College of Engineering & Computer Science
Florida Atlantic University

Time-Efficient Finite Field Microarchitecture Design for Curve448 and Ed448 on Cortex-M4

Mila Anastasova[1], Reza Azarderakhsh[1], Mehran Mozaffari Kermani[2], and Lubjana Beshaj[3]

Computer and Electrical Engineering and Computer Science Department and I-SENSE at Florida Atlantic University, Boca Raton, FL, USA
(manastasova2017, razarderakhsh)@fau.edu

Computer Engineering and Science Department at University of South Florida, Tampa, FL, USA,
mehran2@usf.edu

United States Military Academy West Point, West Point, NY, USA,
lubjana.beshaj@westpoint.edu

Abstract & Introduction

Public Key Cryptography (PKC) protects data confidentiality when transmitting it through an unsecured channel, such as the Internet. We present an efficient design for both Curve448 Elliptic Curve Diffie-Hellman (ECDH) and Ed448 Edwards-curves Digital Signature Algorithm (EdDSA) algorithms:

- We implement ARMv7 highly optimized low-level finite field arithmetic.
- Our design outperforms previous ECDH implementations by more than 48%.
- Our Ed448 DSA shows a speedup of 11%, requiring 6 and 7.4 MCCs for sign and verify.

Curve448 & Ed448

A Montgomery Elliptic Curve Curve448 over a finite field $\mathbb{F}_p$ is defined by:

$$E_M/\mathbb{F}_p : v^2 \equiv u^3 + Au^2 + u$$

where the value of A is defined as 156326 and $p = 2^{448} - 2^{224} - 1$. Montgomery curves have their birationally analogue Edwards curves, where Curve448 can be represented by the solutions to the equation:

$$E_{Ed}/\mathbb{F}_p : ax^2 + y^2 = 1 + dx^2y^2$$

with $d = -39081$ and $a = 1$. The core of Curve448 and Ed448 is the scalar-point multiplication where $P = [k] \cdot Q$ is the addition of point Q to itself k times.

ARMv7-M Architecture

The ARM Cortex-M4 processor's architecture delivers a set of powerful instructions that are devoid of structural hazards.

Table 1. ARMv7-M ISA for memory access and MAC instructions

Instruction	Functionality	Latency (CC)
(V)LDR/ (V)STR	$R_n \leftarrow$ memory memory $\leftarrow R_n$ $S_n \leftarrow$ memory memory $\leftarrow S_n$	2
VMOV	$R_n \leftarrow S_m$ $S_m \leftarrow R_n$	1
UMULL	$Rd_1, Rd_2 \leftarrow R_n \times R_m$	1
UMAAL	$Rd_1, Rd_2 \leftarrow R_n \times R_m + Rd_1 + Rd_2$	1

Proposed Design for Field Arithmetic

We present a novel technique for multi-precision multiplication and squaring, with an emphasis on increasing row (inner loop) size and hence decreasing memory accesses for partial value accumulation.

