SPN-Hash: Improving the Provable Resistance Against Collision Attacks

JIALI CHOY, HUI HUI YAP, KHOONGMING KHOO,
DSO NATIONAL LABORATORIES, SINGAPORE

JIAN GUO
INSTITUTE FOR INFOCOMM RESEARCH, SINGAPORE

THOMAS PEYRIN, AXEL POSCHMANN
NANYANG TECHNOLOGICAL UNIVERSITY, SINGAPORE

CHIK-HOW TAN
TEMASEK LABORATORIES, NATIONAL UNIVERSITY OF SINGAPORE

What is SPN Hash?

- A Hash Function based on well-studied SPN structure.
- Generalized the optimal diffusion of SPN structure
  - So that more block sizes with good differential bounds can be constructed.
- First provable bound for true differential collision probability.
- Speed comparable to Grostl in software.
- Much lighter than SHA-3 candidates in hardware

Motivation

Hash	Proof of Security	Speed
PKC-based , e.g. VSH (very smooth hash)	Collision can be reduced to solving hard problems	Very Slow

Motivation

Hash		Proof of Security	Speed
PKC-based , e.g. VSH (very smooth hash)		Collision can be reduced to solving hard problems	Very Slow
Symmetric Hash	ARX (e.g. Skein, BLAKE)	Hard to determine characteristic DC	Very Fast

Motivation

Hash		Proof of Security	Speed
PKC-based , e.g. VSH (very smooth hash)		Collision can be reduced to solving hard problems	Very Slow
Symmetric Hash	ARX (e.g. Skein, BLAKE)	Hard to determine characteristic DC	Very Fast
	Davies-Meyer (e.g. Whirlpool, ARIRANG)	Low characteristic DC, not correspond to collision resistance	Fast

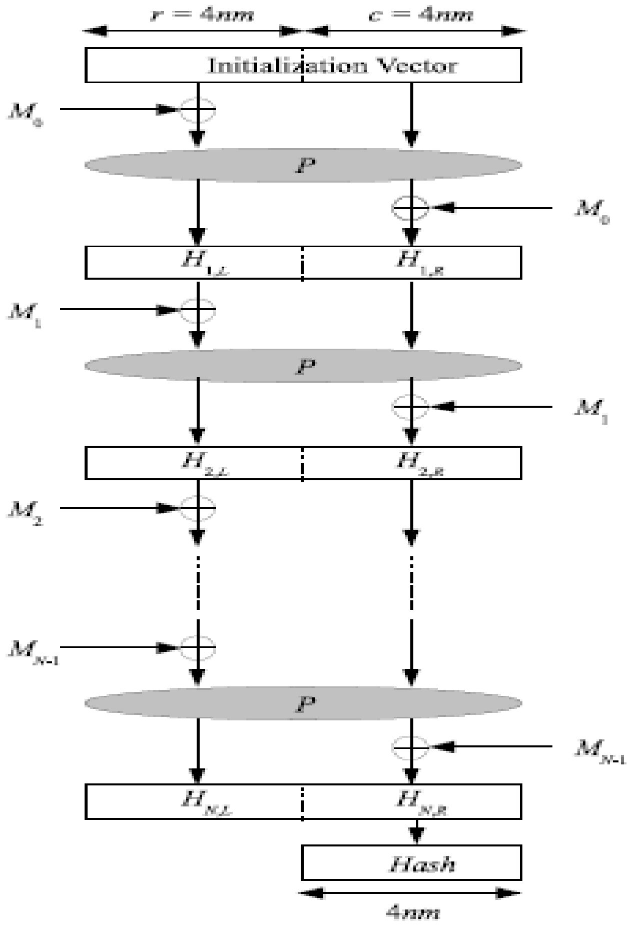
Motivation

Hash		Proof of Security	Speed
PKC-based , e.g. VSH (very smooth hash)		Collision can be reduced to solving hard problems	Very Slow
Symmetric Hash	ARX (e.g. Skein, BLAKE)	Hard to determine characteristic DC	Very Fast
	Davies-Meyer (e.g. Whirlpool, ARIRANG)	Low characteristic DC, not correspond to collision resistance	Fast
	Sponge-like (KECCAK, JH, PHOTON)	Low characteristic DC, correspond to collision resistance	Fast

