# Fast explicit formulae for genus 2 hyperelliptic curves using projective coordinates (Updated)

Thomas Wollinger
Escrypt – embedded security GmbH
twollinger@escrypt.de

Vladyslav Kovtun
Kharkiv Air Force University
vladislav.kovtun@gmail.com

## Abstract

*This contribution proposes a modification of method of divisors group operation in the Jacobian of hyperelliptic curve over even and odd characteristic fields in projective coordinate. The hyperelliptic curve cryptosystem (HECC), enhances cryptographic security efficiency in e.g. information and telecommunications systems (ITS).*

*Index Terms – hyperelliptic curves, explicit formulae.*

## 1. Introduction

The sweeping progress in Information Technologies implies several requirements on modern ITS, especially to secure confidence level, integrity, observability and authenticity of the information that is created, circulates and is being stored. A typical example of such ITS is used for bank applications. ITS can be ensured by applying a data protection based on public key cryptographic.

The transformation in a group of points over an elliptic curve (EC) is accepted as a modern public key primitive [1]. Standard [1] describes the main operator of EC scalar multiplication. Nowadays, the transformation in Jacobian of hyperelliptic curve (HEC) are considered the most promising substitution of EC. The cryptographic transformation in Jacobian is also grounded on scalar multiplication [2] of reduced divisors (hereinafter referred to as divisors).

At the same time, the increasing number of security sensitive applications leads to continuous increase of load pressure on information protection system, and specifically on the public key cryptographic primitives. Thus, it is important to get a significant decrease in computational complexity (hereinafter, complexity) for these primitives. This decrease achieved by reducing the complexity of divisor scalar multiplication, and, therefore by reducing the complexity of group addition and doubling of divisors.

Until recently, arithmetic transforms in Jacobian have been performed using Cantor's method [10], with modifications introduced by Koblitz [2]. HECC were elaborate both in terms of description and in efficient calculations. In this context, the academic community put a lot of attention to enhancing efficiency of the HECC, e.g. [3-10]. The result has been improved methods of arithmetic transforms in Jacobian. Until today, there were many publications trying to improve the HECC. In this paragraph, the authors list only the ones most important for the contribution at hand. Methods of addition and doubling for curves of genus 2 were considered in papers [3, 4]. The first practical implementation of these methods was performed by Harley [5]. The extension of the results [5] for curves over even characteristic fields is given in [6]. Further development of methods of addition and doubling is given in papers [7, 8] and the results have also been extended for curves over even characteristic fields in [9, 10].

The analysis of complexity of the known methods of arithmetic transforms in Jacobian genus 2 HEC over even and odd characteristics demonstrates that the existent methods are already efficient, however there is still room for further improvements, as shown in this contribution.

The most complex field operation during the operators of divisor addition and doubling in the HECC is the inversion, see e.g. [11, 13]. In [7], for the first time an approach for implementing arithmetic operations in Jacobian genus 2 HEC without having to compute field inversion were published. Further development of the proposed approach is given in [12, 13] while the results are improved and spread to a wider class of HEC over even characteristic field. The contribution at hand uses as basis the group operation algorithms as presented in [5, 6, 9].

In compliance with the introduced constructions, the objective of the paper is in providing more efficient group operations on the basis of genus 2 HEC using projective coordinates $\left[U_1, U_0, V_1, V_0, Z\right]$ [6, 12]. We were able to decrease the complexity of the scalar multiplication by 4 % and therefore increase efficiency.

## 2. Efficient Explicit Formulae for HECC Using Projective Coordinates

The proposed improvement in complexity is based on Harley's method [5] and the modifications of Harley algorithm published in [6].

Algorithm of addition and doubling can be computed using operations in the ring of polynomial functions: division, multiplicative inversion, modular reduction, multiplication. Our methods that were used to reduce the number of these field operations are listed below:

- In order to simplify arithmetic operation procedures in the ring of polynomial functions, we performed normalization of these functions [6, 12];
- In order to normalize and minimize Hamming weights of HEC parameters $h(x)$ and $f(x)$ we applied HEC of a special form [7, 9, 14, 15];
- In order to simultaneously invert several field elements, we applied the Montgomery method [5, 6, 9];
- In order to multiply polynomial functions with different powers, we applied the Karatsuba method [6];
- In order to modular reduce polynomial functions with different powers, we applied the Karatsuba method [5];
- In order to exclude inversion over field, we applied projective representation of divisors [6, 7, 12].