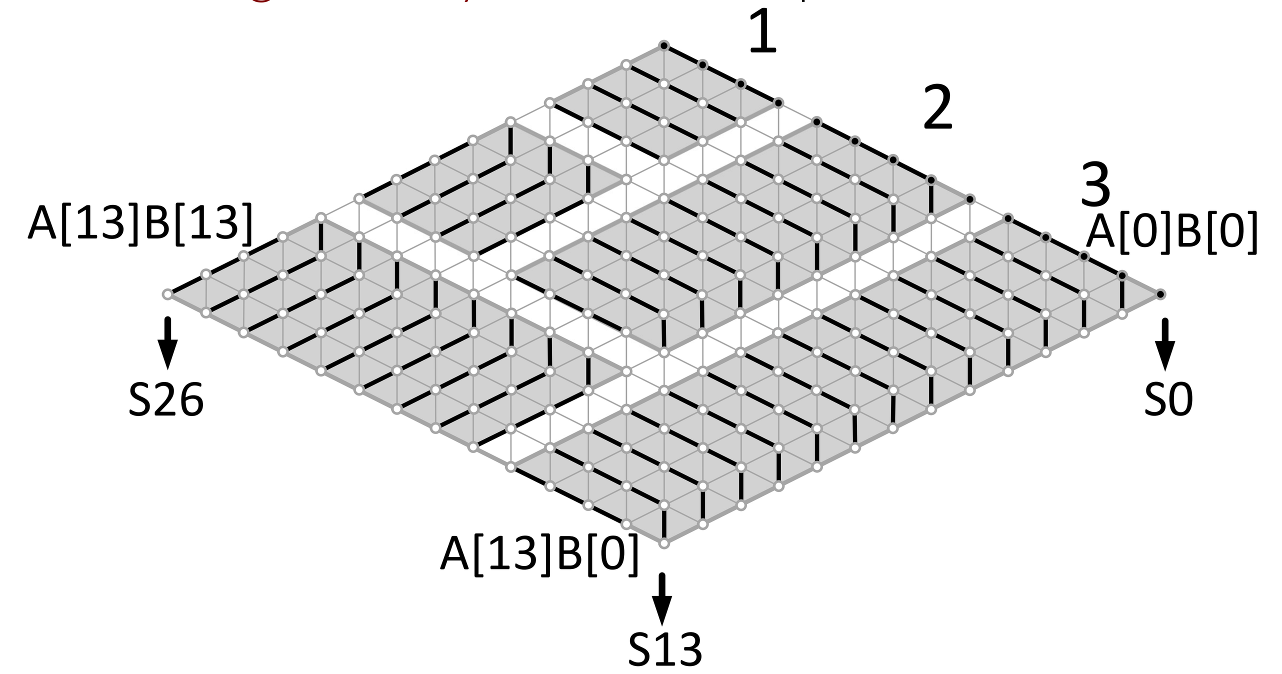

Figure 1. Proposed architecture for 448-bit multi-precision multiplication. Black lines denote inner loop execution.

