GLITCH ATTACKS IN MCUs

DIANA MILENA CEPEDA RIAÑO
FELIPE ALBERTO CASTRO BENAVIDES

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍAS
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA
2021

GLITCH ATTACKS IN MCUs

DIANA MILENA CEPEDA RIAÑO
FELIPE ALBERTO CASTRO BENAVIDES

Trabajo de Grado para optar al título de
Ingeniero Electrónico

Director:
Elkim Felipe Roa Fuentes,
PhD. en Filosofía.

Co-Director:
Luis Eduardo Rueda Guerrero,
MSc. de Ciencia.

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍAS
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA
2021

CONTENIDO

LISTA DE FIGURAS

pág.

RESUMEN

TÍTULO: ATAQUES POR GLITCH EN MICROCONTROLADORES [*]

AUTORES: DIANA MILENA CEPEDA RIAÑO, FELIPE ALBERTO CASTRO BENAVIDES [**]

PALABRAS CLAVE: GLITCH, CANAL LATERAL, ATAQUE, MCU, MICROCONTROLADOR.

DESCRIPCIÓN:

Los ataques de software han sido los más comunes. Sin embargo, últimamente una de las preocupaciones de la industria electrónica es la seguridad del hardware debido al aumento de dispositivos conectados a Internet o IoT. Por lo tanto, es necesario estudiar los ataques existentes. El ataque por glitch es uno de ellos. Su implementación requiere un bajo coste y puede llevar a la fuga de información de los dispositivos. En este trabajo se han realizado múltiples ataques de glitch por alimentación utilizando la tarjeta ChipWhisperer. La tensión de alimentación baja a 0 [V] durante el ataque, y se incrementa la duración del glitch hasta tener un ataque exitoso. Por otro lado, se propuso y se probó un circuito de glitch para gestionar la amplitud y duración del glitch. Un circuito divisor de resistencias permitió controlar la amplitud, y la tarjeta ChipWhisperer ayudaba a manipular la duración. Se encontró que la duración del glitch se mantiene aproximadamente constante a diferentes frecuencias utilizando sólo la ChipWhisperer, y el circuito propuesto permite ver la energía de la señal de glitch para tener un ataque exitoso; esa energía es aproximadamente constante a diferentes frecuencias y amplitudes. El trabajo futuro debe concentrarse en encontrar la causa del salto de instrucción, esto se puede lograr mejorando la simulación de Turpial. Además, debe investigarse la relación entre la energía de la señal de glitch y un ataque exitoso.

---

[*] Trabajo de grado

[**] Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Elkim Felipe Roa Fuentes, PhD. en Filosofía.

ABSTRACT

TITLE: GLITCH ATTACK IN MCUs [*]

AUTHOR: DIANA MILENA CEPEDA RIAÑO, FELIPE ALBERTO CASTRO BENAVIDES [**]

KEYWORDS: GLITCH, SIDE-CHANNEL, ATTACK, MCU, MICROCONTROLLER.

DESCRIPTION:

Software attacks have been the most common ones. However, lastly one of the electronics industry concerns is hardware security because of the increase of devices connected to the Internet or IoT. Therefore, it is necessary to study the existing attacks. The glitch attack is one of them. Its implementation requires low cost and may lead to leak information from devices. In this work, multiple glitch power attacks were done using ChipWhisperer. The supply voltage decreases at 0 [V] during the attack, and the glitch duration is increased until having a successful attack. On the other hand, a glitch circuit was proposed and tested to manage glitch amplitude and duration. A resistor divider circuit allowed to control the amplitude and the ChipWhisperer helped to manipulate the duration. It was found that the glitch duration remains about constant at different frequencies using only the ChipWhisperer, and the proposed circuit leads to see the glitch signal energy to have a successful attack; that energy is about constant at different frequencies and amplitudes. Future work should concentrate on finding the instruction jump cause, it will be achieved enhancing Turpial simulation. Besides, the relationship between the glitch signal energy and a successful attack should be investigated.