## 2.1. Group Operations for HECC over Odd Characteristics Field

Based on the modifications proposed above, we obtain the following group arithmetic for the HECC which is specified by the equation $v^2 + h(u)v = f(u)$ over $\mathbf{F}_q$, considering odd characteristic field and projective coordinates, where $h = h_2x^2 + h_1x + h_0$, $h_i \in \mathbf{F}_2$ and $f = x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$, $f_4 \in \mathbf{F}_2$, $f_i \in \mathbf{F}_q$. Obtained algorithms for specified above curves, are presented in Tables II, III.

## 2.2. Group Operations for HECC over Even Characteristics Field

In this subsection, we consider arithmetic transforms with divisors in Jacobian HEC over even characteristic fields. HEC application over such fields allows for a more efficient group arithmetic if compared to odd characteristics fields.

For this case we used HEC with $h(x) = x$ and $f = x^5 + f_1x + f_0$, $f_i \in \mathbf{F}_2$. Obtained algorithms for the specified above curves are presented in Tables IV, V.

## 3. Analysis of Computational Complexity

In Table I, we give the complexity evaluation, comparing our algorithms and the ones introduced in [5, 6, 9, 12-15]. The algorithm complexity is shown in field operations.

The newly introduced algorithms are the most efficient, when considering projective coordinates over even and odd field. We were able to decrease the complexity of the algorithms, by up to 15 % compared to the fasted algorithm introduced in [6, 14]. We believe that even if our contribution is not to cause a speed increase of magnitudes, an aggregated effort of many contributions we will definitely be able to reach the speed sufficient for practical applications.

## 4. Conclusions

In accordance with the paper objective, there was developed a method of arithmetic transforms in Jacobian genus 2 HEC in projective coordinates which provides a lower complexity if compared to the most efficient methods known [6, 12, 14, 15] and, thus, allowing for increase in the efficiency of scalar multiplication. This modification is characterized by:

- reduction of the number of recomputable values,
- changes in the sequence of the computational steps;
- using dependencies between polynomials in the resultant computation.

The suggested modification of method of arithmetic transforms in Jacobian genus 2 HEC results in 3 to 15 % reduction of complexity dependant on arithmetic operations used and curve type.

Thus, applying the introduced group operation reduces the complexity of the HECC scalar multiplication by 4 % compared to the best known formulae [6, 12, 14].

## 5. References


[1] IEEE P1363-2000. Standard Specifications for Public Key Cryptography. Available at: http://www.ieee.org.

[2] N. Koblitz Hyperelliptic cryptosystems. Journal of cryptology, No 1, 1989, pp.139-150.

[3] A. M. Spallek, “Kurven vom geschlecht 2 und ihre anwendung in public-key-kryptosystemen”, PhD thesis, Universitat Gesamthochschule Essen, 1994.

[4] U. Kriger, “Anwendung hyperellipischer kurven in der kryptographie”, Master’s thesis, Universitat Gesamthochschule, Essen, 2001.

[5] R. Harley, “Fast arithmetic on genus 2 curves”, available at: http://cristal.infra.fr/~harley/hyper. 2000.

[6] T. Wollinger, “Software and hardware implementation of hyperelliptic curve cryptosystems”, PhD dissertation. Bochum, Germany, May 2004.

[7] Y. Miyamoto, H. Doi, K. Matsuo, J. Chao, S. Tsujii, “A fast addition algorithm of genus two hyperelliptic curve”, In the 2002 Symposium on cryptography and information security – SCIS 2002, IEICE Japan, pp. 497-502, 2002. In Japanese.

[8] M. Takahashi, “Improving Harley algorithms for Jacobians of genus 2 Hyperelliptic curves”, In Proc. of SCIS2002, IEICE Japan, 2002. In Japanese.

[9] T. Lange, “Efficient arithmetic on genus 2 hyperelliptic curves over finite fields via explicit formulae”, Cryptology ePrint Archive, report 2002/121,

2002. Available http://eprint.iacr.org.

[10] H. Suguzaki, K. Matsuo, J. Chao, S. Tsujii, "An extension of Harley algorithm addition algorithm for hyperelliptic curves over finite fields of characteristic two", Technical report ISEC2002-9, IEICE Japan, 2002. pp. 49-56.

