Một giải pháp cứng hóa phép nhân điểm Elliptic trên trường GF(p)

Nguyễn Văn Long, Hoàng Văn Thức

***Tóm tắt*— Bài báo này mô tả thuật toán và cấu trúc mạch cho việc tính toán và thực thi phép tính nhân điểm đường cong Elliptic trên trường nguyên tố hữu hạn *GF(p)* có độ dài 256 bit. Cấu trúc mạch được mô tả bằng ngôn ngữ VHDL và được thực thi trên nền tảng chip Zynq xc7z030 và xc7z045.**

***Abstract*— This paper describles an algorithm and structure for computing and implementation point multiplications on Elliptic cuvers defined *GF(p)* with 256 bits length. The circuits have been described in VHDL in implemented on chip Zynq xc7z030 and xc7z045.**

***Từ khóa*— FPGA; Đường cong elliptic trên trường GF(p); nhân điểm.**

***Keywords—FPGA;* Elliptic cuvers over GF(p); Point multiplication.**

I. GIỚI THIỆU VÀ MÔ TẢ THUẬT TOÁN NHÂN ĐIỂM

Phép tính cơ bản và quan trọng nhất trong thuật toán mật mã elliptic là phép nhân một điểm trên đường cong elliptic với một số nguyên dương. Việc tính toán phép tính này phức tạp, tiêu tốn nhiều thời gian với các số lớn do các phép tính phải thực hiện modulo cho đa thức bất khả quy bậc cao. Thực thi cứng hóa phép nhân điểm trên FPGA giúp nâng cao tốc độ giảm thời gian thực hiện cho phép tính, đáp ứng được yêu cầu trong bài toán thực tế Trong nội dung bài báo chúng tôi trình bày, phân tích lựa chọn một số thuật toán dựa trên một số tài liệu khoa học và công trình nghiên cứu trên thế giới, để từ đó làm cơ sở cho việc nghiên cứu thiết kế cứng hóa phép nhân điểm trên hệ đường cong elliptic, được ứng dụng trong giao thức trao đổi khóa: ECDH, ECHMQV,...chữ ký số ECDSA [1][7], mã hóa ECIES [6].

Bài báo được nhận ngày 4/9/2018. Bài báo được nhận xét bởi phản biện thứ nhất vào ngày 28/10/2018 và được chấp nhận đăng vào ngày 8/11/2018. Bài báođược nhận xét bởi phản biện thứ hai vào ngày 10/11/2018 và được chấp nhận đăngvào ngày 21/11/2018.

Phép nhân điểm chính là việc thực hiện phép tính $k*P$, trong đó k là 1 số nguyên và P là một điểm trên đường cong elliptic E đã được định nghĩa trên trường GF(p) [2].

Thuật toán thực hiện phép tính nhân điểm như sau:

Thuật toán 1:

Đầu vào: $k = (k_{t-1},...,k_1,k_0)_2, P \in E(F_p)$

Đầu ra: kP

1. $Q \leftarrow \infty$
2. cho i chạy từ t-1 đến 0 thực hiện
   2.1 $Q \leftarrow 2Q$
   2.2 nếu k_i=1 thì $Q \leftarrow Q + P$
3. Trả về Q

Thuật toán 2:

Đầu vào: $k = (k_{t-1},...,k_1,k_0)_2, P \in E(F_p)$

Đầu ra: kP

1. $Q \leftarrow \infty$
2. cho i chạy từ 0 đến t-1 thực hiện
   2.1 Nếu k_i=1 thì $Q \leftarrow Q + P$
   2.2 $P \leftarrow 2P$
3. trả về Q

Đối với Thuật toán 1 [8], trong vòng lặp tại bước 2.1 và 2.2 đều cho ra một kết quả là giá trị Q. Kết quả đầu ra tại bước 2.1 làm giá trị đầu vào cho bước 2.2. Do vậy quá trình thực hiện phải tính toán nối tiếp kết thúc bước 2.1 rồi mới đến bước 2.2. Trong khi đó, ở Thuật toán 2, trong vòng lặp tại bước 2 kết quả phép tính bước 2.1 là Q và 2.2 là P, hai bước này được thực hiện tính toán độc lập không phụ thuộc nhau nên khi thực hiện ta có thể tính toán song song để nâng cao tốc độ thực thi phép tính. Trong bài báo này chúng tôi lựa chọn thuật toán 2 để thiết kế cứng hóa phép nhân điểm trên nền tảng phần cứng FPGA.

