

Is Elliptic-Curve Cryptography Suitable to Secure RFID Tags?

Workshop on RFID and Light-Weight Crypto

July 14th-15th 2005, Graz (Austria).

Johannes.Wolkerstorfer@iaik.tugraz.at

*Institute for Applied Information Processing
and Communications (IAIK) — VLSI Group*

*Faculty of Computer Science
Graz University of Technology*



Outline

- Motivation
- Elliptic-curve cryptography
- Optimization goals
- Design methodology
- Architecture
- Results
- Conclusions

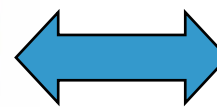


Motivation

- Asymmetric crypto versus symmetric
 - Key distribution in open systems eased
- ECC versus RSA, XTR, NTRU
 - ECC much more compact than RSA
 - XTR, NTRU: well, yes, ...

- Authentication
- Electronic Signature

© Picture: Tagstore

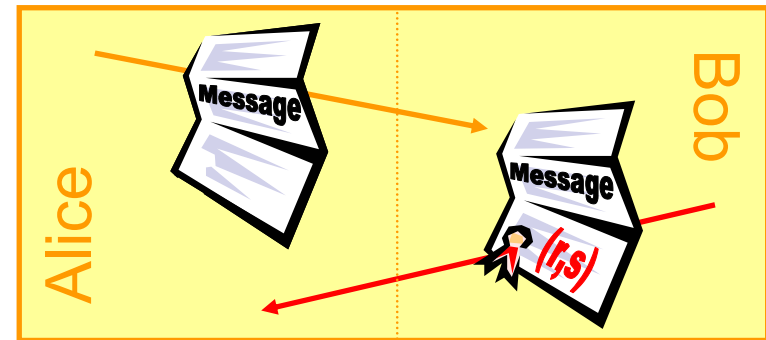


© Picture: TI



Elliptic-Curve Cryptography

- Protocol
 - Challenge-response authentication
- Algorithm
 - ECDSA: elliptic-curve digital signature algorithm
- Computation
 - Scalar multiplication
 - Repeated Doubling and addition of curve points
 - Finite-field operations (160-bit ... 256-bit)



ECDSA

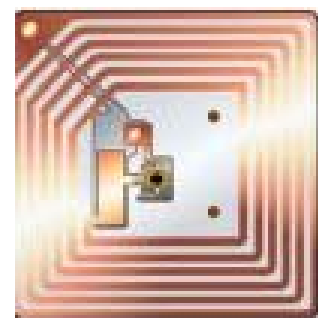
```

e = SHA-1(Message)
k = random(1, n-1)
R = k*(P_x, P_y) = (R_x, R_y)
r = R_x mod n
s = k^-1 * (e + d*r)
  
```

$$\begin{aligned}
 2 \cdot P_1 &= 2 \cdot (x_1, y_1, z_1) = (x_3, y_3, z_3) = P_3 \\
 x_3 &= (3x_1^2 + az_1^4)^2 - 8x_1y_1^2 \\
 y_3 &= (3x_1^2 + az_1^4)(4x_1y_1^2 - x_3) - 8y_1^4 \\
 z_3 &= 2y_1z_1
 \end{aligned}$$

