DeepCover DS28C36: A Hardware Vulnerability Identification and Exploitation Using T-Test and Double Laser Fault Injection

Karim M. Abdellatif and Olivier Hériveaux
Ledger, Donjon
karim.abdellatif@ledger.fr, olivier.heriveaux@ledger.fr

***Abstract*—DeepCover [6] is a secure authenticator circuit family developed by Analog Devices. It was designed to provide cryptographic functions, true random number generation, and EEPROM secure storage. DS28C36 is one of the DeepCover family, which is widely used in secure boot and secure download for IoT. It has been recently deployed in the Coldcard Mk4 hardware wallet [3] as a second secure element to enhance its security. In this paper, we present for the first time, a detailed evaluation for the DS28C36 secure EEPROM against Laser Fault Injection (LFI). In the context of a black box approach, we prove by experimental results that the chip resists single fault attacks. In order to overcome this, we present the use of leakage detection such as Welch's T-test to facilitate finding the correct moments for injecting successful faults, which is not common in Fault Injection (FI) as this method has been used only for Side-Channel Attacks (SCAs). By using this knowledge, we found two moments for injecting laser pulses to extract the protected EEPROM user pages with 99% success rate. The attack can be reproduced within a day. The presented attack negatively impacts the users of DS28C36 (including Coldcard Mk4).**

***Keywords*—DeepCover, DS28C36, Laser Fault Injection, Secure EEPROM.**

I. Introduction

Hardware security continues to be a high priority for several embedded systems vendors. Such systems like secure elements rely on high-level hardware security to prevent all sorts of device-level security. Such threats appear at circuit-level, where an attacker can measure or physically influence the computation/operation performed by the circuit. Side-channel attacks (SCAs) exploit additional sources of information (physical observations) such as observing environmental parameters of the device during its operation such as timing information, power consumption, electromagnetic emissions (EM), and sound. Malicious data modifications are caused by fault attacks, which can be performed by injecting faults using laser/optical [15], electromagnetic [5] [1], and glitches (power and clock) [14]. These attacks pose a serious threat to modern chips.

According to Analog Devices, DeepCover secure authenticators [6] integrate advanced physical security to offer a highest level of protection against physical tampering and reverse engineering. The DS28C36 [7] is a DeepCover secure authenticator that has the following features:

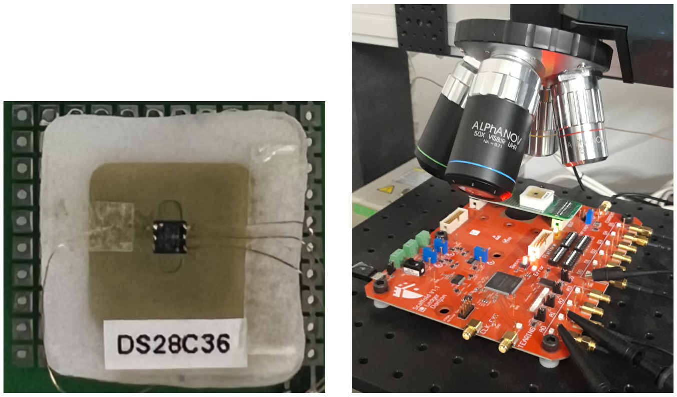
- ECC-256 computation engine
- FIPS 180 SHA-256 computation engine
- SHA-256 OTP (One-Time Pad) encrypted R/W of configurable memory through ECDH established key
- RNG with NIST SP 800-90B compliant entropy source with function to read out
- 17-Bit one-time settable, nonvolatile decrement-only counter with authenticated read
- **8Kbits of EEPROM for user data, keys, and certificates**

According to the vendor's short data sheet [7], to provide the most secure key storage, DeepCover embeds security solutions mask sensitive data under multiple layers of advanced security. Invasive and noninvasive countermeasures are implemented including active die shield, encrypted storage of keys, and algorithmic methods to protect against device-level security attacks.

