Fang-ting HUANG, Dan FENG, Wen XIA, Wen ZHOU, Yu-cheng ZHANG, Min FU, Chun-tao JIANG, Yu-kun ZHOU, 2018. Enhancing security of NVM-based main memory with dynamic Feistel network mapping. *Frontiers of Information Technology & Electronic Engineering*, 19(7):847-863.
https://doi.org/10.1631/FITEE.1601652

Enhancing security of NVM-based main memory with dynamic Feistel network mapping

Key words: Non-volatile memory (NVM); Endurance; Wear leveling; Timing attack

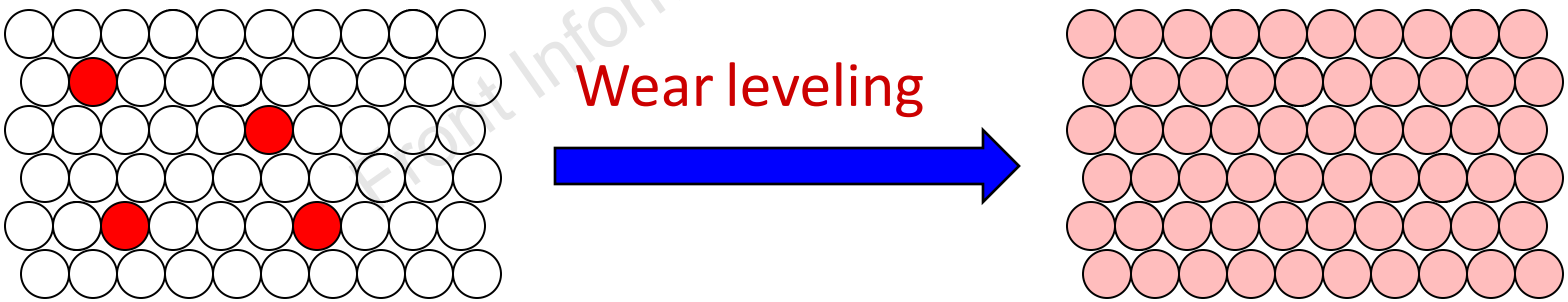
Corresponding author: Dan FENG
E-mail: dfeng@hust.edu.cn

Lifetime problem

Limited endurance (10^7-10^8):

1. Non-uniform write traffic of real world applications (20x less lifetime);

2. Malicious attack: continuously write to a few physical lines ($1000ns * 10^8 = 100s$).



Distribute write traffic evenly

Remapping timing attack (RTA)

Based on two facts:

1. Remapping incurs extra latency;
2. Asymmetry in PCM write time.

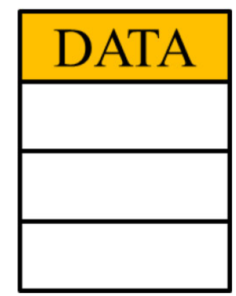
Remapping latency:

1. Wear-leveling schemes need to remap the heavy written lines to the less ones;

2. Remapping: read data from the old location and write to the new one

3. Remapping incurs extra reads and writes!

4. Remapping interval: the number of normal writes before triggering a remapping (expected to be no more than 1%).



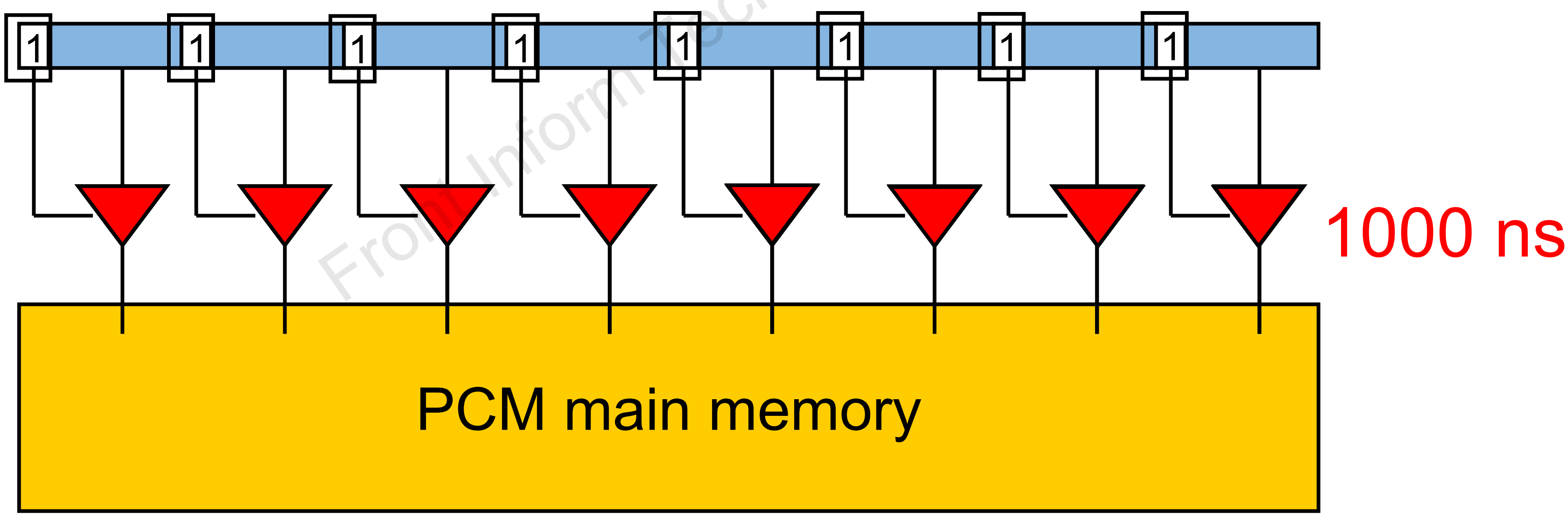
Asymmetry in PCM write time

1. Writing bit '1' is 8x slower than writing bit '0';

2. The duration of a write is determined by the worst-case write time of all cells;

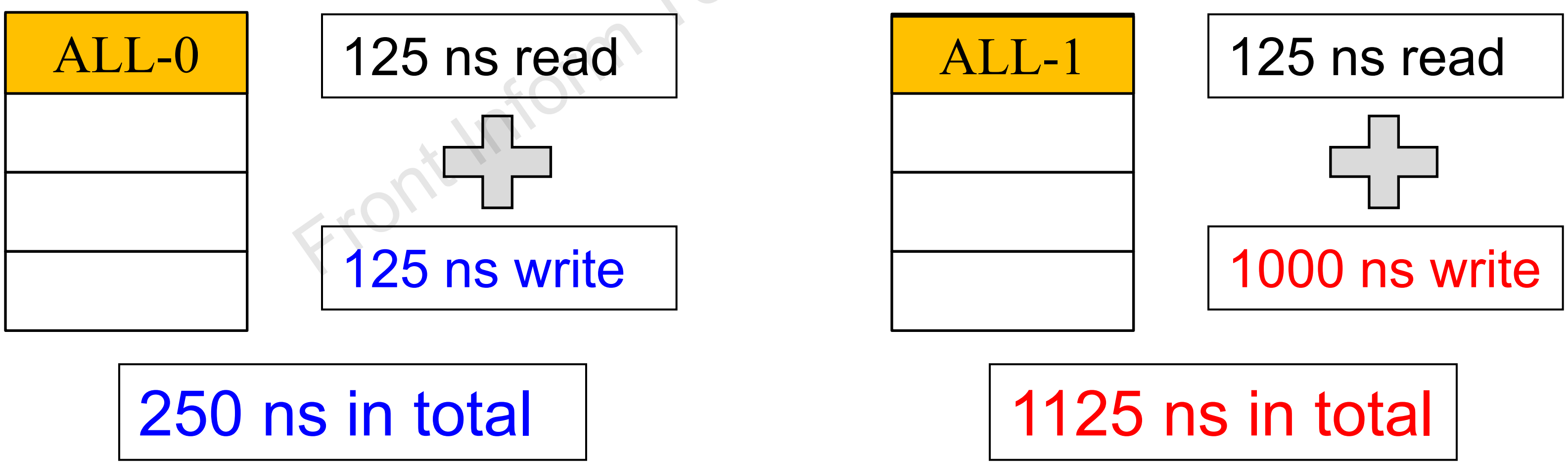
3. The latency of writing ALL-0 is 125 ns;

4. The latency of writing ALL-1 is 1000 ns.

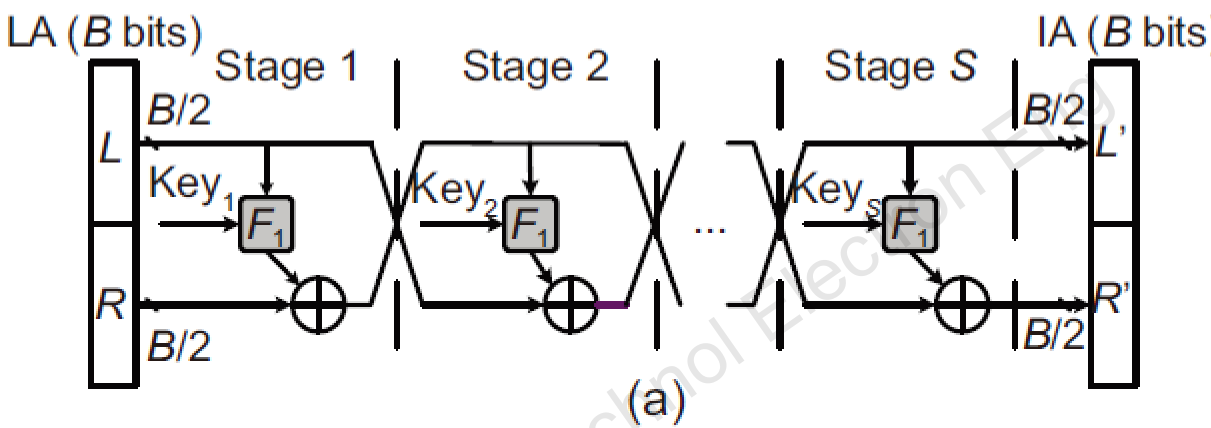


Remapping timing attack

1. Remapping latency differs with different remapping data.
2. It is possible to perform a carefully designed sequence of writes and infer a specific mapping transformation of the wear-leveling schemes.



Feistel network mapping



Encryption of a multi-stage Feistel network

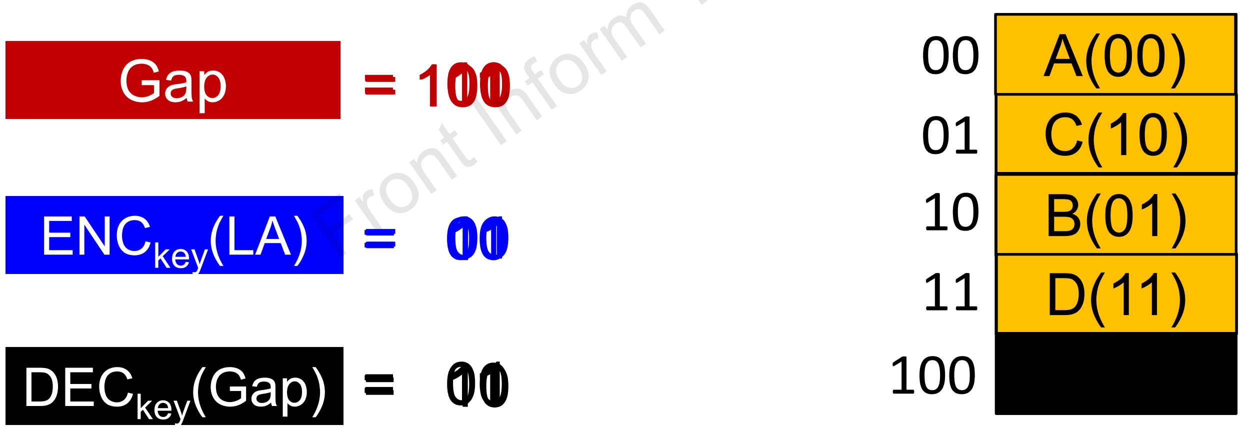
1: Low hardware overhead, easy to implement

2. Enhance the security by adding the number of stages (six stages are enough to resist RTA)

3. Low performance overhead, 1 CPU cycle per stage

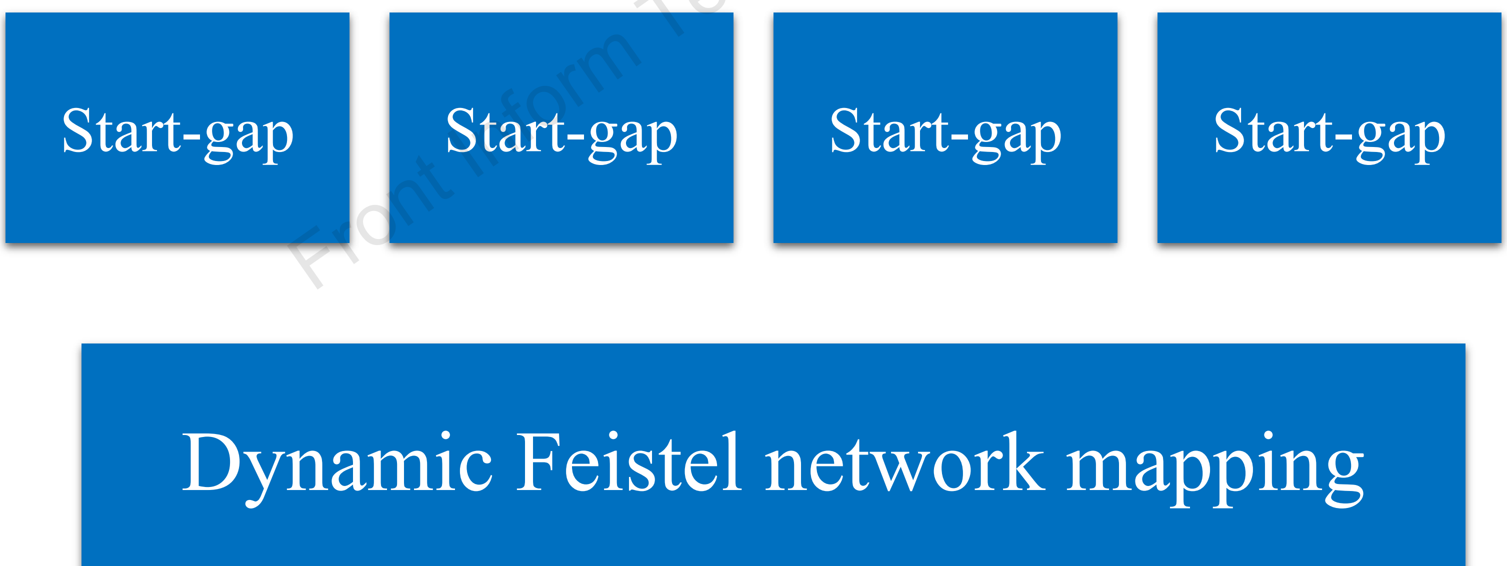
Remapping of dynamic Feistel network mapping

- Remap function: $ENC_{key}(LA)=PA$, $DEC_{key}(PA)=LA$
- ***Gap*** tracks the physical address of the spare line
- The LA will be remapped to the gap line, and can be calculated by $DEC_{key}(Gap)$.
- After remapping, the old location of the remapped LA ($ENC_{key}(LA)$) is the new spare line and is pointed by *Gap*.



Security RBSG

1. Hierarchical, two-level wear-leveling scheme
2. Outer-level: employ dynamic Feistel network mapping in the whole memory space to ensure security
3. The memory space is divided into multiple fixed-sized sub-regions
4. Inter-level: employ start-gap in each sub-region for its low overhead



Major results

(1) Lifetime evaluation

1) We choose the number of Feistel network stages as 7;

2) Security RBSG can achieve comparable lifetime as two-level SR under both RAA and BPA with the recommended configuration.

(2) Lifetime under RAA

1) Lifetime increases as the inner-level remapping interval decreases and the number of regions increases;

2) Security RBSG achieves 67.2% of the ideal lifetime under RAA.

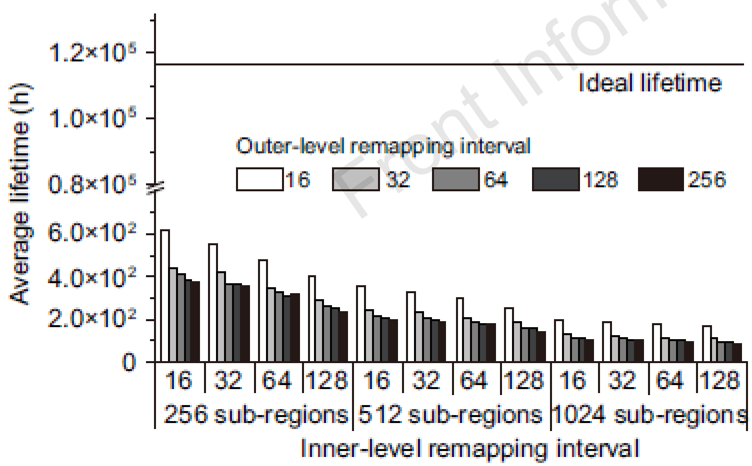


Fig. 11 Average lifetime of two-level SR under an RTA

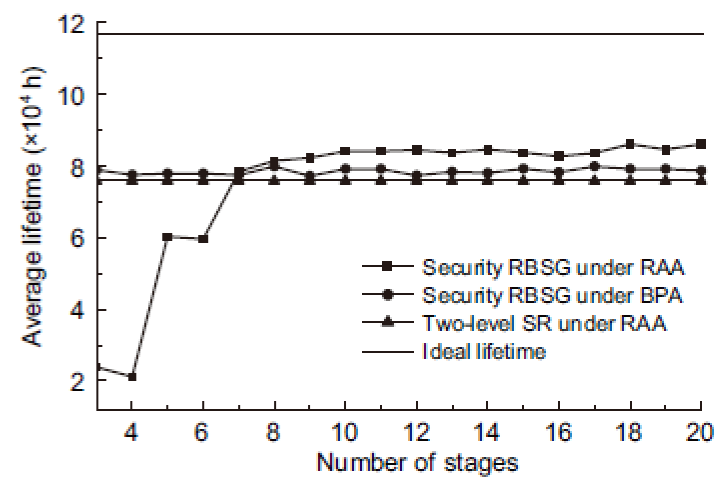


Fig. 14 Average lifetime of different DFN stages

Major results

(3) The IPC degradation of security RBSG shows similar characteristics to that of two-level SR.

(4) The geometric mean of IPC degradation of security RBSG (2.4%) is slightly larger than that of two-level SR (2.1%).

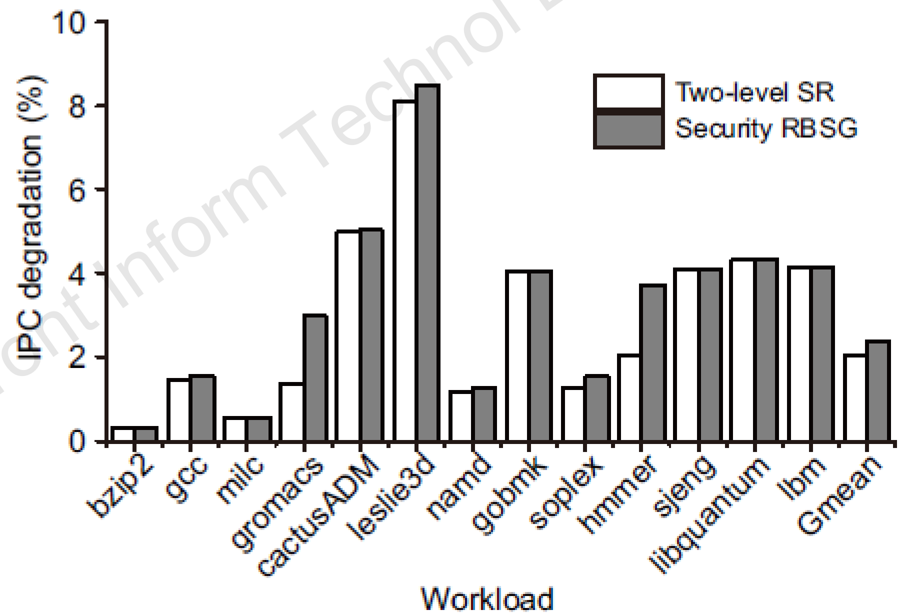


Fig. 17 IPC degradation of two-level SR and security RBSG

Summary

1. Remapping timing attack (RTA);
2. RTA model against SR;
3. Our solution: security RBSG;
4. Evaluation results are given.