Trọng tâm trong hai thuật toán 1 và thuật toán 2 là vòng lặp sử dụng hai phép cộng điểm

và nhân đôi điểm. Hai phép tính này được thể hiện như sau:

Thuật toán nhân đôi điểm: với điểm $P(x_1, y_1) \in E(F_p), P \neq -P$ thì $2P = (x_3, y_3)$ được tính như sau:

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1$$

$$y_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)(x_1 - x_3) - y_1$$

Thuật toán cộng điểm:

$P = (x_1, y_1) \in E(F_p), Q = (x_2, y_2) \in E(F_p),$

$P \neq \pm Q$

Thì $P + Q = (x_3, y_3)$ được tính như sau:

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2$$

$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1$$

Trong phần tiếp theo của bài báo, chúng tôi cũng sẽ trình bày thuật toán và kiến trúc phần cứng của một số phép tính số lớn phục vụ cho việc cứng hóa phép nhân điểm trong Mục II và Mục III. Cụ thể được trình bày ở các mục dưới đây. Mục IV là Kết quả thực hiện và cuối cùng là Mục Kết luận.

II. THIẾT KẾ CỨNG HÓA CÁC PHÉP TÍNH SỐ LỚN CƠ SỞ

A. Thiết kế cứng hóa phép cộng số lớn trên trường GF(p)

Cho hai số tự nhiên *x,y*: $x, y \in \mathbb{Z}_p \{0,1,..., p-1\}$. Ta cần tính toán giá trị tổng của *x* và *y* trên $\mathbb{Z}_p$ như sau: $z = (x + y) \bmod p$. Ta có $0 \leq x + y < 2p$ do đó z phải bằng một trong hai giá trị sau x+y hoặc x+y – p. Từ đây ta xây dựng thuật toán để tính toán z như sau:

Thuật toán tính phép cộng trên GF(p):

z1 := x + y;

z2 := (z1 mod 2^k) + (2^k-p);

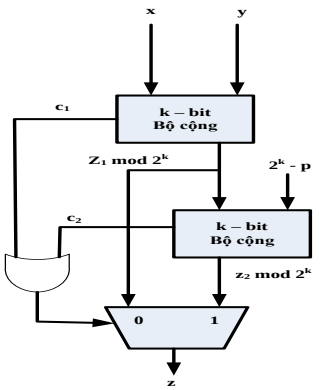
c1 := z1/2^k;

c2 := z2/2^k;

Nếu c1 = 0 và c2 = 0 thì z := z1 mod 2^k;

Không thì z := z2 mod 2^k;

Kết thúc.



Hình 1. Cấu trúc phép cộng số lớn trên GF(p)

B. Thiết kế cứng hóa phép trừ số lớn trên trường GF(p)

Cho hai số tự nhiên *x, y*: $x, y \in \mathbb{Z}_p \{0,1,..., p-1\}$. Ta cần tính toán giá trị tổng của *x* và *y* trên $\mathbb{Z}_p$ như sau: $z = (x - y) \bmod p$. Ta có $-p \leq x - y < p$ do đó z phải bằng một trong hai giá trị sau *x - y* hoặc *x - y + p*. Từ đây ta xây dựng thuật toán để tính toán *z* như sau:

Thuật toán tính phép trừ trên GF(p).

Tổng := x + (2^k-y);

z1 := Tổng mod 2^k;

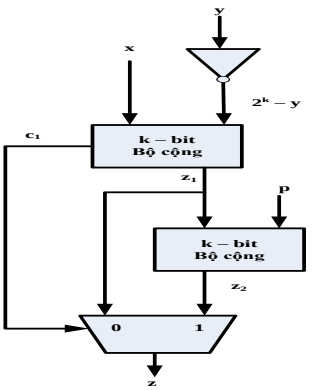
z2 := z1 + p mod 2^k;

c1 := Tổng/2^k;

Nếu c1 = 0 thì z := z1;

Không thì z := z2;

Kết thúc.



