



**HAL**  
open science

# RNS Modular Computations for Cryptographic Applications

Karim Bigou, Arnaud Tisserand

► **To cite this version:**

Karim Bigou, Arnaud Tisserand. RNS Modular Computations for Cryptographic Applications. RAIM: 7ème Rencontre Arithmétique de l'Informatique Mathématique, Apr 2015, Rennes, France. , 2015. hal-01141347

**HAL Id: hal-01141347**

**<https://inria.hal.science/hal-01141347>**

Submitted on 11 Apr 2015

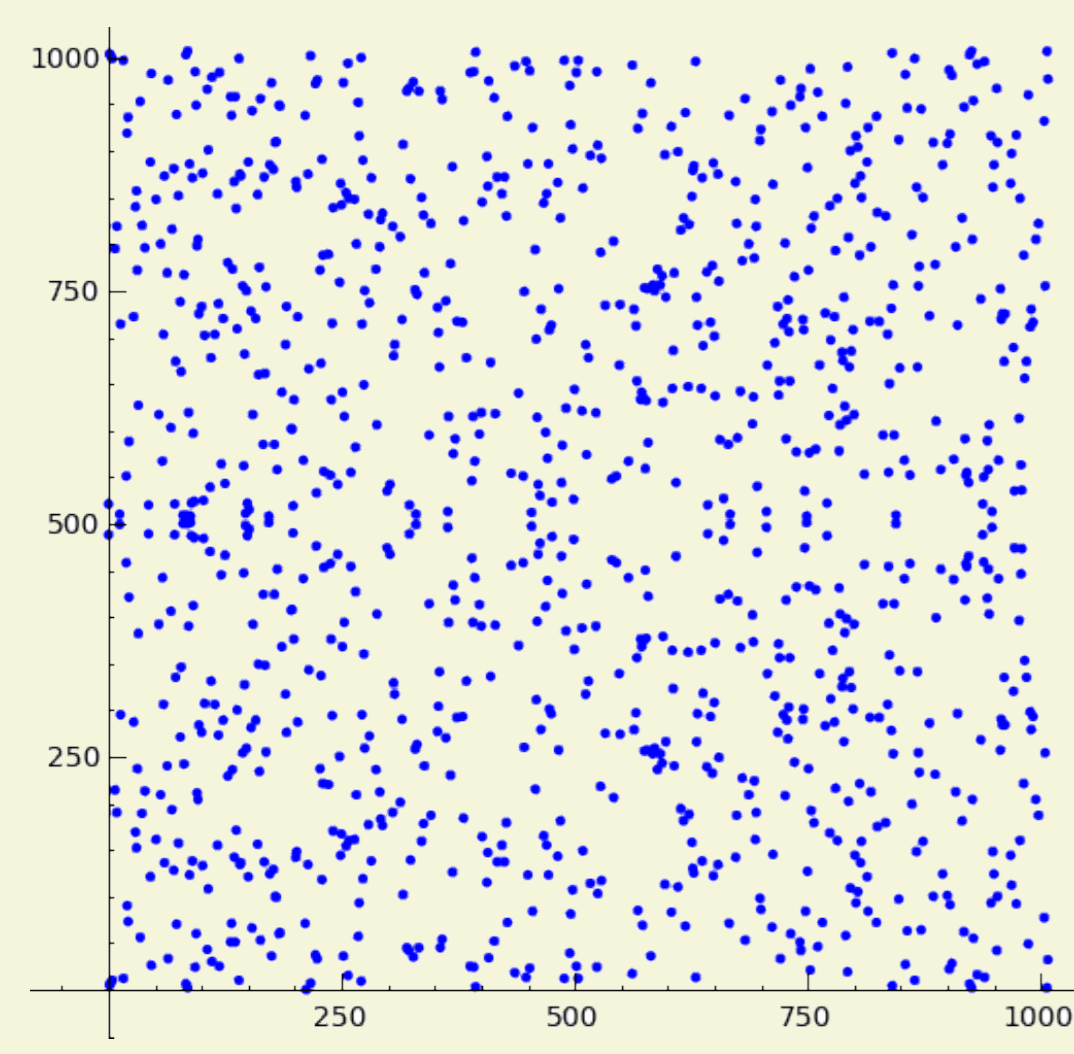
**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## 1. Elliptic Curve Cryptography (ECC)