---

[*] Bachelor Thesis

[**] Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Elkim Felipe Roa Fuentes, Philosophy Doctor.

INTRODUCTION

The Internet of Things or IoT is a network that connects devices to the Internet, which uniquely identify each other and exchange information. It was estimated that by the end of 2020, 25 billion devices were connected to the Internet[1], raising concerns about their security, since sensitive information may be extracted and misused. Considering that IoT or any other network will be as secure as its least secure point, devices handling this type of information cannot be vulnerable to attack.

Software attacks, in electronic systems, are the most known until the present time. Nevertheless, in recent years, hardware attacks have caught the attention of the academy and the industries which are involved. One way to implement hardware attacks consists of glitch injection in the different supply lines of these electronic systems such as power, clock, and ground.

The glitch attacks may jump crucial program execution for the device security, such as, key validation or encrypted algorithms, which, if interrupted, would result in serious consequences to the IoT system. Hence, it is important to study this attack and in future works may find countermeasures.

The glitch parameters such as the duration, the maximum and minimum values can affect the attack performance. For instance, the shape of the glitch was studied in different micro-controllers to leak information and the results showed an increment in the firmware extraction speed and, in particular, a significantly lower amount of injected glitches required to complete the attack[2].

---

[1] W. IQBAL y col. “An In-Depth Analysis of IoT Security Requirements, Challenges and their Countermeasures via Software Defined Security”. En: *IEEE Internet of Things Journal* (2020), págs. 1-1.

[2] C. BOZZATO, R. FOCARDI y F. PALMARINI. “Shaping the Glitch: Optimizing Voltage Fault Injection Attacks”. En: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2019.2 (2019), págs. 199-224. DOI: 10.13154/tches.v2019.i2.199-224.

Accordingly, in this work, supply glitches attacks were implemented controlling the glitch amplitude and duration, in order to violate the security of Turpial MCU. This project is under the guidance of a doctoral project of the OnChip research group, in which the nature of this attack will be investigated and try to discover the consequences at hardware level of glitch attacks in MCUs to be able to propose new ways to counter them.

1. OBJETIVES

1.1. GOAL

To implement glitch attack strategies in microcontrollers.

1.2. SPECIFIC OBJETIVES

- To produce an instruction jump in a microcontroller (MCU) through a glitch attack.
- To design a glitch attack circuit with various parameters, such as glitch duration and glitch amplitude.

2. RELATED WORKS

First, it is important to know that the glitch attack is a transient power drop with a duration typically in the ns to μs range, that occurs at a specific instant of time. It is a fault injection technique that cause a misbehaviour in the target. This is the result of setup time violations that can cause incorrect data to be captured, allowing an attacker to tamper with the regular control flow, e.g., by skipping instructions, influencing a branch decision, corrupting memory locations, or altering the result of an instruction or its side effects. This kind of attack requires little special equipment and can be mounted under lower cost and expertise[2]. It's a non invasive attack and it doesn't damage the device under test[3].

The mainly types of glitch attacks are the clock glitch and power glitch. The clock glitch is an the adversary temporarily shortens the length of a single clock cycle. The disadvantage is that this method requires physical access to an external clock pin[4]. The power glitch focuses on creating disturbances on an otherwise stable power supply line[2].

Other types of hardware attacks are overclocking, in which the adversary persistently applies a higher-frequency clock signal than the nominal clock frequency of the device; underfeeding, in which a lower voltage than the nominal voltage is supplied to the device; overheating, besides to cause setup-time violation on the datapath, it causes modification in memory cells in EEPROM, Flash, and DRAM memories; optical fault injection, where the adversary decapsulates the target integrated circuit

---

[3] S. KAUR, B. SINGH y H. KAUR. "Analytical Classifications of Side Channel Attacks, Glitch Attacks and Fault Injection Techniques: Their Countermeasures". En: *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*. IEEE. 2020, págs. 144-151.