Optimization Goals

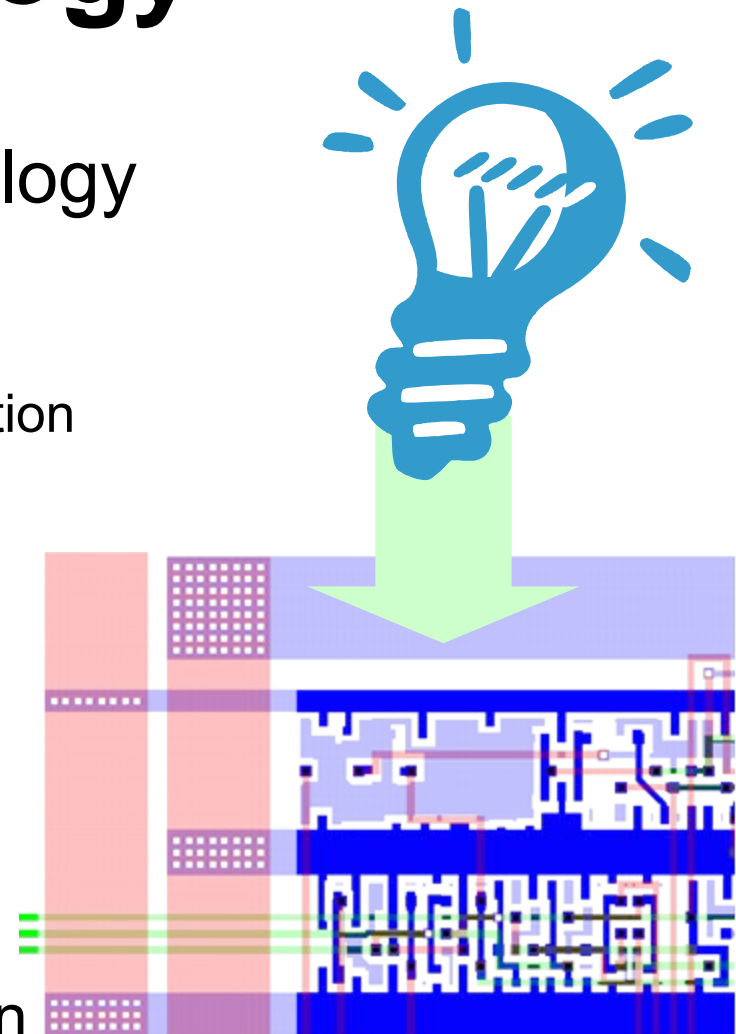
- Goals comply with ISO 18000-3 requirements (13.56 MHz)
 - Area
 - Less than 1 mm²
 - Power
 - $I < 10 \mu\text{A}$ @1.5 V
 - To guarantee 1m operating range
- Security: > 160-bit ECC
 - GF(2¹⁹¹) and/or GF(p₁₉₂)
- Manageable control



© Picture: ippaper.com

Design Methodology

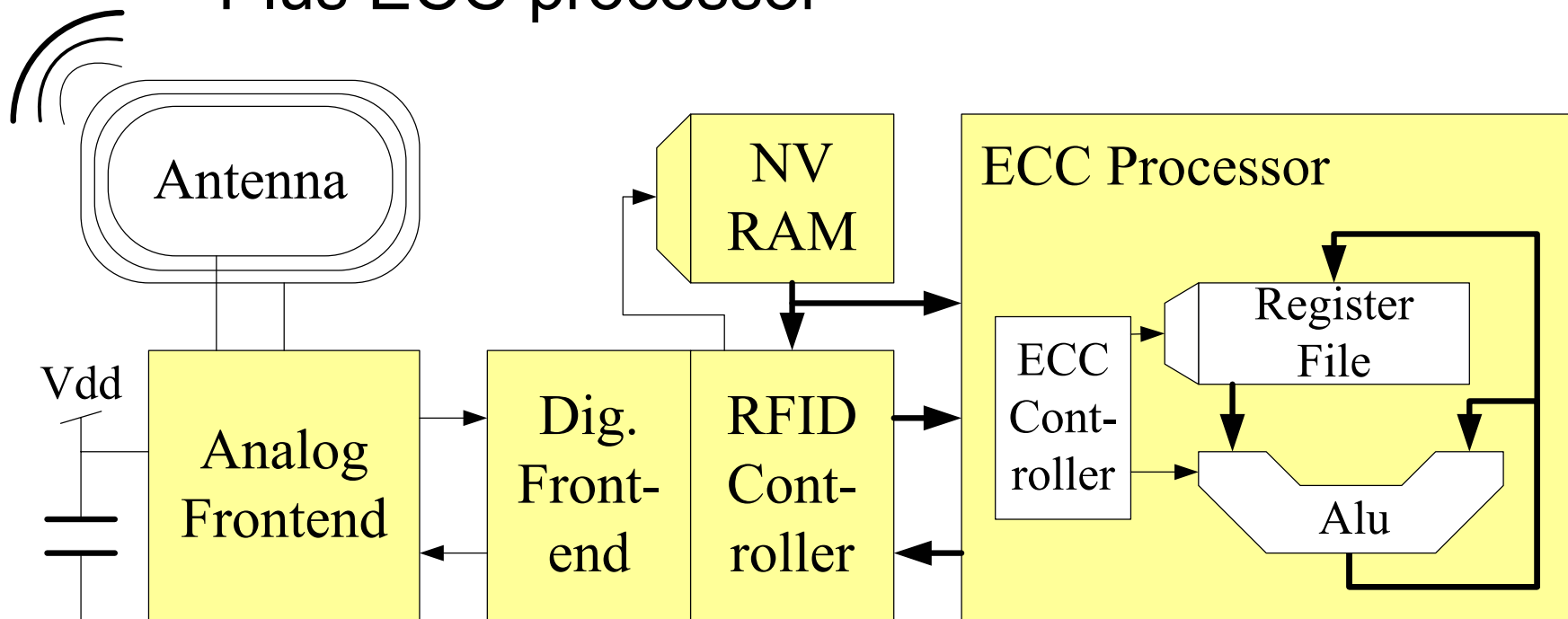
- Top-down design methodology
 - Design space exploration
 - Evaluation of design options
 - Optimization for target application
 - Focus on
 - High-level models
 - Early estimates
- Parameterizable VHDL
- Target technology
 - 0.35 μm CMOS process
 - Standard-cell implementation



