Faster F_p-arithmetic for Cryptographic Pairings on Barreto-Naehrig Curves

Junfeng Fan, Frederik Vercauteren and Ingrid Verbauwhede
Katholieke Universiteit Leuven, COSIC

Sep 8, 2009

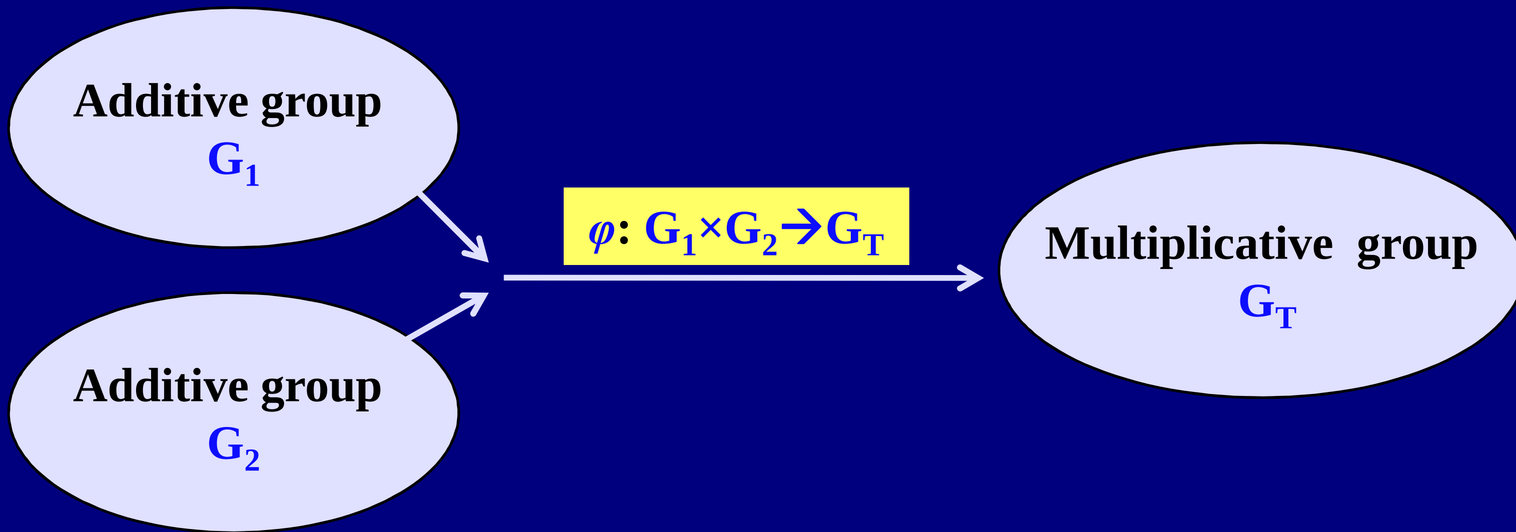
Outline

- Bilinear pairing
- Barreto-Naehrig (BN) curves
- Fast multiplication in F_p
- Hardware implementation
- Conclusion

Outline

- **Bilinear pairing**
- Barreto-Naehrig (BN) curves
- Fast multiplication in Fp
- Hardware implementation
- Conclusion

Bilinear Pairing



- **Bilinear**

  $P \in G_1$, $Q \in G_2$, then $\varphi(aP, bQ) = \varphi(P,Q)^{ab}$.

