Abstract—Elliptical curve cryptography is a complex computational model and before its implementation for wireless sensor network platform, several parameters have to be carefully selected. This paper surveys the complexities of ECC and investigates issues with different implementations of ECC on wireless sensor network platforms. The paper concludes with a critique of inadequacies and how the current research attempts to address some of them with a summary of some early results from the research.


I. INTRODUCTION
The security in wireless sensor networks is currently provided mostly through symmetric key cryptography [1-7]. These protocols are based on the idea of keys before the deployment of the wireless sensor network. However, due to the limitation on memory resources of wireless sensor nodes, these protocols are not able to achieve perfect security and also face a key management problem in large scale wireless sensor networks. On the other hand asymmetric key cryptography offers flexibility to node and clean interface for the security component in the sensor network [8]. This research paper offers analytical study of implementation of ECC for wireless sensor networks.

II. ELLIPTIC CURVE CRYPTOGRAPHY BASICS
Elliptic Curve Cryptography was introduced by Victor Miller [9] and Neal Koblitz [10] independently in the early eighties. The advantage of ECC over other public key cryptography techniques such as RSA is that the best known algorithm for solving ECDLP the underlying hard mathematical problem in ECC takes the fully exponential time and so far there is a lack of sub exponential attack on ECC. An Elliptic curve E(F) is the group of points which satisfies the Weierstrass equation given by

\[ y^2 z + a_1 xyz + a_2 yz^2 = x^3 + a_1 x^2 z + a_2 x z^2 + a_3 z^3, \]

where \( a_1, x, y \) coordinates are elements in the field F. Two types of field are suitable for ECC, namely, characteristic two finite extension fields, where the Weierstrass equation can be simplified to, \( y^2 + xy = x^3 + ax^2 + b \) and large prime characteristic finite fields, where the short Weierstrass form defines the curve, \( y^2 = x^3 + ax + b \). In both cases the field chosen can be defined by its order, denoted by \( q \) and the chosen curve defined by the parameters \( a, b \) to the simplified Weierstrass equation.

Group law on an elliptic curve can be explained with an Elliptic curve E(F) which is shown in Figure 1. The addition of point Q1 and Q2 is \(-R1\). The point \(-R1\) is the mirror point of R1. Point R1 can be obtained by drawing a line in between points Q1 and Q2 which intersects the curve at R1.

III ELLIPTIC CURVE DIFFIE-HELLMAN SCHEME (ECDH) FOR WSN

When adding Q1 and Q2 on the elliptic curve with affine coordinates namely \((x_1, y_1)\) and \((x_2, y_2)\), slope of the line passing through them has to be found out. Slope \( m \) of the line passing through them is given by,

\[ m = (y_2 - y_1) / (x_2 - x_1). \]

The coordinates of the point \(-R1\) which represents the sum of Q1 and Q2 are given by \((x_3, y_3)\)

Where \( x_3 = m^2 - x_1 - x_2 \),

\[ y_3 = (x_1 - x_3)m - y_1. \]
The original Diffie-Hellman algorithm requires 1024 bits to achieve sufficient security but Diffie Hellman based on elliptic curve can achieve the same security level with 160 bit [8]. The classical Elliptic Curve Diffie Hellman scheme works as shown in the Fig. 2.

Initially Alice and Bob agree on a particular curve with base point $P$. They generate their public keys by multiplying $P$ with their private keys namely $Ka$ and $Kb$. After sharing public keys, they generate a shared secret key by multiplying public keys by their private keys. The secret key is $R=Ka*Kb*P$. With the known values of $Qa$, $Qb$ and $P$ it is computationally intractable for an eavesdropper to calculate $Ka$ and $Kb$ which are the private keys of Alice and Bob.

To calculate public and shared secret keys involve most famous scalar multiplication operation which is nothing but point doubling and point addition of point $P$ as shown in the figure 1.

