ON THE DESIGN OF PROJECTIVE BINARY EDWARDS ELLIPTIC CURVES OVER GF (P) BENEFITING FROM MAPPING ELLIPTIC CURVES COMPUTATIONS TO VARIABLE DEGREE OF PARALLEL DESIGN

[1]Mohammad Alkhatib [1]Azmi Jaafar [2]Zuriati Zukarnain [3]Mohammad Rushdan MD. SAID

[1]Department of Information System, Faculty of Computer Science, University Putra Malaysia, 43400 UPM, Serdang, Selangor D.E, Malaysia.
[2]Department of Communication Technology and Networks, Faculty of Computer Science, University Putra Malaysia, 43400 UPM, Serdang, Selangor D.E, Malaysia.
[3]Department of Mathematics, Faculty of Science, University Putra Malaysia, 43400 UPM, Serdang, Selangor D.E, Malaysia.

ABSTRACT— Finding multiplicative inverse (Modular Inversion) operation is the most time-consuming operation in Elliptic Curve Crypto-system (ECC) operations which affects the performance of ECC. Moreover, several factors that affect the design of ECC have not been intensively investigated in the majority of researches related to ECC, Such as system utilization, area, resources-consuming and area*time cost factors, which play significant role in designing efficient ECC for different applications. This work applies Binary Edwards ECC point doubling operation over GF(p) using projective coordinates instead of affine coordinates due to its ability to remove the long time inversion operation by converting it to a number of multiplication operations. We also utilize the inherent parallelism in ECC operations by mapping its computations to parallel hardware design, in order to improve the performance of ECC. Our results show that the shortest time delay is achieved using 7-Parallel Multipliers (PM) design with projection (X/Z, Y/Z), which overcomes both serial design and the design with affine coordinates.

Furthermore, this research proposes a variety of design choices by varying the degree of parallelism to tune-up several factors that affect ECC in order to investigate possible enhancements. It is shown by our experiments that the hardware utilization can be improved by 55%, with less area, and acceptable time-consuming level compared to other designs in the same projection. In other words, we compromise the performance to enhance system utilization degree, and AT cost, and to reduce area and resource-consuming. This trade-off between factors is useful to determine the efficient design to be used for different ECC applications based on their requirements and available resources. Especially, when the time-consuming is not the main priority.

Keywords — Elliptic Curves Crypto-system, Point Doubling Operation, Projective Coordinates systems, Parallel Design ,Hardware Utilization .Time-consuming, area.

1. INTRODUCTION

Elliptic Curves (EC) were introduced in cryptography in 1985 by Koblitz and Miller [1, 2]. The security level of Elliptic Curves Crypto-system (ECC) depends on discrete logarithm problem for elliptic curves. It is worth to mention that there is no known efficient attack on the discrete logarithm problem for EC [3]. ECC is a public key algorithm and uses two kinds of keys: Public key; which is known for all parties in communication network, and private key; which is known for intended recipient. It is mathematically infeasible to compute the private key from the public key [1-3, and 5]. ECC is considered a serious alternative of the well-known RSA algorithm, since it provides a comparable security level to that achieved using RSA algorithm with much smaller key size, which offers a considerable saving in area and resources.

Cryptographic computations need to be fast and accurate. Therefore, ECC uses finite fields arithmetic to perform its operations. Two main finite fields were introduced in the literature: Prime field GF(p), and Binary field GF(2^n) [4-7].

In EC Cryptography, the plain text is mapped into a point on an Elliptic Curve, and then ECC performs it operations on that point to yield new point which represents the ciphertext. There are two main operations in ECC [1-5]:

1. Point Addition; which is adding to different points on the elliptic curve.
2. Point Doubling; which is adding point to itself. Assume J is a point on an elliptic curve such that J= (X_1, Y_1), where $Y_1 \neq o$. Let L be another point and L=2J, where L= (X_3, Y_3), Then X_3= M^2-2X_1, and Y_3= M(X_1-X_3)-Y_1, where: M= (3X_1^2+a)/(2Y_1), M is the tangent at point J and is one of elliptic curves parameters.

Point addition and point doubling operations over GF(p) depend on modular arithmetic operations to perform its computations. These modular operations are: modular addition, modular subtraction, modular multiplication, and the known modular inversion operation. Modular inversion or finding multiplicative inverse is costly operation in terms of time-consuming. Researchers found that the time consumed by each inversion is equivalent to that consumed by 3-4 sequential multiplication steps, where each sequential multiplication is equivalent to 3 sequential addition operations [5, 8-11].

ECC operations are usually applied using affine coordinates (X, Y) [5, 12, and 13]. ECC that uses the affine form suffers the long time inversion operation problem, which affects the performance of ECC. Researchers investigated several methods to address inversion operation. Some of them introduced methods and algorithms to reduce the time-consuming for modular inversion operation, such as: the Extended Binary GCD algorithm, and modular multiplication which were based on the Montgomery's method [14, and 15]. Tawalbeh [16, and 17] introduced a unified algorithm to compute inversion for both prime and binary fields. He used counters instead to keep track of the difference between field elements. However, his method was costly in terms of time and resources. On the other hand, many researches proposed to represent EC points by using different projective coordinates forms instead of the usual affine form. Researchers found that using projective coordinates eliminates inversion operation by converting it to several multiplication operations. Thus, ECC can perform its computations with no inversion operation using projective coordinates form, which contributes to enhance the performance of ECC algorithm [18-21]. Three main projective coordinates were found in the literature:

- Homogeneous (also called standard) projective coordinates system (X/Z, Y/Z).
- López-Dahab projective coordinates system (X/Z, Y/Z^2).
- Jacobean projective coordinates system (X/Z^2, Y/Z^3) [21-24].

Some researchers proposed to apply ECC using serial hardware design, which is considered a time-consuming solution [14, and 25]. On the other side, many researchers used parallel hardware units such as, Multipliers (M) and Adders (A) to perform ECC operations. Using Parallel hardware design achieved great enhances on the performance of ECC, since parallel design utilizes the inherent parallelism in ECC computation in order to perform these computations (Multiplications & Additions) in parallel based on the inherent parallelism in ECC calculations[10, 11, 18-20, and 25-30]. In [30], authors applied Hessian ECC operations over GF(p) using projective coordinates. They used parallel hardware design to achieve maximum gain in terms of time saving. Al-khatib and Abu Alhija in [29] proposed several design choices for standard ECC over GF(p) using projective coordinates. While in [11] they applied Binary ECC over GF(p) using projective coordinates to eliminate inversion. Their designs showed much better performance compared to designs that use normal affine coordinates. In this research, we investigate applying Binary Edwards ECC point doubling operation over GF(p) using the three well-known projective coordinates systems. In order to obtain the best performance (time-consuming results, we use parallel design to map ECC computation, which will be implemented in parallel. Furthermore, we introduce and investigate almost all design choices for Binary Edwards ECC over GF(p) to provide a trade-off between several factors that affect ECC design. In other words, we try to tune-up these factors, which might help in designing the efficient (most suitable) ECC for specific EC applications based on the availability of resources and requirements. For example, Although the 7-PM design can achieve the shortest time-consuming results, it still suffers low hardware utilization degree, resources-consuming, and needs extra area. Our presented designs overcome the 7-PM design [11] in terms of hardware utilization, AT cost, and need less area, with acceptable time-consuming level. A number of factors that affect the design of ECC are studied and used in this research[25-30].

