Compact implementations of pairings

Anthony Van Herrewege

Lejla Batina & Miroslav Knezevic
Prof. Dr. Ir. I. Verbauwhede & Prof. Dr. Ir. B. Preneel

22 May 2009

Outline

Symmetric cryptography

- Pro:
  - High security per bit
  - Very fast implementations
- Contra:
  - How to establish the key?

Asymmetric cryptography

- Pro:
  - No key establishment necessary
  - Central location with everyone's key
- Contra:
  - Need for certificate authorities, ...

Identity-based cryptography

- Pro:
  - Public key deduced from ID
  - No need for certificates
- Contra:
  - How to issue new keys, ...?
- Extra's:
  - Non-interactive key establishment
  - Date-stamped encryption

What?

- Mathematical construction discovered in the 40's
- Allow implementation of ID-based cryptography
- Strength based on discrete logarithm problem

How?

Several available pairings:

Weil, Tate, η_T, Ate, ...

Tate pairing:

$$\hat{e}(P, Q) : E(\mathbb{F}_q)[l] \times E(\mathbb{F}_q)[l] \mapsto \mu_l$$
$$\mu_l \in \mathbb{F}_{q^k}^*$$

Mapping needs to be:

- Bilinear: $\hat{e}(P_1 + P_2, Q) = \hat{e}(P_1, Q) \cdot \hat{e}(P_2, Q)$
- Non-degenerate: $\hat{e}(P, P) \neq 1$
- Well defined

How?

Calculate using optimized version of Miller's algorithm:

Require: $l \in \mathbb{Z}$; $t = \lfloor \log_2(l) \rfloor$; $P, Q \in E(\mathbb{F}_{2^m})[l]$
Ensure: $F = \hat{e}(P, Q) \in \mu_l$

```
F ← 1
V ← P
for i = t − 1 to 0 do
    F ← F^2 · G_{V,V}(φ(Q))
    V ← 2 · V
    if l_i = 1 and i ≠ 0 then
        F ← F · G_{V,P}(φ(Q))
        V ← V + P
    end if
end for
F ← F^((2^{km} − 1)/l)
return F
```

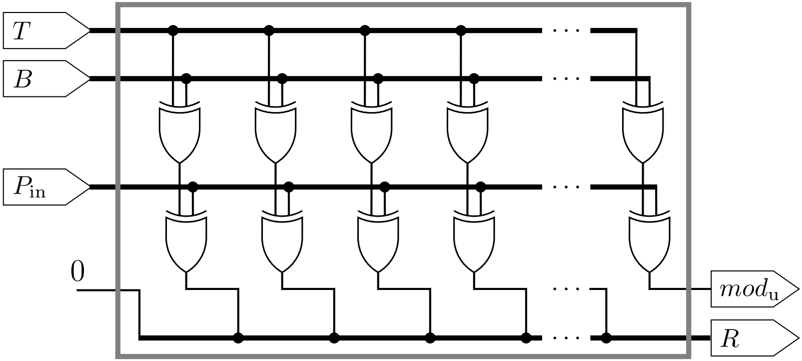
Restrictions

Avoid the use of flip-flops and muxes:

Cell	Area $\left[\frac{\text{gate}}{\text{bit}}\right]$
D flip-flop (reset)	6
D flip-flop (no reset)	5.5
D latch	4.25
3 input MUX	4
2 input XOR	3.75
2 input MUX	2.25
2 input NAND	1
NOT	0.75

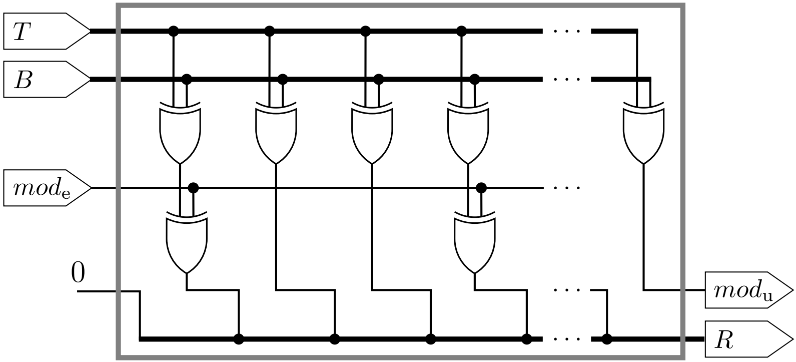
MALU - Addition & Reduction in $\mathbb{F}_{2^m}$

$$R = (T + B \pmod{P_{\text{in}}})_{0:m-2} \ll 1$$
$$mod_{\text{u}} = (T + B \pmod{P_{\text{in}}})_{m-1}$$