IV IMPLEMENTATION ISSUES FOR ELLIPTIC CURVE CRYPTOGRAPHY IN WSN:

The indicative list of relevant issues in implementing ECC is as below:

1. **Seven Tuples**: In order to set up ECDH scheme all users chooses one group of points. This group of points can be specified by an Elliptic Curve parameters which can be defined as seven Tuples [11],

   $$(a, b, q, G, n, h, Fr)$$

   $a$ and $b$ defines the specific elliptic curve, $q$ identifies the order of chosen finite field, $G$ is a base point which generates the subgroup of points on the chosen curve, $n$ gives the number of points in the specified group and $h$ gives the total number of points on the chosen curve. $Fr$ is used to give an indication of the representation to use for the underlying field elements.

2. **Parameter Set Selection**: In case of parameter selection two options involved one is Fixed point selection and other is Random point selection. In the first method you select finite field of your choice, generate curve, followed by selection of a suitable sub group and also a field representation. The next technique involves use of random number generator for selecting the curve, over finite field using an arbitrary seed string. Selection of proper coordinate system also affects the performance of ECC system as computational cost of addition and doubling operations depends on the coordinate system used. Table 1 from [12] shows that addition is faster in affine coordinate system and on the other hand doubling is faster in modified Jacobian coordinate system. For fast scalar multiplication it is advised to switch between these coordinate systems to get better results. Cohen et al [13] has suggested the same approach known as ‘mixed coordinate system’.

3. **Required level of Security**: The Elliptic curve cryptography is based on difficulty of solving Elliptic Curve Discrete Logarithmic Problem. The difficulty of solving ECDLP depends on the size of $n$ which gives the number of points in the specified group. Reasonable size of $n$ give very long time periods for solving ECDLP. The following table gives idea about time required to solve ECDLP problem for given size of $n$.

4. **Interoperability**: To achieve interoperability [11] suggested that all the participating nodes in sensor network will have key pairs based on the same shared EC parameters set. In addition to above, they must use standardised version of the common ECC schemes and protocol during communication.

5. **Performance**: Selection of the ECC parameters is having a large bearing on algorithms available for the implementation of underlying mathematical operations. Using efficient algorithm will always reducing the key calculation time. In wireless sensor network it will directly affects bandwidth, memory and computational cost. Figure 3 shows these three ECC implementation tradeoffs.
V OPTIMIZATION OF ECC FOR WIRELESS SENSOR NETWORK PLATFORM:

The best performance can be obtained by designing the complete system based on ECC for wireless sensor networks. This approach will beyond the scope of this paper. In the second approach there are two levels namely application level and device level where optimization of ECC can be obtained.

Application Level Issues involves following areas [11]:

- Requirement of security level and selection of appropriate EC parameters set.
- Selection of cryptographic scheme.
- Formats for network transfer of keys and other cryptographic primitives.

Device Level issues concern the selection and optimization of the available underlying mathematical algorithms based on the parameters set chosen. It involves,

- Selection of internal field elements representation and associated mathematical algorithm.
- Selection of internal point representation and selection and optimization of point addition algorithm.
- Selection and optimization of a scalar multiplication algorithm taking side channel attacks in to account.
- Utilization of special purpose hardware.

VI PREVIOUS ATTEMPTS OF IMPLEMENTATION OF ECC ON WIRELESS SENSOR NETWORK PLATFORM:

Before implementing elliptical curve cryptography on WSN several choices have to be made as seen in the previous sections. The choice of implementation depends on several parameters, namely the finite field, field representation, elliptic curve algorithms for the field arithmetic [14,26].

Gura et.al implemented ECC on an 8 bit microcontroller by using elliptic curves GF(p) over prime integer field because binary polynomial field arithmetic (specifically multiplication) was insufficiently supported by the contemporary microprocessors and would thus lead to lower performance.