[11] D. Hankerson, J. Lopez Hernandez, A. Menezes, "Software implementation of elliptic curve cryptography over binary fields", Cryptographic Hardware and Embedded Systems, CHES'2000, Springer-Verlag, LNCS 1965, 2001, pp 1-24.

[12] T. Lange, "Inversion-free arithmetic on genus 2 hyperelliptic curves", Cryptology ePrint Archive, report 2002/147, 2002. Available http://eprint.iacr.org.

[13] T. Lange, "Weighted coordinates on genus 2 hyperelliptic curves", Cryptology ePrint Archive, Report 2002/153, 2002. Available http://eprint.iacr.org.

[14] B. Byramjee and S. Duquesne, "Classification of genus 2 curves over $\mathbf{F}_2$ and optimization of their arithmetic", Cryptology ePrint Archive, Report 2004/107, 2002. Available http://eprint.iacr.org.

[15] T. Lange, M. Stevens, "Efficient Doubling on Genus Two Curves over Binary Fields", Selected Areas in Cryptography, Springer-Verlag, LNCS 3357, 2004, pp. 170 – 181.

TABLE I

| Conditions | Addition | | | Mixed addition | | | Doubling | | |
|---|---|---|---|---|---|---|---|---|---|
| | $()^{-1}$ | ^2 | * | $()^{-1}$ | ^2 | * | $()^{-1}$ | ^2 | * |
| **Odd characteristic field** | | | | | | | | | |
| Affine coordinates | | | | | | | | | |
| $f_4 = 0$ [9] | 1 | 3 | 22 | | | | 1 | 5 | 22 |
| Projective coordinates | | | | | | | | | |
| $\deg(h) = 2$, $h_i \in \mathbf{F}_2$ [12] | | 4 | 47 | | 3 | 40 | | 6 | 40 |
| $\deg(h) = 2$, $h_i \in \mathbf{F}_2$ [proposed] | | **4** | **46** | | **4** | **39** | | **6** | **39** |
| **Even characteristic field** | | | | | | | | | |
| Affine coordinates | | | | | | | | | |
| $f_4 = 0$ [6, 9] | 1 | 3 | 22 | | | | 1 | 5 | 22 |
| $h_2 = 0$, $f_4 = 0$ [6, 9] | 1 | 3 | 21 | | | | 1 | 5 | 17 |
| $h(x) = x$, $f_4 = 0$, $f_3 = f_2 = 0$ [6] | | | | | | | 1 | 6 | 9 |
| $h(x) = x$, $f(x) = x^5 + f_3 x^3 + \varepsilon x^2 + f_0$, $\varepsilon \in \mathbf{F}_2$ [14] | 1 | | 24 | | | | 1 | 5 | 13 |
| $\deg(h) = 2$, $h_0 = 0$, $h_1 \in \mathbf{F}_q$, $f(x) = x^5 + \varepsilon x^4 + f_1 x + f_0$, $\varepsilon \in \mathbf{F}_2$ [14] | 1 | | 25 | | | | 1 | 4 | 22 |
| $h_1 \in \mathbf{F}_q$, $h_2 = h_0 = 0$, $f_4 = f_1 = 0$ [15] | | | | | | | 1 | 5 | 9 |
| $h_1 = 1$, $h_2 = h_0 = 0$, $f_4 = f_1 = 0$ [15] | | | | | | | 1 | 6 | 5 |
| $\deg(h) = 2$, $h_0 = 0$, $h_1 \in \mathbf{F}_q$, $f_3 = f_2 = 0$ [15] | | | | | | | 1 | 5 | 17 |
| $\deg(h) = 2$, $h_0 = 0$, $h_1 \in \mathbf{F}_2$, $f_3 = f_2 = 0$ [15] | | | | | | | 1 | 6 | 12 |
| Projective coordinates | | | | | | | | | |
| $h_2 = 0$, $f_4 = 0$ [9] | | 4 | 49 | | 4 | 39 | | 7 | 38 |
| $h(x) = x$, $f_4 = 0$, $f_3 = f_2 = 0$ [6] | | 5 | 45 | | 5 | 38 | | 7 | 31 |
| $h(x) = x$, $f_4 = 0$, $f_3 = f_2 = 0$ [proposed] | | **4** | **44** | | **5** | **37** | | **7** | **29** |
| $\deg(h) = 2$, $h_i \in \mathbf{F}_q$, $f_4 = 0$ [14] | | | | | 3 | 42 | | 6 | 39 |
| $\deg(h) = 2$, $h_0 = 0$, $h_1 \in \mathbf{F}_q$, $f(x) = x^5 + \varepsilon x^4 + f_1 x + f_0$, $\varepsilon \in \mathbf{F}_2$ [14] | | | | | | 45 | | 6 | 38 |
| $h(x) = x$, $f(x) = x^5 + f_3 x^3 + \varepsilon x^2 + f_0$, $\varepsilon \in \mathbf{F}_2$ [14] | | | | | 3 | 39 | | 5 | 26 |