[4] B. YUCE, P. SCHAUMONT y M. WITTEMAN. "Fault attacks on secure embedded software: Threats, design, and evaluation". En: *Journal of Hardware and Systems Security* 2.2 (2018), págs. 111-130.

(IC) and exposes the silicon die to a light pulse, it induces a photo-electric current that causes faulty computations; finally, electromagnetic fault injection, where the adversary applies transient or harmonic EM pulses on the IC through a fault injection probe, which is designed as an electromagnetic coil[4]. All of them are active attacks, i.e., the attacker injects input into the system to cause system malfunction.

On the other hand, there are passive attacks where the attacker only monitors and measures physical characteristics. Passive attacks include leak information from dynamic power consumption. It is proportional to effective capacitance or switching activity and in logical switching, the IC will consume higher power, so by doing simple power analysis (SPA), the attacker can analyze the power pattern on an oscilloscope and can easily detect the secret key. The leakage current is another passive attack, it leaks information because it is related to the input of a device. At a specific input value, there is a specific leakage current, so you can deduce the input value[3].

In sub-100 nm technologies, leakage power is well known to be comparable to dynamic power and is expected to become an even larger portion of the chip power budget in the future. Hence, the leakage (static) supply current can reveal a significant amount of information on the secret key, due to the strong dependence of leakage on the input of digital blocks. Therefore, power analysis attacks based on leakage are expected to be increasingly effective in downscaled technologies[5].

In addition to the SPA, there are other fault analysis techniques, among them are differential fault analysis (DFA), differential power analysis (DPA), fault sensitive analysis (FSA), and cumulative power analysis (CPA)[3]. These side-channel attacks may allow finding the secret encryption key stored on a hardware device that has Advanced Encryption Standard (AES)[6].

---

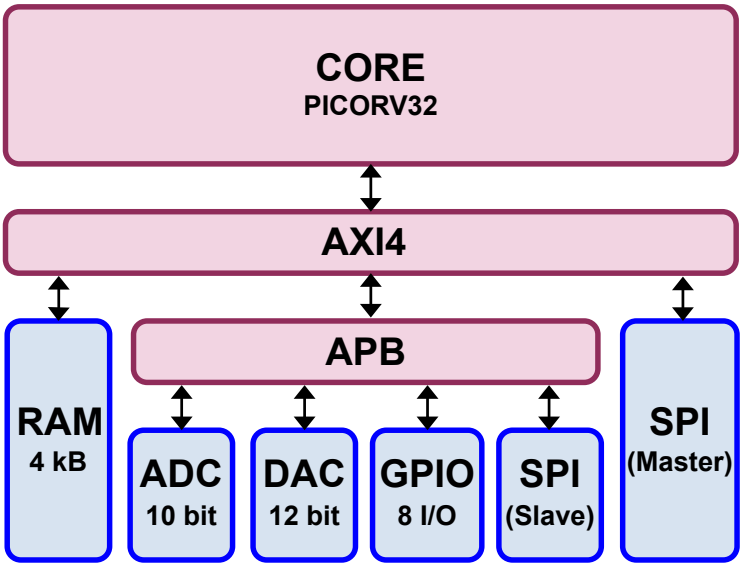
[5] M. ALIOTO y col. “Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variations”. En: *IEEE Transactions on Circuits and Systems I: Regular Papers* 61.2 (2013), págs. 429-442.

[6] F. R. NURADHA y col. “Attack on AES Encryption Microcontroller Devices With Correlation Power Analysis”. En: *2019 International Symposium on Electronics and Smart Devices (ISESD)*. IEEE.

Despite all the attacks that exist, power glitch attacks were selected because as mentioned in the literature, it is a very cheap technique and the adversary does not have a high expertise to do the attack. Furthermore, this project is under the guidance of a doctoral project of the OnChip research group, in which the nature of this attack will be investigated.