Hình 2. Cấu trúc phép trừ số lớn trên GF(p)

C. Thiết kế cứng hóa phép nhân số lớn trên trường GF(p)

Phép nhân trên trường GF(p) được định nghĩa như sau:

$$C = a.b \bmod p,\ 0 \le a,b < p$$

Để thực hiện cứng hóa phép nhân đa thức trên trường GF(p) cần thực hiện hai bước, bước thứ nhất tính toán phép nhân giữa hai số *a* và *b*, bước thứ hai tính phép modulo kết quả phép nhân với *p*.

Có nhiều thuật toán khác nhau sử dụng để thực thi phép nhân trên trường GF(p), trong số đó có thuật toán thích hợp cho phần cứng, phần mềm hoặc phần sụn... Các thuật toán sử dụng cho phần cứng cần thiết kế sao cho quá trình nhân chỉ sử dụng các phần tử cơ bản trong phần cứng là AND, XOR,... thanh ghi, MUX... tương ứng với các phần tử cơ bản trong FPGA là CLBs và LUTs. Với tiêu chí này trên thế giới đã có nhiều bài báo công bố các giải pháp khác nhau để thực hiện thuật toán nhân trên trường GF(p), nhưng có thể tóm lại trong một số phương pháp chính sau:

- Phương pháp nhân rồi chia (multiply and then divide)

- Phương pháp nhân đan xen (Interleaving)

- Phương pháp nhân Montgomery (*nhân Montgomery*). Hiện tại, chúng tôi thực hiện cứng hóa phép nhân theo phương pháp nhân đan xen, phương pháp này dễ thực hiện trên nền tảng phần cứng chỉ sử dụng phép cộng modulo và phép nhân 2. Trong thời gian sắp tới chúng tôi sẽ nghiên cứu về hai phương pháp còn lại. Phương pháp nhân đan xen sẽ được làm rõ hơn trong thuật toán nhân đan xen trích dẫn trong tài liệu [4], [5]. Thuật toán nhân đan xen được trình bày như sau:

Cho hai số tự nhiên *x*, *y*: $x, y \in \mathbb{Z}_p = \{0,1,..., p-1\}$. Ta cần tính toán giá trị tích của *x* và *y* trên $\mathbb{Z}_p$ như sau: $z = x.y \bmod p$. Ta có

$$x.y = \left(x_{k-1}.2^{k-1} + x_{k-2}.2^{k-2} + ... + x_0.2^0\right)y$$

$$= (...(0.2 + x_{k-1}y)2 + x_{k-2}y)2 + ... + x_1y)2 + x_0y$$

với cách tính như trên, tính toán phép nhân có thể sử dụng bằng phép cộng và nhân đôi sau đó thực hiện phép rút gọn (mod p) ta sẽ được kết quả phép nhân trên trường GF(p), $z = x.y \bmod p$

Thuật toán tính phép nhân trên GF(p):

r := 0;

với i in 0 to k-1 lặp:

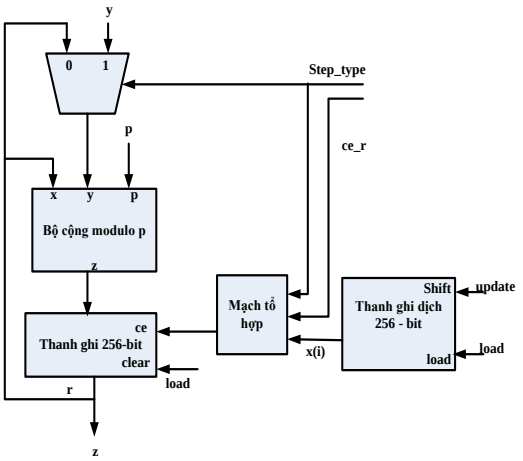
r := (r +r) mod p;

if x(k-i-1)=1 thì r := r + y mod p;

kết thúc nếu;

kết thúc lặp;

kết quả := r;



Hình 3. Cấu trúc phép nhân số lớn trên GF(p)

D. Thiết kế cứng hóa phép chia/nghịch đảo trên trường GF(p)

Phép tính nghịch đảo là trường hợp riêng của phép chia khi *a*/*b* với *a* = 1. Ta xét trường hợp tổng quát, cho hai số tự nhiên *a*, *b*: $a,b \in \mathbb{Z}_p \{0,1,..., p-1\}$. Ta cần tính toán giá trị thương của *a* và *b* trên $\mathbb{Z}_p$ như sau: $z = a / b \bmod p$. (1)

Để tính toán biểu thức (1) có nhiều thuật toán khác nhau (*thuật toán Euclidean thuật toán nhị phân, thuật toán cộng-trừ và thuật toán tính nghịch đảo theo định lý Fermat's Little*) trong phần này chúng tôi lựa chọn thuật toán nhị phân để thiết kế module nghịch đảo trên trường GF(*p*).

