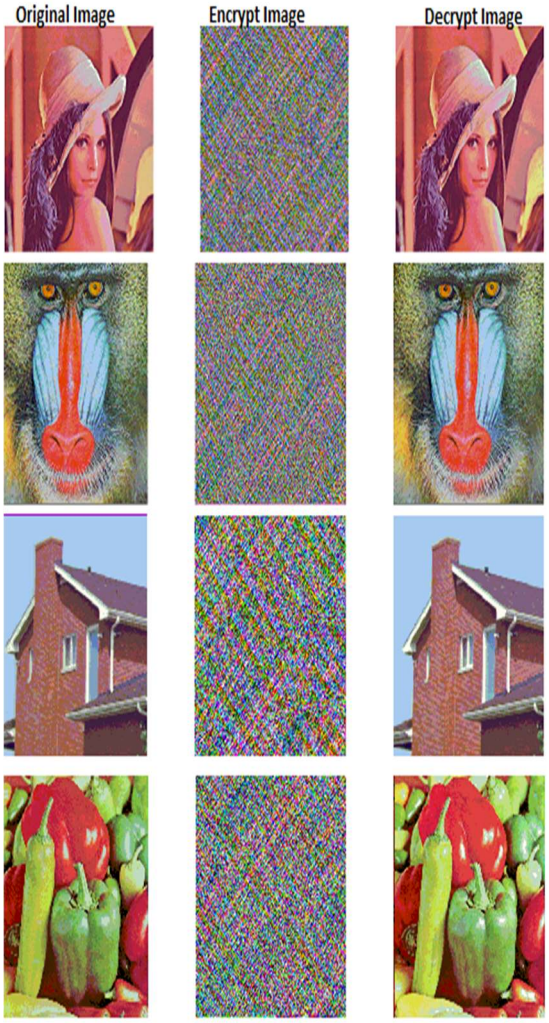

Figure 6: Results obtained.

An interpretation of Figure 7, where the histograms of the original images, encrypted and decrypted respectively, shows that the histograms of the encrypted image are separated randomly and evenly distributed in all gray levels. The flat histograms observed are taken from the encrypted images.

To evaluate these encryption algorithms, we must study their efficiency which is an important task, we must analyze the image entropy, the mean square error (MSE), the peak signal to noise ratio (PSNR) and the statistical and differential analysis. Such an evaluation of these encryption algorithms proposed in this work was carried in an article entitled: " Image encryption algorithm based on elliptic curves "

Journal of Theoretical and Applied Information Technology
15th August 2022. Vol.100. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

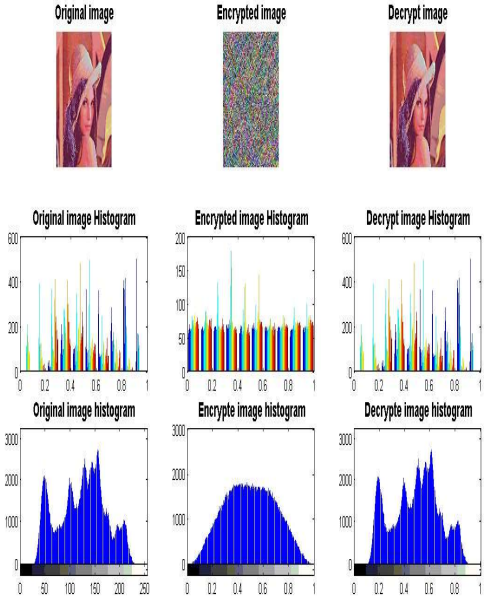

Figure 7: Input, Output and Encrypted Image Histograms.

5.2 Summary of device usage

The resource requirement of the encryption and decryption design is given in detail in Figure 8, for input / output interfaces and synchronization, it should be noted that in practice additional blocks are required.

Device Utilization Summary			
Slice Logic Utilization	Used	Available	Utilization
Number of Slice LUTs	48	69,120	1%
Number used as logic	48	69,120	1%
Number using O6 output only	48		
Number of occupied Slices	28	17,280	1%
Number of LUT Flip Flop pairs used	48		
Number with an unused Flip Flop	48	48	100%
Number with an unused LUT	0	48	0%
Number of fully used LUT-FF pairs	0	48	0%
Number of slice register sites lost to control set restrictions	0	69,120	0%
Number of bonded IOBs	144	640	22%

Figure 8: Summary of device usage (Estimated values)

5.3 RTL schematic

The top-level RTL scheme for the encryption and decryption algorithm implemented on FPGA is shown in Figure 8. This is a schematic representation of the design presented at the register transfer level (RTL). These system blocks are designed for the Virtex5 xc5vlx110t-3ff1136 card.