Architecture

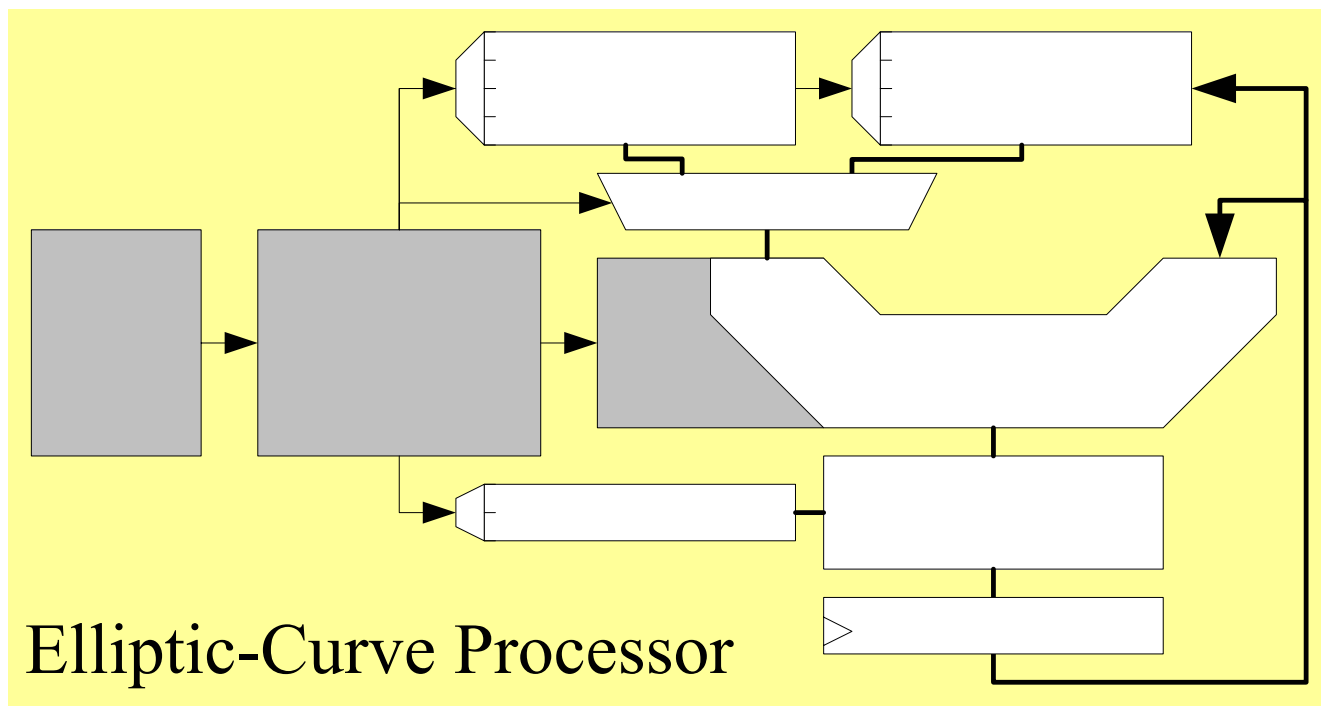
ECC-Enabled RFID Tag

- Conventional tag architecture
 - Plus ECC processor



Architecture

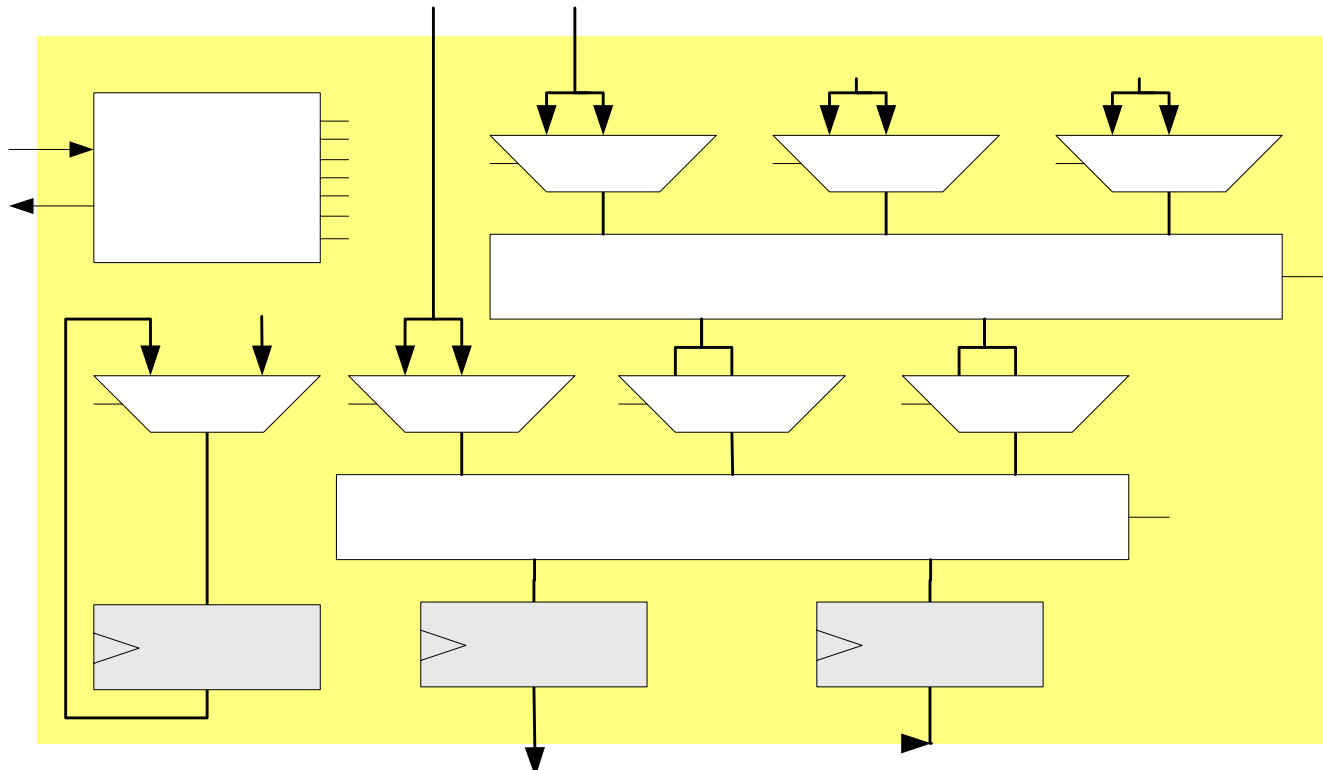
ECC Processor



- Full-precision architecture
- Supports different finite fields

Architecture

Dual-Field Arithmetic Unit

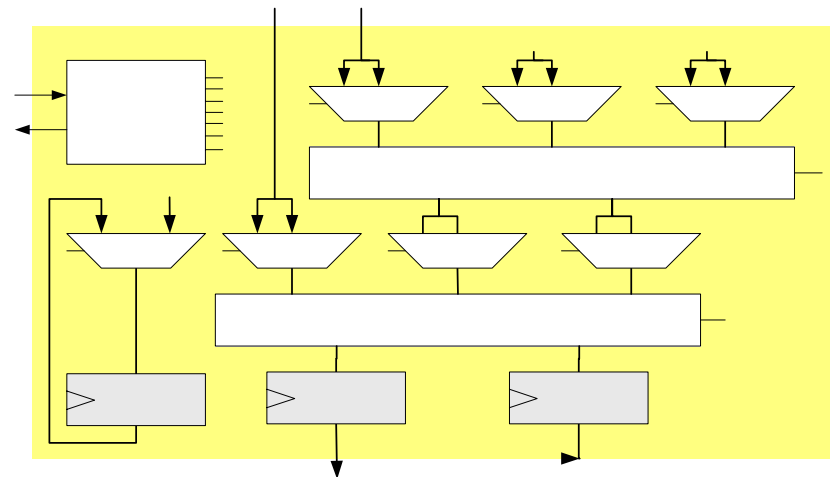


- Operates in $GF(2^m)$ and $GF(p)$
- Redundant representation of $GF(p)$

Results: Arithmetic unit

- Operations supported by arithmetic unit
- Many HW resources reused
- Dual-field capability at almost no overhead
- Uses Montgomery multiplication