Cho hai số tự nhiên *a*, *b*: $a,b \in \mathbb{Z}_p \{0,1,..., p-1\}$

- Nếu cả hai số đều chẵn, khi đó ta có: GCD(*a*,*b*) = 2.GCD(*a/2*, *b/2*)

- Nếu chỉ có một số chẵn, giả sử b chẵn thì GCD(*a*, *b*) = GCD(*a*, *b/2*).

- Nếu không có số nào chẵn và giả sử $a \ge b$

thì khi đó GCD(*a*,*b*) = GCD(*a, a -b*) và khi đó *a – b* là số chẵn.

Lặp lại quá trình trên sau một số hữu hạn *m* bước ta sẽ tìm được một số xác định GCD(*a* ,*b*) = GCD(a_p , *0*). Từ đây ta có thể xây dựng thuật toán tính $z = \frac{a}{b} \bmod p$ như sau:

Thuật toán nhị phân tính nghịch đảo trên GF(p):

a := p; b := y; c := 0; d := x;

Trong khi a > 1 lặp

Trong khi (b mod 2) = 0 lặp

b := b/2; d := Divide_By_2(d, P);

Kết thúc lặp;

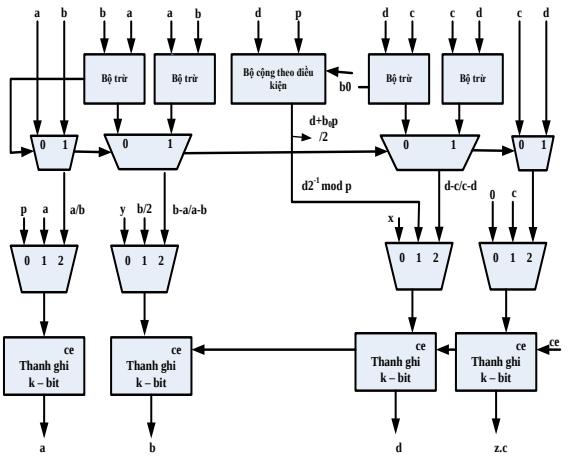
Nếu b >= a thì b := b-a; d := (d-c) mod P; Nếu không thì

Old_b := b; b := a-b; a := Old_b;

Old_d := d; d := (c-d) mod P; c := Old_d; Kết thúc nếu;

Kết thúc lặp;

Z := c;



Hình 4. Cấu trúc phép chia/nghịch đảo theo thuật toán nhị phân trên trường GF(p)

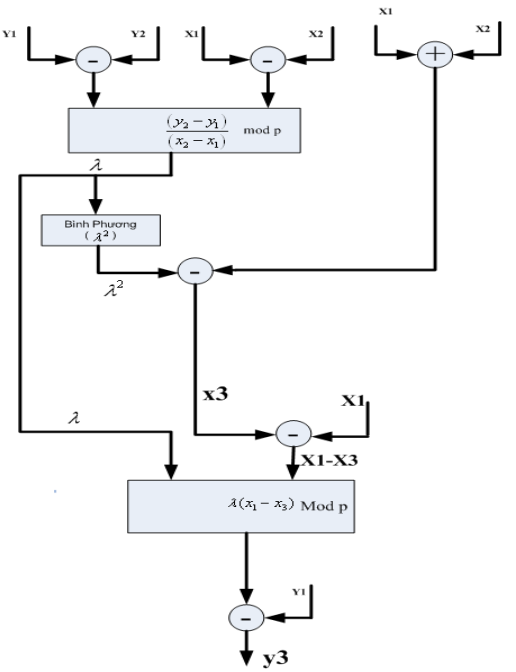
III. THIẾT KẾ CỨNG HÓA PHÉP NHÂN ĐIỂM ELLIPTIC TRÊN TRƯỜNG *GF(p)*