Motivation

Hash		Proof of Security	Speed
PKC-based , e.g. VSH (very smooth hash)		Collision can be reduced to solving hard problems	Very Slow
Symmetric Hash	ARX (e.g. Skein, BLAKE)	Hard to determine characteristic DC	Very Fast
	Davies-Meyer (e.g. Whirlpool, ARIRANG)	Low characteristic DC, not correspond to collision resistance	Fast
	Sponge-like (KECCAK, JH)	Low characteristic DC, correspond to collision resistance	Fast
SPN Hash		Low (true) DC, Provable collision resistance	Fast

SPN Hash – Mode of Operation



1. Uses the JH mode of operation.
2. It is a sponge variant.
3. (a) **Sponge:**M_i only XORed to input.

   (b) **JH:** M_i is XORed to both input and output.
4. Reason for using JH:
   (a) DC of $P \Rightarrow$ collision resistance (similar to sponge).

   (b) Pre-image resistant (similar to sponge).

   (c) 2nd pre-image attack on sponge. No effective 2nd pre-image attack on JH.

SPN Hash - Permutation P

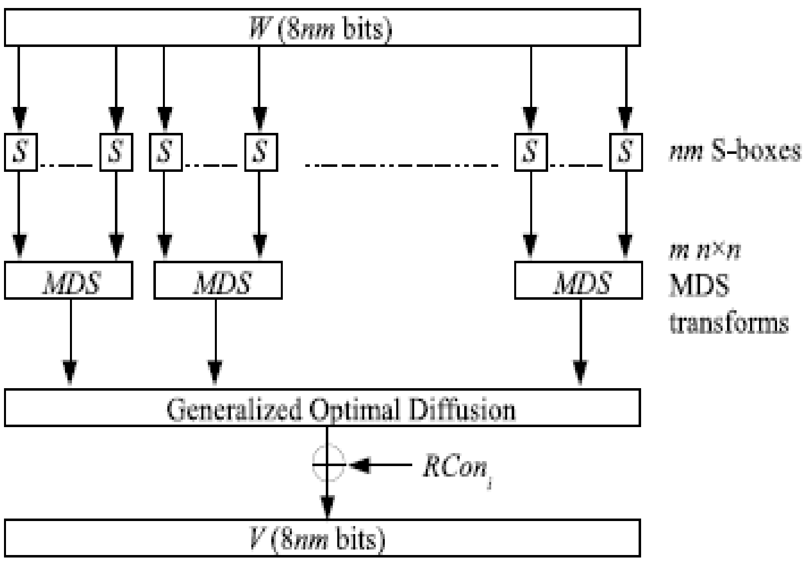


Fig. 4. The round function in permutation *P*

1. *P iterates SPN structure* **10 rounds.**

2. The substitution layer uses the AES S-box.

3. SPN similar to that used in AES:
   (a) There are *m* MDS's
   (b) Each MDS takes in *n* S-boxes

3. Known AES result, *m*=*n*.

4. In SPN hash, *m* divides *n*.

   We design new component **Generalized Optimal Diffusion** to achieve non-square block size.

SPN Hash - Permutation P

- **Q:** Why consider SPN design with non-square block size?
  A: So that we can design more block sizes.

MDS Size (n)	Block Size ($n \times n$) AES-like SPN	Hash Size
8 bytes	8×8 = 64 bytes = 512 bit	256 bit
16 bytes	16×16=256 bytes = 2048 bit	1024 bit

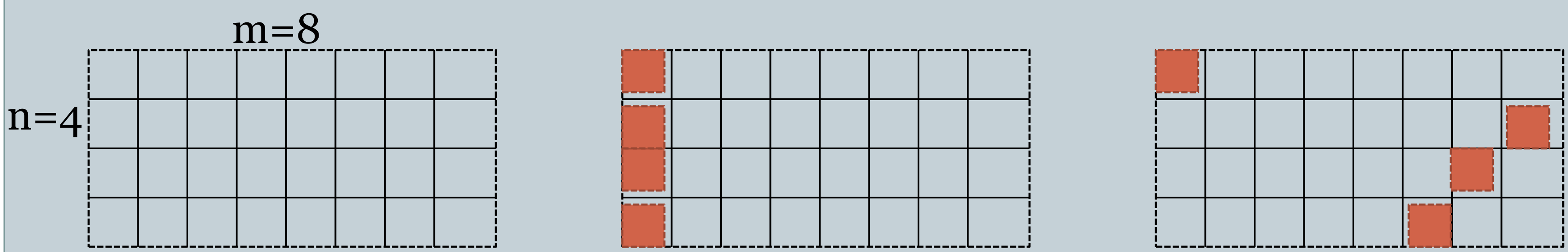
SPN Hash - Permutation P

- **Q:** Why consider SPN design with non-square block size?
  A: So that we can design more block sizes.