In this work, the device under test is Turpial microcontroller. It is composed by a 32b RISC-V core (PicoRV32), a 4KB SRAM, a 10b ADC, a 12b DAC, 8 GPIO, and two SPI interfaces (master and slave). All the modules are connected using two different buses: AXI4 and APB. This organization is shown in the figure 1. The MCU supports a maximal clock frequency of 100 MHz and the supply core voltage is the 1.2 V. The core has a two-stage instruction pipeline with one fetch stage and one execute stage[7].

Figura 1. Turpial MCU



2019, págs. 1-4.

7 C. DURAN y col. “A 32-bit RISC-V AXI4-lite bus-based microcontroller with 10-bit SAR ADC”. En: *2016 IEEE 7th Latin American Symposium on Circuits Systems (LASCAS)*. 2016, págs. 315-318. DOI: 10.1109/LASCAS.2016.7451073.

To apply the glitch attack, the most common setup is using the ChipWhisperer (CW) operating principles, as shown in Figure 2. The CW is an open source toolchain that makes learning about side-channel attacks. It is mostly focused on power analysis attacks, voltage and clock glitching, which disrupt a device's power supply or clock signal to cause unintended behaviour[8].

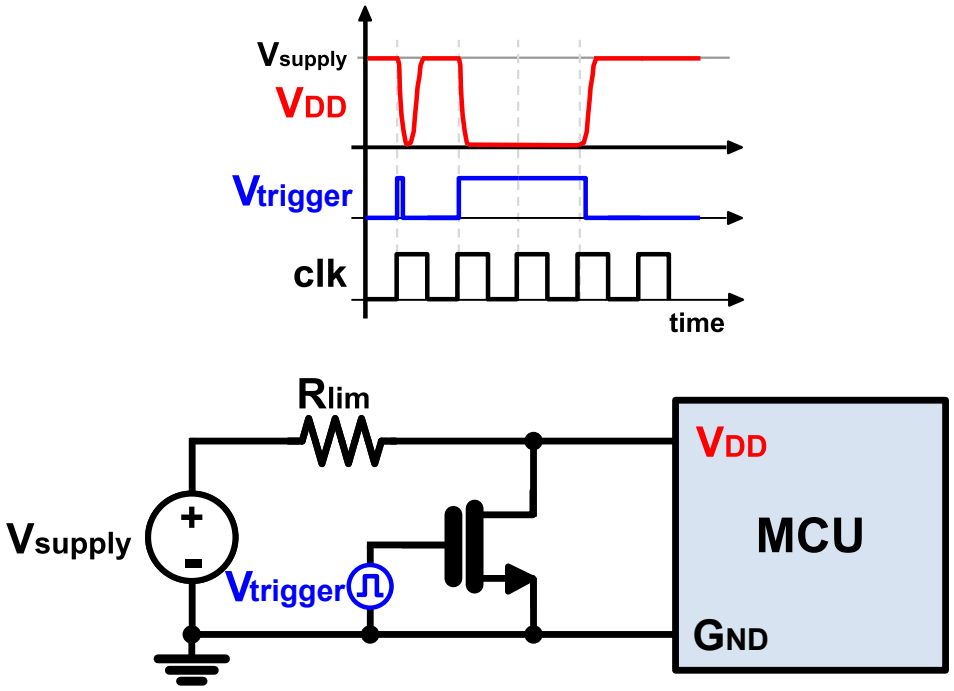
CW target uses a transistor, typically MOSFET, that is placed in parallel to the power supply line and it is used to briefly short-circuit VDD to ground. Then, the glitch is triggered by a Field Programmable Gate Array (FPGA) managing the attack timing. The main limitations of this technique are the reduced control over the attack parameters: for instance, additional equipment is required for controlling the voltage levels, and the generated glitch can be unpredictable, due to variations in both MOSFET and target electronic properties[2].