In this work, we deal with LFI, which is considered one of the most efficient and precised fault injection techniques. Skorobogatov et al. [15] proved that laser is well suited for fault attacks. After that, LFI has been considered as a benchmark for fault injection because it allows to inject faults with maximum feasible precision in both timing and location on the chip. Several examples in the literature presented practical attacks using LFI. Viera et al. [16] presented a permanent modification into the flash of an STM32 chip using LFI. Obermaier et al. [13] managed to break the protection mechanism used in STM32F0 by shedding the light on the chip. Another practical example was presented by Hériveaux [10] to break the secure EEPROM of ATECC using LFI.

Motivations: Coldcard Mk4 has recently used DS28C36 as a second secure element in addition to Microchip ATECC608B to enhance the product security [4]. This was done after attacking ATECC608B (the only secure element in Mk3 version) using LFI as shown in [11]. In the Mk4 version, the user seed is encrypted and stored in the ATECC608B. The encryption key is shared between the MCU (which was also attacked before in [11]), and user protected EEPROM pages of DS28C36 (pages 14 and 15). So, an attacker must attack three different devices in order to recover the protected master key, which is considered as a hard challenge.

Contributions: We present for the first time a vulnerability identification and exploitation of the protected user EEPROM



(a) Decapped chip (b) Laser setup

Fig. 1. Setup

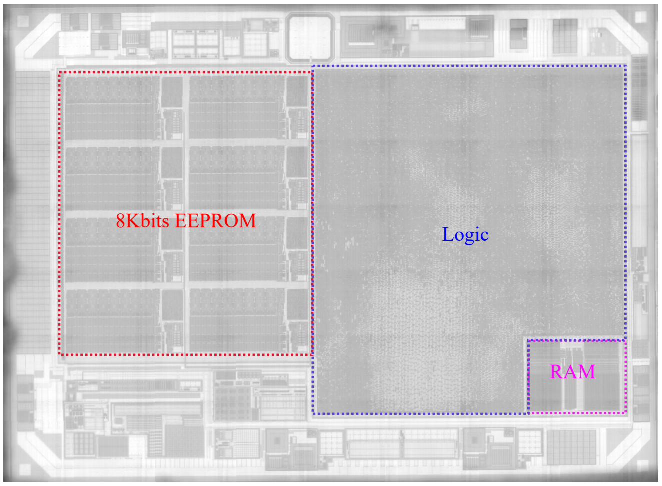


Fig. 2. Infrared backside image of the circuit

pages of the DS28C36 chip. First, we highlight a deep evaluation for the read page command using LFI and show by experimental results that this chip resists single fault attacks. Second, we present how leakage detection techniques such as T-test can be useful to detect the manipulation timing of the page protection setting during the execution of the read page command. Third, we invest locating the page protection setting in order to inject multiple laser pulses to attack successfully the user protected EEPROM pages.

This paper is organized as follows. **Section II** describes the experimental setup used in this work. **Section III** discuses the read page command and the attack scenario. **Section IV** highlights the use of single laser pulse in characterizing the read page command. **Section V** proposes using leakage detection for better understanding the read page command. **Section VI** shows how multiple laser pulses are used to successfully extract the protected user EEPROM pages of DS28C36. **Section VII** shows our responsible disclosure to the vendor. **Section VIII** concludes this work.

II. Setup

A. Sample preparation

In order to perform LFI, backside package decapsulation is needed to access to the silicon substrate of the device. Therefore, we performed this decapsulation and we used infra-red imaging to capture a detailed picture of the chip and have a look at its internal layout. Fig. 1a shows the chip after being decapped from the backside. Fig. 2 shows the internal structure of the chip resulting from the infra-red imaging. According to the short data sheet, we are able to identify three main blocks: EEPROM, RAM, and logic.

B. Our setup

In order to inject laser pulses, we used an infra-red pulsed laser source and a microscope for focusing. We used a 10X objective (laser beam is about 5μm diameter). A scaffold board [9] was used to communicate with chip by sending I^{2}C commands and also for controlling the synchronization during the fault injection. Fig. 1b shows the DuT fixed on the scaffold board and also the laser objectives. In addition, a Tektronix MSO44 200 MHz digital oscilloscope with a maximum sampling rate of 6.25 GS/s, was used to measure and capture the instantaneous power consumption of the DUT during the experiment.