2. ECC Algorithms & Architectures

This section propose the hardware algorithms and crypto-architectures for ECC crypto-processor that emanated from using new different coordinate systems with different projection systems to show their benefits when computed using parallel multipliers [39].

2.1 Homogeneous Projective Coordinates(X/Z, Y/Z)

In Homogeneous projection, the usual affine coordinates (X, Y) is replaced by the projective coordinates (X/Z, Y/Z), and then we use point doubling equations in order to compute M, X_3, $Y_{3,}$ and Z_3 as following:
First, we calculate the slope M:

$$M = \frac{\left(\frac{Y}{Z}+\frac{Y^2}{Z^2}\right)\left(1+2\frac{X}{Z}\right)-2b\frac{X}{Z}}{a+2b\frac{Y}{Z}-\left(\frac{X}{Z}+\frac{X^2}{Z^2}\right)\left(1+2\frac{Y}{Z}\right)} \Longrightarrow \frac{(ZY+Y^2)(Z+2X)-2bXZ^2}{Z^2(aZ+bY)-(XZ+X^2)(Z+2Y)} \qquad \frac{A_2*A_1-2bXZ^2}{Z^2*A_3-A_4A_5} = \frac{A_6}{A_7}$$

To simplify the computations, we use the following symbols:
A_1=ZY+Y^2, A_2=Z+2X, A_3=aZ+bY, A_4=XZ+X^2, A_5=Z+2Y, A_6=A_1*A_2-2bXZ^2, A_7=Z^2*A_3-A_4*A_5

Then, we use point doubling equations for ECC over GF(p) to compute X_3, Y_3, and Z_3:

$$X_3 = A_7\left[Z*A_6^{\,2} - 2X*A_7^{\,2}\right],$$
$$Y_3 = A_6\left[3X*A_7^{\,2} - Z*A_6^{\,2}\right] - Y*A_7^{\,3} ,$$
$$Z_3 = Z*A_7^{\,3}$$

Note we use the symbols M, and A to represent the multiplication, and addition operations respectively in our designs.

2.1.1 Using 7 Parallel Multipliers:

In this design, we use 7 parallel multipliers (PM), and 5 parallel adders (PA) to perform point doubling operation for Binary Edwards ECC over GF(p).
It is notable that this design overcomes the other designs in terms of time-consuming results for the three projective coordinates systems used in our research. it takes 4 sequential multiplication (SM), and 3 sequential addition (SA) steps. However, this design consumes more resources and needs more area. Furthermore, 6 multipliers remain Idle (unused) when using the 7-PM design, which affects the degree the system will be utilized. As a consequence, this design obtains low hardware utilization results. Figure 1 presents the 7-PM design for Binary Edwards ECC over GF(p) using projection (X/Z, Y/Z). .

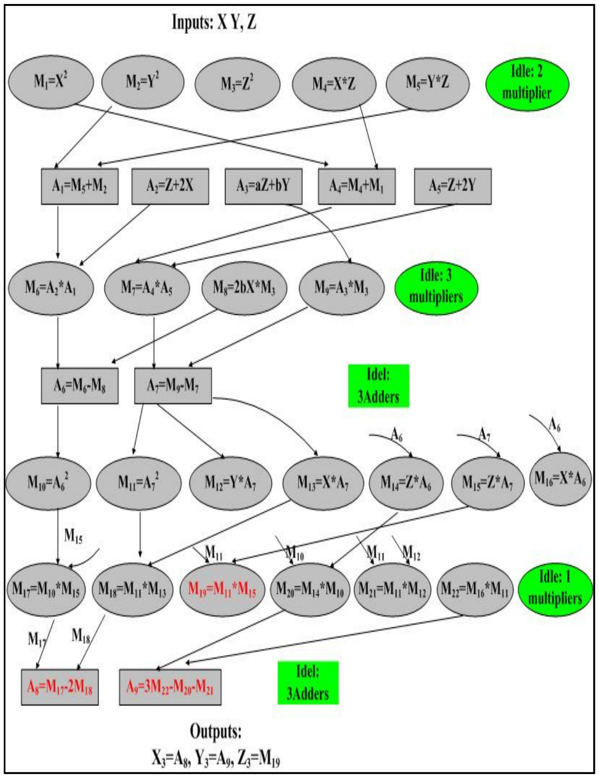

Figure 1: The 7-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/Z).

2.1.2 Using 6 Parallel Multipliers:

Figure 2, it shows the 6-PM design using standard projective coordinates. In this design, we reduce the area to 6-PM and 5-PA, which affects the performance (time-consuming) for ECC. The 6-PM design takes 5 SM and 3 SA steps to perform point doubling operation over GF(p). It consumes more time and needs less area compared to the 7-PM design.

In other words, we compromise the performance in favor of area and resources-consuming.

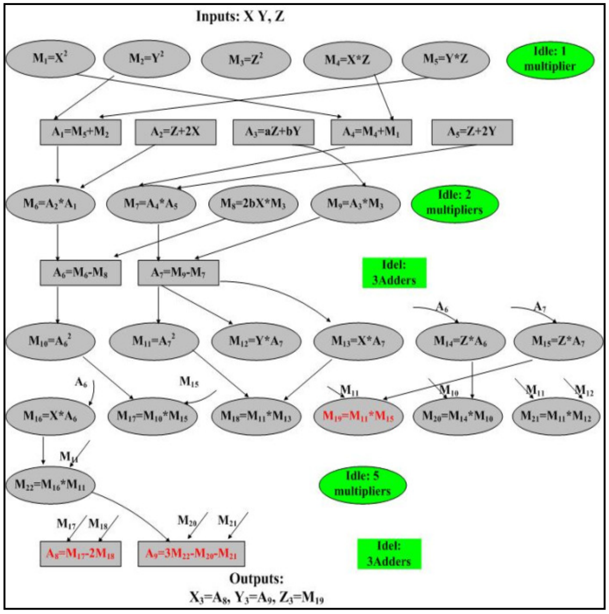

Figure 2: The 6-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/Z).

2.1.3 Using 5 Parallel Multipliers:

Figure 3 presents the 5-PM design for Binary Edwards ECC over GF (p) using projection (X/Z, Y/Z). Although this design uses less area than the 6-PM design, it shows comparable performance results. Furthermore, it enhances the hardware utilization for the ECC

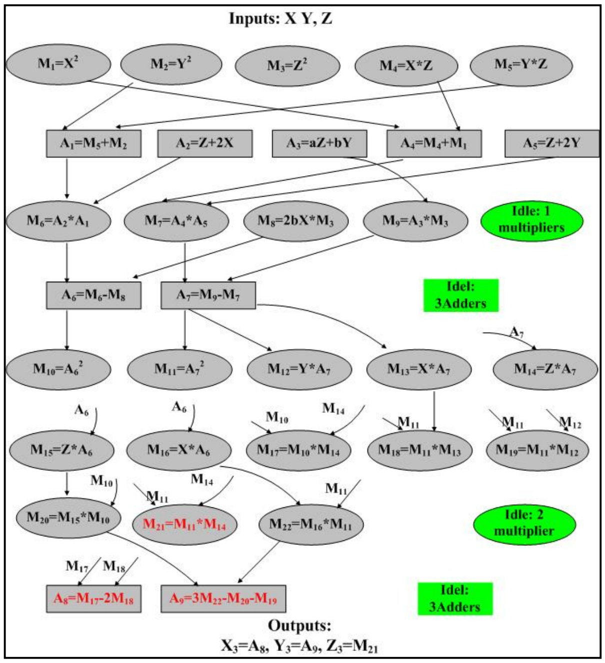

Figure 3: The 5-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/Z).