Generally, the MCUs integrate a voltage regulator for providing a stable power supply to the internal processor and memory. Depending on the regulator technology, an external filtering capacitor can be required. It is suggested that the voltage glitch source be connected directly to the capacitor pin, in order to bypass the internal regulator and avoid any interference of this component during the attack.[2]

---

[8] NewAE Technology Inc. *NEWAE HARDWARE PRODUCT DOCUMENTATION*. 2020. URL: https://rtfm.newae.com/.

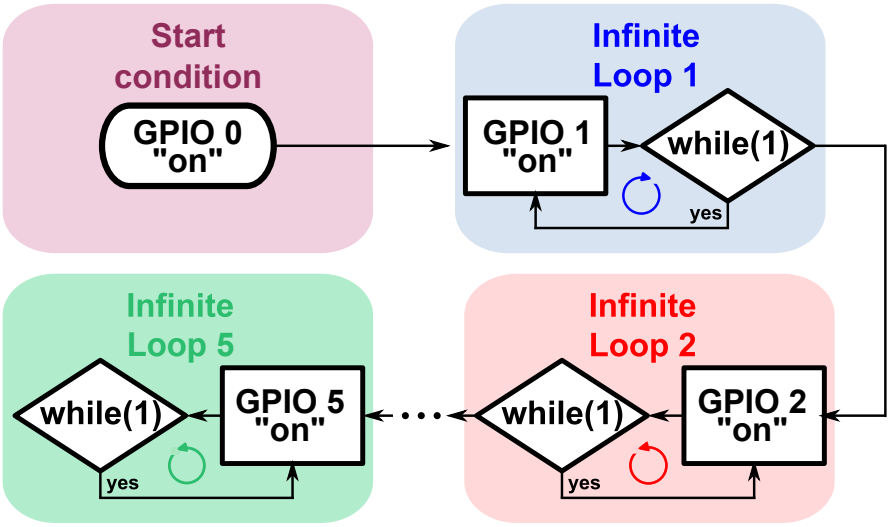
Figura 2. Glitch circuit and signal nodes



3. DESIGN AND IMPLEMENTATION OF GLITCH POWER ATTACK

First of all, it is necessary to know that a successful glitch attack produces a jump instruction in the MCU memory. It allows skipping a password or some security mechanism. To emulate it, the MCU is going to be in an infinite loop unless the glitch attack causes a jump instruction. Figure 3 shows in more detail the written MCU program in the memory. There are a start configuration and five infinite loops, each one, turns on a GPIO. In the start configuration, the GPIO 0 turns on, and the others are off. Then, it enters an infinite loop and turns on GPIO 1. There are other four infinite loops, so if there is an instruction jump, the GPIOs between 2 to 5 could be on.