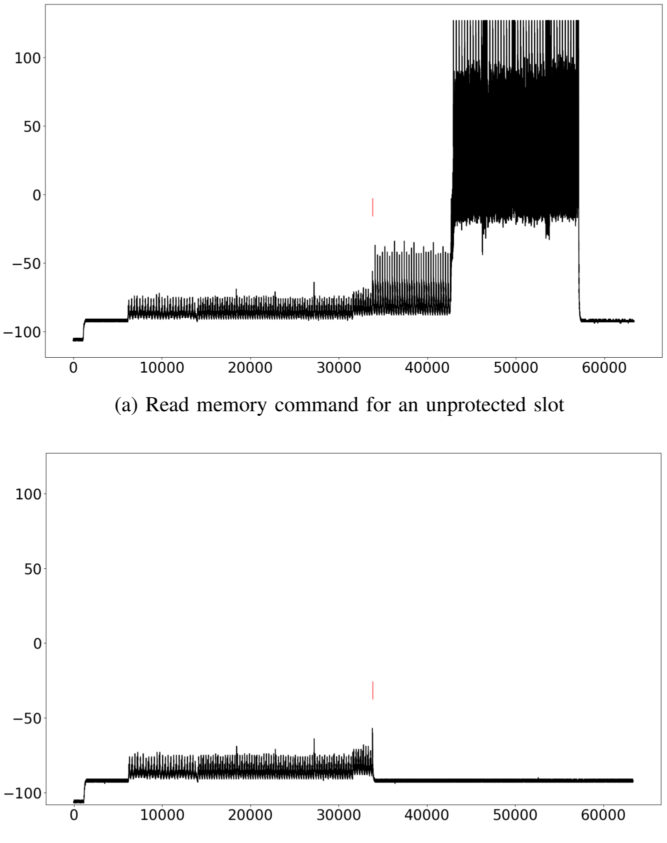
III. Read Page command and attack scenario

A. Read page command

According the open source project of the Coldcard Mk4 version [4], the secure EEPROM of DS28C36 has 32 pages and the page length is 32 bytes. Pages from 0 to 15 are classified as user pages. Pages from 22 to 24 map to private keys over the Nist P-256 curve, and pages 16 to 21 store the X and Y components of the corresponding public keys. Pages from 25 and 26, are marked as secret pages. Pages from 27 to 29 are dedicated to decrement counter, random number, GPIO, respectively. The last two pages are reserved to RAM buffers. Since not all the EEPROM memory bits are visible in this page mapping, we suspect some EEPROM memory to be used for storing the security configuration, such as pages read and write fuses. Note: from [4], the hardware wallet manufacturer uses the protected user pages 14 and 15 to store the secrets.

After the chip preparation, we started to monitor the power consumption of the chip during the read page command. The main idea of this step is to differentiate between the chip behavior when the page is unprotected and after being protected. We selected the user page number 7 as an example.

Fig. 3(a) shows the power consumption of the chip in case of reading this page before enabling the read-protection. Fig. 3(b) presents the power consumption during the read page command of the same page after being protected (locked). From the two figures, we can conclude that the chip executes the same state machine before the red line and the divergence starts at the red line before reading the EEPROM page. Moreover, we observed the absence of any jitter during the command execution.



