An Efficient ASIC Implementation of Public Key Cryptography Algorithm SM2 Based on Module Arithmetic Logic Unit

Danyang Yang*, Zibin Dai, Wei Li, Tao Chen
Institute of Information Science and Technology, Zhengzhou 450001, China
* Email:dyyang1024@163.com

Abstract

SM2 public key cryptography, proposed by China, is widely used to ensure the security in communication. In this paper, based on the module arithmetic logic unit, SM3 unit, verify unit and XOR unit, we implemented a dual-field processor over SM2 public key cryptography, which can complete digital signature, verification, information encryption and decryption. In addition, the SM2 processor is described by Verilog HDL, and synthesized in CMOS 55nm process. Experimental results show that the SM2 processor runs at high frequency of 476 MHz with area of 245K gates. What's more, the processor requires only 0.56ms to compute a 256-bit point multiplication in GF(p), and 0.59ms in GF(2^m).

Keywords

SM2 processor, module arithmetic logic unit ASIC, dual-field

1. Introduction

Neal Koblitz and Victor Miller proposed the application of the ellipse curve in cryptography in 1985，independently [1]. As one of the three major public key cryptosystems, the Elliptic Curve Cryptosystem (ECC) is more difficult to break down and more secure with the same length and granularity. Thus, ECC has become the standard of public key cryptography algorithm.

On the basis of the existing research results of ECC algorithm, SM2 public key cryptography was published by China in December 2010. Compared with ECC algorithm, SM2 is slightly better in security and implementation efficiency [2]. It will have wide application prospect in public key cryptosystems.

The existing researches on elliptic curve are mainly based on ECC algorithm in [6-9], but few on SM2 algorithm. And almost all of them are about the implementation of point multiplication such as [5], rarely involving the hardware implementation of SM2 algorithm protocol.

The basic operations of SM2 algorithm are mainly the modular operations of finite field as same as ECC. In this paper, we realize point multiplication and subsequent modular operations by executing two module arithmetic logic units(MALU) in parallel. Based on unified operation module, our processor could implement SM2 digital signature algorithm and public key encryption algorithm.

The rest of the paper is organized as follows: Section 2 introduces the background information of public key cryptography algorithm SM2. Section 3 describes our hardware implementation in detail. The experimental results are shown in Section 4. Section 5 draws a conclusion to this work.

2. Background

2.1 SM2 algorithm

SM2 elliptic curve cryptography consists of several levels, such as protocol layer, algorithm layer, group operation layer and finite field layer, as shown in Fig. 1. The protocol layer is used to complete a variety of elliptic curve cryptography services, including digital signature, key distribution, information encryption, etc. The algorithm layer implements point multiplication. The group operation layer mainly includes point addition (PA) and point doubling (PD), which are realized by modular operations. The finite field layer which plays a basic role in the SM2 algorithm, also known as the modular operation layer, includes module add-sub, module square, module multiplication, module inverse and other module operations.

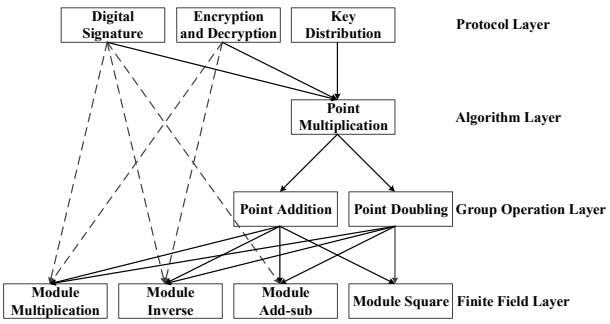


Fig 1. The layer of SM2 public key cryptography

2.2 Point multiplication

In ECC algorithm, point multiplication is the most time-critical function whose speed determines the efficiency of the ECC processor. Elliptic curve scalar multiplication can be realized by calling PA and PD operation. When completing point multiplication, we will try our best to reduce the number of modular division for its long operation time. Through the comparative analysis of various projective coordinates, this paper uses Jacobian projective coordinates [4] to calculate PA and PD in GF(p). In GF(2^m), Lopez and Dahab projective coordi-

978-1-7281-0735-6/19/$31.00 ©2019 IEEE

nates are used to calculate the PA and PD.