- **Non-degenerate**

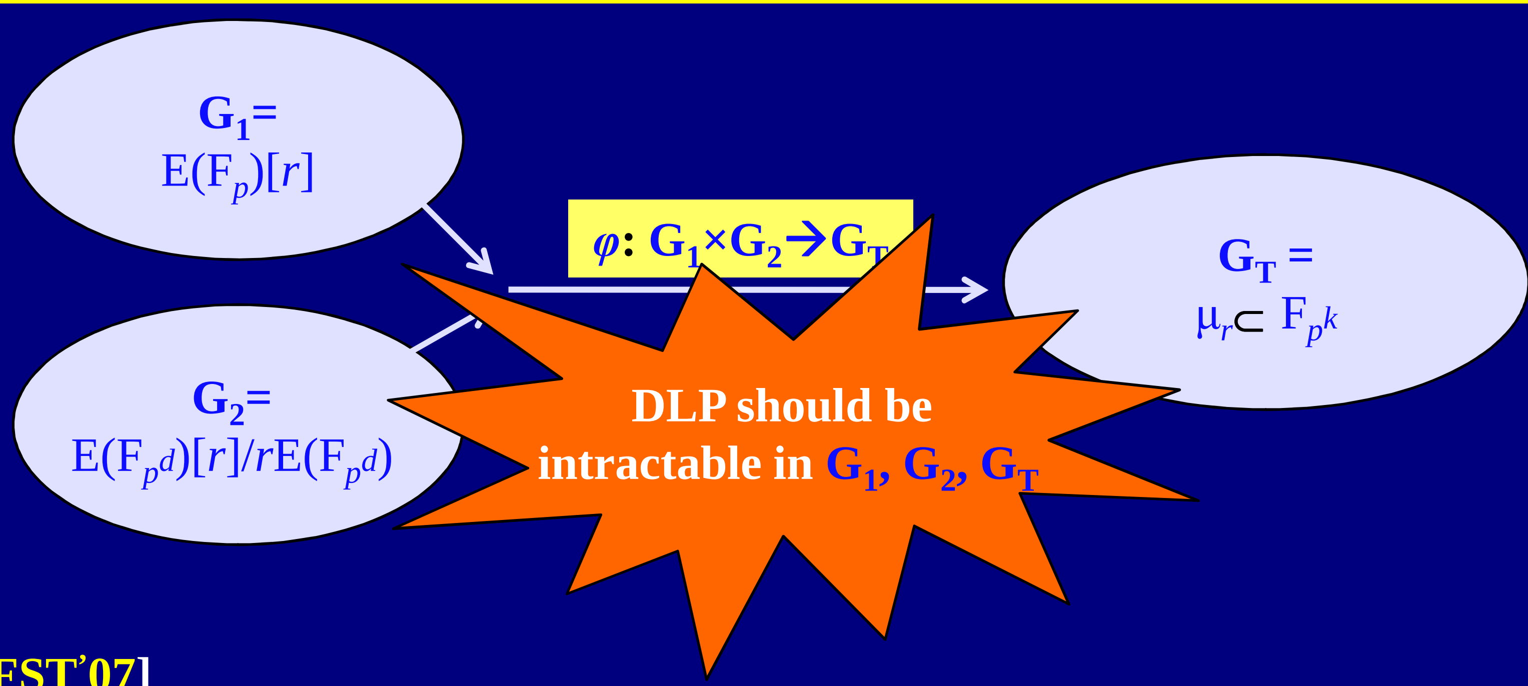
  $P \in G_1 \backslash 0, \exists\ Q \in G_2$, such that $\varphi(P,Q) \neq 1$.

- **Computable**

Application

- One-round three-way key exchange [Joux'00]
- Identity-based encryption [Sakai+01, Boneh+01]
- Identity-based signature [Cha+03, Paterson'02]
- Short signature [Boneh+01]
- …

Tate pairing



[FST'07]

Security level (in bits)	Subgroup size r (in bits)	Extension field size q^k (in bits)	Embedding Degree k	
			$\rho \approx 1$	$\rho \approx 2$
80	160	960 -1280	6 - 8	2,3 - 4
112	224	2200 - 3600	10 - 16	5 - 8
128	256	3000 - 5000	12 - 20	6 -10
192	384	8000 – 10000	20 - 26	10 -13
256	512	14000 - 18000	28 - 36	14 - 18

Barreto-Naehrig Curves

- Elliptic curve

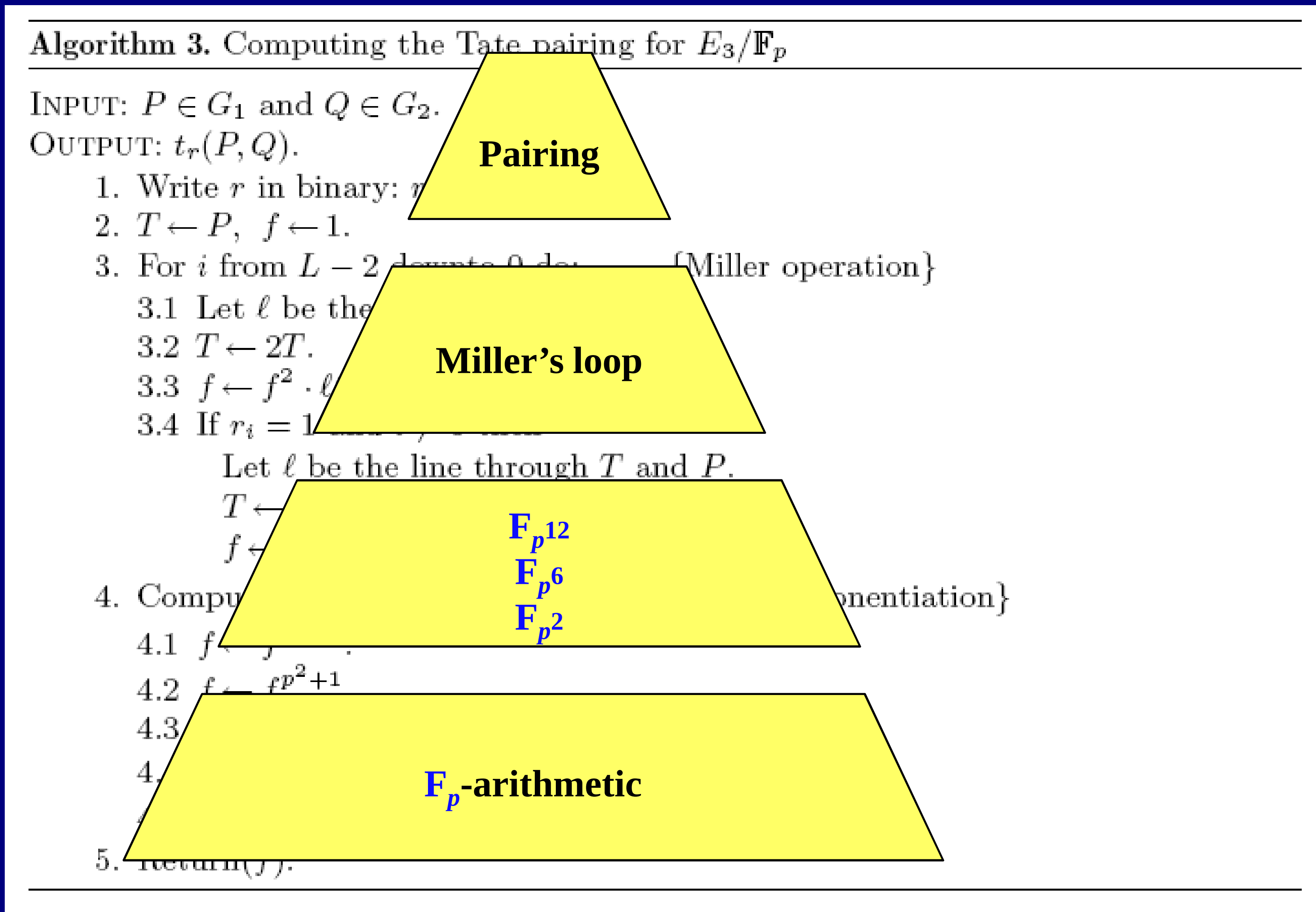
$$E : y^2 = x^3 + b \text{ over } F_p,$$