(a) Read memory command for an unprotected slot

(b) Read memory command for a protected slot

Fig. 3. Power consumption during read memory command

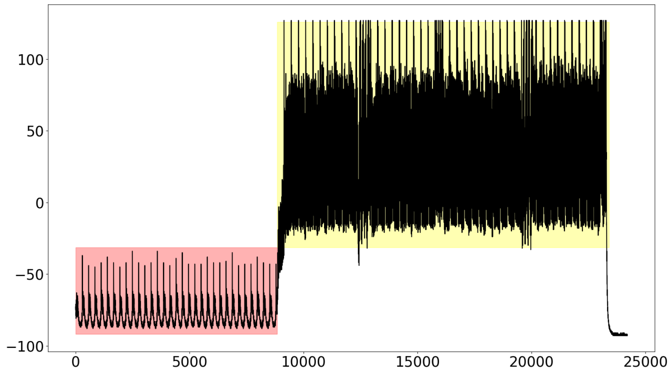


Fig. 4. Zoom on the EEPROM reading

According to the data sheet [7], the chip supports encrypted EEPROM storage. This means that the EEPROM data is encrypted. During the successful reading, the chip decrypts the encrypted page content. We can understand that from focusing on the EEPROM reading in case of the unprotected slot as shown in Fig. 4. The red zone indicates the 32-byte reading (32 peaks). Then, there are four identical patterns that indicate the decryption process. There are no details about this process in their short data sheet.

Algorithm 1: Attack methodology

```
while True do
    PrepareFault();
    ChipRestart();
    ReadPage();
    SaveLog();
    MoveLaser()
```

B. Attack scenario

After the initial study on the read page command, the next step is dedicated to perform the LFI. We use the infra-red pulsed laser source and a microscope for focusing. We used a 10X objective (laser beam is about 5μm diameter). The main idea is to write data to a specific page (a user page for example) and attack this page after activating the read-protect command. The attack scenario is shown in Algorithm 1. $PrepareFault()$ prepares the fault parameters such as pulse width, offset, and laser power. Before starting the attack, the chip is restarted and the read page command is executed ($ChipRestart()$ and $ReadPage()$). After faulting the read page command, the attack log is saved, including the response of the chip, the laser beam position, the power consumption trace and the fault injection timings. Then, the laser beam moves to another spot to scan another chip location.

IV. Single fault injection

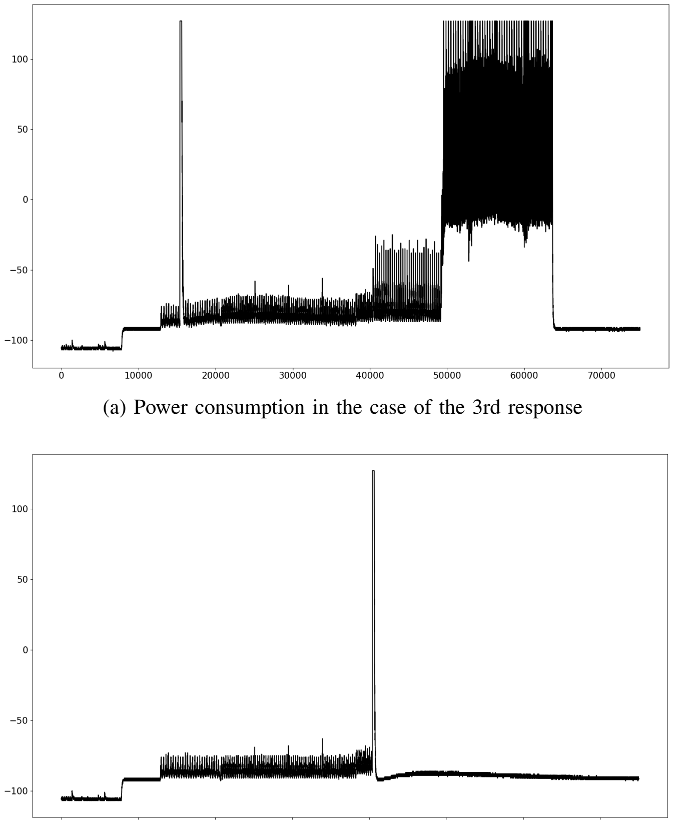
In this section, we will highlight the first experiment using single laser pulse during the execution of read page command. The single fault laser pulse was injected randomly before the divergence which is highlighted by the red line in Fig. 3(b). We scanned the overall chip. We randomized the laser power source between 20% and 80% from it's maximum value (2.4W).

We obtained five different responses shown in Table I. The first response is the normal response which is obtained when the page is protected (locked) and starts with `2155`. The second response is obtained when the chip gets crashed. `NACK` means that there is an error during the I^{2}C communication. We can note that the only interesting results are the last two results. Both start with `21aa` which is the indication of accepting the read page command. Unfortunately, no one of them is equivalent to the stored value. We discovered that third response is the value stored in page 17 (a public key page), and it's permanently unprotected.