PA and PD can be computed by several modular operations, such as modular multiplication and modular addition/subtraction in finite field.

2.3 Modular operation

Modular operations are the main parts and basic operations of the ECC algorithm. The modular multiplication is implemented based on adder, which is shown in algorithm 1 [3]. The modular division is based on the extended Euclidean algorithm, as shown in algorithm 2. The algorithms of binary field are similar to that of prime field. But, δ will be initialized with 1 for GF(2^m) in algorithm 2.

Algorithm 1: Modified Radix-4 Interleaved Multiplication

Input : $A,\ B,\ P$
Output : $V = A * B \mod P$

$V \leftarrow b_0 \bullet A,\ U \leftarrow 2 \bullet A \ (mod\ P),\ A \leftarrow 4 \bullet A \ (mod\ P),\ B \leftarrow B/2$
For i from 0 $up\left\lceil \frac{m}{2} \right\rceil do$
  $V = V + b_0 \bullet U + b_1 \bullet A (mod\ P);$
  $U \leftarrow 2 \bullet A \ (mod\ P),\ A \leftarrow 4 \bullet A \ (mod\ P),\ B \leftarrow B/4;$
end for
retern V

Algorithm2: Radix-4 Binary GCD Modular Division Algorithm

Input : Prime P and $X, Y \in [1, P-1]$
Output : $Z = X / Y \mod P$

$A \leftarrow P, B \leftarrow Y, U \leftarrow X, V \leftarrow 0, \rho \leftarrow m, \delta \leftarrow 0;$
while $\rho > 0$ *do*
 if $B \mod 4 = 0$ then
  $B \leftarrow B/4;\ U \leftarrow U/4 \bmod P;$
  if$\delta \le 0$ then $\rho \leftarrow \rho - 2;$
  else if $\delta = 1$ then $\rho \leftarrow \rho - 1;$
  end if
  $\delta \leftarrow \delta - 2;$
 else if $B \mod 2 = 0$ then
  $B \leftarrow B/2;\ U \leftarrow U/2 \bmod P;$
  if $\delta \le 0$ then $\rho \leftarrow \rho - 1;$end if
  $\delta \leftarrow \delta - 1;$
 else
  $Next\_A \leftarrow B;\ Next\_V \leftarrow U;$
  if $(A+B) \mod 4 = 0$ then
   $B \leftarrow (B+A)/4;\ U \leftarrow (U+V)/4 \bmod P;$
  else
   $B \leftarrow (B-A)/4;\ U \leftarrow (U-V)/4 \bmod P;$
  end if
  if $\delta < 0$ then $A \leftarrow Next\_A; V \leftarrow Next\_V; \delta \leftarrow -\delta - 1;$
  else if $\delta = 0$ then $\delta \leftarrow \delta - 1;\ \rho \leftarrow \rho - 1;$
  else $\delta \leftarrow \delta - 1;$
  end if
end if
end while
if $V < 0$, then $V \leftarrow V + p;$ end if
if $A = 1$, then $Z \leftarrow V$, else $Z \leftarrow P - V$;end if
return Z

3. Proposed hardware architecture

3.1 SM2 architecture

The architecture of the proposed SM2 public key cryptography processor in this work is shown in Fig 2.

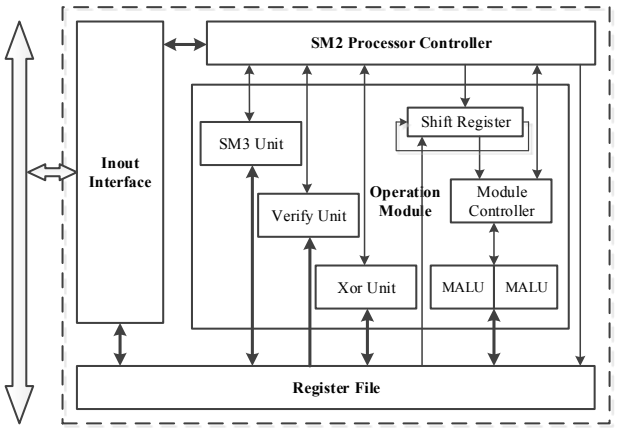


Fig 2. SM2 processor architecture

The Input/output Interface Module stores input data and output data respectively, judges input data, and produces the operation start signal according to the different functions, such as digital signature, verification, encryption and decryption.