where

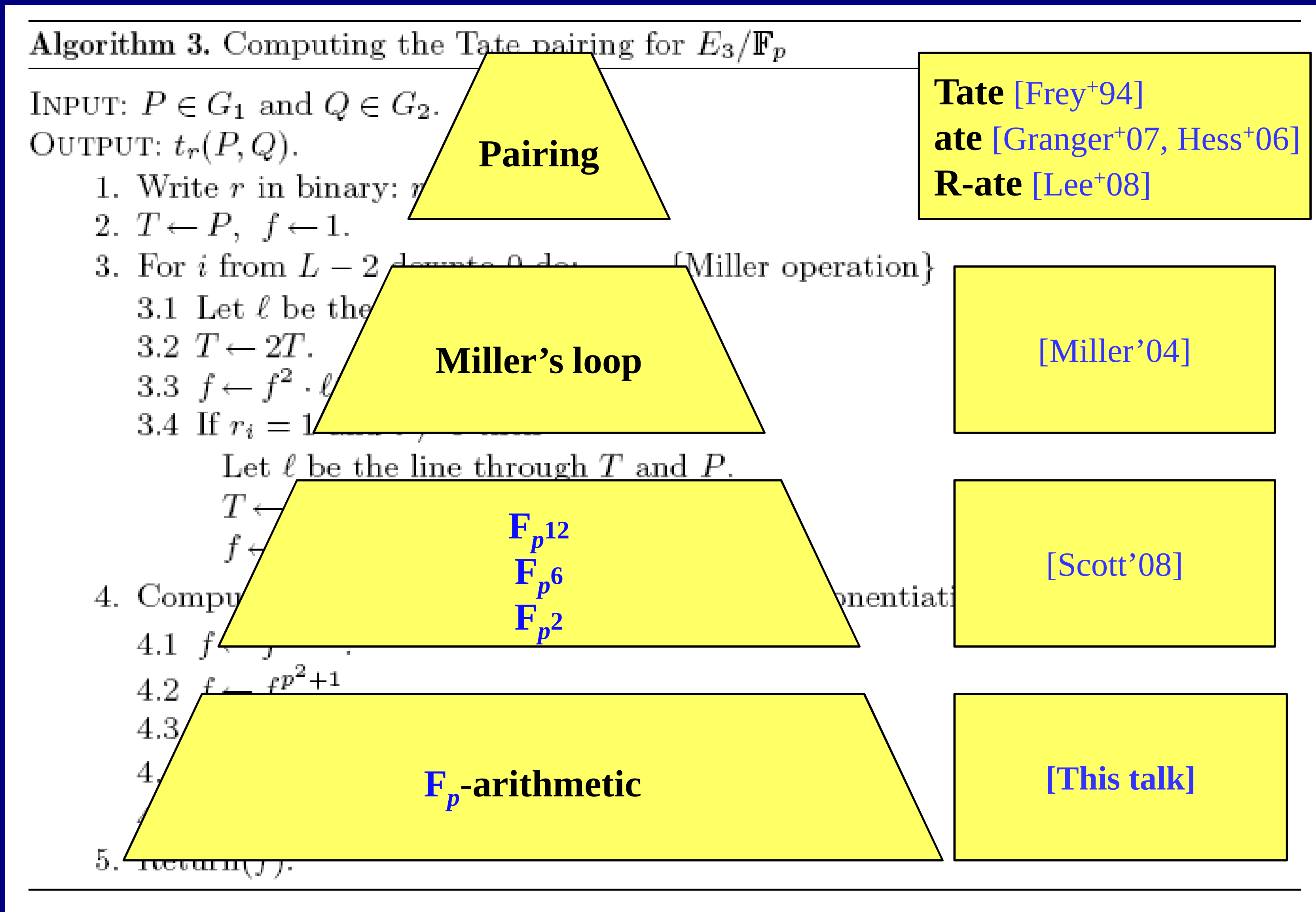
$$p(z) = 36z^4 + 36z^3 + 24z^2 + 6z + 1,$$
$$r(z) = 36z^4 + 36z^3 + 18z^2 + 6z + 1.$$