For better understanding, we monitored the power consumption of the chip during the last two responses. The power consumption in case of third response shown in Fig. 5(a), is equivalent to the case when the page is unprotected and confirms our previous finding that indicated the similarity of this value to slot 17. Regarding the 4th response (see Fig. 5(b), the EEPROM reading is not executed as the power consumption looks like the protected case. Therefore, It's not an interesting fault. We located the last two responses on the layout of the chip as shown in Fig. 6.

TABLE I
SINGLE FAULT RESPONSES

Number	Chip response
0	2155ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff
1	ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff
2	NACK
3	21aab8289516978a7b25eb1d8a317f6c6a71718b4d47de4754ac32a1d1c5adb7d324
4	21aa208cfc9a7dc7fcdb5437775fea79aa2c95f5795ed2bfe883082a2ada0585694f



(a) Power consumption in the case of the 3rd response

(b) Power consumption in the case of the 4th response

Fig. 5. Power consumption during single fault

We spent several weeks scanning the overall chip using the single laser pulse and we always obtained the same outputs (the five different responses shown in Table I). Therefore, we decided to understand deeply the difference between the unprotected and protected page on the level of the power consumption.

V. LEAKAGE DETECTION

In the previous section, we observed by experimental results that using a single laser pulse (single fault) is not efficient against attacking the read page command. As this is a secure chip, we consider it could be protected against single LFI and includes a multiple checking counter-measure.

To solve this challenge, we studied the difference between the unprotected and protected page, statistically. The main idea is to find the timing when the page protection setting (bit/bits) is manipulated. Hence, we decided to focus on using leakage detection techniques used for side-channel attacks (SCAs).

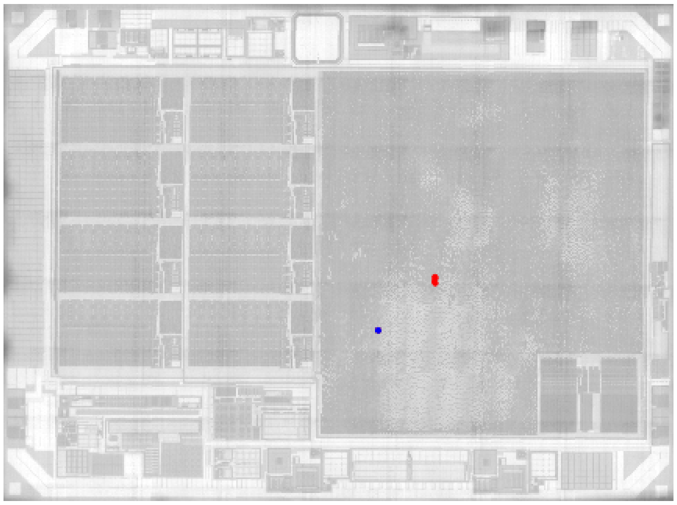


Fig. 6. Single fault positions: 3rd and 4th responses are located in red and blue, respectively

Leakage detection is a methodology to identify leakage moments which contain sensitive information. It is widely used in SCAs to reduce the computation complexity of security evaluations and improve the efficiency of the SCAs. Several methods have been used to identify the amount of leakage such as T-test [8] and NICV [2]. In this paper, we will focus on the T-test. The main idea of using T-test in SCAs is to compare the leakages of sensitive operations such as block cipher, with fixed plaintexts (and key) to the leakages of the same implementation with random plaintexts (and fixed key). If a significant difference of means is observed between the leakages, it is concluded that the device leaks during this operation. The Welch's T-test is calculated as shown in Eq. 1, where μ, S^2 and N are the mean, variance, and number of traces, respectively, for the two sets of data (0 and 1)

$$t = \frac{\mu_0 - \mu_1}{\sqrt{\frac{S_0^2}{N_0} + \frac{S_1^2}{N_1}}} \tag{1}$$

We will apply the T-test, which is always used in SCAs to detect sensitive operations, in fault injection (FI). The main purpose is to detect when sensitive bits are processed. More precisely, we will try to locate on the power consumption trace, the manipulation of the page protection bit/bits.