2.1.4 Using 4 Parallel Multipliers:

In this design, we use 4 PM and 2 PA to design the ECC. It consumes 6 SM and 4 SA time units.

Note that there are 2 Idle multipliers in the step number 6 of sequential multiplication, and 7 Idle adders in steps 2, 3, and 4. The 4-PM design obtains better hardware utilization results than previous designs in this projection. The 4-PM design is presented in Figure 4

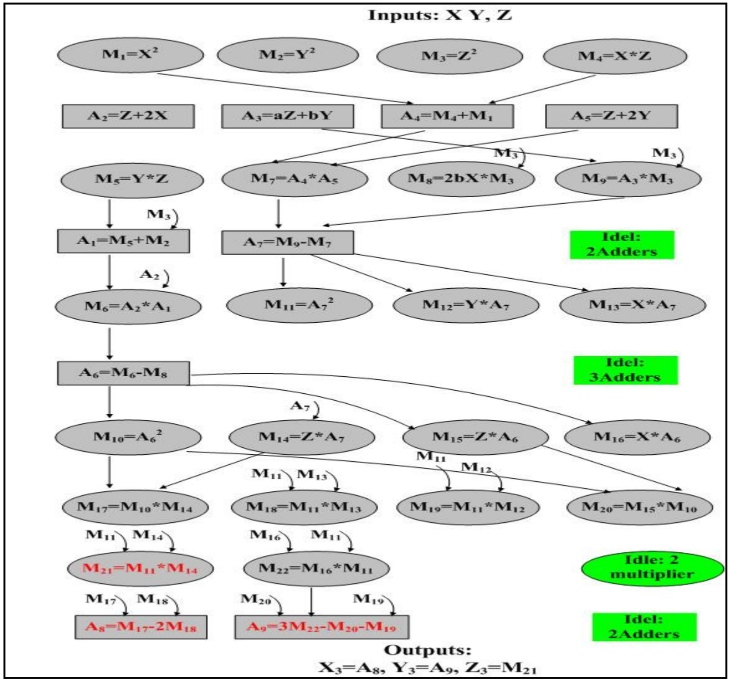

Figure 4: The 4-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/Z).

2.1.5 Using 3 Parallel Multipliers:

This design takes longer time than the 4-PM design. It consumes 8 SM and 4 SA steps. However, the 3-PM design needs less area and resources and obtains comparable hardware utilization results in comparison to 4-PM design. Note that we use 3 PM and 2 PA in this design. Figure 5 shows the dataflow for the 3-PM design.

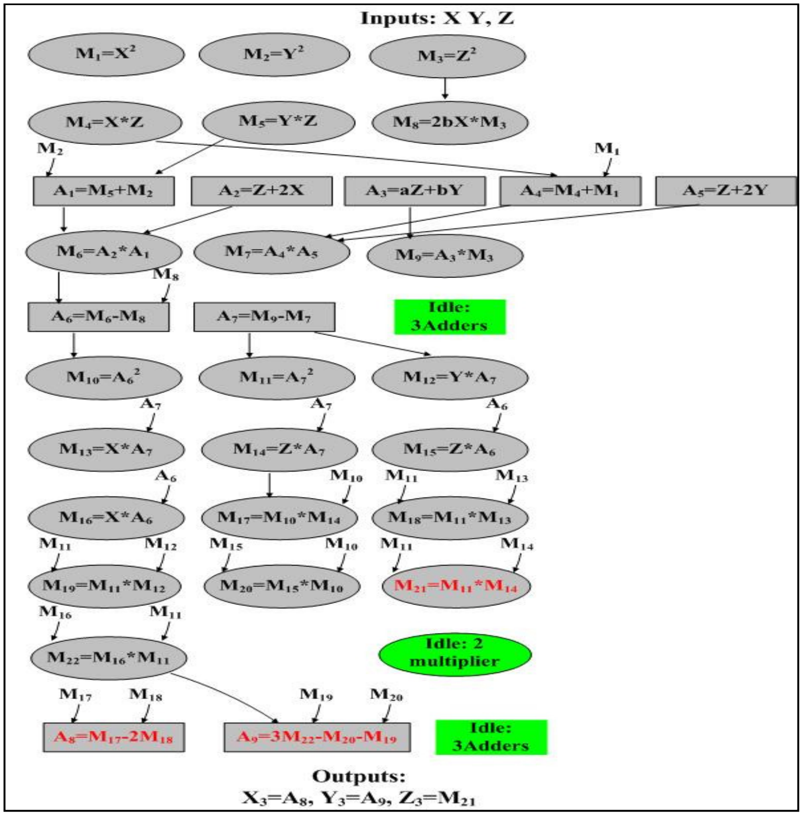

Figure 5: The 5-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/Z).

2.1.6 Using 2 Parallel Multipliers:

This design uses 2 PM and 2 PA to perform point doubling operation. The 2-PM design overcomes the serial design in terms of time-consuming with comparable hardware utilization results. Moreover, it achieves the best hardware utilization ratio with less area in comparison with the previous designs in this projection. However, this design needs more area than the serial design, and takes longer time than the other designs in standard projective coordinates system. Dataflow for the 2-PM design of ECC presented in figure 6 shows that computations in each sequential operation step commence after the completion of the previous step. For example, ECC cannot perform the computations in sequential step number 10, which includes M_{10} and M_{11}, unless results from addition operation A_6 in previous sequential operation have been received. It is also notable that arithmetic operations (multiplications, and additions) in each sequential operation step are performed in parallel. Thus, time-consuming is measured by the number of sequential operations whereas, the time-consuming for one sequential multiplication operation is equivalent to three sequential addition operations.