- Some nice features:
  - $r = \#E(F_p)$
  - DLPs in G_1 and G_T are almost equally hard (128-bit security)

Pairing computation



Pairing computation



Outline

- Bilinear pairing
- Barreto-Naehrig (BN) curves
- **Fast multiplication in F_p**
- Hardware implementation
- Conclusion

Modular multiplication

- Target: Compute "*ab* mod *p* "
- Fast reduction method
  - Use pseudo-Mersenne number
    - $p = 2^m - s$, where *s* is *small.*
  - Montgomery
  - Barrett
  - Chung-Hasan
    - If $p=f(t)$ and $f(t)$ is monic, then $c(t)/f(t)$ is efficient.

Montgomery method

- Given $p < 2^m$ and $a,b < p$, output $ab2^{-m}$ mod p.
  - $p' = -p^{-1}$ mod 2^m [precomputed]
  - 1: $c = ab$ — ***m*-bit multiplication**
  - 2: $\mu = c$ mod 2^m
  - 3: $q = \mu p'$ mod 2^m — ***m*-bit multiplication**
  - 4: $r = (c + qp) / 2^m$ — ***m*-bit multiplication**
  - 5: $r = r - p$ if $r > p$
  - Return r

What is special for BN Curves?

- E : $y^2 = x^3 + b$ over F_p, where

$$p=p(z) = 36z^4 + 36z^3 + 24z^2 + 6z + 1.$$

- Some observations on p:
  - Can not be pseudo-Mersenne number ☹
- However,
  - $p(z)$ has small coefficients
  - $p^{-1}(z) = -324z^4+36z^3+12z^2-6z+1$ mod z^5
  - $p^{-1}(z) = 1$ mod z

Montgomery multiplication

In integer ring

- Given p, a and b,

  output $ab2^{-m} \bmod p$.

  - Precompute $p' = -p^{-1} \bmod 2^m$
  - 1: $c = ab$
  - 2: $\mu = c \bmod 2^m$
  - 3: $q = \mu p' \bmod 2^m$
  - 4: $r = (c+qp) / 2^m$
  - 5: $r = r-p$ if $r>p$

In polynomial ring

- $p(z) = 36z^4+36z^3+24z^2 + 6z + 1$, $a(z)$ and $b(z)$,

  output $a(z)b(z)z^{-5} \bmod p(z)$.

  - $p'(z) = 324z^4-36z^3-12z^2+6z-1$
  - 1: $c(z) = a(z)b(z)$
  - 2: $\mu(z) = c(z) \bmod z^5$
  - 3: $q(z) = - p'(z)\mu(z) \bmod z^5$
  - 4: $r(z) = (c(z)+ q(z)p(z)) / z^5$

Montgomery multiplication

In integer ring

- Given p, a and b,

  output $ab2^{-m} \bmod p$.

- Precompute $p' = -p^{-1} \bmod 2^m$
- 1: $c = ab$
- 2: $\mu = c \bmod 2^m$
- 3: $q = \mu p' \bmod 2^m$
- 4: $r = (c+qp) / 2^m$
- 5: $r = r-p$ if $r>p$

In polynomial ring

- $p(z) = 36z^4+36z^3+24z^2 + 6z + 1$, $a(z)$ and $b(z)$,

  output $a(z)b(z)z^{-5} \bmod p(z)$.

- $p'(z) = 324z^4-36z^3-12z^2+6z-1$
- 1: $c(z) = a(z)b(z)$
- 2: $\mu(z) = c(z) \bmod z^5$
- 3: $q(z) = -p'(z)\mu(z) \bmod z^5$
- 4: $r(z) = (c(z)+ q(z)p(z)) / z^5$

$p'(z)$ and $p'z)$ **have** small **coefficients**

Montgomery multiplication (DS)

In integer ring

- Given p, a and b,

output $ab2^{-m} \bmod p$.

- Precompute $p' = -p^{-1} \bmod 2^k$
- For $i=0$ to d
- 1: $c = c + ab_i$
- 2: $\mu = c \bmod 2^k$
- 3: $q = \mu p' \bmod 2^k$
- 4: $c = (c+qp) / 2^k$
- End for
- $c = c-p$ if $c>p$

In polynomial ring

- $p(z) = 36z^4+36z^3+24z^2 + 6z + 1$, $a(z)$ and $b(z)$,

output $a(z)b(z)z^{-5} \bmod p(z)$.