The SM2 Processor Controller Module generates control signals according to the external commands and feedback signals, and controls operation unit and register file to perform various operations and choose the store data.

Resister File is made up of multiple sets of registers for storing and outputting intermediate calculation results.

The Operation Module integrates all operation units of the SM2 algorithm processor, including shift register unit, MALUs, SM3 unit, verify unit and XOR unit. The SM3 Unit can hash the data. The XOR unit performs XOR operation on data. These two units are used in the SM2 encryption and decryption algorithm. The verify unit is used to verify whether the data meets the requirements of the algorithm.

The Shift Register Unit shifts the random number K in the point multiplication according to the shift signal given by the SM2 processor controller. Under the control of SM2 processor controller and shift register unit, module controller could control MALUs to complete the point multiplication, the coordinate system transformation and subsequent modular operations.

3.2 Parallel arithmetic

In our design, point multiplication and various modular operations are executed by calling two MALUs in parallel. The modular controller controls two MALUs to take operands from different registers and complete different modular operations according to the data flow graph of PA, PD or subsequent modular operations.

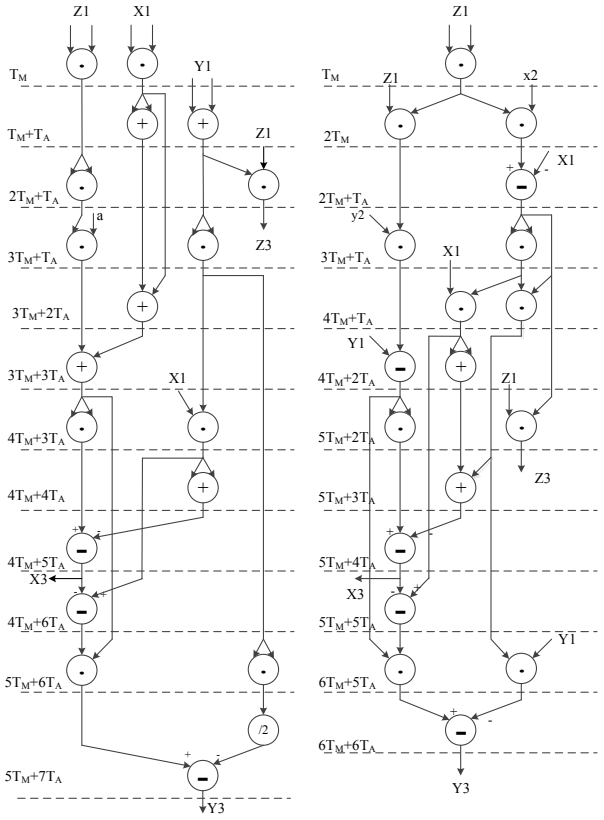


Fig 3. The data flow graph in GF(p)

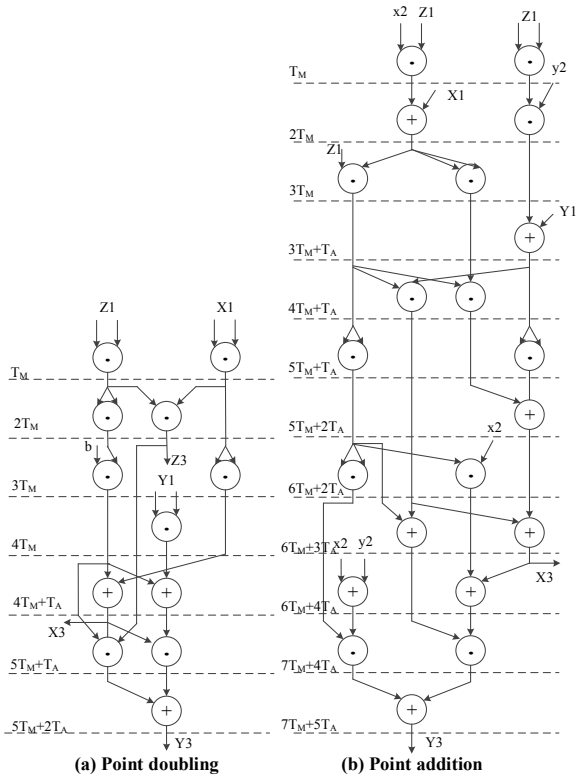
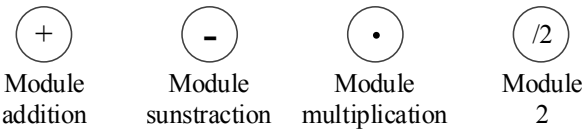