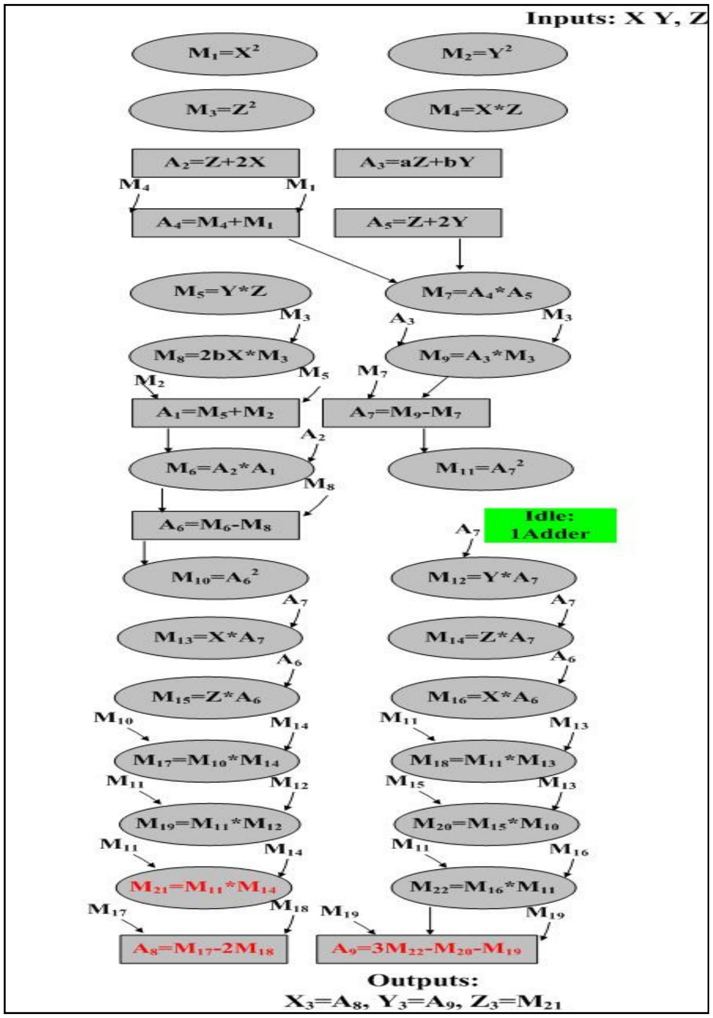

Figure 6: The 2-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/ Z).

2.2 L´opez-Dahab Projective Coordinates(X/Z, Y/Z^2)

In this projection, the usual affine coordinates (X, Y) is are represented by the projective coordinates (X/Z, Y/Z^2). We calculate the slop M first:

$$M = \frac{\left(\frac{Y}{Z^2}+\frac{Y^2}{Z^4}\right)\left(1+2\frac{X}{Z}\right)-2b\frac{X}{Z}}{a+2b\frac{Y}{Z^2}-\left(\frac{X}{Z}+\frac{X^2}{Z^2}\right)\left(1+2\frac{Y}{Z^2}\right)} \Longrightarrow$$

$$\frac{(Y*Z^2+Y^2)*(Z+2X)-2bXZ^4}{Z^3*(aZ^2+2bX)-Z*(X*Z+X^2)*(Z^2+2Y)} \Longrightarrow$$

$$\frac{A_1*A_2-2bX*Z^4}{Z^3*A_3-Z*A_4*A_5} = \frac{A_6}{A_7}$$, Where:

A_1= Y*Z^2+Y^2, A_2= Z+2X, A_3= aZ^2+2bX, A_4= X*Z+X^2, A_5= Z^2+2X, A_6= A_1*A_2-2bX*Z^4, A_7= Z^3*A_3-Z*A_4*A_5

and then we use point doubling equations over GF(p) in order to find X_3, Y_3, and Z_3 as following:
X_3= Z*A_6^2-2X*A_7^2
Y_3= Z*A_6*A_7*[3X*A_7^2-Z*A_6^2]-Y*A_7^4
Z_3= Z*A_7^2

In the following designs, we apply this projection using variable number of parallel hardware units to provide a trade-off between several factors that affect the design of ECC, such as: Speed, Hardware Utilization, Area, and Security.

2.2.1 Using 5 Parallel Multipliers:

In this design, we use 5 PM and 4 PA. Based on the inherent parallelism in ECC computations, we find that using more than 5 PM will result in at least 1 Idle multiplier in all sequential multiplications, which leads to consume more recourses and additional area. The 5-PM design gives the best performance results among the other designs that use L´opez-Dahab projective coordinates. However, this design shows relatively low hardware utilization results. Note that there are 7 Idle multipliers and 10 Idle adders in the 5-PM design. This design is shown in figure 7.

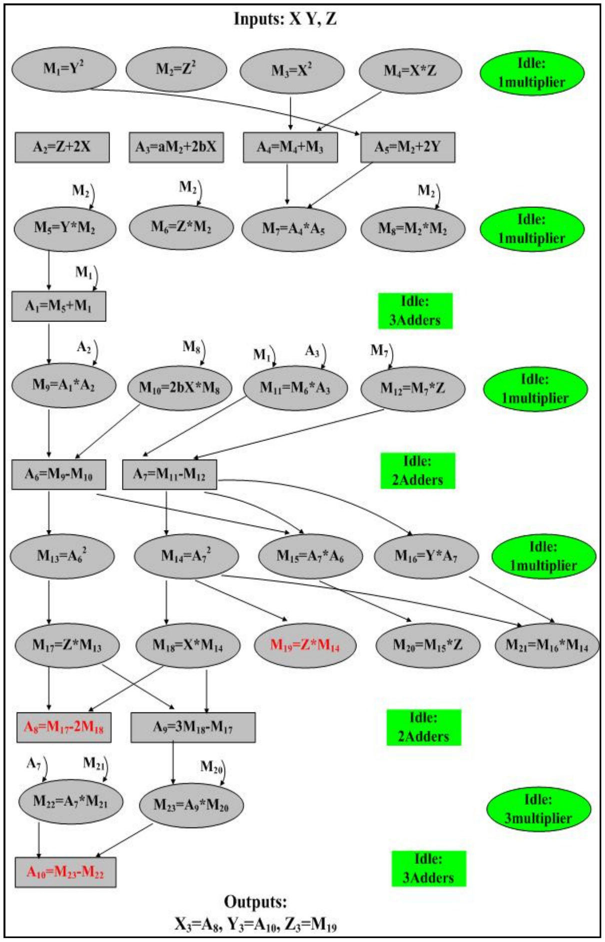

Figure 7: The 5-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/ Z^2).

2.2.2 Using 4 Parallel Multipliers:

Figure 8 presents the 4-PM design for ECC point doubling operation over FG(p). It is worth to mention that this design overcomes the 5-PM design in terms of hardware utilization ratio, and uses less area. Moreover, it achieves comparable performance level, which makes this design a preferable choice when designing Binary Edwards ECC over GF(p) using projection (X/Z, Y/Z^2). Note that the 5-PM design was introduced by our previous research work in [11], which suffers low hardware utilization results.

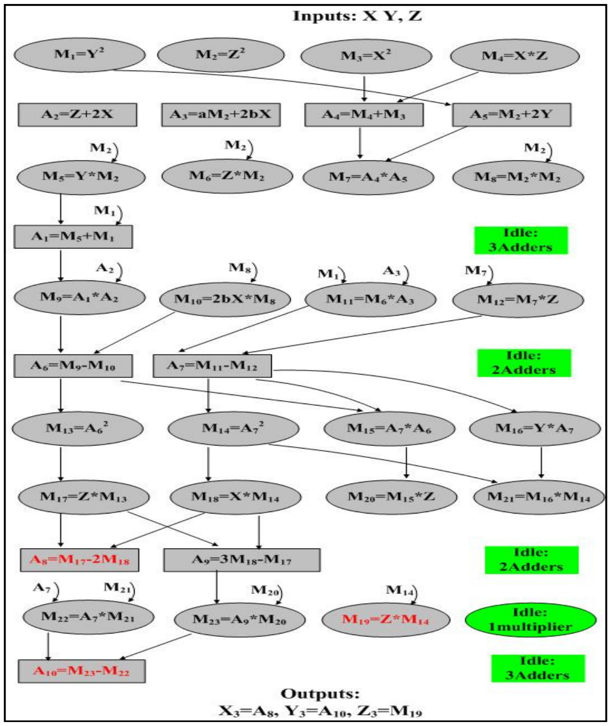

Figure 8: The 4-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/ Z^2).