TABLE II. Algorithm 1. Mixed addition of divisors

| | |
|---|---|
| **Input:** | $[U_{11}, U_{10}, V_{11}, V_{10}, 1], [U_{21}, U_{20}, V_{21}, V_{20}, Z_2]$ |
| **Output:** | $[U'_1, U'_0, V'_1, V'_0, Z'] = [U_{11}, U_{10}, V_{11}, V_{10}, 1] + [U_{21}, U_{20}, V_{21}, V_{20}, Z_2]$ |

| | Operations | Cost |
|---|---|---|
| 1 | Precomputation: $\tilde{U}_{11} = Z_2 \cdot U_{11},\ y_1 = \tilde{U}_{11} - U_{21},\ y_2 = U_{20} - U_{10} \cdot Z_2$. | 2M |
| 2 | Computation of $r$ for $u_1$ and $u_2$: $y_3 = U_{11} \cdot y_1 + y_2,\ r = y_2 \cdot y_3 + y_1^2 \cdot U_{10}$. | 1S, 3M |
| 3 | Computation $inv = r/u_2 \bmod u_1,\ inv = inv_1 x + inv_0$: $inv_1 = y_1,\ inv_0 = y_3$. | |
| 4 | Computation of $s = (v_1 - v_2) inv \bmod u_1,\ s = s_1 x + s_0$: $w_0 = V_{10} \cdot Z_2 - V_{20},\ w_1 = V_{11} \cdot Z_2 - V_{21},\ w_2 = inv_0 \cdot w_0,$ $w_3 = inv_1 \cdot w_1,\ s_1 = (inv_0 + inv_1) \cdot (w_0 + w_1) - w_2 - w_3 \cdot (U_{11} + 1),\ s_0 = w_2 - U_{10} \cdot w_3$. If $s_1 = 0$ then <Special case is considered> | 7M |
| 5 | $R = r \cdot Z_2,\ s_2 = s_0 \cdot Z_2,\ s_3 = s_1 \cdot Z_2,\ \tilde{R} = R \cdot s_3,\ w_0 = s_1 \cdot s_0,\ w_1 = s_1 \cdot s_3,\ w_2 = s_0 \cdot s_3,\ w_3 = w_1 \cdot U_{21},$ $w_4 = R \cdot s_1$. | 9M |
| 6 | Computation of $l = su_2$: $l_0 = w_0 \cdot U_{20},\ l_2 = w_3 + w_2,\ l_1 = (w_1 + w_0) \cdot (U_{21} + U_{20}) - l_0 - w_3$. | 2M |
| 7 | Computation of $u' = (s(l + h + 2v_1) - k) u_1^{-1},\ k = (f - v_1 h - v_1^2) / u_1$: $\tilde{U}'_1 = 2w_2 - s_3 \cdot s_1 y_1 + h_2 \tilde{R} - R^2,$ $\tilde{U}'_0 = s_2^2 + s_1 \cdot y_1 \cdot (s_1 \cdot \tilde{U}_{11} - 2s_2) + y_2 \cdot w_1 + 2w_4 \cdot V_{21} + h_1 \tilde{R} + R \cdot [h_2 (s_2 - s_1 \tilde{U}_{11}) + r \cdot (y_1 + 2U_{21} - f_4 Z_2)]$. | 2S, 8M |
| 8 | Correction: $U'_0 = \tilde{U}'_0 \cdot \tilde{R},\ U'_1 = \tilde{U}'_1 \cdot \tilde{R},\ Z' = s_3^2 \cdot \tilde{R}$. | 1S, 3M |
| 9 | Computation of $v' \equiv -(h + s_1 l + v_2) \bmod u',\ v' = v'_1 x + v'_0$: $V'_1 = \tilde{U}'_1 \cdot (l_2 - \tilde{U}'_1 + h_2 \tilde{R}) + s_3^2 \cdot (\tilde{U}'_0 - h_1 \tilde{R} - w_4 V_{21} - l_1),\ V'_0 = \tilde{U}'_0 \cdot (l_2 - \tilde{U}'_1 + h_2 \tilde{R}) - s_3^2 \cdot (l_0 + h_0 \tilde{R} + w_4 \cdot V_{20})$. | 5M |
| | | 4S, 39M |