Fig 4. The data flow graph in GF(2^m)



T_M :The time of a modular multiplication operation

T_A :The time of a modular add-sub operation

Fig 5. The meaning of symbols

Under the Jacobian projected coordinates and LD projected coordinates, we finish PA and PD in GF(p) and GF(2^m) by using two MALUs, respectively. The data dependence graph for parallel computation of point operations in GF(p) and GF(2^m) are shown in Fig 3 and Fig 4. The different symbols in Fig 3 and 4 represent different operations, as shown in Fig 5. The source of the arrow represents the source of the operands, and the direction the arrow points to represents the direction of the calculation.

As depicted in Fig 4 and $5,5T_M+7T_A$ and $6T_M+6T_A$ are required to complete PD and PA in prime field, while $5T_M+2T_A$ and $7T_M+5T_A$ in binary field.

3.3 Modular arithmetic logic unit

The SM2 Elliptic Curve processor realizes point multiplication and other modular operations by calling two MALUs which can realize modular operations such as modular multiplication, modular division and modular add-sub in prime field and binary field, respectively to execute SM2 algorithm.

The structure of MALU is shown in Fig. 6. The proposed scheme can be divided into two parts: data path and control unit. The data path mainly includes four structures: $2A \bmod P$, $4A \bmod P$, $x+y+z \bmod P$ and $x^2 \bullet \mathrm{A(x)} \bmod \mathrm{F(x)}$. The $x+y+z \bmod P$ structure is also responsible for realizing the operations of $A \pm B \bmod p$ in modular add-sub and $B \pm A/4$ and $U \pm V/4 \bmod p$ in modular division. The results of N and O units, used in the modular division, are fed back to the MALU controller. The MALU controller produces the control signals, controls the data flow direction of the data path based on the finite state machine.

4. Experimental Result

This paper describes the proposed architecture by Verilog HDL at RTL-level, and uses VCS to simulate and verify the function. Our design is synthesized in CMOS 55 nm process. The results show that proposed SM2 Elliptic Curve Cryptograph processor occupies 245K gates and 0.47mm^2 core area. In addition to the modular operation unit, SM2 processor also includes a SM3 unit, an XOR unit, and a verify unit. Among them, the core area of SM3 unit is 0.04 mm^2, which costs the 18.9K gates. It could reach a high frequency of 435MHz. It only needs 0.56ms for a 256-bit point multiplication in GF(p), and 0.59ms in GF(2^m).