2.2.3 Using 3 Parallel Multipliers:

In this design, we assume that the availability of resources is limited to 3 PM and 3 PA. Although the 3-PM design needs less area and resources with similar system utilization results compared to the 4-PM design, it consumes more time units. The 3-PM design is shown in figure 9.

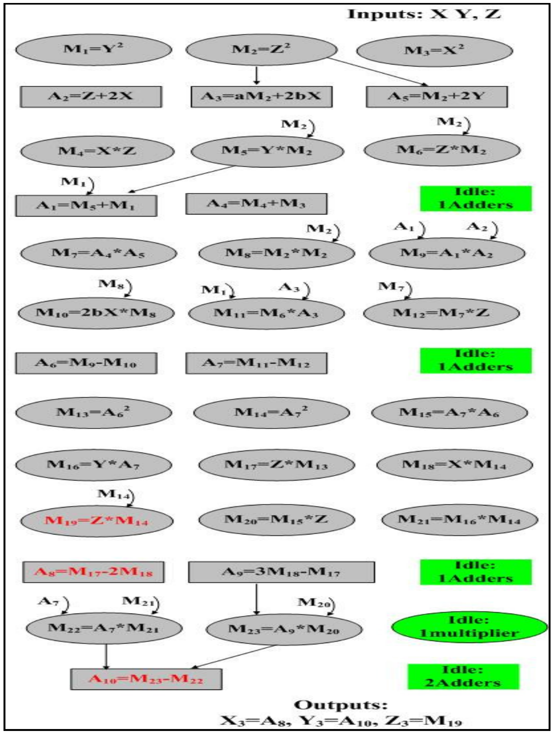

Figure 9: The 3-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/ Z^2).

2.2.4 Using 2 Parallel Multipliers:

Figure 10 shows the dataflow for Binary Edwards ECC design. We reduce the area and resources-consuming, which compromises the performance of the system. The 2-PM provides a notable trade-off in terms of time-consuming, area, hardware utilization, security, and resources-consuming when compared the other designs including the serial design in this projection.

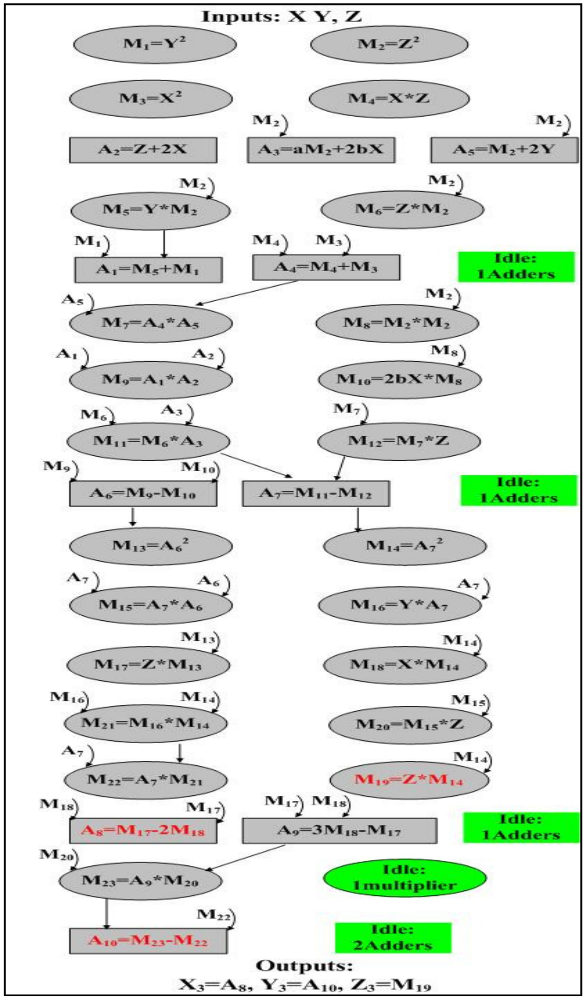

Figure 10: The 2-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z, Y/ Z^2).

2.3 Jacobean Projective Coordinates (X/Z^2, Y/Z^3)

In this projection, we represent the affine coordinates (X, Y) by the projective form (X/Z^2, Y/Z^3). We start by calculating the slope M and then use point doubling equations over GF(p) in order to compute X_3, Y_3, and Z_3 as following:

$$M = \frac{\left(\frac{Y}{Z^3}+\frac{Y^2}{Z^6}\right)*\left(1+2\frac{X}{Z^2}\right)-2b\frac{X}{Z^2}}{a+2b\frac{Y}{Z^3}-\left(\frac{X}{Z^2}+\frac{X^2}{Z^4}\right)*\left(1+2\frac{Y}{Z^3}\right)} \Longrightarrow$$

$$\frac{(Y * Z^3 + Y^2) * (Z^2 + 2X) - 2bX * Z^6}{Z^5 * (aZ^3 + 2bY) - Z * (X * Z^2 + X^2) * (Z^3 + 2Y)}$$

$$\Longrightarrow \frac{A_1*A_2-2bX*Z^6}{Z^5*A_3-Z*A_4*A_5} = \frac{A_6}{A_7}$$, Where:

A_1= Y*Z^3+Y^2, A_2= Z^2+2X, A_3= aZ^3+2bY, A_4= X*Z^2+X^2, A_5= Z^3+2Y, A_6= A_1*A_2-2bX*Z^6, A_7= Z^5*A_3-Z*A_4*A_5

We find X_3, Y_3, and Z_3 by substituting in point doubling equations:

X_3= Z^2*A_6^2-2X*A_7^2,

$Y_3= Z*A_6*[3X*A_7^2-Z^2*A_6^2]-Y*A_7^3$, and
$Z_3= Z*A_7$

In the following, we discuss several design solutions for Binary Edwards ECC using projection (X/Z^2, Y/Z^3), we also use a variable degree of parallelism to tune-up different factors that affect ECC.

2.3.1 Using 5 Parallel Multipliers:

Figure 11 presents the 5-PM design of Binary Edwards ECC over GF (p) using projection (X/Z^2, Y/Z^3). Our experiments emphasized that adding more than 5 PM only consumes extra area and has no benefit in terms of performance and hardware utilization due to the inherent parallelism in ECC point doubling calculations. This designs performs doubling operation within 7 SM and 6 SA time units which is considered the best time-consuming results in this projection. However the 5-PM design suffers low system utilization level.

