TABLE 3. COMPUTATIONAL COSTS FOR THE UNIFIED LAW UNDER DIFFERENT CONFIGURATIONS

Curve	Coordinates	Addition / Unified Law
Edwards	affine	$2I + 6M + 4H + 1\bar{c} + 1\bar{d}$
	standard projective	$10M + 1S$
	inverted Edwards	$9M + 1S$
extended Jacobi quartic	affine	$2I + 5M + 5S + 7H + 2\bar{\epsilon} + 1\bar{\delta}$
	projective	$8M + 3S$

When Edwards curves are represented according to the inverted Edwards coordinates system, the neutral element and the inverse of (X, Y, Z) do not belong to the curve (#9). The constraint $XYZ \neq 0$ stems from such condition [13]. Indeed, given two points $P_1\ (X_1, Y_1, Z_1)$ and $P_2\ (X_2, Y_2, Z_2)$ belonging to the Edwards curve (#9), one has that:

the unified law $P_3(X_3, Y_3, Z_3) = P_1(X_1, Y_1, Z_1) + P_2(X_2, Y_2, Z_2)$ is expressed by the following equations [13]:

$$X_3 = (E + B)H \ \ mod\ p; \qquad (\#13)$$
$$Y_3 = (E - B)I \ \ mod\ p;$$
$$Z_3 = AHI \ \ mod\ p.$$

where

$$A = Z_1 Z_2 \ \ mod\ p;\ \ B = dA^2 \ \ mod\ p;$$
$$C = X_1 X_2 \ \ mod\ p;\ \ D = Y_1 Y_2 \ \ mod\ p;$$
$$= CD \ \ mod\ p;\ \ H = C - D \ \ mod\ p;$$
$$\text{I} = (X_1 + Y_1)(X_2 + Y_2) - C - D \ \ mod\ p;$$

The important fact is that moving to the inverted Edwards coordinates allows one to avoid modular inversions. Again, a computational overhead stems from the need to map a three-dimensional projective system back to the two-dimensional affine system. In the case of inverted Edwards coordinates, this procedure involves two modular inversion and two modular multiplications.

3.2. Extended Jacobi quartic curves in projective coordinates

A point (x_a, y_a) belonging to an extended Jacobi quartic curve represented in affine coordinates can be transformed in the point (X_s, Y_s, Z_s) belonging to the same curve represented in projective coordinates. The mapping rule can be formalized as follows:

$$x_a = \frac{X_s}{Z_s} \ \ mod\ p, \qquad y_a = \frac{Y_s}{Z_s^2} \ \ mod\ p, \qquad (\#14)$$
$$Z_s \neq 0 \ \ mod\ p$$

The curve equation in projective coordinates then becomes [14]:

$$Y^2 = \epsilon X^4 - 2\delta X^2 Z^2 + Z^4 \ \ mod\ p \qquad (\#15)$$

where parameters δ and ϵ should be set by following the rules already reported in section 2.

In the new coordinate system the neutral element is (0,1,1), and the inverse of (X, Y, Z) is $(-X, Y, Z)$. Moreover, given two points $P_1\ (X_1, Y_1, Z_1)$ and $P_2\ (X_2, Y_2, Z_2)$ that belong to the extended Jacobi quartic curve (#15), one has that:

the unified law $P_3(X_3, Y_3, Z_3) = P_1(X_1, Y_1, Z_1) + P_2(X_2, Y_2, Z_2)$ is expressed by the following equations [14]:

$$X_3 = X_1 Z_1 Y_2 + Y_1 X_2 Z_2 \ \ mod\ p; \qquad (\#16)$$
$$Y_3 = [B + G][D - \delta F] + \epsilon F(X_1^2 Z_2^2 + X_2^2 Z_1^2) \ \ mod\ p; \qquad (\#28)$$
$$Z_3 = B - G \ \ mod\ p.$$

where

$$A = Z_1 Z_2 \ \ mod\ p;\ B = A^2 \ \ mod\ p;$$
$$C = X_1 X_2 \ \ mod\ p;\ \ D = Y_1 Y_2 \ \ mod\ p;$$
$$= C^2 \ \ mod\ p;\ \ F = 2AC \ \ mod\ p;$$
$$G = \epsilon E \ \ mod\ p;$$

As a result, moving from affine coordinates to the projective coordinates allows one to avoid modular inversions. The coordinate transformation process again involves the affine, bi-dimensional coordinates system and a projective, three-dimensional coordinates system. The transformation from projective coordinates back to affine coordinates requires one to complete two modular inversions and three modular multiplications.

