High-Speed Implementation of the SHA-3 Candidate Shabal

2nd SHA-3 Conference

Julien Francq and Céline Thuillet
{julien.francq, celine.thuillet}@eads.com

EADS Defence & Security,
Cyber Security Customer Solutions Center

2010, August the 23th

Introduction

2 ways of studying SHA-3 candidates hardware cost on FPGA

- "Fair and comprehensive comparison" of all the candidates.
  - [Tillich *et al.*, ePrint 2009], [Kobayashi *et al.*, ePrint 2010], [Henzen *et al.*, CHES 2010], [Gaj *et al.*, CHES 2010], [Baldwin *et al.*, 2nd SHA-3 Conf.]
- Investigate hardware optimizations of some ones.

Why have we focused on Shabal and its throughput?

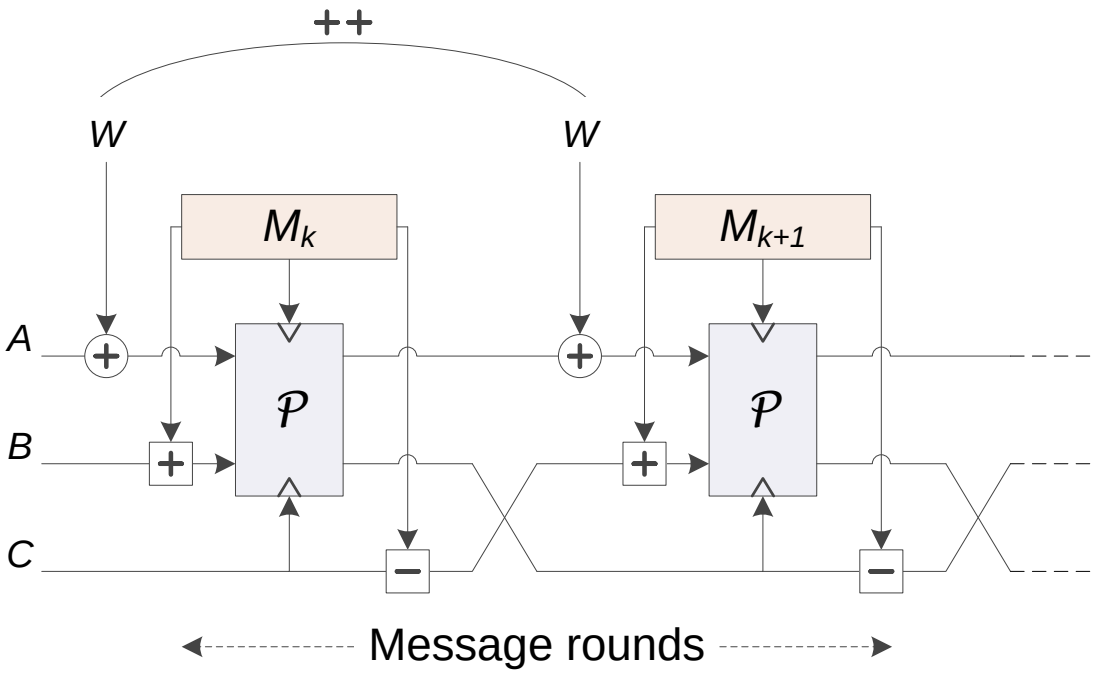
- It can be implemented in resource-constrained devices [Detray *et al.*, ePrint 2010].
- At first sight, its restricted parallezibility seems to limit its throughput (TP).
  $\Rightarrow$ Fortunately, a neat method allows to improve its TP!
  $\Rightarrow$ Unfolding Method