- $p'(z) = -1$
- For i=0 to 5
- 1: $c(z) = c(z) + a(z)b_i$
- 2: $\mu(z) = c(z) \bmod z = c_0$
-
- 4: $r(z) = (c(z) – c_0 p(z)) / z$
- End for

Montgomery multiplication (DS)

In integer ring

- Given p, a and b,

output $ab2^{-m} \bmod p$.

- Precompute $p' = -p^{-1} \bmod 2^k$

$$36\, c_0 = (2^5 + 2^2)\, c_0$$
$$24\, c_0 = (2^4 + 2^3)\, c_0$$
$$6\, c_0 = (2^2 + 2)\, c_0$$

- 4: $c = (c + qp) / 2$
- End for
- $c = c - p$ if $c > p$

In polynomial ring

- $p(z) = 36z^4 + 36z^3 + 24z^2 + 6z + 1$, $a(z)$ and $b(z)$,

output $a(z)b(z)z^{-5} \bmod p(z)$.

- $p'(z) = -1$
- For i=0 to 5
- 1: $c(z) = c(z) + a(z)b_i$
- 2: $\mu(z) = c(z) \bmod z = c_0$
-
- 4: $r(z) = (c(z) - c_0 p(z)) / z$
- End for

There is one problem…

Choose $z=137$,

Input $a(z) = 35z^4 + 36z^3 + 7z^2 + 6z + 103$

$b(z) = 5z^4 + 136z^3 + 34z^2 + 9z + 5$

- 1: $c(z) = a(z)b(z)$
- 2: $\mu(z) = c(z) \bmod z^5$
- 3: $q(z) = p'(z)\mu(z) \bmod z^5$
- 4: $r(z) = (c(z) + \mu(z)p(z)) / z^5$

Result : $r(z) = 2243z^4 - 820648z^3 - 964511z^2 - 616127z - 173978$

But we need $r_i<z$, thus, **division by z** is needed.

$r(z) = -28z^5 + 37z^4 + 32z^3 + 120z^2 + 62z + 12$

Choose $z=2^m+s$

- For BN-curves, $p(z)$ and $r(z)$ should be prime.

$$p(z) = 36z^4 + 36z^3 + 24z^2 + 6z + 1,$$

$$r(z) = 36z^4 + 36z^3 + 18z^2 + 6z + 1,$$

$$t(z) = 6t^2 + 1.$$

We can choose $z = 2^m + s$, where s is small.

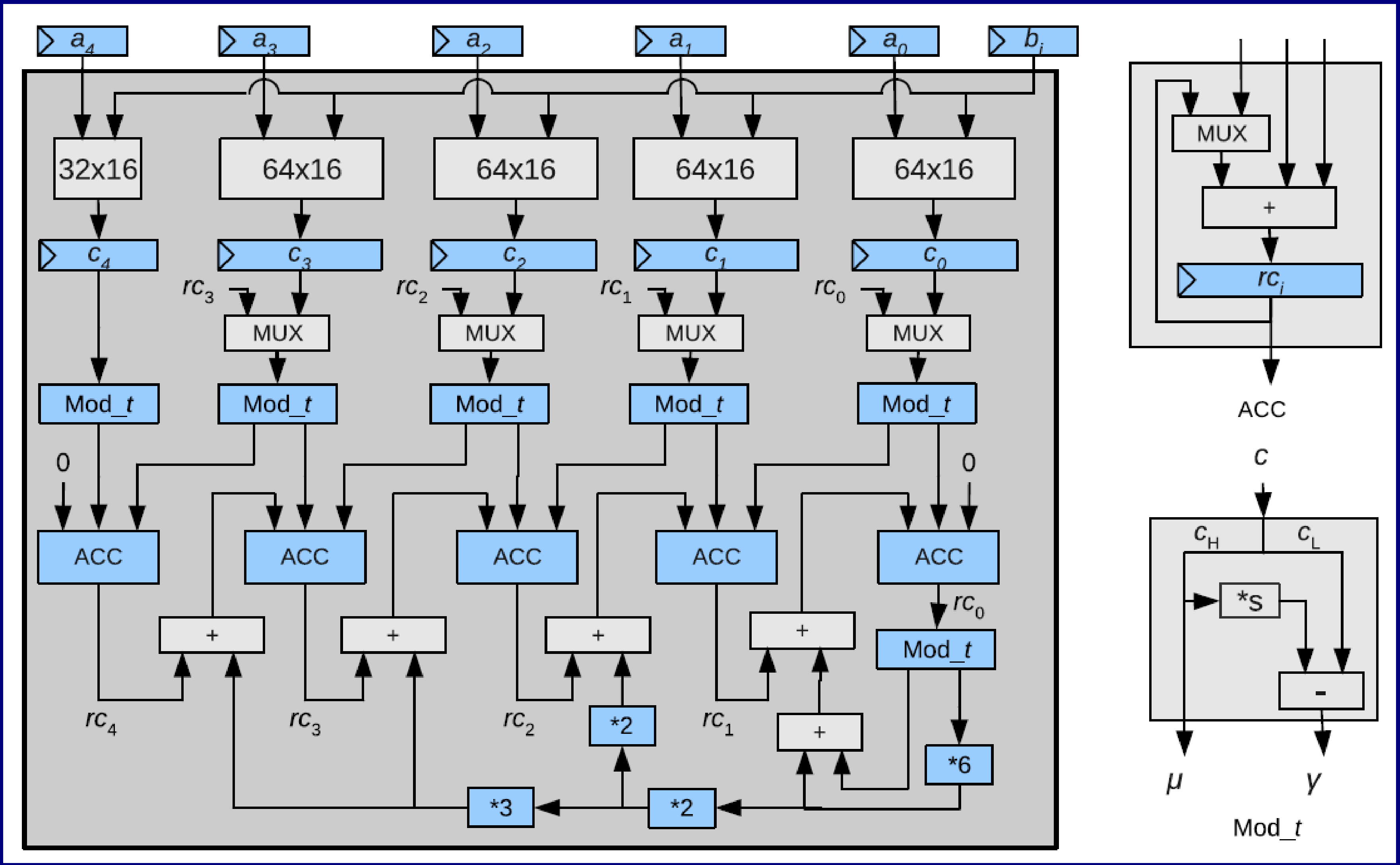
For 128-bit security, we choose $z = 2^{63} + s$, where $s=857$, and