A. Thiết kế cứng hóa phép cộng điểm Elliptic trên trường GF(p)

Phép cộng điểm elliptic được định nghĩa như ở phần trên khi đó tọa độ điểm $R = P + Q$ được xác định như sau:

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1 \text{ với } \lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

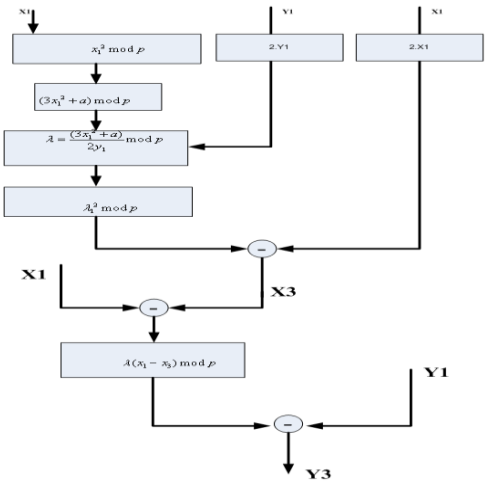


Hình 5. Cấu trúc phép cộng điểm elliptic trên trường GF(p)

B. Thiết kế cứng hóa phép nhân đôi điểm Elliptic trên trường GF(p)

Phép nhân đôi elliptic được định nghĩa như ở phần trên khi đó tọa độ điểm $R = 2P$ được xác định như sau: $x_3 = \lambda^2 - 2x_1$;

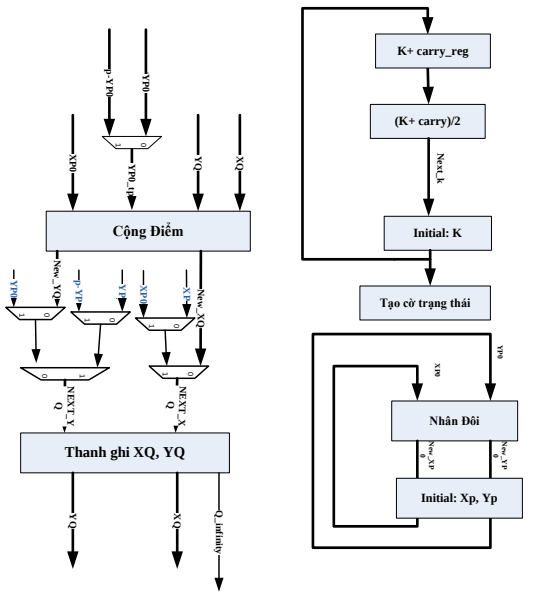
$$y_3 = \lambda(x_1 - x_3) - y_1 \text{ với } \lambda = \frac{3x^2_1 + a}{2y_1}$$



Hình 6. Cấu trúc phép nhân đôi điểm elliptic trên trường GF(p)

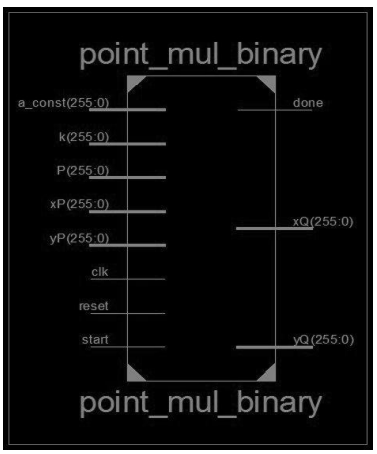
C. Thiết kế cứng hóa phép nhân điểm Elliptic trên trường GF(p)

Việc cứng hóa phép nhân điểm elliptic trên trường GF(p) là quá trình ghép nối các module đã thiết kế cứng hóa ở các mục trên, cấu trúc module thực hiện phép nhân điểm elliptic trên GF(p) như sau:



Hình 7. Cấu trúc phép nhân điểm elliptic trên trường GF(p)

Giao diện module thực hiện phép nhân điểm elliptic trên trường GF(p) dựa trên công nghệ FPGA:



Hình 8. Giao diện module phép nhân điểm elliptic trên trường GF(p)