Outline

Outline

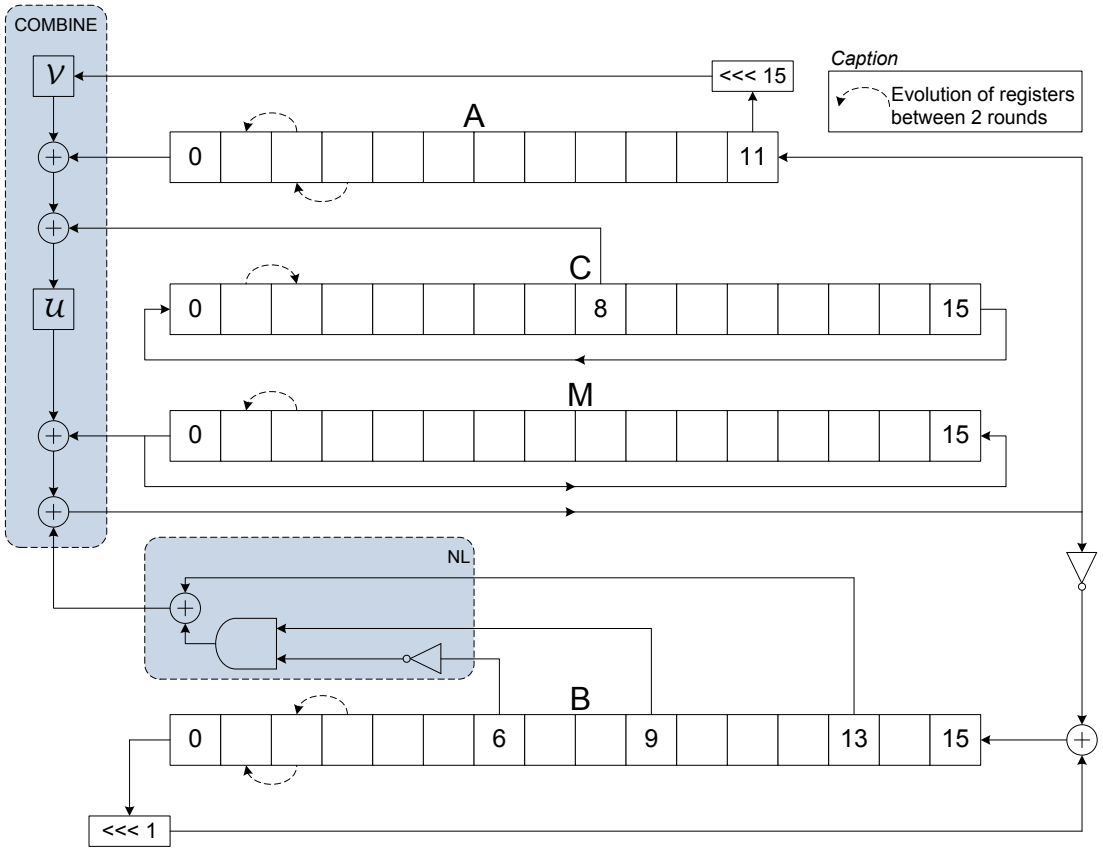
Parameters and Generic Construction

- Iteration of permutation $\mathcal{P}$ on padded message blocks M_k (16 $\times$ 32-bit words).
- Internal state (A, B, C) initialized by an Initial Vector.
- W: 2 $\times$ 32-bit counter.
- $\oplus$: XOR, $\boxplus$ ($\boxminus$): Wordwise addition (subtraction) modulo 2^{32}.



Permutation $\mathcal{P}$

- 48 iterations of a Non-Linear Feedback Shift Register (NLFSR)
- A: 12 $\times$ 32-bit words, B and C: 16 $\times$ 32-bit words.
- $\mathcal{U}(x) = 3 \times x \bmod 2^{32}$, $\mathcal{V}(x) = 5 \times x \bmod 2^{32}$, $\lll$: left rotation.

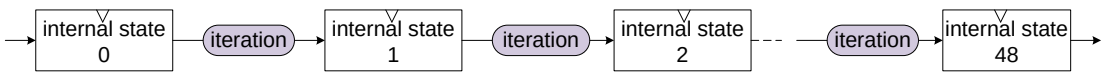


- (Final update of A omitted on the figure)

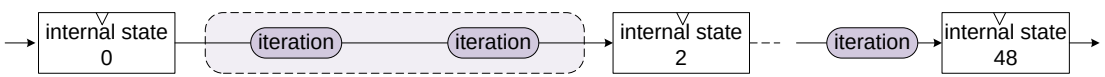
Outline

Unfolding Method

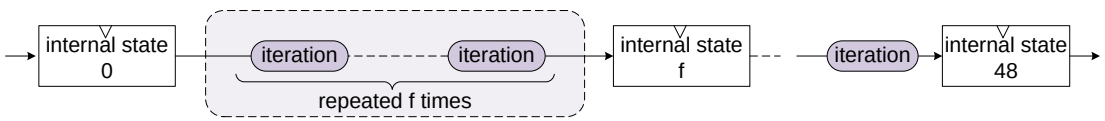
- Unroll the Shabal core
- Method already used for hash functions [Hoare *et al.*, IASTED 2002] [Lien *et al.*, CT-RSA 2004] [Lee *et al.*, JSPP 2008].
- Factor 1