- $p(z)$ is 258-bit prime
- $r(z)$ is 258-bit prime

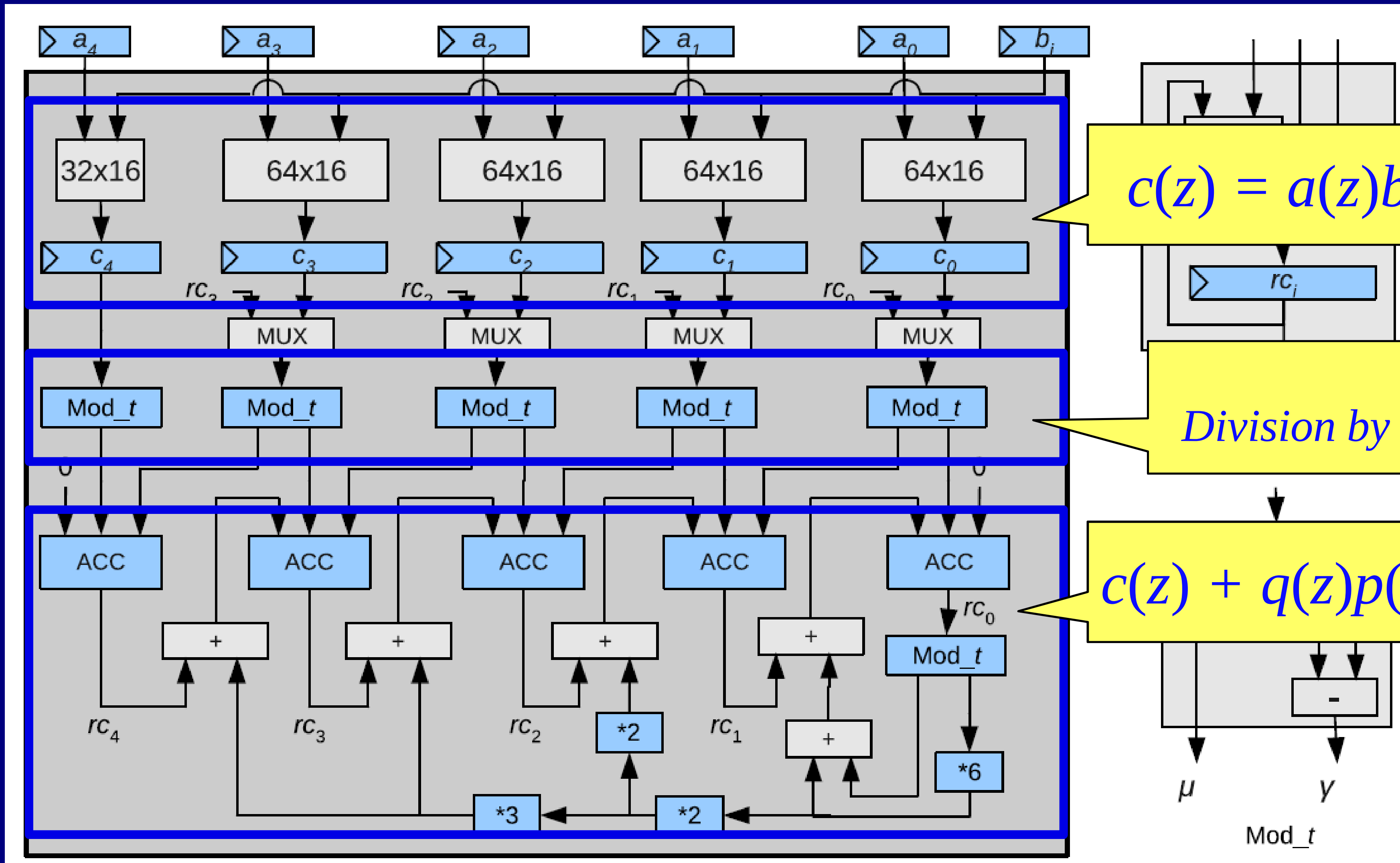
Outline

- Bilinear pairing
- Barreto-Naehrig (BN) curves
- Fast multiplication in F_p
- **Hardware implementation**
- Conclusion