Figura 3. Turpial program

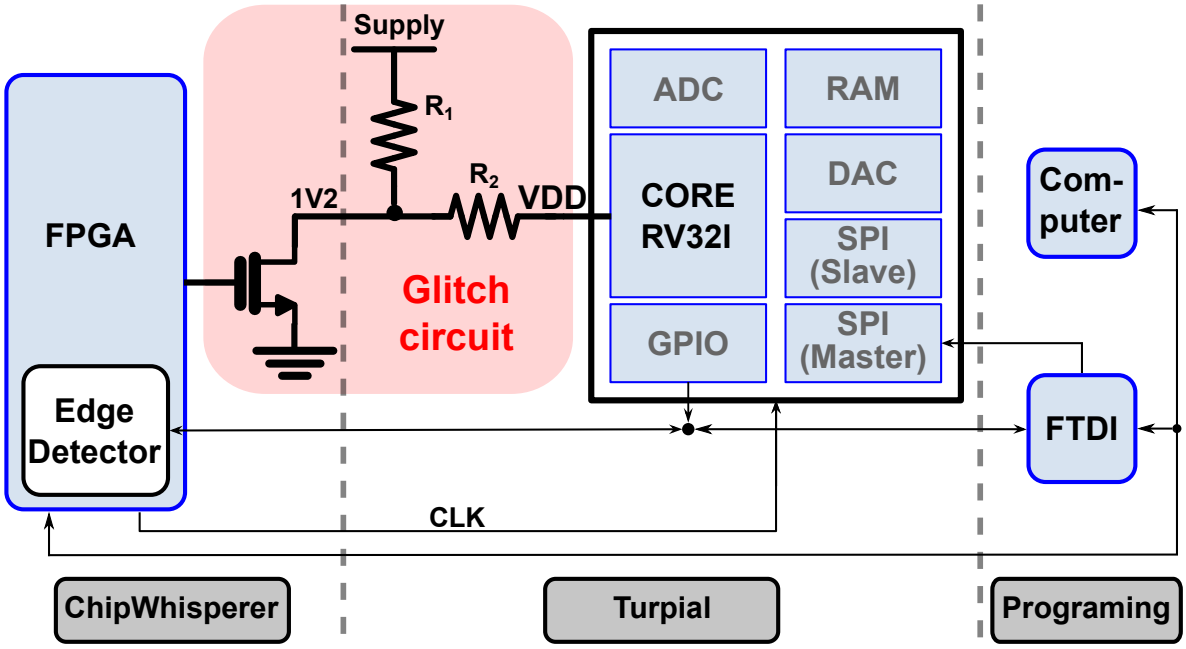


3.1. DESIGN GLITCH ATTACK IMPLEMENTATION WITH THE CW

First, it is worth mentioning that it is not strictly necessary to use the CW to do the next implementation. Any FPGA or MCU can be used to achieve the glitch attack, hence the ease of applying it. It was decided to use the CW because of its facilities to apply glitch attacks.

At the beginning, the glitch attack in the MCU is applied through the supply line using the CW. The Figure 4 shows the glitch implementation with the CW.

Figura 4. CW Implementation. Power glitch attack with fixed amplitude using a FTDI and a computer to program the MCU and the CW.



The glitch circuit is composed of a power N-MOS transistor (IRF7807ZTRPBF) and two resistors. R1 is a current limiter, and R2 has a low value and is used to measure the core current consumption. The circuit is connected to the supply voltage of the Turpial core. The transistor is triggered by the CW FPGA.

The Turpial board has a voltage regulator and coupling capacitances at the input supply core voltage. For that reason, the capacitances were removed and the core supply was disconnected from the regulator, before injecting the glitch.

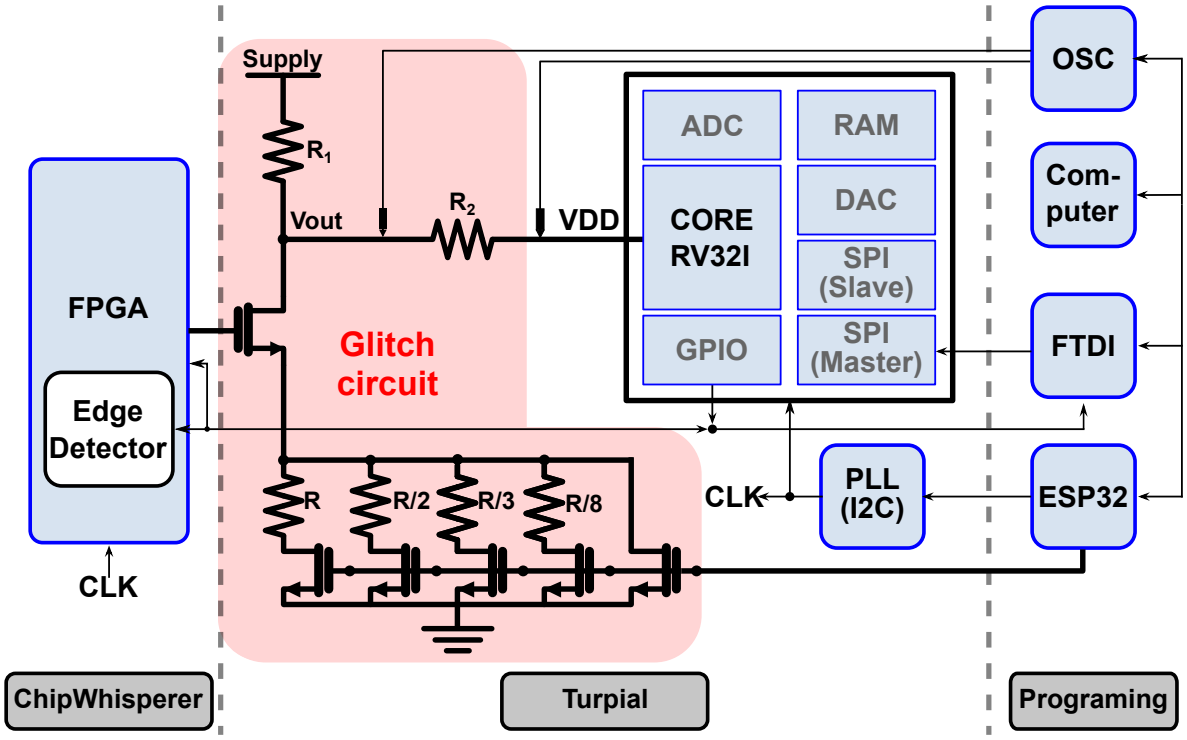
The CW FPGA was modified to add an Edge Detector to read the Turpial GPIOs. The FTDI (C232MH) is used to write the MCU memory, and to read the GPIOs and the Edge Detector output. The computer operates the CW and the FTDI.