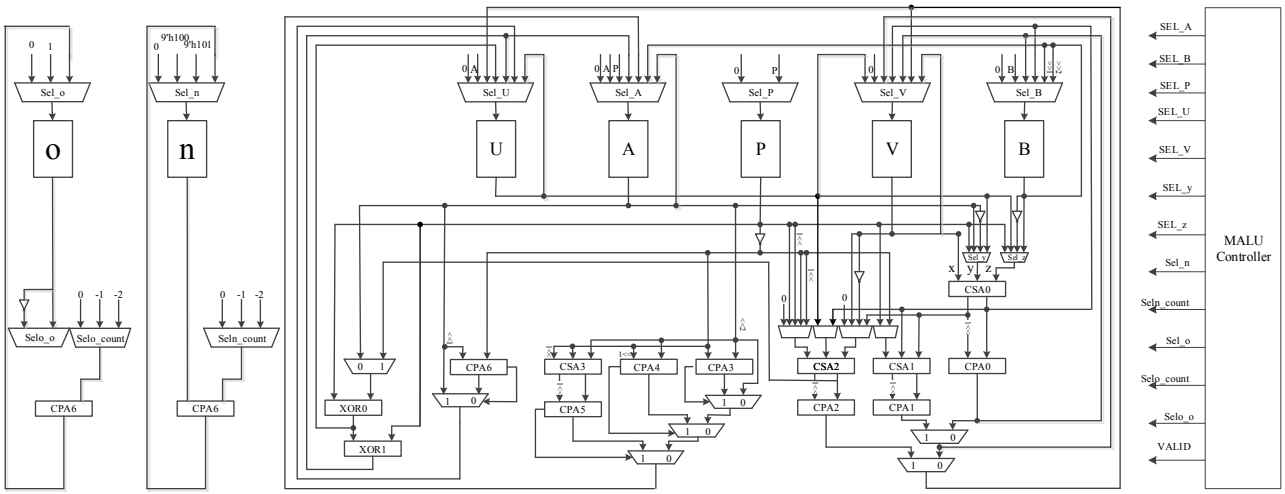


Fig 6. The arithmetic of MALU

Table 1 Comparison with Other Related work

Paper	Technology	Area(gates)	Core size(mm^2)	Field	Size(bits)	f_{max}(MHz)	Time(ms)
This work	55nm	245K	0.47	Dual field	256	476	0.56/0.59
[3]	55nm	189K	0.35	Dual field	571 256	316	6.75 1.45
[6]	90nm	170K	0.55	Dual field	256	147	4.4
[7]	90nm	168K	0.58	Dual field	256	256	1.89
[8]	0.13um	179K	1.35	Dual field	160	141.3/158.1	3.09/3.5
[9]	65 nm	447K	0.93	Prime field	256	546.5	0.73

In order to evaluate the performance of SM2 processors, the performance of this paper is compared with the related works, and the results are provided in Table 1. Compared with [3][6][7][8], our SM2 processor has a great advantage in speed, although it has a larger area. The speed of our design is similar to [9], but it supports dual-field operations and has a great advantage in area.

5. Summary

A dual-field SM2 elliptic curve cryptography processor proposed and implemented in this work has comprehensive advantage in speed and area, which could complete SM2 digital signature and verification, information encryption and decryption.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 61404175.

References

[1] Koblitz, Neal. "Elliptic curve cryptosystems." Mathematics of Computation 48.177:203-209(1987).

[2] Zhaohui, Wang, and Z. Zhenfeng. "Overview on Public Key Cryptographic Algorithm SM2 Based on Elliptic Curves." Journal of Information Security Research (2016).

[3] Zilong, Liu, D. Liu, and X. Zou. "An Efficient and Flexible Hardware Implementation of the Dual-Field Elliptic Curve Cryptographic Processor." IEEE Transactions on Industrial Electronics:1-1(2016).

[4] Hankerson, Darrel, A. J. Menezes, and S. A. Vanstone. "Guide to elliptic curve cryptography." (2004).

[5] Zhang, Dan, and G. Bai. "High-performance implementation of SM2 based on FPGA." 2016 8th IEEE International Conference on Communication Software and Networks (ICCSN) IEEE, (2016).

[6] Lee, Jen Wei, et al. "A 521-bit dual-field elliptic curve cryptographic processor with power analysis resistance." ESSCIRC, 2010 Proceedings of the IEEE, (2010).

[7] Lee, Jen Wei, et al. "An Efficient DPA Countermeasure With Randomized Montgomery Operations for DF-ECC Processor." IEEE Transactions on Circuits and Systems II: Express Briefs 59.5:0-5(2012).

[8] Lai, Jyu Yuan, and C. T. Huang. "Energy-Adaptive Dual-Field Processor for High-Performance Elliptic Curve Cryptographic Applications." IEEE Transactions on Very Large Scale Integration (VLSI) Systems 19.8:1512-1517(2011).

[9] Hossain, Md Selim, et al. "High-performance elliptic curve cryptography processor over NIST prime fields." IET Computers & Digital Techniques 11.1:33-42(2017).