Multiplier (digit-serial)



Multiplier (digit-serial)

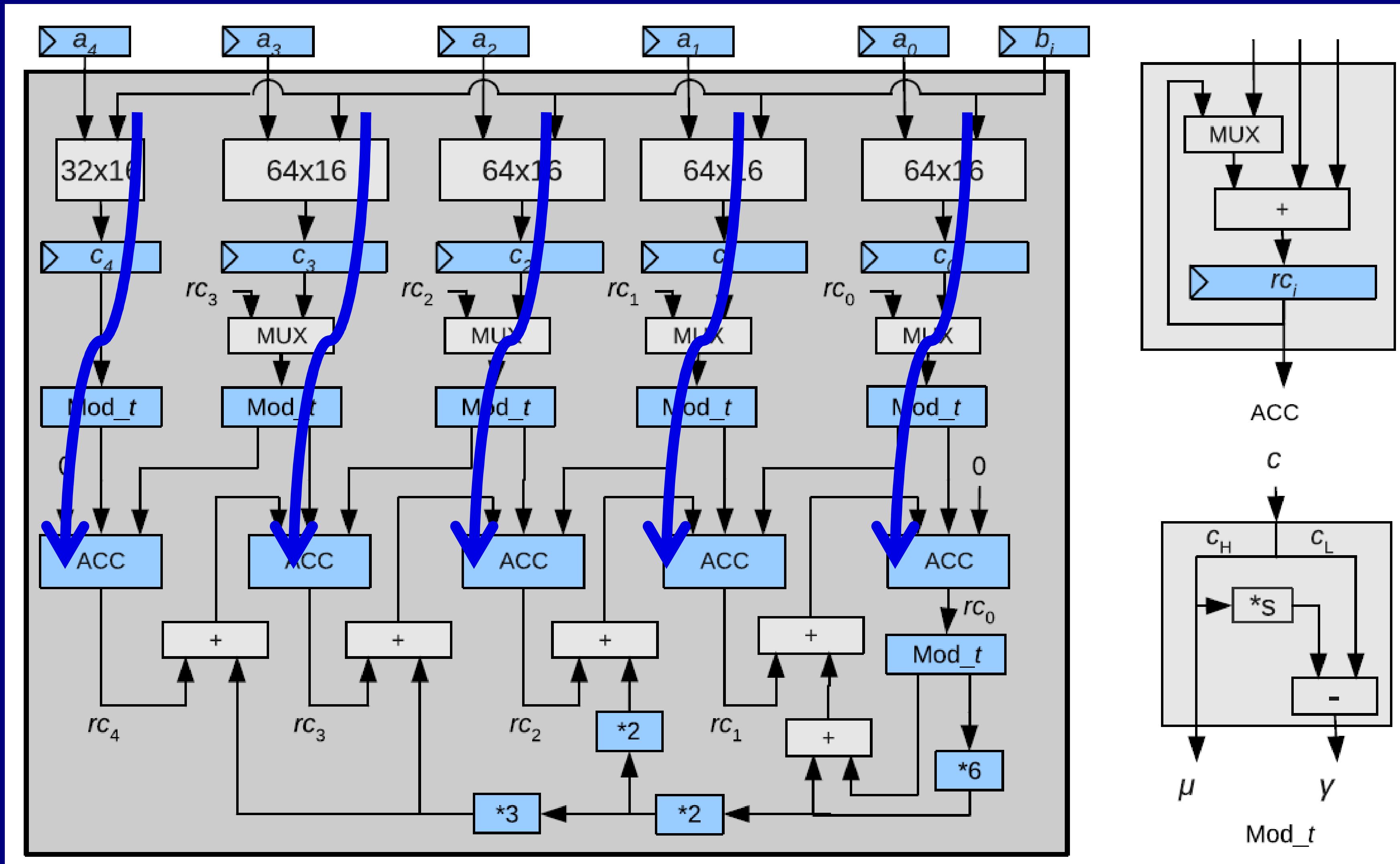


$c(z) = a(z)b_i$

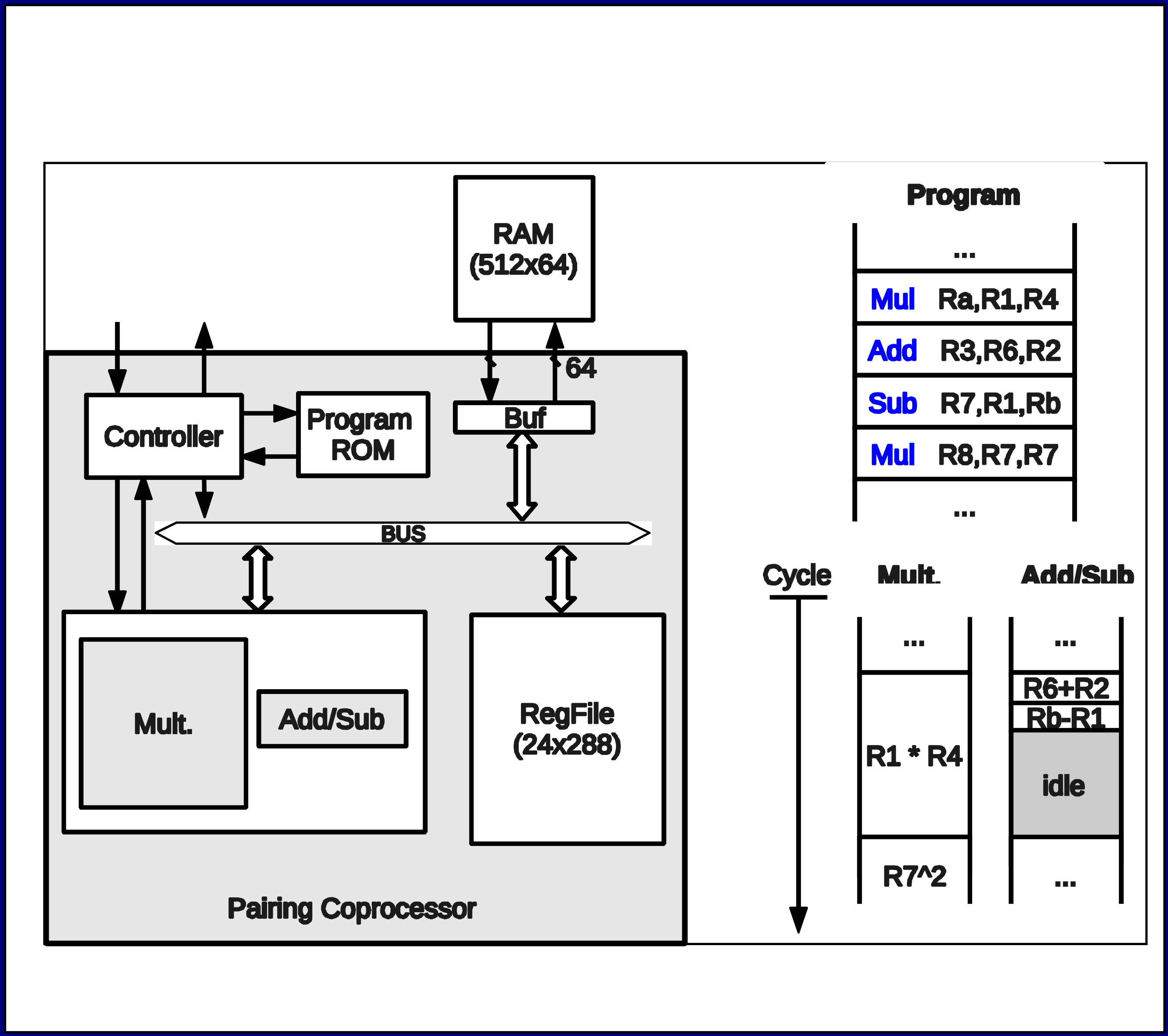
Division by Z

$c(z) + q(z)p(z)$

Multiplier (digit-serial)



Hardware implementation



Results & Comparison

Design	Pairing	Security [bit]	Platform	Area	Frequency [MHz]	Performance [ms]
this design	ate	128	130 nm ASIC	183 kGates	204	4.22
	R-ate					2.91
[12]	Tate					34.4
	ate	128	130 nm ASIC	97 kGates	338	22.8
	R-ate					15.8
[10]	ate	128	64-bit core2	-	2400	6.25
	R-ate					4.17
[9]	ate	128	64-bit core2	-	2400	6.01
[18]	η_T over $\mathbb{F}_{2^{239}}$	67	XC2VP100-6	25278 slices	84	0.034
	η_T over $\mathbb{F}_{2^{283}}$	72		37803 slices	72	0.049
[3]	η_T over $\mathbb{F}_{3^{97}}$	66	XC4VLX60-11	18683 slices	N/A	0.0048
	η_T over $\mathbb{F}_{3^{193}}$	89	XC4VLX100-11	47433 slices	N/A	0.010

Other F_p?

- For any irreducible $p(z)$ defined as

$$p(z) = p_n z^n + p_{n-1} z^{n-1} + .. + p_1 z \pm 1,$$

when p_i is integer, then $p^{-1}(z) \bmod z^n$ has integer coefficients, and

$$p^{-1}(z) = \pm 1 \bmod z.$$

Conclusion

- A new method to perform in F_p multiplication for BN-curves
  - Montgomery multiplication in polynomial ring
  - $z=2^n+s$, where s is small
- This algorithm works for all irreducible $p(z)$ if
  - $p(z) = p_n z^n + p_{n-1} z^{n-1} + .. + p_1 z \pm 1$
  - $z = 2^n+s$, where s is small

Thanks for your attention!