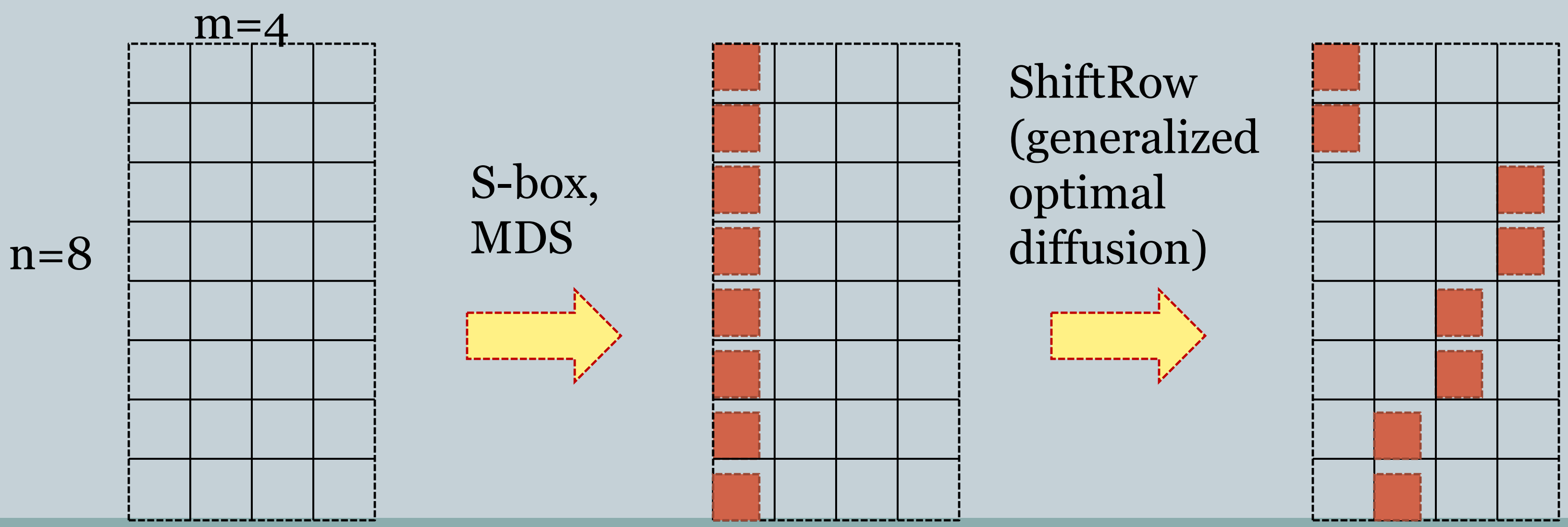
MDS Size (n)	Block Size ($n \times n$) Square SPN	Hash Size	Block Size ($m \times n$), m divides n Our SPN	Hash Size
8 bytes	8×8 = 64 bytes = 512 bit	256 bit	2×8 = 16 bytes=128 bit	64 bit
			4×8 = 32 bytes=256 bit	128 bit
			8×8 = 64 bytes=512 bit	256 bit
16 bytes	16×16=256 bytes = 2048 bit	1024 bit	2×16 = 32 bytes=256 bit	128 bit
			4×16 = 64 bytes=512 bit	256 bit
			8×16 = 128 bytes=1024 bit	512 bit
			16×16 = 256 bytes=2048 bit	1024 bit

How to Construct Non-Square Block Size

Rijmen-Daemen's Construction (optimal diffusion)



Our Construction (generalized optimal diffusion)