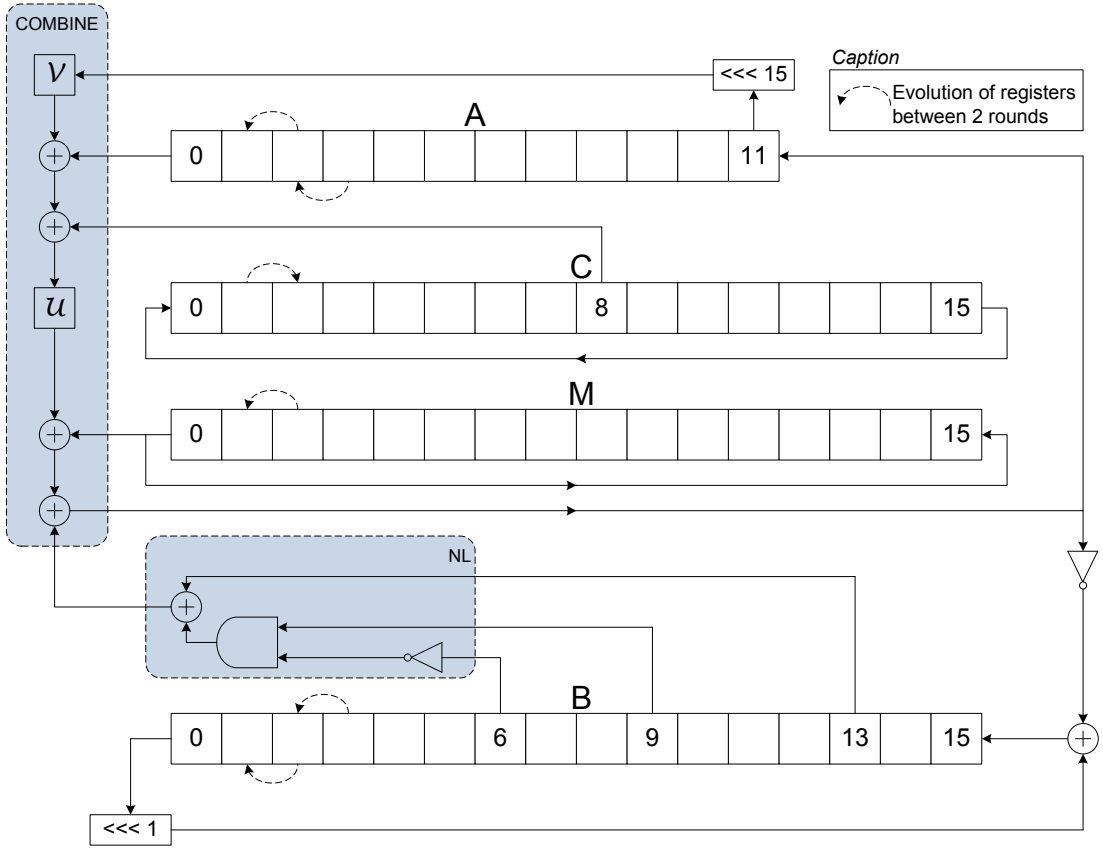
- Factor 2



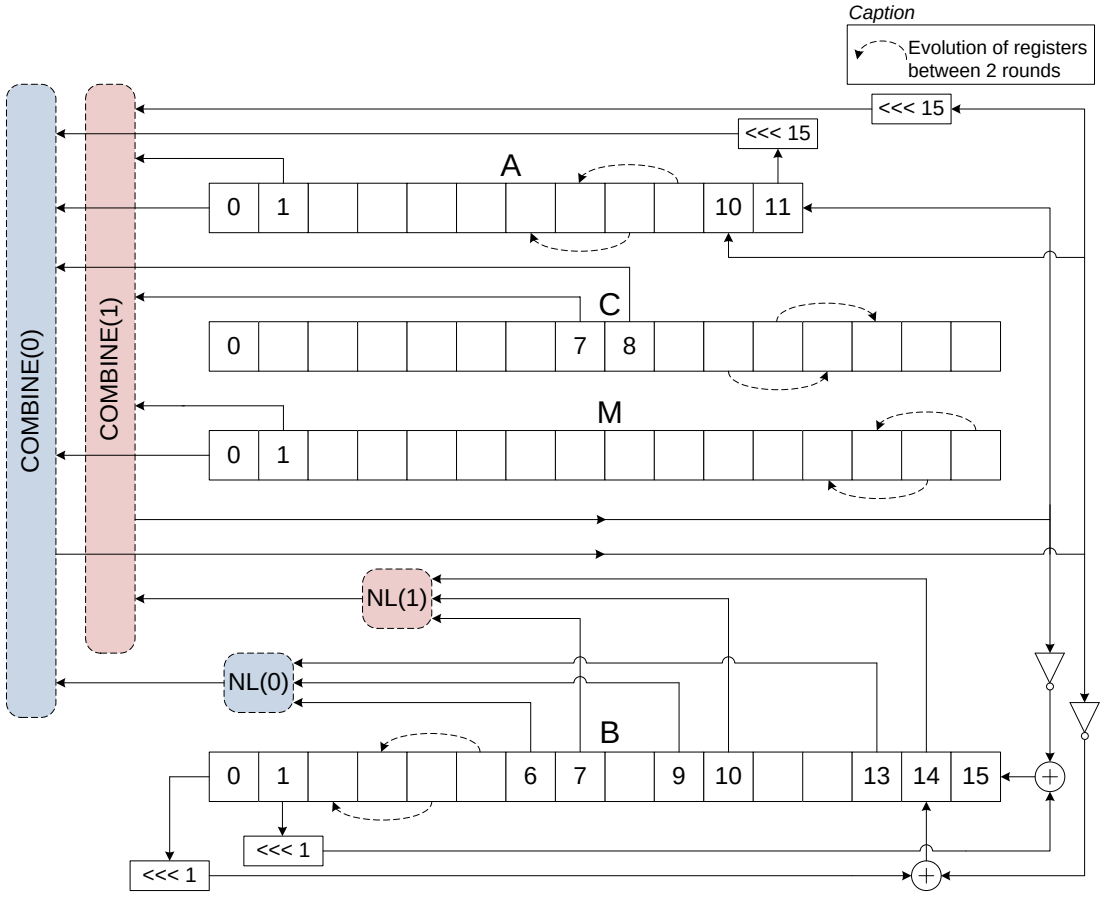
- ...
- Factor f



Let's Recall $\mathcal{P}$...



...Unfolding Method on $\mathcal{P}$ with Factor 2



- We let you imagine the complexity of circuits with growing f...

Why Using Unfolding Method?

1. Save register loading time.
   $\Rightarrow$ Impact on Frequency.
2. $\searrow$ # Clock Cycles per message block

- Yet, TP (for long messages) $= \dfrac{\text{Message block length} \times \text{Frequency}}{\#\text{ Clock Cycles per message block}}$.
- How can we explain an hypothetical gain?
- Ex.: when $f = 1 \rightarrow 2$, # Clock Cycles per message block/2.
  $\rightarrow$ To get TP $\nearrow$, Frequency $(f = 2) >$ Frequency $(f = 1)$ / 2.

Outline

Implementation Results (1/2)

- VHDL
- Xilinx Virtex-5 and ISE Tools (v11.1i)
- Optimization: Speed High
- Software padding
- Post-Place and Route Static Timing Report

Implementation Results (2/2)

Unfolding factor f	Area (slices)	Max. Freq. (MHz)	TP (Gbps)	TP/Area (Mbps/slice)
1	**1533**	**247**	2.634	1.718
2	1556	144	3.072	1.974
3	1607	101	3.232	**2.011**
4	1715	76	**3.242**	1.890
6	1884	50	3.200	1.698
8	1958	37	3.157	1.612

- More efficient: factor 3.
- Highest TP: factor 4.
- Method limit > 4.

Improvements on Previous Works

Reference	Area (slices)	Max. Freq. (MHz)	TP (Gbps)	TP/Area (Mbps/slice)
[Baldwin *et al.*]	2768	138	1.450	0.523
[Feron *et al.*]	1171	126	2.588	2.210
[Kobayashi *et al.*]	1251	214	2.282	1.390