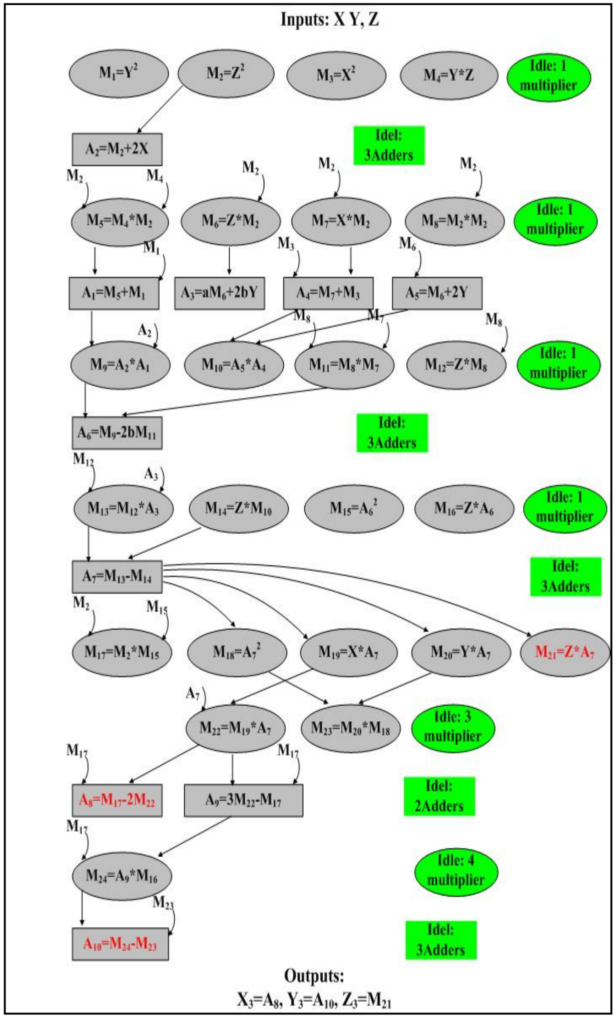

Figure 11: The 5-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z^2,Y/ Z^3).

2.3.2 Using 4 Parallel Multipliers:

Figure 12 shows the 4-PM design. Note that it uses less area with comparable time-consuming level compared to the previous design. Furthermore, this design enhances the system utilization considerably. This makes this design a serious alternative for the 5-PM when designing Binary ECC over GF(p) using Jacobean projective coordinates systems (X/Z^2, Y/Z^3). This design achieves 75% utilization ratio, which overcomes results obtained by the 5-PM design.

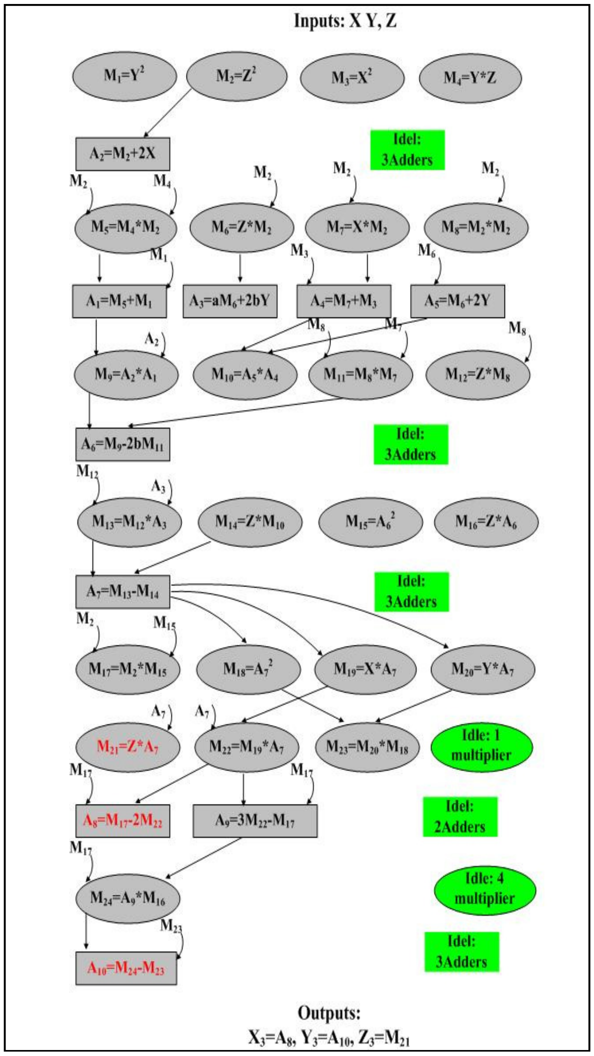

Figure 12: The 4-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z^2,$Y/\ Z^3$).

2.3.3 Using 3 Parallel Multipliers:

In this design, we use 3 PM and 2 PA to map ECC point doubling computations. Note that each sequential computation step cannot commence unless the previous step has been completed due to data dependency of computational mapping for ECC design. The 3-PM design achieves 97% hardware utilization with less area, which makes this design a preferable trade-off between area and performance. Figure 13 shows dataflow of the 3-PM design for Edwards ECC over GF(p). Not the data can immigrate to the next sequential level after completing the computation of the current multiplication/ addition level due to data dependency and the inherent parallelism in ECC point doubling computation.

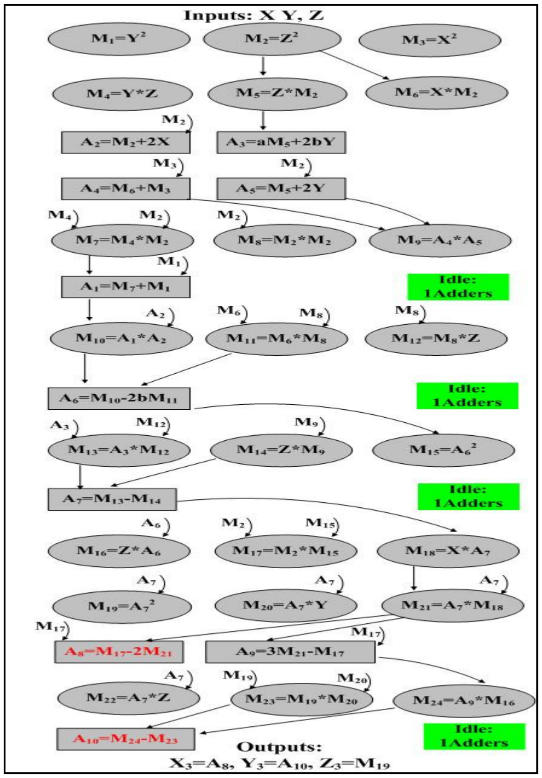

Figure 13: The 3-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z^2,Y/ Z^3).

2.3.4 Using 2 Parallel Multipliers:

Figure 14 shows the 2-PM design. In comparison with the 3-PM design, we find that both designs obtains the same system utilization results approximately. Furthermore, the 2-PM design has the priority in terms of area and resources-saving. However, this design still needs extra 4 SM time units compared to the 3-PM. It is notable that the 2-PM design overcomes the serial design in terms of time-consuming with comparable hardware utilization results.

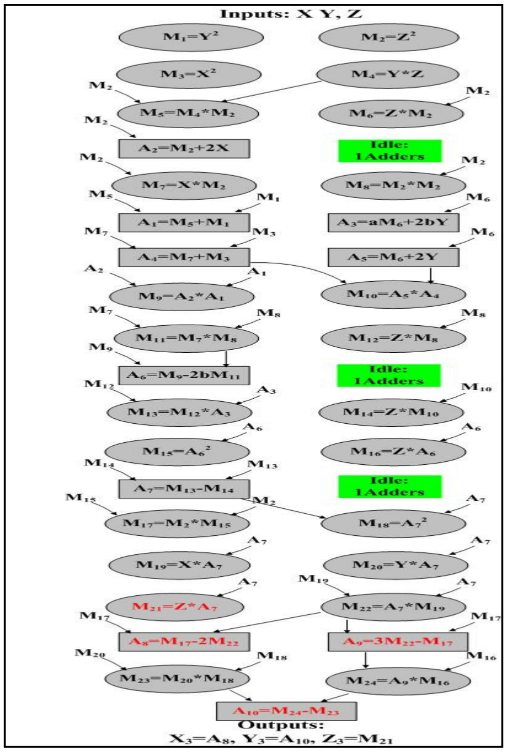

Figure 14: The 2-PM design for Binary Edwards ECC for point doubling operation over GF (P) using Projection (X/Z^2,Y/ Z^3).