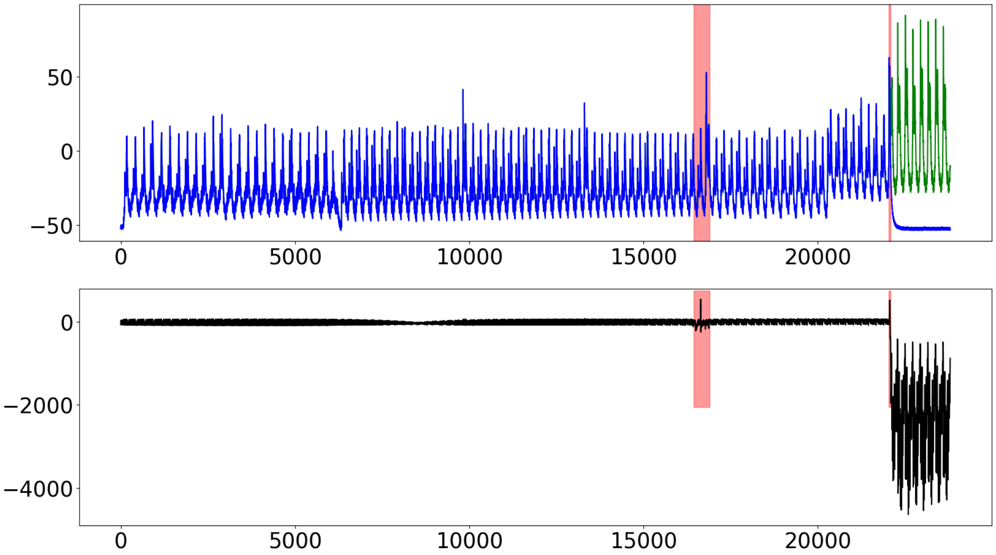


Fig. 7. Upper: green curve indicates the unprotected slot and the blue indicates the same page when it's protected. Below: T-test result

This can be done by performing the T-test between two sets of data. The first set is collected when the page is unprotected and the second set is collected when the same page is protected. By performing the T-test between the two different sets, we will look for the significant difference, that may guide us to the correct timing for injecting faults. Also, it will indicate if one laser pulse (single fault) or many laser pulses (multiple faults) are needed.

We collected the two sets of traces for page 8 (Note: page 7 was locked previously and can't be unlocked). The number of collected traces per each set is 100K traces. We limited the number of samples until a slight increase after the red line (see Fig. 3). In addition, we increased the vertical resolution for reducing the measurement noise and improving the statistical result. Prior to performing the T-test, we precisely aligned all the traces using cross-correlation [12]. Fig. 7 shows the average of each data set (upper) and the result T-test is shown below. We can see two zones where there are significant peaks, highlighted in red. Note: we are not interested in the peaks after the divergence because the two sets are already different during these moments. The two highlighted red zones indicates the significant difference between the two sets which indicate the manipulation of the page protection bit/bits.

The obtained results from the T-test confirms what we concluded in **Section IV**, single laser pulse (single fault) is not sufficient to extract the protected page. In addition, it concludes that the chip is protected against single fault attacks by performing an algorithmic verification in the two red zones highlighted by the T-test.

VI. Multiple laser pulses

After finding the correct timing for injecting multiple laser pulses using the T-test as described previously. The next step is to validate this result practically by scanning the overall chip surface with injecting multiple laser pulses in the timing where the T-test showed the significant difference (two red zones in Fig. 7).