Name	Function (s,c)=	Name	Function (s,c)=
Clear	(0, 0)		
Hold	(s', c') → (s'', 0)	Load	(a, 0)
Add	(s+a, c)	Sub	(s-a, c)
Shftl	(2s, 2c)	Shftr	((s+p·q)/2, c/2)
Mul ₀	(a·b ₀ , 0)	Mul _i	((s+p·q)/2+a·b _i +, c/2)



Results

- Size, performance, and power

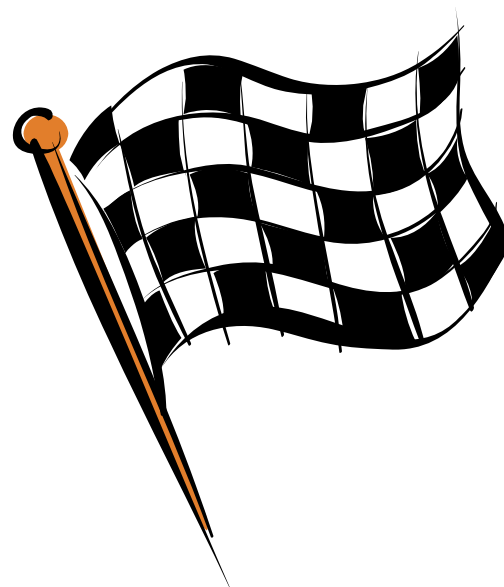
CMOS	ECC Processor				
I_{gate} [nm]	Area [mm ²]	Power [μW]	Clock [kHz]	GF(p ₁₉₂) [s]	GF(2 ¹⁹¹) [s]
350	1.31	30	60	11.3	7.1
180	0.35	30	175	3.9	2.5
90	0.09	30	545	1.3	0.8

- Smallest stand-alone ECC processor
 - Reported in literature so far!

Results

Does ECCU fit RFID?

- Area on 0.35 μm CMOS
 - No – too large: 1.31 mm^2
- Area on 180 nm CMOS
 - **YES** – 0.35 mm^2 is feasible
- Power
 - **YES!** Constraints can be met by
 - Lowering clock frequency (e.g. 175 kHz @180nm)
- Performance
 - Is poor: > 1 second for an operation (@ 175 kHz)
 - But: 100 ops / second (@ $f_{\text{max}} = 68.5 \text{ MHz}$)



Conclusions



- Achievements
 - Novel arithmetic unit
 - Dual-field operation: $GF(p)$ and $GF(2^m)$
 - Area (and power consumption)
 - Suitable for RFID implementation
- Outlook
 - Hardwired control
 - More efficient register file
 - Random number generation, Hashing(!)