TABLE III. Algorithm 2. Doubling of divisor

| | |
|---|---|
| **Input:** | $[U_1, U_0, V_1, V_0, Z]$ |
| **Output:** | $[U'_1, U'_0, V'_1, V'_0, Z'] = 2[U_1, U_0, V_1, V_0, Z]$ |

| | Operations | Cost |
|---|---|---|
| 1 | Precomputation: $Z_2 = Z^2,\ \tilde{V}_1 = h_1 Z + 2V_1 - h_2 U_1,\ \tilde{V}_0 = h_0 Z + 2V_0 - h_2 U_0$. | 1S |
| 2 | Computation of $r$ for $u$ and $h + 2v$ (while $\tilde{v} \equiv (h + 2v) \bmod u$): $w_0 = V_1^2,\ w_1 = U_1^2,$ $w_2 = \tilde{V}_1^2 = h_1^2 Z_2 + 4w_0 - h_2^2 w_1,\ w_3 = \tilde{V}_0 \cdot Z - U_1 \cdot \tilde{V}_1,\ r = \tilde{V}_0 \cdot w_3 + w_2 \cdot U_0$. | 2S, 4M |
| 3 | Computation $inv \equiv r/\tilde{v} \bmod u,\ inv = inv_1 x + inv_0$: $inv_1 = -\tilde{V}_1,\ inv_0 = w_3$. | |
| 4 | Computation of $k \equiv \lfloor (f - hv - v^2) / u \rfloor \bmod u,\ k = k_1 x + k_0$: $w_3 = f_3 \cdot Z_2 + w_1,\ k_1 = 2w_1 + w_3 - Z \cdot (w_4 + 2 f_4 U_1 + h_2 V_1),\ w_4 = 2U_0,$ $k_0 = U_1 \cdot (Z \cdot (2w_4 + f_4 U_1 + h_2 V_1) - w_3) + + Z \cdot (Z \cdot (f_2 \cdot Z - h_1 V_1 - h_2 V_0 - 2 f_4 U_0) - w_0)$. | 7M |
| 5 | Computation of $s = k \cdot inv \bmod u,\ s = s_1 x + s_0$: $w_0 = k_0 \cdot inv_0,\ w_1 = k_1 \cdot inv_1,\ s_0 = w_0 - Z \cdot U_0 \cdot w_1,$ $s_3 = (inv_0 + inv_1) \cdot (k_0 + k_1) - w_0 - w_1 \cdot (1 + U_1),\ s_1 = s_3 \cdot Z$. If $s_1 = 0$ then <Special case is considered> | 7M |
| 6 | $R = r \cdot Z_2,\ \tilde{R} = R \cdot s_1,\ w_0 = s_1 \cdot s_3,\ w_1 = s_0 \cdot s_3,\ w_3 = w_1 \cdot Z,\ w_4 = R \cdot s_3$. | 6M |
| 7 | Computation of $l = su$: $l_0 = U_0 \cdot w_1,\ l_2 = U_1 \cdot w_0,\ l_1 = (w_1 + w_0) \cdot (U_1 + U_0) - l_0 - l_2$. | 3M |
| 8 | Computation of $u' = \lfloor l^2 + \frac{1}{s} l (2v + h) - \frac{1}{s^2} (f - vh - v^2) \rfloor / u^2$: $\tilde{U}'_1 = 2w_3 + h_2 \tilde{R} - R^2,$ $\tilde{U}'_0 = s_0^2 + 2w_4 \cdot V_1 + R \cdot (h_1 s_1 + U_1 \cdot (2r \cdot Z - h_2 s_3) + h_2 s_0 - f_4 R)$. | 2S, 4M |
| 9 | Correction: $U'_0 = \tilde{U}'_0 \cdot \tilde{R},\ U'_1 = \tilde{U}'_1 \cdot \tilde{R},\ Z' = s_1^2 \cdot \tilde{R}$. | 1S, 3M |
| 10 | Computation of $v' \equiv -(h + s_1 l + v_2) \bmod u',\ v' = v'_1 x + v'_0$: $V'_1 = \tilde{U}'_1 \cdot (l_2 - \tilde{U}'_1 + w_3 + h_2 \tilde{R}) +$ $s_1^2 \cdot (\tilde{U}'_1 - h_1 \tilde{R} - w_4 V_1 - l_1),\ V'_0 = \tilde{U}'_0 \cdot (l_2 - \tilde{U}'_1 + w_3 + h_2 \tilde{R}) - s_1^2 \cdot (l_0 + h_0 \tilde{R} + w_4 \cdot V_0)$. | 5M |
| | | 6S, 39M |