IV. KẾT QUẢ THỰC HIỆN

Module phép nhân điểm elliptic trên trường GF(p) đã được chúng tôi tổng hợp trên 02 nền chip xc7z030 và xc7z045 và đang được ứng dụng trong đề tài nhà nước "Nghiên cứu thiết kế, chế tạo module bảo mật phần cứng HSM, ứng dụng trong các hệ thống bảo mật và xác thực thông tin". Dưới đây là kết quả tổng hợp phép nhân điểm trên 02 nền tảng phần cứng khác nhau

Bảng 1. Kết quả cài đặt thuật toán nhân điểm trên nền tảng phần cứng FPGA

Thuật toán nhân điểm	Chip FPGA	Chiều dài chuỗi bit	Tần số hoạt động (MHz)	Tài nguyên thiết kế (LUTs)
Thuật toán 1	Xc7z030	256	136.1	34472
	Xc7z045		157.3	34486

V. KẾT LUẬN

Trong nội dung bài báo nhóm tác giả trình bày giới thiệu, phân tích, lựa chọn thuật toán nhân điểm và một số thuật toán thực hiện các phép tính cơ sở trên hệ mật đường cong elliptic. Để từ đó làm cơ sở cho việc thiết kế cứng hóa phép tính cơ sở trong mục II và phép nhân điểm trong mục III. Triển khai các phép toán trên FPGA bằng ngôn ngữ mô tả phần cứng HDL VHDL. Đưa ra kết quả tổng hợp về tài nguyên thiết kế, tần số hoạt động của phép tính trên 02 chip FPGA xc7z030 và xc7z045. Kết quả phép tính hoạt động ổn định, đáp ứng được yêu cầu đề ra, được ứng dụng trong đề tài nhà nước "Nghiên cứu thiết kế, chế tạo module bảo mật phần cứng HSM, ứng dụng trong các hệ thống bảo mật và xác thực thông tin".

TÀI LIỆU THAM KHẢO

[1]. American Bankers Association. ANSI X9.62-1998: Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA).

[2]. N. Koblitz, S. Vastone, and A. Menezes. The State of Elliptic Curve Cryptography, *Design, Codes and Cryptography,* 19(2/3):173-193, March 2000.

[3]. J. Lutz. High Performance Elliptic Curve Cryptographic co-processor. Master's thesis, University of Waterloo, 2003.

[4]. Đề tài cấp Ban "Nghiên cứu thiết kế, chế tạo module bảo mật cài đặt an toàn, cứng hóa các thuật toán GOST (28147-89, R34.11-2012, R34.10-2012) dựa trên công nghệ FPGA". Ban Cơ yếu Chính phủ, Thực hiện 2015- 2016. Chủ nhiệm Nguyễn Biên Cương.

[5]. SEC1. Elliptic Curve Cryptography: Standards for Eficient Cryptography Group, http://www.secg.org

[6]. TC03-2:2015, "*Thuật toán chữ ký số ECDSA*", Ban cơ yếu Chính phủ.

[7]. The FIPS 186-3 Elliptic Curve Digital Signature Algorithm Validation System (ECDSA2VS), *January 17, 2012.*

[8]. Cryptographic Algorithms on Reconfigurable Hardware, Springer.

SƠ LƯỢC VỀ TÁC GIẢ

KS. Nguyễn Văn Long

Đơn vị công tác: Viện Khoa học – Công nghệ mật mã, Ban Cơ yếu chính phủ

Email: longyenkk2@gmail.com

Quá trình đào tạo: Nhận bằng kỹ sư năm 2014.

Hướng nghiên cứu hiện nay: Công nghệ vi mạch, FPGA, công nghệ nhúng Linux.

TS. Hoàng Văn Thức

Đơn vị công tác: Viện Khoa học - Công nghệ mật mã, Ban Cơ yếu Chính phủ.

Email: thuchv@yahoo.com

Quá trình đào tạo: Nhận bằng kỹ sư năm 1998 và Thạc sĩ năm 2004 chuyên ngành Kỹ thuật mật mã, Học viện Kỹ thuật mật mã. Nhận bằng Tiến sĩ Toán học, Viện Khoa học - Công nghệ quân sự năm 2012.

Hướng nghiên cứu hiện nay: Khoa học - Công nghệ Mật mã.