The point multiplication of positive integer and point P on an elliptic curve decomposed into sequence of point additions and point doublings. Gura et.al also proposed hybrid method for multiplication by combining advantages of row wise and column wise multiplication techniques. The column wise strategy was used as the outer algorithm and the row wise strategy was used as the inner algorithm. The curves referred were curves standardised by NIST [16]. They also employed mixed coordinated system by using a combination of Modified Jacobian and Affine coordinates. The Non Adjacent Forms (NAF) method was exploited for recoding the positive integer k in point multiplication in order to reduce point additions.

Shantz.et.al [17] presented an efficient technique to calculate a modular division. The idea is to compute \( y/x \) in one operation instead of computing \( 1/x \) first and then multiplying it with \( y \). This scheme has reduced one multiplication in modular division operation. The new algorithm can be applied in both GF (p) and GF (2^m) fields.

Woodbury et al [18] introduced another ECC system over optimal extension field GF(p^m) where \( p \) is chosen of the form \( 2^n \pm c \). The author implemented ECC in less than 2 seconds.

Cohen et al [19] analysed the impact of a coordinate system and proposed new modified Jacobian coordinates which achieves the fastest doubling operation. Moreover they introduced a mixed coordinate system, which divides exponentiation into sub operations and chose the best coordinate representation for each sub operation.

Malan.et.al. [20] implemented ECC over GF(2^m) binary extension field curves as it allows space and time efficient algorithms. Implementation was carried out on MICA2 platform with Tiny OS and nes C language. They used polynomial basis and multiplication of point was obtained by Algorithm IV.1 in Blake et.al [21]. Addition was achieved with Algorithm 7 and multiplication of elements in \( GF(2^m) \) was implemented as Algorithm 4 in Lopez and Dahab while inversion was implemented as Algorithm 8 in Hankerson et al[22].

Mallan’s [20] EccM2.0 obeys NIST recommended curves [16] over binary extension field. The time required to generate a shared key with EccM 2.0 averaged over 100 trails is about 34.61 Sec with standard deviation of 0.921 seconds. For some applications this time may not be practicable. Further reduction of EccM 2.0’s [20] running time through source or assembly level enhancement is of course of interest. Worthy of consideration for improvement of this work is a “Normal Basis” which can be implemented by using ANDs, XORs, and cyclic Shifts which will ultimately expedite multiplication and squaring operations. Hybrid of Normal and polynomial bases can be used to exploit the advantage of each simultaneously [20], or as course work by Gura et.al [15]suggests that EccM 2.0 module may be re-implemented over GF(p) as expensive
inversions operations could be avoided by use of projective as opposed to affine coordinates [20].

VII POSSIBLE PATHWAYS FOR THE ENHANCEMENT

As discussed above, previous researchers have used prime integer field because binary polynomial field arithmetic was insufficiently supported by the contemporary microprocessors. Our research suggests binary polynomial field provides better results on modern sensor nodes like Imotes as they are better equipped with memory resources. In addition, binary field is more microprocessor friendly than prime integer field. Also instead of stressing on NAF method for recoding of positive integer, method based on one’s complement subtraction for recoding is very attractive and simple option. NAF may offer lowest Hamming weight but it also gives penalty on memory and engages microcontroller for more time on calculation rather than sensing applications as it involves modular operations. The one’s complement subtraction method for recoding of positive integer combined with Cohen’s mixed Jacobian coordinates show much promise and initial results show that the combination gives a significant improvement in results. This gives a better choice for wireless sensor networks as Jacobian coordinates’ give fastest doubling operation and one’s complement subtraction reduces point additions at the cost of point doublings.

VIII CONCLUSION

From the above discussion it is clear that elliptic curve cryptography is complex and there are a number of practical issues to be resolved when integrating the technology into wireless sensor networks security system. One must consider the performance overheads in terms of the time, memory and bandwidth penalty for the use of authentication and encryption/decryption in WSN applications. This paper has provided a brief view into the complex topic of elliptic curve cryptography. It has enormous potential for WSN provided that proper algorithms have been used for scalar multiplication process.

REFERENCES