TABLE IV. Algorithm 3. Mixed addition of divisors

| | | |
|---|---|---|
| **Input:** | $[U_{11}, U_{10}, V_{11}, V_{10}, 1], [U_{21}, U_{20}, V_{21}, V_{20}, Z_2]$ | |
| **Output:** | $[U'_1, U'_0, V'_1, V'_0, Z'] = [U_{11}, U_{10}, V_{11}, V_{10}, 1] + [U_{21}, U_{20}, V_{21}, V_{20}, Z_2]$ | |
| | Operations | Cost |
| 1 | Precomputation: $\tilde{U}_{11} = Z_2 \cdot U_{11}$, $y_1 = \tilde{U}_{11} + U_{21}$, $y_2 = U_{20} + U_{10} \cdot Z_2$. | 2M |
| 2 | Computation of $r$ for $u_1$ and $u_2$: $y_3 = y_1 \cdot U_{11} + y_2$, $r = y_2 \cdot y_3 + y_1^2 \cdot U_{10}$. | 1S, 3M |
| 3 | Computation of $inv = r/u_2 \bmod u_1$, $inv = inv_1 x + inv_0$: $inv_1 = y_1$, $inv_0 = y_3$. | |
| 4 | Computation of $s = (v_1 - v_2) inv \bmod u_1$, $s = s_1 x + s_0$: $w_0 = V_{10} \cdot Z_2 + V_{20}$, $w_1 = V_{11} \cdot Z_2 + V_{21}$, $w_2 = inv_0 \cdot w_0$, $w_3 = inv_1 \cdot w_1$, $s_0 = w_2 + U_{10} \cdot w_3$, $s_1 = (inv_0 + inv_1) \cdot (w_0 + w_1) + w_2 + w_3 \cdot (U_{11} + 1)$. If $s_1 = 0$ then <Special case is considered> | 7M |
| 5 | $R = r \cdot Z_2$, $s_2 = s_0 \cdot Z_2$, $s_3 = s_1 \cdot Z_2$, $\tilde{R} = R \cdot s_3$, $w_0 = s_1 \cdot s_0$, $w_1 = s_1 \cdot s_3$, $w_2 = s_0 \cdot s_3$, $w_3 = w_1 \cdot U_{21}$, $w_4 = R \cdot s_1$. | 9M |
| 6 | Computation of $l = su_2$: $l_0 = w_0 \cdot U_{20}$, $l_2 = w_3 + w_2$, $l_1 = (w_1 + w_0) \cdot (U_{21} + U_{20}) + l_0 + w_3$. | 2M |
| 7 | Computation of $u' = (s(l + h + 2v_1) - k) u_1^{-1}$, $k = (f - v_1 h - v_1^2) / u_1$, $u' = x^2 + u'_1 x + u'_0$: $\tilde{U}'_0 = s_2^2 + s_1^2 \cdot y_1 U_{11} + y_2 \cdot w_1 + \tilde{R} + R \cdot r \cdot y_1$, $\tilde{U}'_1 = w_1 \cdot y_1 + R^2$. | 3S, 5M |
| 8 | Correction: $U'_0 = \tilde{U}'_0 \cdot \tilde{R}$, $U'_1 = \tilde{U}'_1 \cdot \tilde{R}$, $Z' = s_3^2 \cdot \tilde{R}$. | 1S, 3M |
| 9 | Computation of $v' \equiv -(h + s_1 l + v_2) \bmod u'$, $v' = v'_1 x + v'_0$: $V'_1 = \tilde{U}'_1 \cdot (l_2 + \tilde{U}'_1) + s_3^2 \cdot (\tilde{U}'_0 + w_4 \cdot V_{21} + l_1)$, $V'_0 = \tilde{U}'_0 \cdot (l_2 + \tilde{U}'_1) + s_3^2 \cdot (l_0 + w_4 \cdot V_{20})$. | 6M |
| | | 5S, 37M |