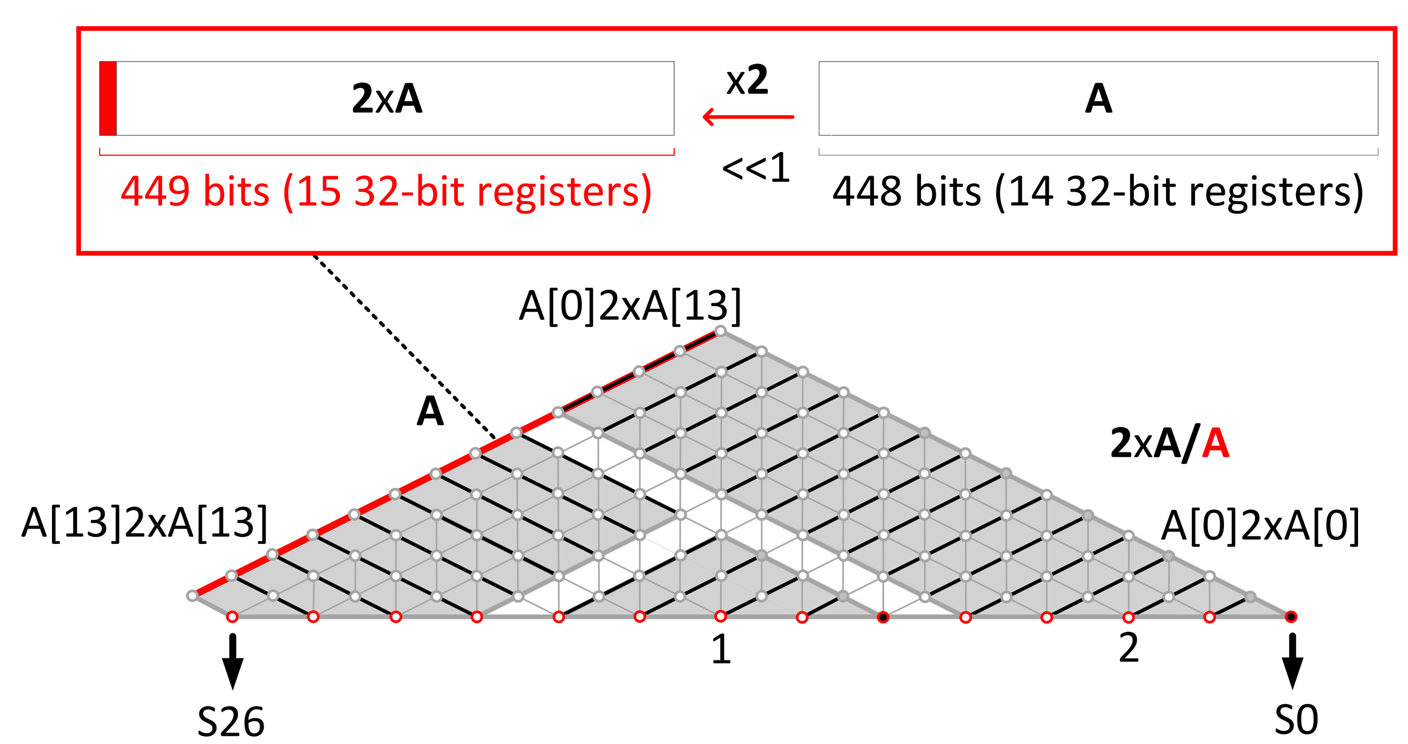

Figure 2. Proposed architecture for 448-bit multi-precision squaring. Red line denotes additional simulated lane for increased word size of the doubled operand.