Elliptic curve over  $\mathbb{F}_P$ :  $y^2 = x^3 + ax + b$  with  $P$  a  $\ell$ -bit prime



Security levels:  $\ell \in \{160, \dots, 600\}$  bits

Curve level operations:

- ▶ point addition (ADD):  $Q + Q'$
- ▶ point doubling (DBL):  $Q + Q$
- ▶ scalar multiplication:  $[k]Q = \underbrace{Q + Q + \dots + Q}_{k \text{ times}}$

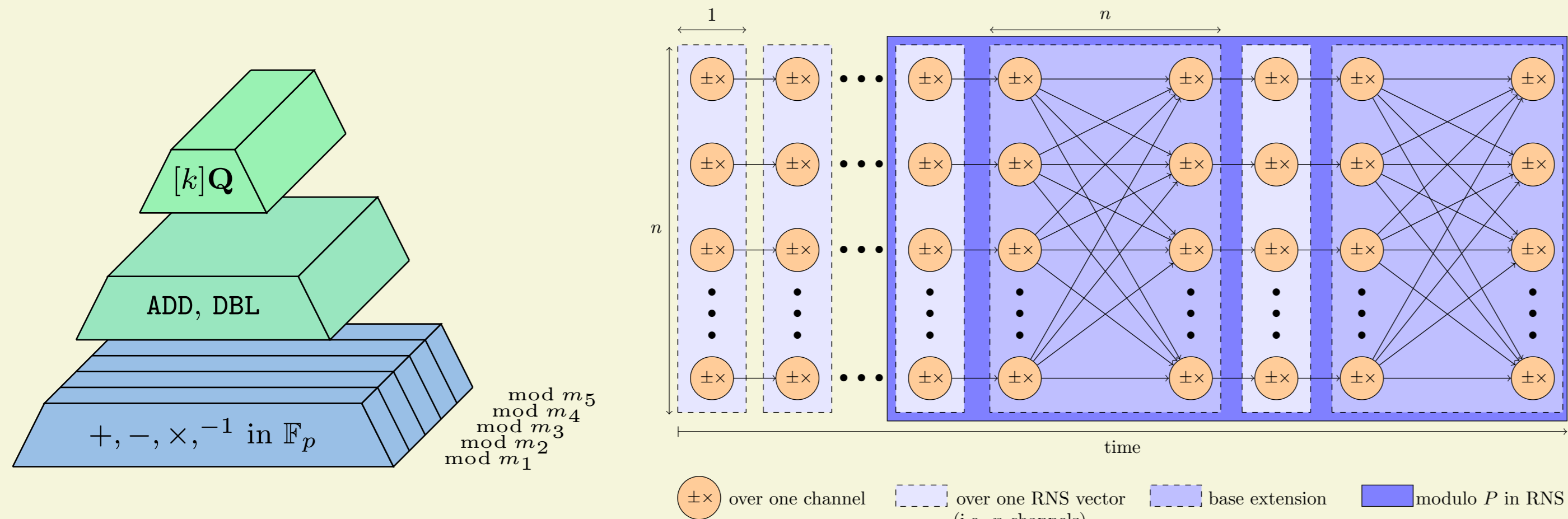
Security (ECDLP): knowing  $Q$  and  $[k]Q$ ,  $k$  cannot be recovered