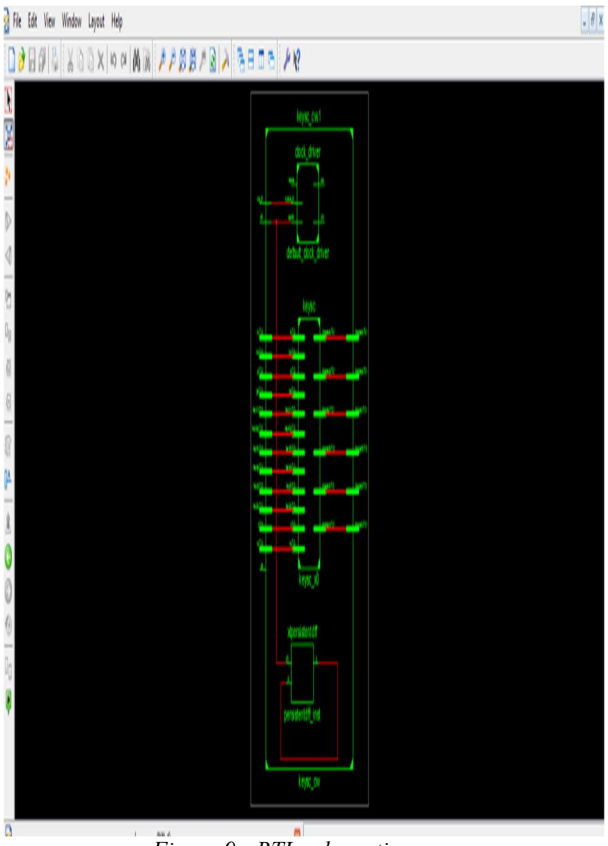

Figure 9: RTL schematic.

6. CONCLSION

By software and hardware co-simulation using Matlab Simulink System Generator and Xilinix, an ECC image encryption technique was proposed.

This study by the Xilinx system generator which provided means for the hardware implementation of these encryption and decryption algorithms with minimum resources and minimum delay.

Thus, we have implemented the important tools generating systems Matlab, Simulink and Xilinx in the field of cryptography, because the security of embedded systems evolves rapidly and its encryption systems allow easy hardware implementation.

The effectiveness of this type of encryption was investigated in another review article.

Open question: Giving a cryptography method for images that is efficient, fast and better than existing methods is still a hot topic. Implement these methods in real time and study the attacks on these encryption methods.

REFERENCES:

[1] Diffie, W., Hellman, M. , "New directions in cryptography", *IEEE Transactions on Information Theory, 1976*.

[2] Chillali, S., Oughdir, L., "A diagram of confidentiality of information during a traffic offence", *AIP Conference Proceedings 2074, 020028, 2019.*

[3] Chillali, S., Oughdir, L., "ECC Image Encryption Using Matlab Simulink Blockset", *ICDTA2021, Lecture Notes in Networks and Systems, vol 211. Springer, Cham, 2021.*

[4] Zeriouh, M., Chillali, A., Boua, A., "Cryptography Based on the Matrices", *Bol. Soc. Paran. Mat. 37(3), pp.75–83, 2019.*

[5] Boulbot, A., Chillali, A., Mouhib, A.:, "Elliptic curves over the ring R", *Bol. Soc. Paran. Mat. 38(3), pp.193-201, 2020.*

[6] Hua, Z.Y., Zhou, Y.C., Pun, C.M., Chen, C.L.P:, "2D sine logistic modulation map for image encryption", *Inf. Sci., 297, 80–94, 2015.*

[7] Zhang, Y:, "The unified image encryption algorithm based on chaos and cubic", *S-Box. Inf. Sci., 450, 361–377, 2018.*

[8] L. Zhang, F. Zhang:, "A New Certificate less Aggregate Signature Scheme", *Comput Commun. Vol 32 (6),1079 –1085, 2009.*

[9] H. Xiong, Z. Guan, Z. Chen, F Li:, "An Efficient Certificate less Aggregate Signature with Constant Pairing Computation. Inform", *Sci. vol 219, pp 225–235, 2014.*