Differential Results on our SPN

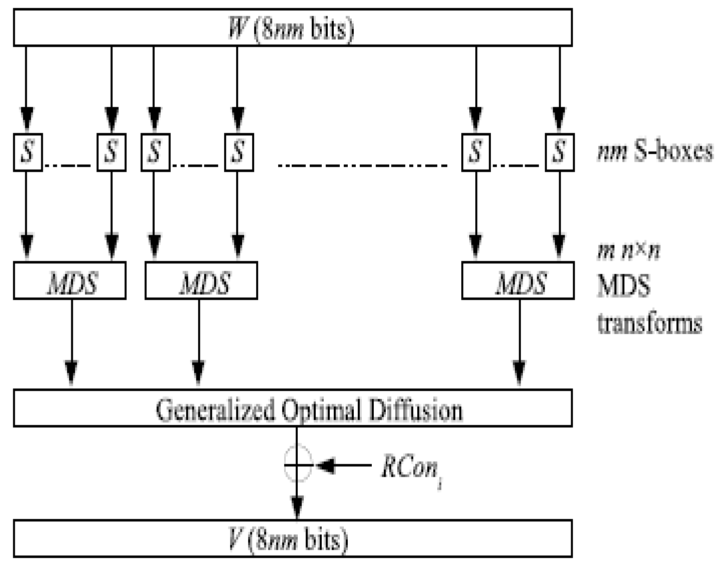


Fig. 4. The round function in permutation P

- **Q:** Why care about differential probability?

  A: Collision ⇔ Zero Output Differential.

- **Rijmen-Daemen result:** $m \geq n$.

  Every 4 rounds ⇔ $(n+1)^2$ active S-boxes

- **Our construction:** m divides n.

  Every 4 rounds ⇔ $(m+1)\times(n+1)$ active S-boxes

- **Example:** Construct 32-byte block.
  - AES Result(m=8,n=4): **25 active S-box.**
  - Our Result (m=4,n=8): **45 active S-box**

Differential Results on our SPN

- Counting Active S-boxes ⇔ Characteristic Differential Probability (Uses Wide-trail strategy of Rijmen-Daemen in [IMA Conference on Crypto and Coding 2001, Springer LNCS 2260, pp.222])

- We want:

  True Differential Probability ⇔ Actual Collision Probability (Uses Park et al.'s SDS result [FSE 2003, Springer LNCS 2887, pp. 247])

True Differential of SPN Hash

- **SPN Hash-128:** Block size = 256 bit. Hash output = 128-bit.
  - n=8, m=4 [32 AES S-box , Four 8×8 MDS].
  - True differential probability (256-bit block) ≤ $2^{-214.7}$.
  - Differential collision probability ≤ $2^{128} \times 2^{-214.7} = 2^{-86.7} < 2^{-64}$.

- **SPN Hash-256:** Block size = 512 bit. Hash output = 256-bit.
  - n=8, m=8 [64 AES S-box, Eight 8×8 MDS]
  - True differential probability (512-bit block) ≤ $2^{-429.5}$.
  - Differential collision probability ≤ $2^{256} \times 2^{-429.5} = 2^{-173.5} < 2^{-128}$.

- **SPN Hash-512:** Block size = 1024 bit. Hash output = 512-bit.
  - n=16, m=8 [128 AES S-box, Eight 16×16 MDS].
  - True differential probability (1024-bit block) ≤ 2^{-816}.
  - Differential collision probability ≤ $2^{512} \times 2^{-816} = 2^{-304} < 2^{-256}$.

Comparison with Existing Hash

- Among SHA-2 and SHA-3 hashes, only one have true differential bound and that is ECHO.

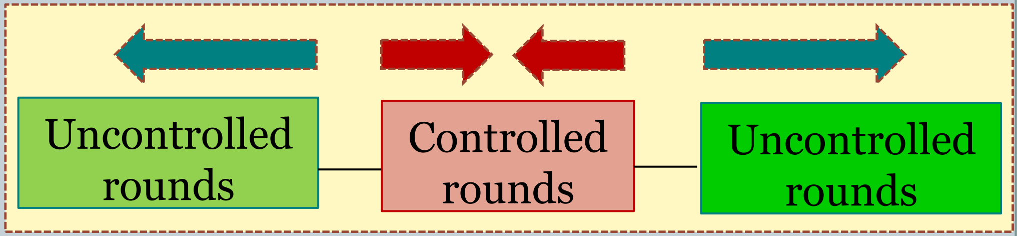
	SPN Hash-512	ECHO-512
Block Size	1024 bit	2048 bit
True Diff Probability of Block	2^{-816}	2^{-452}
Output (after truncation)	512 bit	512 bit
True Differential Collision Probability (after truncation)	2^{-304} (truncate 1024→512)	- (truncate 2048→512)

- True differential of ECHO block worse than SPN-hash block.
- ECHO truncate more bits, differential probability suffer even more.