3. RESULTS & DISCUSSION

This section presents comparisons between all designs in this research. A number of factors that affect the design and implementation of ECC has been considered in these comparisons as following [5, 10, 11, 26-30]:

1. Parallel Units(PU): Which represents the number of parallel hardware units used in the design; this factor plays significant role the design implementation, since it determines the required resources and area for implementation.
2. Sequential Operations(SO): Which determines the performance (time-consuming) for the design; where the time units are measured by calculating the number of sequential multiplication (SM) and addition (SA) steps.
3. Hardware Utilization(HU): To determine the degree the system will be utilized.
4. Idle Units(IU): To compute the number of hardware components that remains idle when using the design.
5. Area*Time^2 (AT^2)and Area*Time(AT): These two factors can be used as a measure of cost (also called cost factor). AT^2 is used When the priority is given to the time-consuming in the design implementation. On the other hand, if the main concern is the area and required resources, AT will be a preferable measure.
6. Parallelization Enhancement(PE): The ratio of enhancements in parallel computations obtained using parallel design to the serial design.

Tables 1, 2, and 3 showed summary of results for Binary Edwards ECC designs over GF(p) using Homogeneous , L´opez-Dahab, and Jacobean projective coordinates respectively. Our results show that the 7-PM design shown in table 1 achieves the best time-consuming results compared to other designs throughout projections used in this research, it needs 4 SM and 3 SA (time units) to perform point doubling, which is considered the shortest critical path delay. Furthermore, this design enhances AT^2(cost factor) to reach 112. Therefor, the 7-PM design is a considerable choice when the system performance has the main priority in ECC design. The Other designs: 6-PM-serial design, compromise the performance to reduce the area, and resources-consuming, and to enhance system utilization for ECC. For example: Hardware Utilization can be enhanced by 100% approximately using 2-PM design. On the other hand, this affects the performance and AT^2 cost; time-consuming is increased to 11 SM, and 5 SA. However, since the aim of the 2-PM design is to save area and resources, we focus on the AT cost factor which obtains the best results using 2-PM design and the serial design. Moreover, it is notable from table 1 that 2-PM design achieves 50% enhancements in terms of system performance compared to the serial design. This provides a considerable trade-off between performance, area, resources-consuming, hardware utilization, and cost.

Results in table 1 show that 5-PM design is preferable choice for 5 SM performance level, since it receives the same time-consuming results with less area and much better utilization degree.

Design	PU	SO	IU	HU	PE	AT^2	AT
7-PM Design[2]	7M, 5A	4 SM, 3 SA	6M, 6A	76%	550%	112	28
6-PM Design	6M, 5A	5 SM, 3 SA	8M, 6A	71|%	440%	150	30
5-PM Design	5M, 5A	5 SM, 3 SA	3M, 6A	83%	440%	125	25
4-PM Design	4M, 4A	6 SM, 4 SA	2M, 7A	86%	367%	144	24
3-PM Design	3M, 5A	8 SM, 3 SA	2M, 6A	86%	275%	192	24
2-PM Design	2M, 2A	11 SM, 5 SA	1A	100%	200%	242	22
Serial Design[1]	1M, 1A	22 SM, 9 SA	-	100%	-	484	22

Table 1: Comparison between different designs in projection (X/Z, Y/Z).

Designs that use L´opez-Dahab projective coordinates are shown in Table 2. Note that the shortest time delay for this projection is 6 SM and 5 SA, which can be achieved using 5-PM and 4-PM designs. Although both designs obtains the same performance level, the 4-PM designs shows much better system utilization degree, and with less area. Furthermore, both cost factors AT&AT^2 were improved using 4-PM design.

As expected, results obtains from the other designs in this projection show a trade-off between several factors. While 2-PM, and 3-PM overcome the serial design in respect to time-consuming and AT^2 cost, hardware utilization degree, and AT cost were better in the serial design. This trade-off provides a variety of design choices which might be useful in different EC applications based on the requirements and required resources.

Design	PU	SO	IU	HU	PE	AT^2	AT
5-PM Design[11]	5M, 4A	6 SM, 5 SA	7M, 10A	73%	383%	180	30
4-PM Design	4M, 4A	6 SM, 5 SA	1M, 10A	87%	383%	144	24
3-PM Design	3M, 3A	8 SM, 5 SA	1M, 5A	90%	288%	192	24
2-PM Design	2M, 3A	12 SM, 5 SA	1M, 5A	90%	192%	288	24
Serial Design[14]	1M, 1A	23 SM, 10 SA	-	100%	-	529	23

Table 2: Comparison between different designs in projection (X/Z, Y/Z^2).

Table 3 presents results of designs using Jacobean projective coordinates system. The 4-PM design overcomes the 5-PM, and the other designs in this projection in terms of hardware utilization, and performance respectively. It is worth mentioning, that 3-PM design obtains much better hardware utilization (which means the design will be saturated in this point), AT, and AT^2 results with less area. However, it consumes extra 1 SM, and 1 SA steps in comparison with 4-PM and 5-PM designs. The best parallelization enhancements results are accomplished using 4-PM design.

Design	PU	SO	IU	HU	PE	AT^2	AT
5-PM Design[11]	5M, 4A	7 SM, 6 SA	11M, 14A	63%	343%	245	35
4-PM Design	4M, 4A	7 SM, 6 SA	4M, 14A	75%	343%	196	28
3-PM Design	3M, 3A	8 SM, 7 SA	4A	97%	300%	192	24
2-PM Design	2M, 3A	12 SM, 7 SA	4A	97%	200%	288	24
Serial Design[14]	1M, 1A	24 SM, 10 SA	-	100%	-	576	24

Table 3: Comparison between different designs in projection (X/Z^2, Y/Z^3).

4. CONCLUSIONS

Several hardware designs for Binary Edwards Elipti Curve Crypto-system point doubling computations over GF(p) are discussed in this paper. Our designs are introduced and investigated using the three well-known projective coordinates systems: Homogeneous, L´opez-Dahab, and Jacobean coordinates. All our designs perform ECC point doubling operation with no inversion operation due to the ability of projective coordinates to convert inversion to a number of multiplications which enhances the performance of ECC considerably. Furthermore, we provides almost all design choices for Binary Edwards ECC point doubling using the mentioned projections. These designs consider a trade-off among different factors that affect ECC design, such as: time-consuming, hardware utilization, area, resources-consuming, and the cost factors. This will be useful to select the most suitable hardware design to several ECC application based on design requirements and available resources.