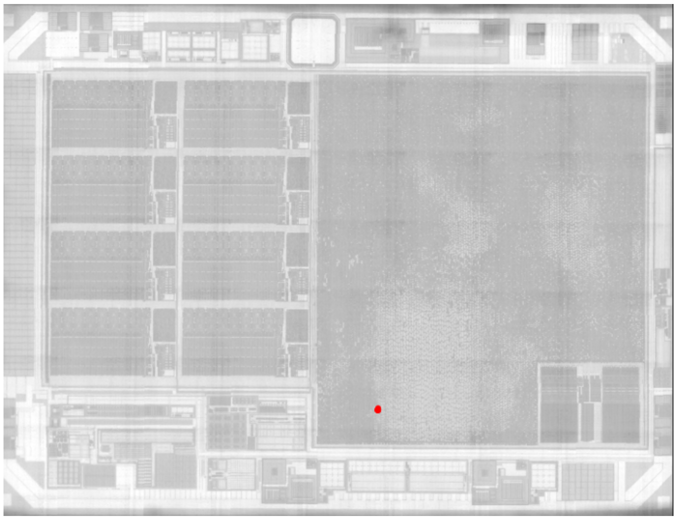
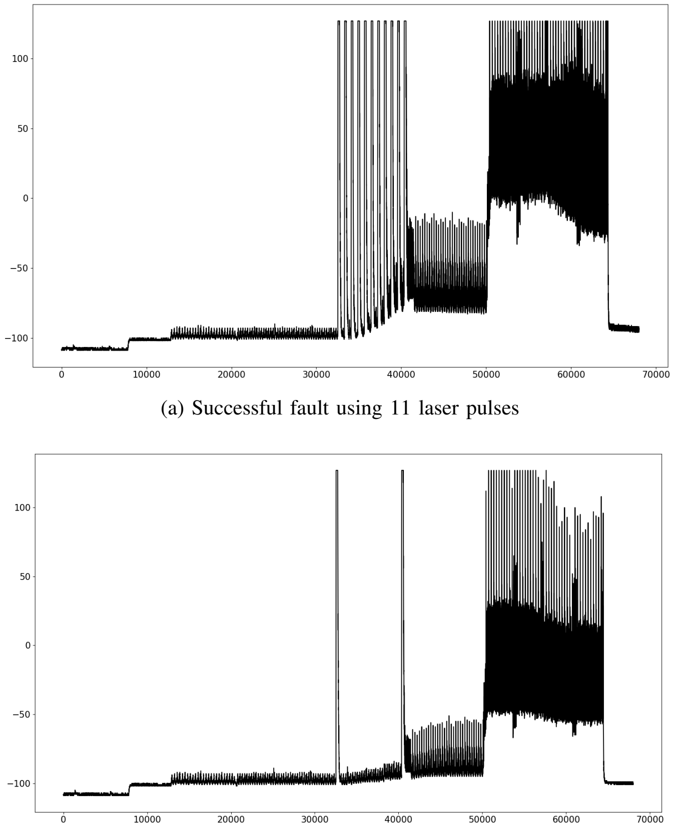


Fig. 8. Positions of successful faults

We repeated the same attack scenario shown in Algorithm 1 with randomizing the laser power source from 20% to 80% during the chip scan. In order to reduce the search for the fault parameters, we decided to inject a row of multiple faults that cover the two red timing zones highlighted in the T-test result (Fig. 7) and also in between. By this way, we managed to dump the same data stored in page 8 when we inject this row of multiple pulses in the locations shown in Fig. 8. The number of pulses was 11 and the interval between each pulse is 2 μs. The power consumption in case of the successful fault



(a) Successful fault using 11 laser pulses

(b) Refined double fault

Fig. 9. Power consumption during successful faults

is shown in Fig. 9(a). This successful fault was obtained with 55% of the laser power source. After fixing the successful location, we refined again the number of pulses and we found that the attack is also successful with a double fault attack on the same two peaks shown in the T-test result. Fig. 9(b) shows the power consumption of the successful fault in case of injecting only two laser pulses. The success rate of the attack after focusing on the correct location is 99%.

We tried all the user pages (from 0 to 15) and the attack worked successfully. However, in case of permanent-protected pages used for P256 curve private keys, the chip passed a fixed unidentified value for these pages. This means that the presented attack is applicable only to the user protected pages (the case of Coldcard Mk4).

A. Discussion

From the above results, we can conclude that the leakage detection using Welch's T-test added a significant contribution in finding the correct moments for injection faults and allowed us to perform a successful attack on the protected user pages of the EEPROM, using multiple pulses. Without this method, the time consumed in finding the correct number of faults is very long and it could be difficult to find. Therefore, we advise the vendors to use this technique during the evaluation phase against fault injection to help in having robust designs against fault injection.