3.3. Theoretical analysis of the computational load

Table 3 provides the computational load of the unified law for the different configurations of the ECC system analyzed in this research. Thus, the first column of the table gives the curve family; the second column indicates the coordinate system adopted; finally, the third report the computational costs of the unified law operation when the ECC system is based on the specific set up {curve, coordinate system}. The following notations have been used in the formalization of the computational costs:

- I: cost of a modular inversion;
- M: cost of a modular multiplication;
- S: cost of a modular squaring;
- H: cost a modular addition/subtraction.

It is worth noting that in the literature one finds that $I \approx 100 \cdot M$, while $M \approx S$.

When the unified law involves rational points belonging to an Edward curve, the following quantities should also be defined:

- $\bar{c}$: cost of a modular multiplication by c (Edwards curves);
- $\bar{d}$: cost of a modular multiplication by d (Edwards curves).

When the operation involves rational points belonging to an extended Jacobi quartic curve, the following quantities should be defined:

- $\bar{\epsilon}$: cost of a modular multiplication by ϵ (extended Jacobi quartic curves);
- $\bar{\delta}$: cost of a modular multiplication by δ (extended Jacobi quartic curves).

The values of the computational costs, $\bar{c}$, $\bar{d}$, $\bar{\epsilon}$, and $\bar{\delta}$ actually depend on the number of bits required to represent the parameters c, d, ϵ and δ, respectively, and are set accordingly.

An analysis of the computational costs reported in Table 3 leads to a first, important conclusion: the unified law always proves less demanding when carried out in a projective coordinates system, irrespectively of the specific family of curves adopted. The crucial issue is the modular inversion, which cannot be avoided when using affine coordinates. Table 3 shows that the computational effort required to complete the operation in projective coordinates does not exhibit large variations among the various families. However, one should take into account that any point multiplication procedure eventually iterates several times this formula; as a result, a small saving in the computational load on the single operation may lead to significant improvements on the complete ECC system.

3.4. Overall cryptosystem

The previous Sections showed that the design of an ECC cryptosystem to be hosted by a programmable microprocessor requires addressing a few aspects: the family of curves to be adopted, and the coordinate system to be used. Each one of these design options is relevant to the computational efficiency of the eventual cryptosystem.

The proposed cryptosystem supports the point multiplication that is the core operation of the standard elliptic curve Diffie–Hellman (ECDH) key agreement protocol [1]. The general scheme of the ECC cryptosystem is the following:

- coordinate transformation (from affine to projective coordinates);
- point multiplication;
- coordinate transformation (from projective to affine coordinates).

The following Section presents experiments to compare the performance of the different implementations of the ECC cryptosystem that can be obtained by setting the two features: type of curve and coordinate system.

4. Experimental results

In order to test all the possible approaches that can be made to produce an elliptic curve point multiplication, different embedded platforms were considered:

- ARM9 running at 210 MHz;
- ARM11 running at 700 MHz;
- ARM Cortex A8 running at 1.0 GHz;

Experimental results show the associate timings required to complete the computing process.

The experimental session involved six different Edwards curves and as many extended Jacobi quartic curves. In both cases, three different key sizes have been used: 256 bit, 384 bit, and 512 bit. All the curves are cryptographically secure, as they belong to the curve database provided in [15].

Table 4. Edwards curve cryptosystem on ARM9@ 210Mhz (Values are in msec).

ID	bits	Affine multiplication	Standard projective multiplic.	Inverted multiplic.
E-1	256	242.1	153.6	145.9
E-2	256	247.5	158.9	149.8
E-3	384	521.3	352.1	335.7
E-4	384	519.3	325.4	309.5
E-5	512	955.1	582.4	544.5
E-6	512	946.4	604.4	563.6

Table 5. Edwards curve cryptosystem on ARM11@ 700Mhz (Values are in msec).