ECDLP: Elliptic Curve Discrete Logarithm Problem

$y^2 = x^3 + 4x + 20$  over  $\mathbb{F}_{1009}$

## 3. RNS Computation Flow in ECC Applications

RNS allows to perform some field level operations in parallel



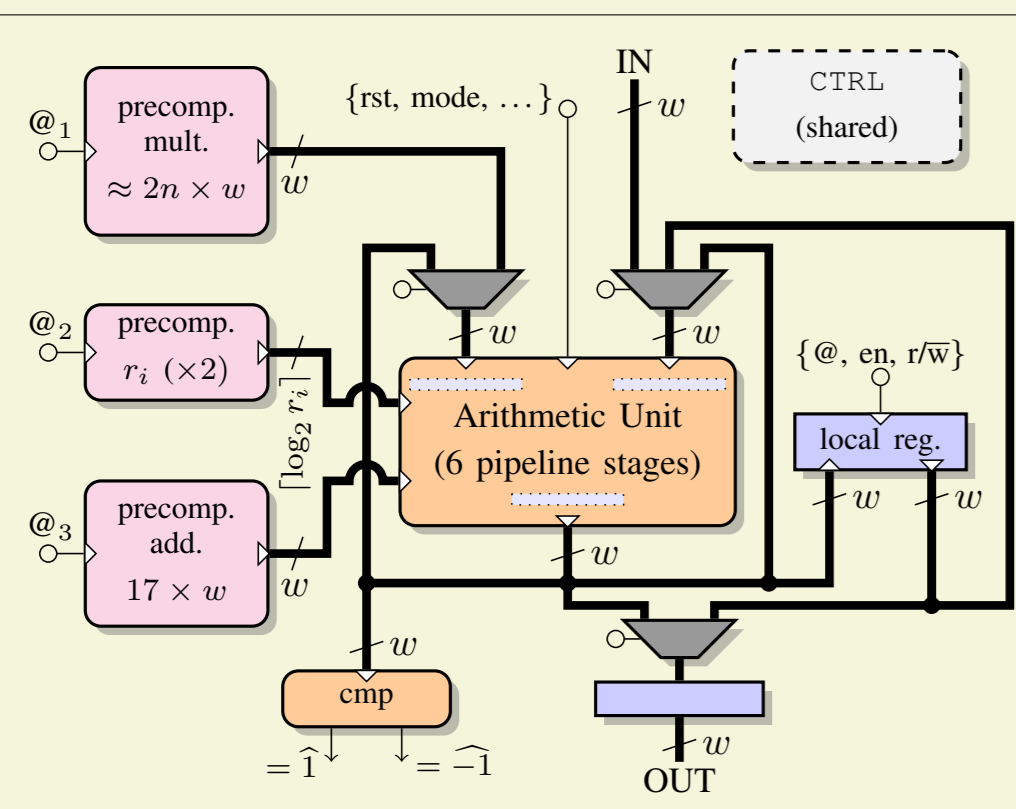
## 5. New RNS Modular Inversion (MI) (CHES 2013)

State-of-the-art RNS MI methods:

- ▶ based on **Fermat's Little Theorem** (FLT-MI):  $X^{-1} = X^{P-2} \text{ mod } P$  i.e. a large exponentiation with a lot of modular reductions which costs  $O(\log_2 P \times n^2)$  EMMS
- ▶ **very limited** parallelization due to internal data dependencies