Why no collision prob for ECHO?

Rebound Attack – Overview

- Divide an attack into two phases: Controlled rounds and Uncontrolled rounds

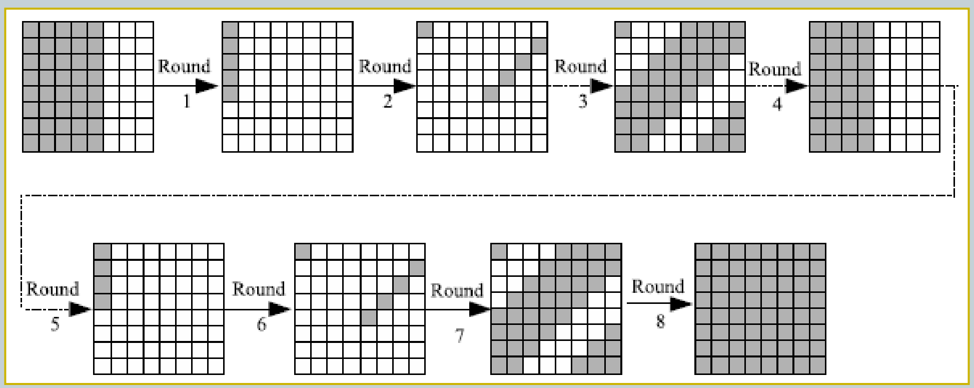


- Controlled rounds
  - Efficient meet-in-the-middle
  - Exploits available freedom degrees in the middle of a differential path
  - Non Full Super-Sbox Analysis
- Uncontrolled rounds
  - Mainly probabilistic
  - Solutions of the controlled rounds are computed backwards and forwards
- Can result in a **distinguishing attack**

Rebound Attack

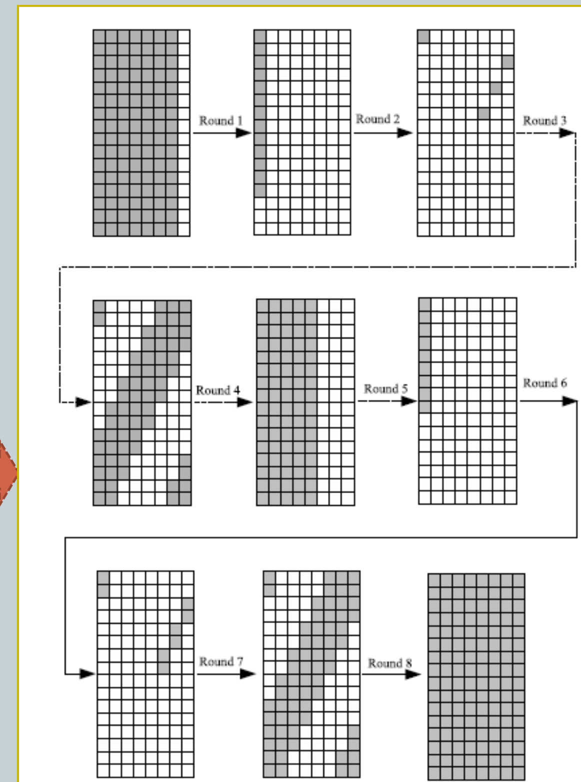
- View 512-bit and 1024-bit internal state of P as a 8×8 and 16×8 matrix of bytes
- 8-round differential paths
- Coloured cell: active byte;

  White cell: passive byte



512-bit P

1024-bit P

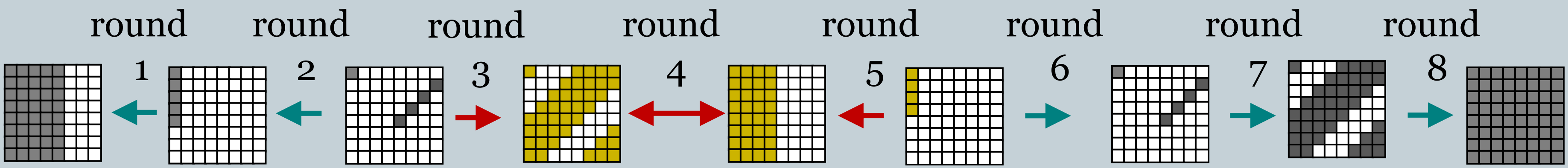


Rebound Attack- Non-Full Active Super-Sbox

- The non-full active Super-Sbox method allows attacker to **control 3 rounds in the middle** (**controlled rounds**): A starting point can be obtained with time 1 on average and 2^8 memory (512-bit P) / 2^{16} memory (1024-bit P)

- The rest of the path is fulfilled probabilistically (**uncontrolled rounds**): In the example of 512-bit P below, we have to pay a probability of approximately 2^{-48}

- Need to ensure **enough freedom degrees** to find a pair of values following the path: In example, need 2^{48} starting points but can choose 2^{72} differences at the start of controlled rounds



8-round differential path for 512-bit P

Rebound Attack

- **Q**: How does this translate to a distinguishing attack?

  A: We obtained distinguishers:
  - 512-bit P: Finding a valid pair for the whole 8-round path requires 2^{48} operations and 2^{8} memory. Ideal case requires 2^{96} computations.
  - 1024-bit P: Finding a valid pair for the whole 8-round path requires 2^{88} operations and 2^{16} memory. Ideal case requires 2^{256} computations.

$\Rightarrow$ Secure against rebound attack since P comprises 10 round functions.

Hardware Implementation

- Implement lightweight SPN Hash 128-bit and 256-bit.
- **Optimization:** Serialize the 8 by 8 MDS matrix over GF(2^8).
- **Problem:** Not easy to find byte-based serialized 8 by 8 MDS matrix, by using method of PHOTON hash design.
- **Our Solution:** Use parallel copies of the PHOTON 8 by 8 MDS matrix over GF(2^4).

Serialized matrix over GF(2^8)

$$Q = (A_{256})^8 = \begin{pmatrix} 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \\ 2&4&2&11&2&8&5&6 \end{pmatrix}^8 = \begin{pmatrix} 2&4&2&11&2&8&5&6 \\ 12&9&8&13&7&7&5&2 \\ 4&4&13&13&9&4&13&9 \\ 1&6&5&1&12&13&15&14 \\ 15&12&9&13&14&5&14&13 \\ 9&14&5&15&4&12&9&6 \\ 12&2&2&10&3&1&1&14 \\ 15&1&13&10&5&10&2&3 \end{pmatrix}$$

$$X = (X_1 || X_2) \rightarrow (Q \cdot X_1 || Q \cdot X_2),$$

where $X \in GF(2^8), X_1, X_2 \in GF(2^4)$

Lightweight implementation

- Besides Serialized MDS, we also use other optimizations like compact AES S-box, efficient use of registers, etc…
- Comparison with SHA-3 candidates:

Digest size	Alg.	Ref.	Msg. size	Technology	Area [GE]	Latency [clk]	T'put@100KHz [kbps]	FOM [nbps/GE^2]
128	SPN-Hash-128		256	UMC 0.18	2 777	710	36.1	2 338
	SPN-Hash-128		256	*estimate*	4 600	230	55.7	2 627
256	SPN-Hash-256		512	UMC 0.18	4 625	1 430	35.8	837
	SPN-Hash-256		512	*estimate*	8 500	230	111.3	1 541
	BLAKE-32	[23]	512	UMC 0.18	13 575	816	62.8	340
	GROSTL-224/256	[34]	512	AMS 0.35	14 622	196	261.2	1 222
	SKEIN-256-256	[34]	256	AMS 0.35	12 890	1 034	24.8	149

Software Implementation

- Expect speed of SPN-Hash 256 comparable to Grostl-256 (22 cycles/byte).
  - Use same number of AES S-boxes.
  - T-Table implementation independent of MDS coefficients.
  - ShiftByte is done implicitly in T-table look-up.
  - SPN-Hash process 256-bit message in 10 rounds compared to Grostl-256 which process 512-bit message in 20 rounds.
- SPN-Hash 128 should run at similar speed.
  - Takes half the message bit, process half the operations.

Conclusion

- We have designed new hash function SPN-hash.
  - Output Sizes: 128-bit, 256-bit, 512-bit
- Provable differential collision bound for all these sizes.
- Also secure against pre-image, 2^{nd} pre-image and rebound attacks.
- Much lighter than existing SHA-3 candidates in Hardware.
- Efficiency comparable to Grostl in Software.

Thank You!

THE FULL VERSION OF THE PAPER CAN BE FOUND ON EPRINT 2012/234

HTTP://EPRINT.IACR.ORG/2012/234