TABLE V. Algorithm 4. Doubling of divisor

| | | |
|---|---|---|
| **Input:** | $[U_1, U_0, V_1, V_0, Z]$ | |
| **Output:** | $[U'_1, U'_0, V'_1, V'_0, Z'] = 2[U_1, U_0, V_1, V_0, Z]$ | |
| | Operations | Cost |
| 1 | Precomputation: $Z_2 = Z^2$, $w_0 = V_1^2$, $w_1 = U_1^2$, $w_2 = Z \cdot U_1$. | 3S, 1M |
| 2 | Computation of $r$ for $u$ and $h + 2v$ (while $\tilde{v} \equiv (h + 2v) \bmod u$): $R = U_0 \cdot Z_2^2$. | 1S, 1M |
| 3 | Computation of $inv \equiv r/\tilde{v} \bmod u$, $inv = inv_1 x + inv_0$: $inv_1 = Z$, $inv_0 = w_2$. | |
| 4 | Computation of $k \equiv \lfloor (f - hv - v^2) / u \rfloor \bmod u$: $k_1 = w_1$, $k_0 = U_1 \cdot w_1 + Z \cdot (Z \cdot V_1 + w_0)$. | 3M |
| 5 | Computation of $s = k \cdot inv \bmod u$, $s = s_1 x + s_0$: $w_0 = k_0 \cdot inv_0$, $w_1 = k_1 \cdot Z$, $s_0 = w_0 + Z \cdot U_0 \cdot w_1$, $s_3 = (inv_0 + Z) \cdot (k_0 + k_1) + w_0 + w_1 \cdot (1 + U_1)$, $s_1 = s_3 \cdot Z$. If $s_1 = 0$ then <Special case is considered> | 7M |
| 6 | $\tilde{R} = R \cdot s_1$, $w_0 = s_1 \cdot s_3$, $w_1 = s_0 \cdot s_3$, $w_3 = w_1 \cdot Z$, $w_4 = R \cdot s_3$. | 5M |
| 7 | Computation of $l = su$: $l_0 = U_0 \cdot w_1$, $l_2 = U_1 \cdot w_0$, $l_1 = (w_1 + w_0) \cdot (U_1 + U_0) + l_0 + l_2$. | 3M |
| 8 | Computation of $u' = \lfloor l^2 + \frac{1}{s} l(2v + h) - \frac{1}{s^2}(f - vh - v^2) \rfloor / u^2$: $\tilde{U}'_0 = s_0^2 + \tilde{R}$, $\tilde{U}'_1 = R^2$. | 2S |
| 9 | Correction: $U'_0 = \tilde{U}'_0 \cdot \tilde{R}$, $U'_1 = \tilde{U}'_1 \cdot \tilde{R}$, $Z' = s_1^2 \cdot \tilde{R}$. | 1S, 3M |
| 10 | Computation of $v' \equiv -(h + s_1 l + v_2) \bmod u'$, $v' = v'_1 x + v'_0$: $V'_0 = \tilde{U}'_0 \cdot (l_2 + \tilde{U}'_1 + w_3) + s_1^2 \cdot (l_0 + w_4 \cdot V_0)$, $V'_1 = \tilde{U}'_1 \cdot (l_2 + \tilde{U}'_1 + w_3) + s_1^2 \cdot (\tilde{U}'_0 + \tilde{R} + w_4 \cdot V_1 + l_1)$. | 6M |
| | | 7S, 29M |