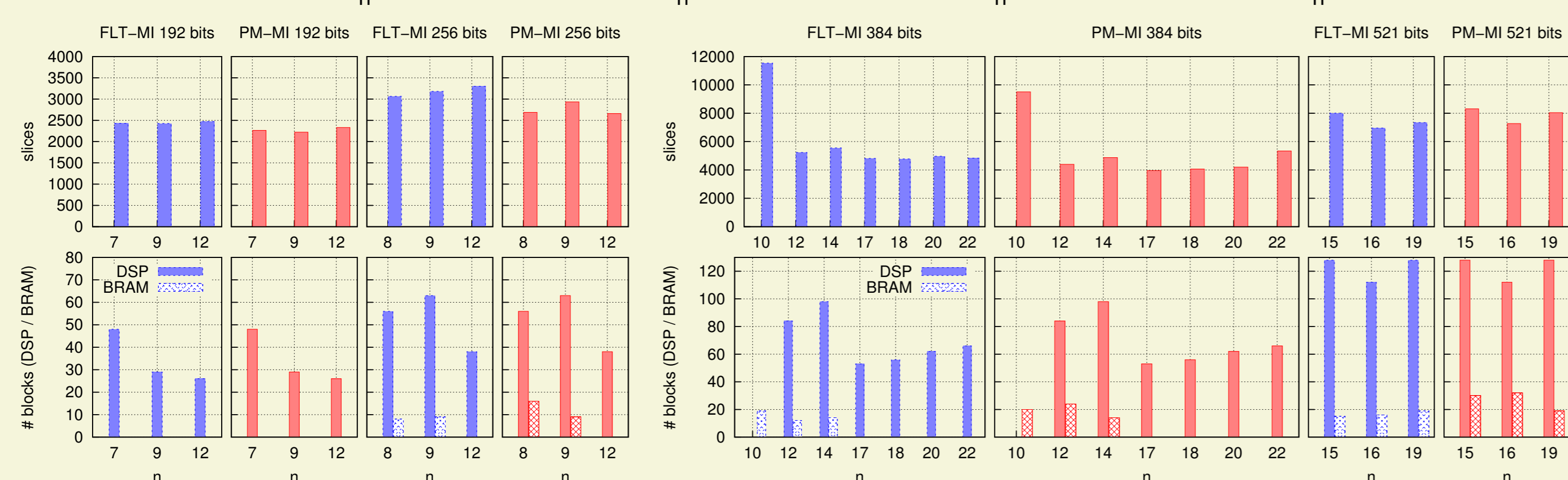
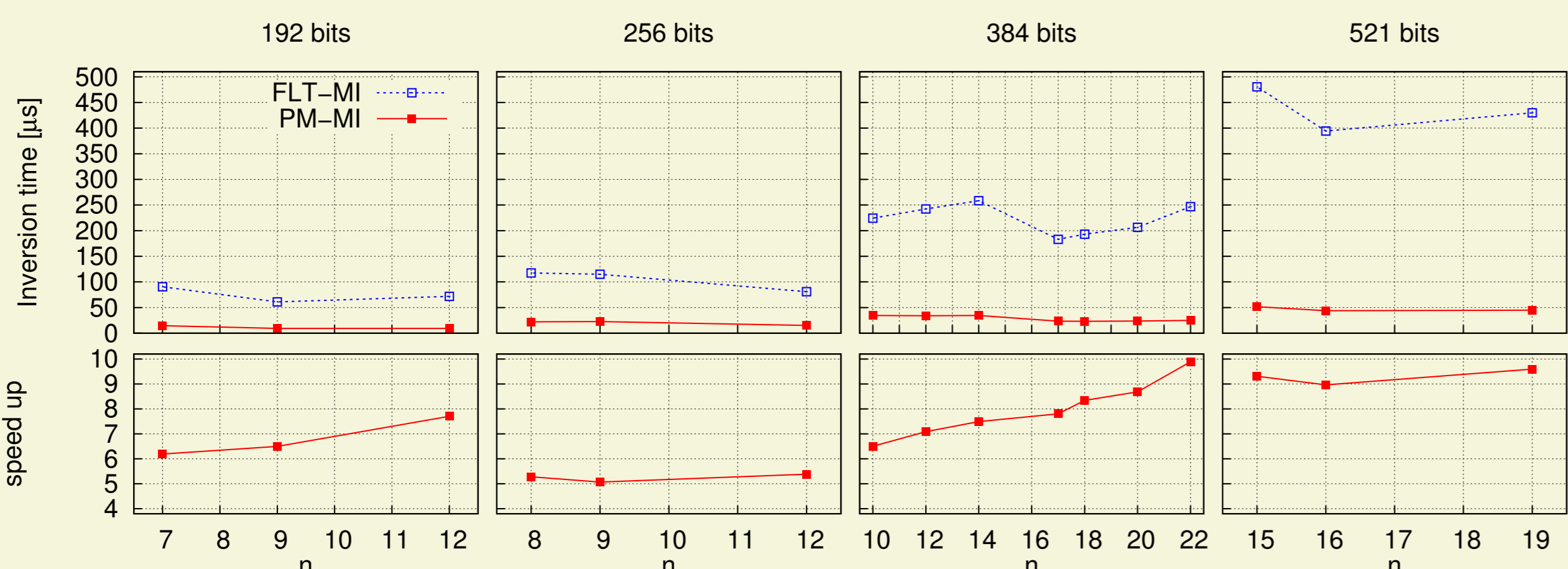
Proposed method PM-MI:

- ▶ extended binary **Euclidean algorithm** (binary-ternary version)
- ▶ uses the **plus-minus trick**: if  $X$  and  $Y$  are odd then  $X + Y = 0 \text{ mod } 4$  or  $X - Y = 0 \text{ mod } 4$
- ▶ PM-MI works **without BE** and costs  $O(\log_2 P \times n)$  EMMS



Example: # EMMS for  $\ell = 192$  bits

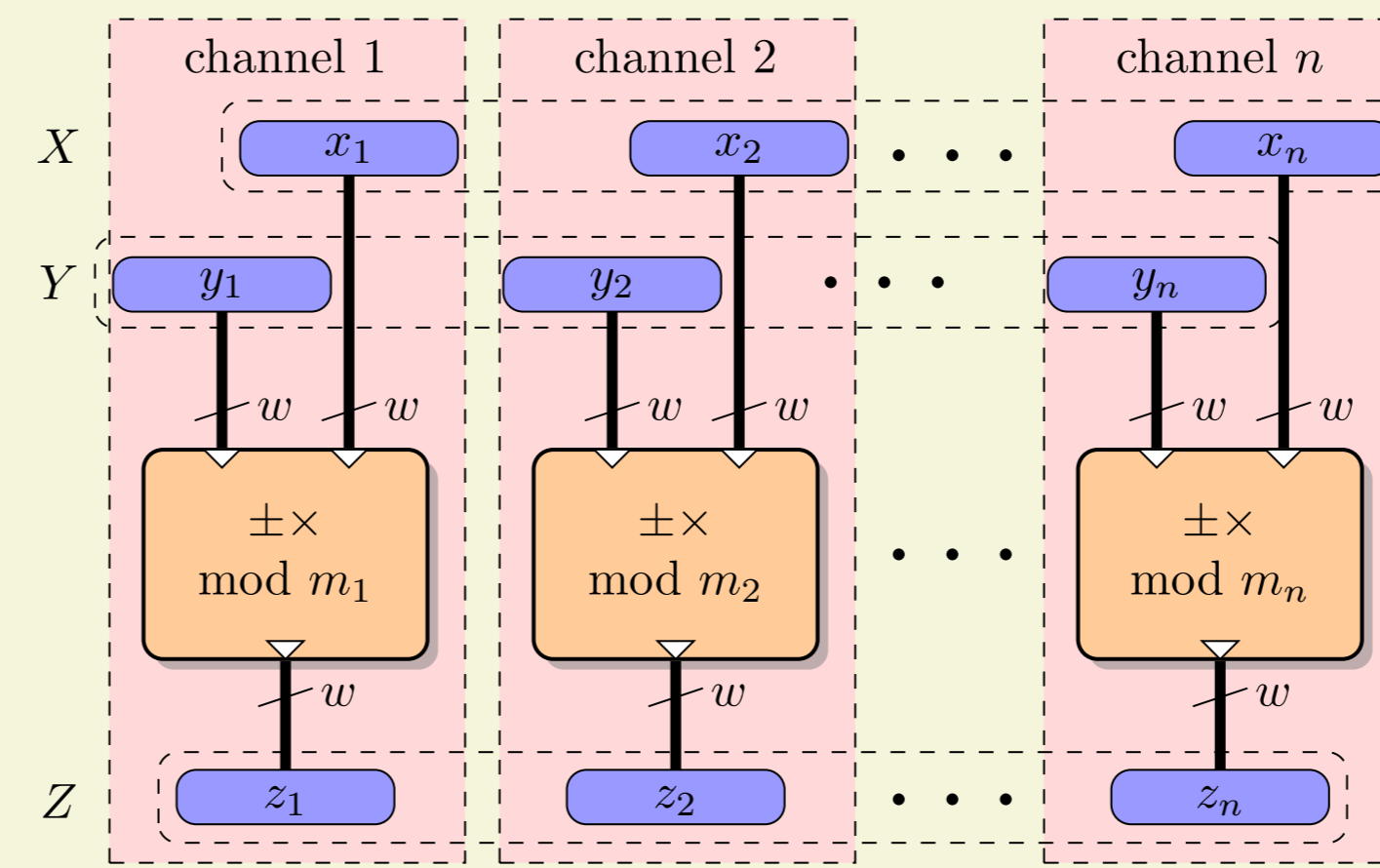
$n \times w$	FLT-MI	PM-MI	Gain Factor
$12 \times 17$	103140	5474	18
$9 \times 22$	61884	4106	15
$7 \times 29$	40110	3193	12