[Detrey *et al.*]	**153**	**256**	2.051	**13.407**
Our work ($f = 4$)	1607	101	**3.232**	2.011

- Highest TP of the state-of-the-art

Outline

Conclusion and Future Works

- At first sight, the restricted parallelibility of Shabal seems to be a factor of TP limitation
- $\Rightarrow$ Fortunately, unrolling Shabal improves its TP (+23%)!
- Future Works
  - Unroll the 48 iterations and insert flip-flops (retiming)
  - Manual Place and Route
- Our work is available for free evaluation
  - shabal.com
- Significant contribution in the hardware benchmarking of the SHA-3 candidates

A Funny Conclusion...

Contrary to Chabal, Shabal can be fast and lightweight.

Thanks for your attention and your future questions, comments, suggestions, etc.!

SAPHIR2 Project

- SAPHIR2 project supported by the French Agence Nationale de la Recherche (ANR-08-VERS-014)
  - Security and Analysis of Primitives of Hashing Innovatory and Recent 2
- **Partners:** ANSSI, Cryptolog International, EADS Defence & Security (CSCSC), France Télécom, Gemalto, INRIA, LIENS, Sagem Sécurité, UVSQ.
- **Goals:**
  - follow and participate to the NIST SHA-3 Contest (cryptanalysis, implementations, optimizations, etc.),
  - to support the candidates proposed by the SAPHIR projects 1 and 2.

Partenaires SAPHIR2

CRYPTOLOG
Créateur de confiance

ANSSI

EADS
DEFENCE
& SECURITY

Ens

france telecom

gemalto
security to be free

INRIA

Sagem Sécurité
SAFRAN Group

UNIVERSITÉ DE VERSAILLES
SAINT-QUENTIN-EN-YVELINES