VII. Disclosure

The presented attack in this paper has been reported to Analog Devices before any publication. We would like to thank them for their collaboration during the responsibility disclosure. In addition, we also reported this work to the hardware wallet manufacturer Coinkite.

VIII. Conclusion

This paper presented for the first time, a black box attack on the user protected EEPROM pages of DS28C36, which is from the DeepCover family developed by Analog Devices. We proved by experimental results that the chip has been protected against single fault attacks. Thanks to leakage detection techniques that helped us to identify the manipulation of the page protection bit/bits during the read page command. With this knowledge, we managed to extract the user protected pages using multiple laser pulses with 99% success rate. Future work includes further research to investigate another attack path to extract the P256 curve private key pages.

References

[1] Karim M Abdellatif and Olivier Hériveaux. Silicontoaster: A cheap and programmable em injector for extracting secrets. In *2020 Workshop on Fault Detection and Tolerance in Cryptography (FDTC)*, pages 35–40. IEEE, 2020.

[2] Shivam Bhasin, Jean-Luc Danger, Sylvain Guilley, and Zakaria Najm. Nicv: normalized inter-class variance for detection of side-channel leakage. In *2014 International Symposium on Electromagnetic Compatibility, Tokyo*, pages 310–313. IEEE, 2014.

[3] COLDCARD. COLDCARD Mk4 Improvements. https://coldcard.com/docs/coldcard-mk4.

[4] COLDCARD. Dual Secure Elements. https://github.com/Coldcard/firmware/blob/master/docs/mk4-secure-elements.md.

[5] Amine Dehbaoui, Jean-Max Dutertre, Bruno Robisson, and Assia Tria. Electromagnetic Transient Faults Injection on a Hardware and a Software Implementations of AES. In *2012 Workshop on Fault Diagnosis and Tolerance in Cryptography*, pages 7–15. IEEE, 2012.

[6] Analog Devices. DeepCover: Embedded Security. https://www.analog.com/media/en/technical-documentation/tech-articles/deepcover-embedded-security.pdf.

[7] Analog Devices. DS28C36 Data Sheet. https://www.analog.com/media/en/technical-documentation/data-sheets/ds28c36.pdf.

[8] Benjamin Jun Gilbert Goodwill, Josh Jaffe, Pankaj Rohatgi, et al. A testing methodology for side-channel resistance validation. In *NIST non-invasive attack testing workshop*, volume 7, 2011.

[9] O. Heriveaux. Scaffold. https://github.com/Ledger-Donjon/scaffold?files=1/.

[10] Olivier Hériveaux. Defeating a secure element with multiple laser fault injections. In *Symposium sur la sécurité des technologies de l'information et des communications-SSTIC 2021*, 2021.

[11] Olivier Hériveaux. Triple exploit chain with laser fault injection on a secure element. In *2022 Workshop on Fault Detection and Tolerance in Cryptography (FDTC)*, pages 9–17. IEEE, 2022.

[12] Bozhi Liu, Kemeng Chen, Minjun Seo, Janet M Roveda, and Roman Lysecky. Methods and analysis of automated trace alignment under power obfuscation in side channel attacks. *Journal of Hardware and Systems Security*, 5(2):127–142, 2021.

[13] Johannes Obermaier and Stefan Tatschner. Shedding too much light on a microcontroller's firmware protection. In *WOOT*, 2017.

[14] Colin O'Flynn. Fault Injection using Crowbars on Embedded Systems. *IACR Cryptol. ePrint Arch.*, 2016:810, 2016.

[15] Sergei P Skorobogatov and Ross J Anderson. Optical Fault Induction Attacks. In *International workshop on cryptographic hardware and embedded systems*, pages 2–12. Springer, 2002.

[16] Raphael Viera, Jean-Max Dutertre, Mathieu Dumont, and Pierre-Alain Moëllic. Permanent laser fault injection into the flash memory of a microcontroller. In *2021 19th IEEE International New Circuits and Systems Conference (NEWCAS)*, pages 1–4. IEEE, 2021.