ID	bits	Affine multiplication	Standard projective multiplic.	Inverted multiplic.
E-1	256	51.86	31.84	31.48
E-2	256	54.57	34.94	34.46
E-3	384	115.21	70.71	68.97
E-4	384	115.76	67.18	65.02
E-5	512	212.02	129.32	120.22
E-6	512	208.39	129.86	120.83

Table 6. Edwards curve cryptosystem on ARM CortexA8@ 1Ghz (Values are in msec).

ID	bits	Affine multiplication	Standard projective multiplic.	Inverted multiplic.
E-1	256	24.63	17.96	17.42
E-2	256	24.62	18.51	18.11
E-3	384	55.52	37.73	35.77
E-4	384	55.74	36.82	35.58
E-5	512	107.01	68.15	64.60
E-6	512	106.37	73.43	65.83

Table 7. Extended Jacobi quartic curve cryptosystem on ARM9@ 210Mhz (Values are in msec).

ID	bits	Affine multiplic.	Standard projective multiplication
JQ-1	256	333.7	197.3
JQ-2	256	358.7	212.1
JQ-3	384	790.9	503.4
JQ-4	384	779.0	492.9
JQ-5	512	1298.3	831.1
JQ-6	512	1313.0	818.3

Table 8. Extended Jacobi quartic curve cryptosystem on ARM11@ 700Mhz (Values are in msec).

ID	bits	Affine multiplic.	Standard projective multiplication
JQ-1	256	77.08	47.14
JQ-2	256	81.62	49.43
JQ-3	384	168.12	106.90
JQ-4	384	167.90	106.66
JQ-5	512	286.73	185.16
JQ-6	512	298.31	190.34

Table 9. Extended Jacobi quartic curve cryptosystem on ARM CortexA8@ 1Ghz (Values are in msec).

ID	bits	Affine multiplic.	Standard projective multiplication
JQ-1	256	45.19	23.64
JQ-2	256	37.56	24.74
JQ-3	384	80.49	53.28
JQ-4	384	78.14	51.47
JQ-5	512	148.05	100.04
JQ-6	512	145.80	97.68

Table 10. Edwards curves coordinate conversions on ARM11@ 700Mhz (Values are in msec).

ID	bits	Standard projective to affine coord's	Inverted to affine coordinates
E-1	256	0.04	0.08
E-2	256	0.05	0.09
E-3	384	0.07	0.12
E-4	384	0.06	0.13
E-5	512	0.10	0.18
E-6	512	0.09	0.18

Table 11. Extended Jacobi quartic curve coordinate conversions on ARM11@ 700Mhz. (Values are in msec).

ID	bits	Projective to affine coordinates
JQ-1	256	0.09
JQ-2	256	0.09
JQ-3	384	0.14
JQ-4	384	0.14
JQ-5	512	0.18
JQ-6	512	0.19

Tables 4-9 report on the results obtained by evaluating point multiplication with the binary/right-to-left algorithm. Each table refers to a specific embedded architecture. For each curve, each Table gives: the curve identifier, including the bit-length of curve parameters; the time spent to accomplish a point multiplication on the Edwards curve, and the same quantity for the extended Jacobi quartic curves, for different coordinate systems. All timings are expressed in msec.

The empirical evidence obtained on the various hardware platforms highlights some interesting outcomes. First, the use of projective coordinate systems allows one to obtain consistent improvements on the point multiplication procedure. Secondly, the Edwards curves in both coordinates seem to outperform in general the extended Jacobi quartic curves, in terms of computing speed. But the best performance was obtained by the Edwards curves in inverted coordinates. Finally, from the tables 10 and 11, one can note that the computational time required by coordinate conversions is in general very low with respect to the projective multiplication phase regardless of the type of curves.

The extensive experimental session aims to demonstrate the flexibility of presented framework, allowing different hardware platform to complete cryptographic primitives over both Edwards and extended Jacobi quartic curves and different coordinate systems.