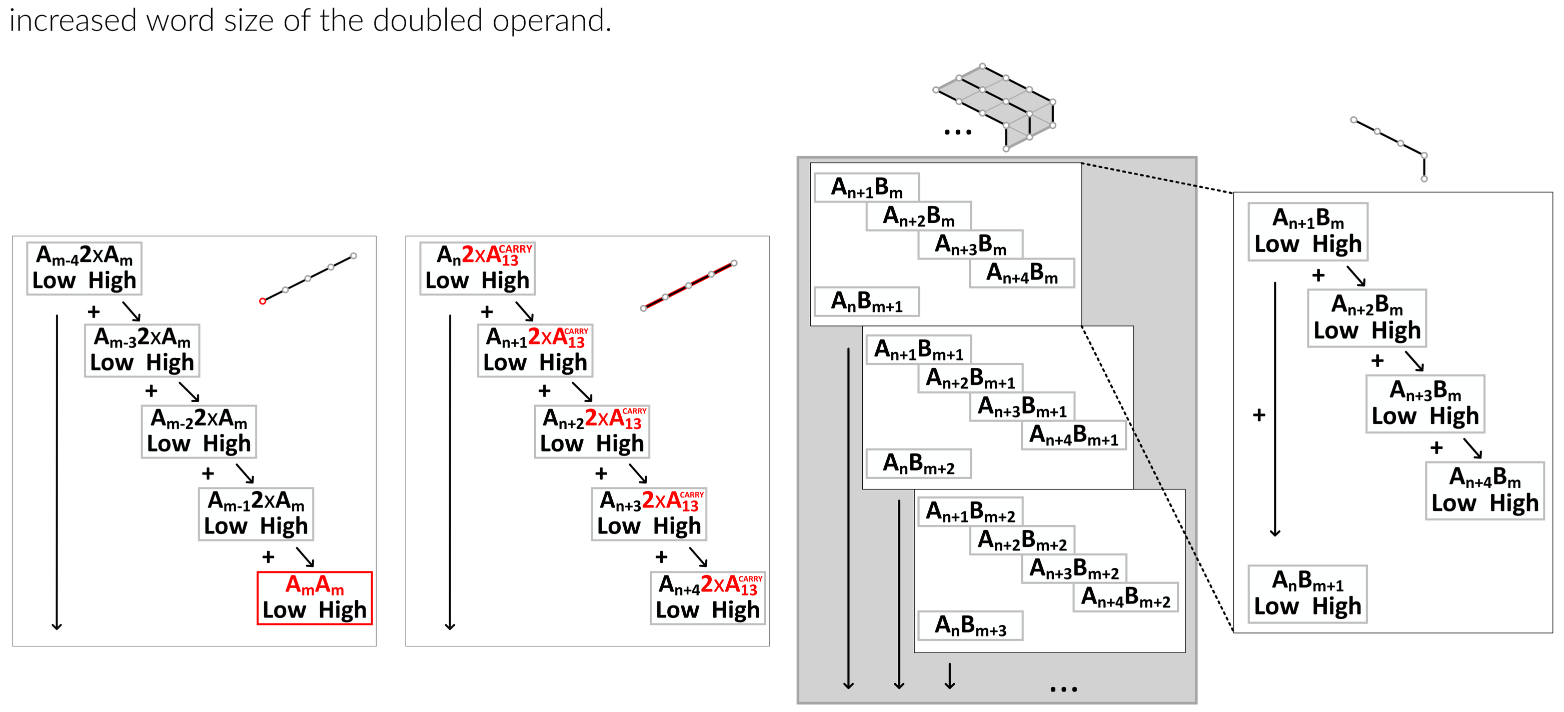

Figure 3. Proposed design for the inner execution loop for multi-precision multiplication and squaring showing.

Performance Evaluation & Conclusions

We compare our work with the best-known counterparts in the literature targeting the same platform for Curve448- and Ed448-based algorithms and we present the latency results in number number of clock cycles in Tables 2 and 3. We mark around 48.2% and 36.8% of speedup for X448 @24MHz and @168MHz, respectively. We report a speedup of 13.1%, 8.1%, and 12.4% for Ed448 EdDSA.

Table 2. Finite field operations for Curve448/Ed448 targeting ARMv7-M

Ref.	Arithmetic Performance Evaluation					
	Fp			Group		
	Mul	Sqr	Inv	Add	Double	Multiply
Curve448						
Seo et al. [4]	821	821	363,485	6,566	6,567	6,218,135
This work	**613** 25.33%	**532** 35.20%	**247,707** 31.85%	**6,640(total)** 49.44%		**3,220,682** 48.21%
Ed448						
Anastasova et al. [1]	705	705	325,997	8,465(total)		3,703,755
This work	**613** 13.05%	**532** 24.54%	**247,934** 23.95%	**7,323(total)** 13.49%		**3,259,379** 12.00%

Table 3. Curve 25519 and Curve448 key exchange and digital signature computational latency on IoT platforms

Work	Platform	Freq. [MHz]	X448	Ed448 KeyGen	Ed448 Sign	Ed448 Verify
Curve25519 [2]	Cortex-M4	84	894	390	544	1,331
Curve448 [3]	AVR	32	103,229	-	-	-
	MSP	25	73,478	-	-	-
Curve448 [4]	Cortex-M4	24	6,218	-	-	-
		168	6,286	-	-	-
Ed448 [1]	Cortex-M4	24	-	4,069	6,571	8,452
		168	-	4,195	6,699	8,659
This work	Cortex-M4	24	**3,221**	**3,536**	**6,038**	**7,404**
		168	**3,975**	**4,282**	**6,787**	**8,854**

In this work, we present a novel design for time-efficient finite field arithmetic over Curve448 and Ed448.

References

[1] Mila Anastasova, Mojtaba Bisheh-Niasar, Hwajeong Seo, Reza Azarderakhsh, and Mehran Mozaffari Kermani. Efficient and Side-Channel Resistant Design of High-Security Ed448 on ARM Cortex-M4. In *2022 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 93–96. IEEE, 2022.

[2] Hayato Fujii and Diego F Aranha. Curve25519 for the Cortex-M4 and beyond. In *International Conference on Cryptology and Information Security in Latin America*, pages 109–127. Springer, 2017.

[3] Hwajeong Seo. Compact implementations of Curve Ed448 on low-end IoT platforms. *ETRI Journal*, 41(6):863–872, 2019.

[4] Hwajeong Seo and Reza Azarderakhsh. Curve448 on 32-bit ARM Cortex-M4. In *International Conference on Information Security and Cryptology*, pages 125–139. Springer, 2020.