It is worth to mention that the 7-PM design using projection (X/Z, Y/Z) achieves the best time consuming and AT^2 results, which means that it will be a preferable design choice, if the performance (speed) was the main priority. On the other hand, the other designs tune-up different factors to introduce the most efficient design according to requirements of ECC application. For example, in some designs, we compromise the speed to enhance system utilization and AT cost, and to reduce area. This will be useful in many ECC applications such as digital signature, and key exchange

5. REFERENCES

[1] N. Koblitz, Elliptic curve cryptosystem, Mathematics of Computation 48 (1987) 203–209.Menezes, A.J., P.C. Van Oorschot, and S.A. Vanstone, "Handbook of Applied Cryptography", CRC Press, Boca Raton, Florida, 1996.

[2] V. Miller, Uses of elliptic curves in cryptography, in: Cryptology Conference (CRYPTO), LNCS 218, 1985, pp. 417–426.

[3] Darrel Hankerson, Alfred Menezes, Scott Vanstone,"Guide to Elliptic Curve Cryptography," Springer-Vl-Rlag New York, Inc., 175 I-'Ifth Avenue, New York, Ny 10010, USA, 2004.

[4] A. Menezes, E. Teske, A. Weng, Weak fields for ECC, in: The Cryptographer's Track at RSA Conference (CT-RSA), LNCS 2964, 2004, pp. 366–386.

[5] Wade Trappe, And Lawrence C. Washington, "Introduction to Cryptography with Coding Theory," By Prentice Hall, 2002, 1: 1-176.

[6] P. Gaudry, F. Hess, N.P. Smart, Constructive and destructive facets of Weil descent on elliptic curves, Journal of Cryptology 15 (2002) 19–46..

[7] IEEE P1363, Standard specifications for public key cryptography, 1999.

[8] Guerric Meurice de Dormale, Jean-Jacques Quisquater, "High-speed hardware implementations of Elliptic Curve Cryptography: A survey" Journal of Systems Architecture 53 (2007) 72-84, by Elsevier

[9] Kimmo Järvinen, and Jorma Skyttä, "On Parallelization of High-Speed Processors for Elliptic Curve Cryptography", IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 16, NO. 9, SEPTEMBER 2008.

[10] Qasem Abu Al-Haija and Mohammad Al-Khatib , "Parallel Hardware Algorithms & Designs for Elliptic Curves Cryptography to Improve Point Operations Computations using new projective coordinates", Journal of Information Assurance and Security (JIAS), By Dynamic Publishers Inc., USA, April 2010, Vol.4, No.1, Paper 6: (588-594).

[11] Mohammad Al-Khatib, Qasem Abu Al-Haija , and Ramlan Mahmud, "Performance Evaluation of Projective Binary Edwards Elliptic Curve Computations with Parallel Architectures," Accepted for publication at Journal of Information Assurance and Security (JIAS), By Dynamic Publishers Inc., USA, July 2010.

[12] Orlando,G., and Paar,C., "A High-Performance Reconfigurable Elliptic Curve Processor for GF(2^m)", Workshop on Cryptographic Hardware and Embedded Systems - CHES 2000, Massachusetts, August 2000.

[13] Stinson, D. R., "Cryptography: Theory and Practice", CRC Press, Boca Raton, Florida, 1995.

[14] Orlando,G., and Paar,C., "A scalable GF(p) elliptic curve processor architecture for programmable hardware," Cryptographic Hardware and Embedded Systems - CHES 2001, Paris, France, May 14-15, 2001.

[15] E. Savas, C. K. Koc ¸"The Montgomery Modular Inverse - Revisited" IEEE- Transactions On Computers, July 2000, 49.

[16] L. Tawalbeh And A. Tenca, "An Algorithm And Hardware Architecture For Integrated Modular Division And Multiplication In GF(P) And GF(2N)," IEEE International Conference On Application-Specific Systems, April, 2004.

[17] L. Tawalbeh, "A Novel Unified Algorithm And Hardware Architecture For Integrated Modular Division And Multiplication In GF (P) And GF (2N) Suitable For Public-Key Cryptography.", Ph.D. Thesis, School Of Electrical Engineering And Computer Science, Oregon State University, October 28, 2004.

[18] Adnan Gutub, "Efficient Utilization of Scalable Multipliers in Parallel to Compute GF (p) Elliptic Curve Cryptographic Operations," Kuwait Journal of Science & Engineering (KJSE), December 2007, 34(2): 165-182.

[19] Hakim Khali, MIEEE and Ahcene Farah," Cost-Effective Implementations of GF (P) Elliptic Curve Cryptography Computations," Ijcsns International Journal of Computer Science and Network Security, August 2007, 7(8).

[20] Adnan Gutub, Mohammad Ibrahim, and Turki Al-Somani, "Parallelizing GF(P) Elliptic Curve Cryptography Computations for Security and Speed", IEEE International Symposium on Signal Processing and its Applications in conjunction with the International Conference on Information Sciences, Signal Processing and their Applications (ISSPA), Sharjah, United Arab Emirates, February 12-15, 2007.

[21] Darrel Hankerson, Alfred Menezes, Scott Vanstone,"Guide To Elliptic Curve Cryptography," Springer-Vl-Rlag New York, Inc., 175 I-'Ifth Avenue, New York, Ny 10010, USA, 2004.

[22] Tanja Lange, "A note on L´opez-Dahab coordinates", Faculty of Mathematics, Technical University of Denmark, 2006.

[23] David, Nigel, and Jacques, "Projective coordinates Leak", Applied research and security center, France.

[24] H. Cohen, G. Frey, R. M. Avanzi, C. Doche, T. Lange, K. Nguyen and F. Vercauteren, "Handbook of Elliptic and Hyperelliptic Curve Cryptography," Discrete Mathematics and Its Applications, vol. 34, Chapman & Hall/CRC, 2005.

[25] Blake, I., Seroussi, G., and Smart, N., "Elliptic Curves in Cryptography, "Cambridge University Press: New York, 1999.

[26] Qasem Abu Al-Haija and Lo'ai Tawalbeh, " Efficient Algorithms & Architectures for Elliptic Curve Crypto-Processor Over GF (P) Using New Projective Coordinates Systems", Accepted for publication at Journal of Information Assurance and Security (JIAS), By Dynamic Publishers Inc., USA, July 2010.

[27] Akira Higuchi and Naofumi Takagi, " A fast addition algorithm for elliptic curve arithmetic in GF(2n) using projective coordinates ", Information Processing Letters, Volume 76, Issue 3, 15 December 2000, Pages 101-103.

[28] Lo'ai tawalbeh and Qasem Abu Al-Haija, "Enhanced FPGA Implementations for Doubling Oriented and Jacobi-Quartics Elliptic Curves Cryptography," Accepted for publication at Journal of Information Assurance and Security (JIAS), By Dynamic Publishers Inc., USA, Sep 2010.

[29] Mohammad Al-khatib, Qacem Sbu Al-Haija, and Azmi Jaafar, "choices on designing gf (p) elliptic curve coprocessor benefiting from mapping homogeneous curves in parallel multiplications", Accepted for publication at International Journal on computer science and engineering, By engg publisher, India, Jan 2011.

[30] Lo'ai Tawalbeh and Qasem Abu Al-Haija, "Speedup Elliptic Curve Cryptography Computation By Adopting Edwards Curves Over GF (p) On Parallel Multipliers," International Journal of Security (IJS), CSC Press, October, 2009.