## 2. Residue Number System (RNS)

$X$  a large  $\ell$ -bit integer is represented by:

$$\vec{X} = (x_1, \dots, x_n) = (X \text{ mod } m_1, \dots, X \text{ mod } m_n)$$



RNS base  $\mathcal{B} = (m_1, \dots, m_n)$   
 $n$  pairwise  $w$ -bit co-primes with  $n \times w \geq \ell$

The Chinese remainder theorem (CRT) is the base of RNS

EMM: elementary modular multiplication ( $w$  bits)

Pros:

- ▶ **carry free** between channels
- ▶ **fast parallel**  $+$ ,  $-$ ,  $\times$  and some exact divisions
- ▶ **non-positional** number system, randomization against SCAs
- ▶ **flexibility** for hardware implementations

Cons:

- ▶ comparison, **modular reduction** and division are **much harder**

## 4. State-of-the-Art Algorithms and Architectures

RNS Montgomery Reduction

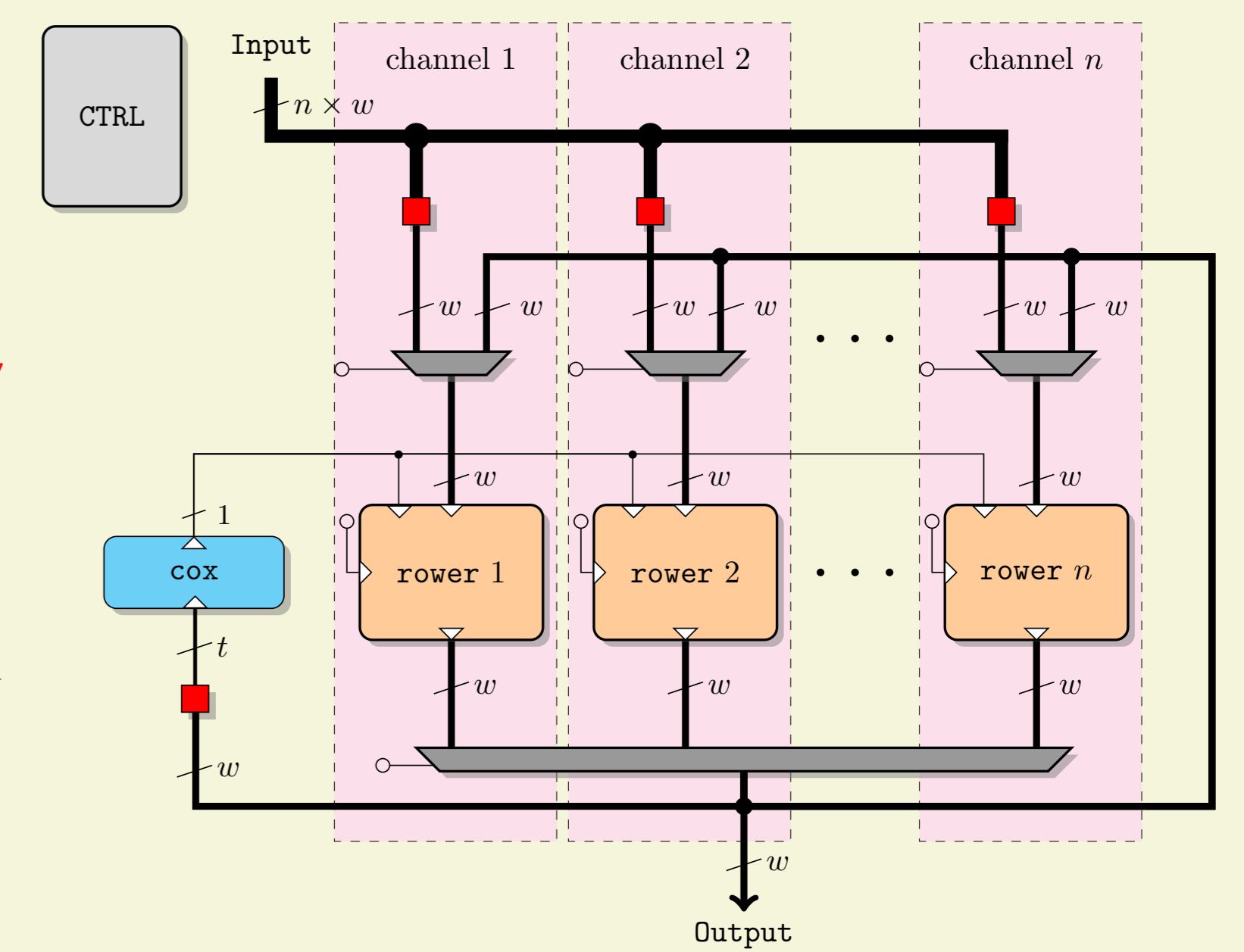
Input:  $\vec{X}, \vec{X}'$

Output:  $(\vec{w}, \vec{w}')$  with  $w \equiv X \times M^{-1} \text{ mod } P$

$$\begin{aligned} \vec{Q} &\leftarrow \vec{X} \times (-\vec{P}^{-1}) && \text{(in base } \mathcal{B}) \\ \vec{Q}' &\leftarrow \text{BE}(\vec{Q}, \mathcal{B}, \mathcal{B}') && \text{--- BE ---} \\ \vec{S}' &\leftarrow \vec{X}' + \vec{Q}' \times \vec{P}' && \text{(in base } \mathcal{B}') \\ \vec{w}' &\leftarrow \vec{S}' \times \vec{M}^{-1} && \text{(in base } \mathcal{B}') \\ \vec{w} &\leftarrow \text{BE}(\vec{w}', \mathcal{B}', \mathcal{B}) && \text{--- BE ---} \end{aligned}$$

BE: base extension

$$M = \prod m_i$$



## 6. Fast Patterns for RNS Computations (ASAP 2014)

Cost of standard and modular multiplications in RNS:

- ▶ standard:  $n$  EMMS fully parallel
- ▶ modular:  $2n^2 + O(n)$  EMMS 1 mult. & 1 red.

Proposed method:

- ▶ **splits** operands into 2 parts:  $\vec{X} = (\vec{K}_x) \times (\vec{M}_a) + (\vec{R}_x)$  allows to replace  $2n$  moduli by only  $\frac{3}{2}n$
- ▶ **reuses** split result in various computation patterns
- ▶ requires an hypothesis on  $P$ : OK for ECC/DH, but not for RSA

Cost for some patterns (#EMMS):

Operations	s-o-t-a	our
$AB \text{ mod } P$	$2n^2 + 4n$	$2.5n^2 + 12.5n$
$A^2 \text{ mod } P$	$2n^2 + 4n$	$1.75n^2 + 10.5n$
$Cst \times A \text{ mod } P$	$2n^2 + 4n$	$1.75n^2 + 7n$
$Cst \times A^2 \text{ mod } P$	$4n^2 + 8n$	$2.75n^2 + 16.5n$

Usage for Diffie-Hellman or ElGamal:

