

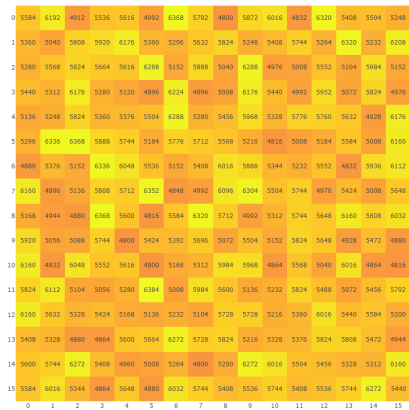
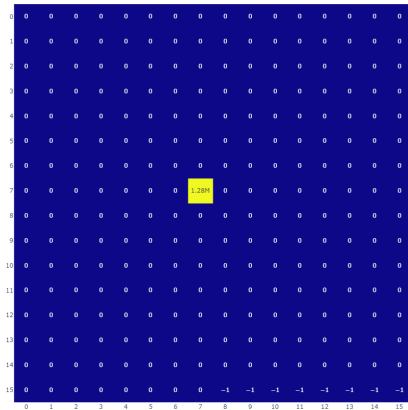
Tool for Management of Flash Memory Wear-Leveling on Embedded System Device

Martin Havlík

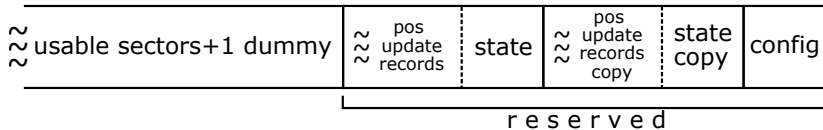
Supervisor: Ing. Václav Šímek



June 14, 2023

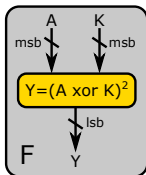
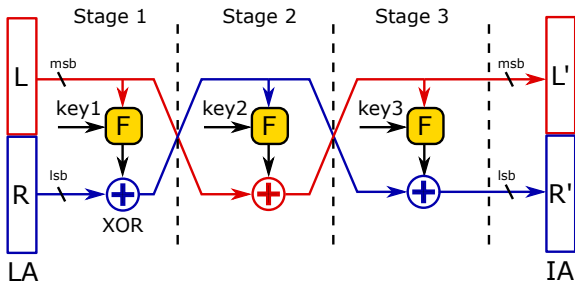


- ESP32, SPI NOR Flash ($\approx 100\,000$ erases lifetime)
- existing wear-levelling component
- **goal:** memory wear monitoring tool

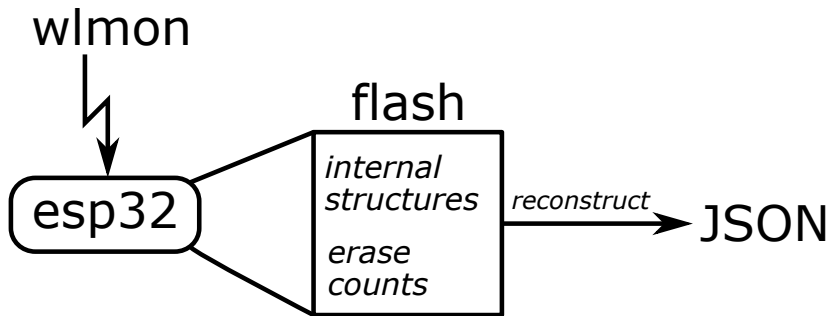


!!! *long term monitoring not possible in current ESP-IDF implementation*

- beyond the thesis assignment
- `wl_advanced` – fix & extend WL from ESP-IDF
- + **per sector erase count tracking**
- + **1:1 address randomization** based on unbalanced Feistel Network



- 1 wlmon embedded C/C++ back-end



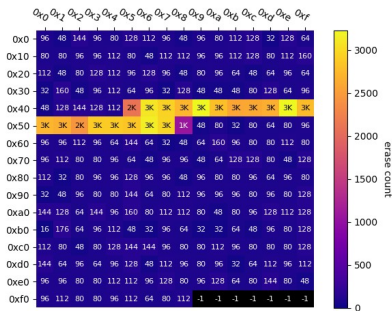
2 espw1mon PC side Python front-end

espw1mon v0.4

wl_mode: advanced

start_addr: 0
full_mem_size: 1048576
page_size: 4096
sector_size: 4096
update_rate: 16
wr_size: 16
version: 2
temp_buff_size: 32
crc: 1337287392

pos: 208
max_pos: 249
move_count: 18
access_count: 0
max_count: 16
block_size: 4096
version: 2
device_id: 4141172863
cycle_count: 0
feistel_keys: (162, 53, 119)
crc: 2523486302



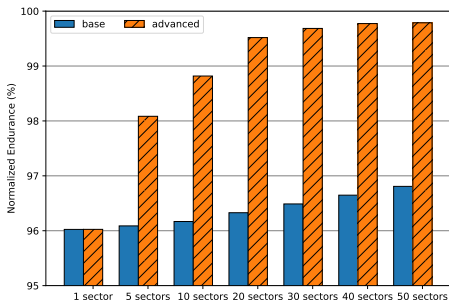
Sector count: 248

Overall erase count: 75040

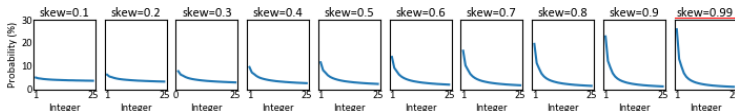
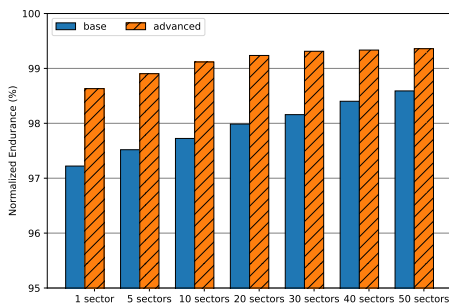
[Toggle erase count visibility](#)

[Export heatmap with Plotly](#)

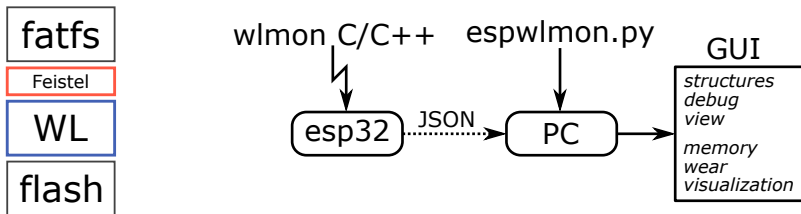
Constant address



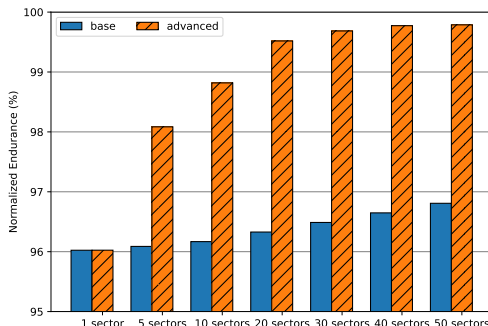
Zipfian distribution



<https://github.com/omnitex/espwlmon> under Apache 2.0



`wl_advanced` – per sector
erase count tracking &
address randomization



- 1 Jak jste se vypořádal s **problémem identifikace paměti typu NOR flash** (viz "specific device ID still could not be matched", konec 2.3.2, str. 8) ?

- ① Jak jste se vypořádal s **problémem identifikace paměti typu NOR flash** (viz "specific device ID still could not be matched", konec 2.3.2, str. 8) ?
- **esptool.py flash_id** → manufacturer 0x20 device 0x4016
 - **JEP106BG** STANDARD MANUFACTURERS IDENTIFICATION CODE → XMC
 - <https://github.com/flashrom/flashrom/blob/master/include/flashchips.h>
 - 32Mbit, 100K PE/sector, XIP, 20 years data retention

- 2 Z technické zprávy plyne (viz např. podkap. 4.2, str. 16), že **stávající řešení je vhodné spíše pro paměti typu NOR flash s kapacitami řádově v jednotkách MB**. Máte nějaké náměty, **jak toto řešení přizpůsobit pro interní/externí paměti typu NOR flash o vyšších kapacitách** (např. 16 MB na ESP32 s 24 či 32 bitovou adresací v SPI) ?

2 Z technické zprávy plyne (viz např. podkap. 4.2, str. 16), že **stávající řešení je vhodné spíše pro paměti typu NOR flash s kapacitami řádově v jednotkách MB**. Máte nějaké náměty, **jak toto řešení přizpůsobit pro interní/externí paměti typu NOR flash o vyšších kapacitách** (např. 16 MB na ESP32 s 24 či 32 bitovou adresací v SPI) ?

- Feistel Network max 2^{16} sectors (256MB of 4KB), could be extended
- Max safe number of sectors = 6250 ($6250 \cdot 16 = 100\,000$)
→ (≈ 24 MB of 4KB) → regions
- JSON might be too verbose (erase counts)

- 3 Vyjádřete se k **nejhoršímu možnému případu počtu opakování** "This must be repeated *until the network produces a valid IA.*", viz podkap. 4.2.2, str. 18.

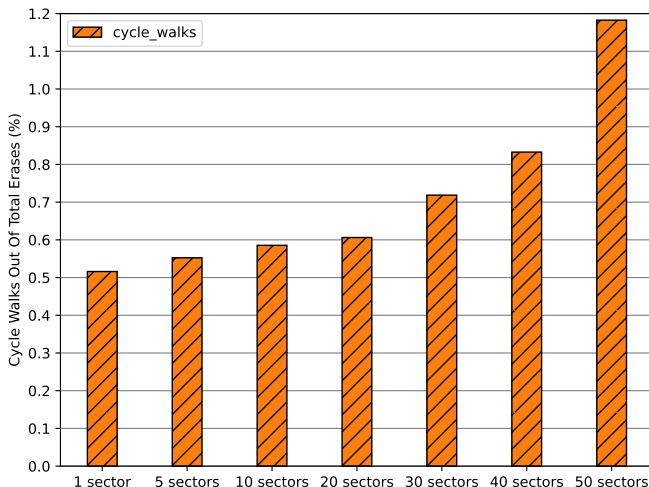
- 3 Vyjádřete se k **nejhoršímu možnému případu počtu opakování** "This must be repeated *until the network produces a valid IA.*", viz podkap. 4.2.2, str. 18.

- full simulated partition erase, until lifetime, 100 runs

| 0 cw | 1 cw | 2 cw | 3 cw | 4 cw |
|-----------|-----------|----------|-----------|------|
| 43.6681 % | 43.6681 % | 10.917 % | 1.74672 % | 0 % |

- cycle walk (cw) – additional Feistel Network iteration on invalid Intermediate Address (IA)

- 3 Vyjádřete se k **nejhoršímu možnému případu počtu opakování** "This must be repeated *until the network produces a valid IA.*", viz podkap. 4.2.2, str. 18.



- 4 Jak náročné by bylo využít stávající realizační výstup pro práci se **sektory o jiné velikosti** (např. 512 B) než stávající (4 kB) ?

- 4 Jak náročné by bylo využít stávající realizační výstup pro práci se **sektory o jiné velikosti** (např. 512 B) než stávající (4 kB) ?
- most of `wl_advanced` should be dynamic and flexible
 - integrating with 512 B modes – `safe`, `performance`
 - correct sector size propagation, `fatfs`

QURESHI, M. K., KARIDIS, J., FRANCESCHINI, M., SRINIVASAN, V., LASTRAS, L. et al. **Enhancing Lifetime and Security of PCM-Based Main Memory with Start-Gap Wear Leveling**. In: *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*. New York, NY, USA: Association for Computing Machinery, 2009, p. 14–23. MICRO 42. DOI: 10.1145/1669112.1669117. ISBN 9781605587981.

YANG, Y. AND ZHU, J. **Write Skew and Zipf Distribution: Evidence and Implications**. ACM Trans. Storage. New York, NY, USA: Association for Computing Machinery. 2016, vol. 12, no. 4. DOI: 10.1145/2908557. ISSN 1553-3077.

Mr. Havlík has accomplished great results in his Bachelor's thesis project TOOL FOR MANAGEMENT OF FLASH MEMORY WEAR LEVELING ON EMBEDDED SYSTEM DEVICE. He's been able to learn all the necessary context of the wear-levelling discipline in a reasonably short time, successfully **analyzed the design and real function of the ESP-IDF wear-levelling component** implementation and **created the required tooling for its monitoring and management**. In addition to the basic project requirements, he has **proposed and implemented his own improvements** for extending the wear-levelled media life. The theoretical part covers the topic at a very good quality too. Therefore I appraise Mr. Havlik's project at the highest possible grade.

Martin Vychodil – Software manager, Espressif Systems (Czech)