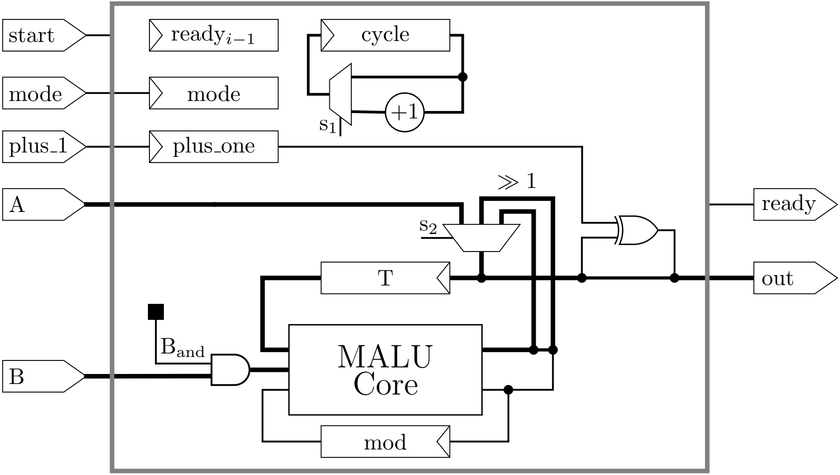


MALU - Addition & Reduction in $\mathbb{F}_{2^m}$

Optimized MALU needs $\Delta = m - (\text{Hamm}(P) - 1)$ less XOR's:

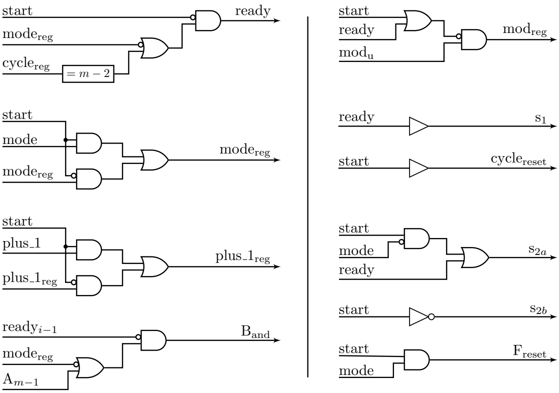


$\mathbb{F}_{2^m}$ Multiplication & Addition



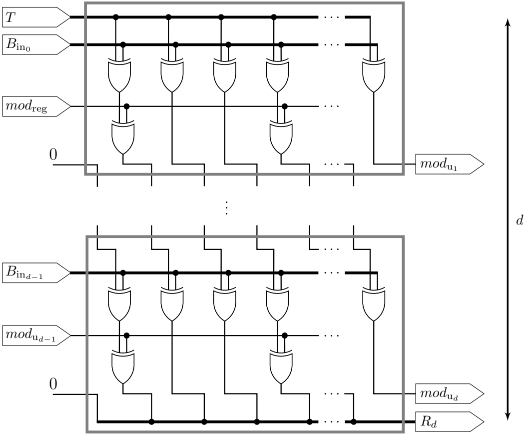
$\mathbb{F}_{2^m}$ Multiplication & Addition

No FSM needed, simple logic:

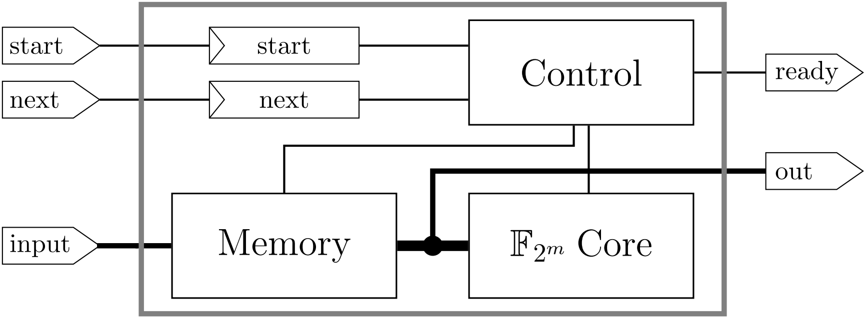


$\mathbb{F}_{2^m}$ Multiplication & Addition

Speed up calculation through daisy-chaining MALU's:



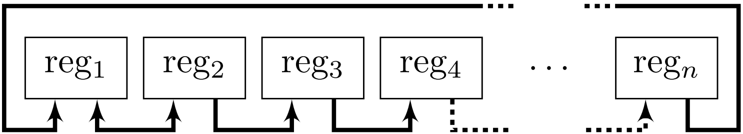
Controller for Miller's algorithm



Memory design

Starting design:

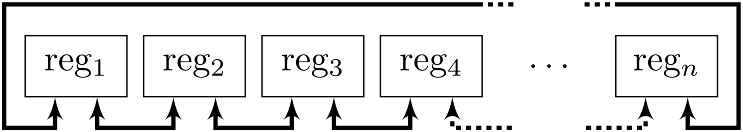
$$\bar{t} = O\left(\frac{n^2}{3}\right) \qquad \overline{w} = O\left(\frac{n^3}{3}\right)$$



Memory design

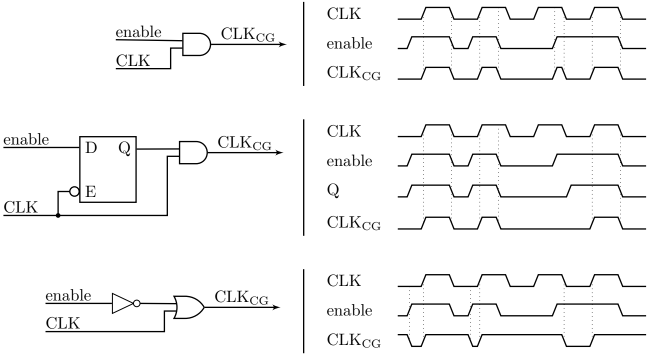
Final design:

$$\bar{t} = O\left(\frac{n}{4}\right) \qquad \overline{w} = O\left(n\right)$$



Optimizations

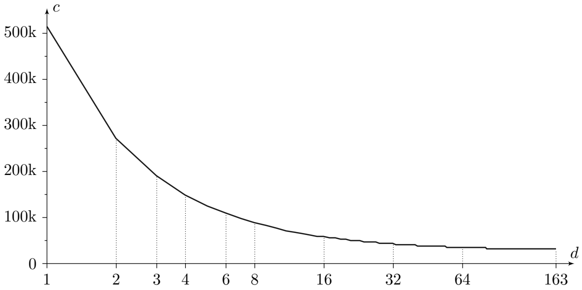
- Remove reset from registers $\left(-0.5\ \frac{\text{gate}}{\text{bit}}\right)$
- Implement clock gating:



Runtime

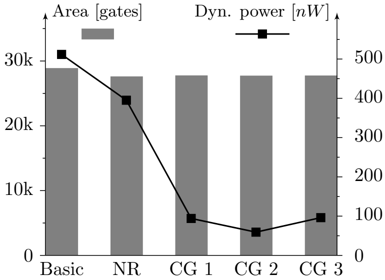
- FSM with 553 states
- Total n° of clockcycles c for one pairing:

$$c = 21681 + 4322 + 2998 \cdot \left\lceil \frac{m}{d} \right\rceil.$$



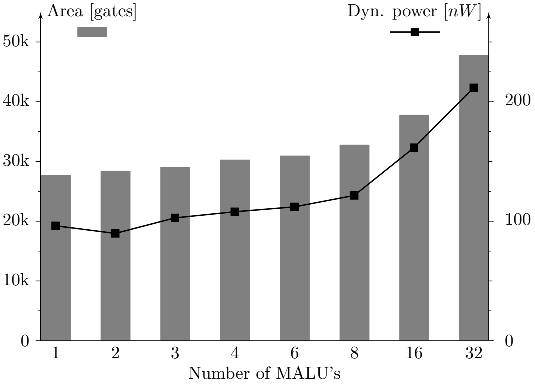
Synthesis

Implementation	Area [gates]		Power @ 10 kHz $[nW]$			
			Dynamic		Leakage	
Basic	28 876		512		117	
No Reset	27 596	96%	395	77%	107	92%
CG 1	27 751	96%	94	18%	109	94%
CG 2	27 713	96%	59	12%	102	88%
CG 3	27 734	96%	96	19%	110	94%



Synthesis - Continued

Component	Opp. [gates]	
MALU	458	1.7%
$\mathbb{F}_{2^m}$ core		
Logic	783	2.8%
Registers	962	3.5%
Controller		
Logic	13 044	47%
Registers	12 487	45%
Total	27 734	100%



Comparison

	This work		Beuchat *et al.*
	1 MALU	2 MALUs	
Field	$\mathbb{F}_{2^{163}}$	$\mathbb{F}_{2^{163}}$	$\mathbb{F}_{3^{97}}$
Pairing	Tate	Tate	η_T
Security [bit]	652	652	922
Technology [μm]	0.13	0.13	0.18
Area [gates]	27 430	28 155	193 765
f [MHz]	10.3	5.44	200
Calc. time [μs]	$50 \cdot 10^3$	$50 \cdot 10^3$	46.7
Power [mW]	$98.3 \cdot 10^{-3}$	$48.6 \cdot 10^{-3}$	672
Efficiency $\left[\frac{nJ}{\text{bit}}\right]$	7.54	3.73	34.0

Conclussion

The end

Questions?