5. Conclusions

Elliptic curve cryptography provides an appealing alternative to conventional public-key algorithms, as the former cryptosystem can obtain a comparatively higher security level per key-bit. Since the computational complexity of a cryptographic machine correlates with the size of the key, low-resource, low-power embedded devices mostly benefit from that property. However, the actual implementation of elliptic-based cryptosystems requires one to deal with a range of design options, which in turn may affect the eventual computational complexity of the system.

The paper showed that Edwards curves can effectively support cryptosystems in low-resource, low-power embedded devices. In principle, both Edwards curves and extended Jacobi quartic curves provide interesting features that allow one to optimize the number of operations to be completed for implementing ECC. Indeed, the experimental session proved that the former configuration can attain remarkable results in terms of computational efficiency. In this regard, it is worth noting that a cryptosystem based on Edwards curves attains satisfactory performances even in devices running at a clock frequency of 200 MHz.

6. Acknowledgments

The authors would like to thank S. Decherchi for his research and his teaching that made possible this work.

7. References

[1] Menezes AJ, van Oorschot PC, Vanstone SA: *Handbook of Applied Cryptography*. CRC Press 1996.

[2] E. Barker, W. Barker, W. Burr, W. Polk, M. Smid, NIST special publication 800-57, "Recommendation for Key Management – Part 1: General (Revision 3)". July 2012. Available "http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57_part1_rev3_general.pdf".

[3] Fact Sheet NSA Suite B Cryptography, National Security Agency, http://www.nsa.gov/ia/ programs/suiteb_cryptography/, retrieved November 17, 2012.

[4] D. Hankerson, A. Menezes, S. Vanstone, "Guide to Elliptic Curve Cryptography", Springer, 2004.

[5] Quisquater, J.J., Samyde, D.: ElectroMagnetic Analysis (EMA): Measures and Countermeasures for Smart Cards. e-smart 2001, LNCS 2140 (2001) 200–210.

[6] P. Montgomery, "Modular Multiplication Without Trial Division", Mathematics of Computation, vol. 44, pp. 519–521, 1985.

[7] H.M. Edwards, "A normal form for elliptic curve", Bulletin of the American Mathematical Society - 2007.09.04.

[8] H. Hisil, K. K. Wong, G. Carter, E. Dawson. "Jacobi Quartic curves Revisited", ACISP 2009, LCNS Vol. 5594, pp 452-468, Springer-Verlag, 2009.

[9] D.J.Bernstein, P.Birkner, T.Lange and C.Peters. "Optimizing double-base elliptic curve single-scalar multiplication". INDOCRYPT'07 Proceedings of the cryptology 8th international conference on Progress in cryptology, pp 167-182, Springer-Verlag Berlin, 2007.

[10] Guerric Meurice de Dormale, Jean-Jacques Quisquater. "High-speed hardware Implementations of Elliptic Curve Cryptography: A survey". Journal of Systems Architecture 53 (2007), pp. 72–84, 2006.

[11] D.J. Bernstein, T. Lange, "Faster addition and doubling on elliptic curves", University of Chicago, Chicago, IL 60607-7045,USA, Technische Universititeit Eindoven, P.O.,Box 513, 5600 MB Eindhoven, Netherlands - 2007.09.06.

[12] S. A. Vanstone, "Next generation security for wireless: elliptic curve cryptography", Computers and Security, Vol 22, No 5, Aug. 2003.

[13] D.J.Bernstein and T.Lange. "Inverted Edwards coordinates". AAECC'07 Proceedings of the 17th international conference on Applied algebra, algebraic algorithms and error-correcting codes, pp 20-27, Springer-Verlag Berlin, 2007.

[14] O. Billet, M. Joye. "The Jacobi Model of an Elliptic Curve and Side-Channel Analysis" Proceedings of the 15th international conference on Applied algebra, algebraic algorithms and error-correcting codes AAECC'03, pp 34-42 Springer-Verlag Berlin, 2003.

[15] H. Ivey-Law, R. Rolland. "Constructing a database of cryptographically strong elliptic curves". Proc. of 5th Conf. on Network Architectures and Information Systems Security, SAR-SSI 2010.