When the transistor is triggered, the 1V2 node decreases to almost 0 [V] during the glitch time configured in the CW FPGA.

3.2. NEW GLITCH CIRCUIT DESIGN

A hypothesis of the doctoral thesis, under which this work is guided, specifies that the jump instruction is related to the glitch signal energy applied to the MCU, independent of its duration or amplitude. Therefore, the proposed circuit allows controlling these parameters as shown in the Figure 5.

Figura 5. Proposed Implementation. Power glitch attack with variable amplitude using a FTDI and a computer to program the MCU and CW, ESP32 to program the PLL and a resistor divider.



From the Figure 4, we added an oscilloscope, a PLL, an ESP32, buffers, and a digital configurable resistor divider. The new implementation is shown in the Figure 5.

The oscilloscope is used to get the voltage wave to measure the signal energy. The CW and the Turpial's clock are generated by the PLL (CY22150). The resistor divider and the PLL are configured by the ESP32. The CW FPGA is used as a buffer to read the Turpial GPIOs. Finally, the resistor divider helps to get different glitch amplitudes in the MCU supply voltage.

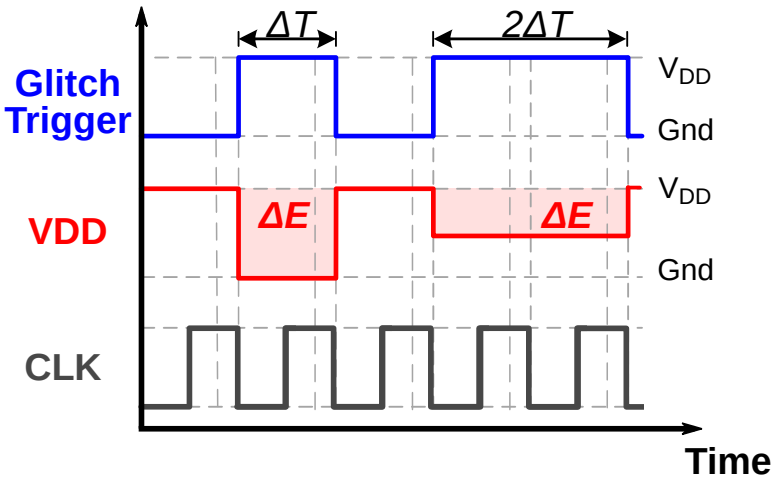
The digital configurable resistor divider follows the next equation:

$$Vout = \frac{1.2}{1 + frac} \tag{1}$$

Where *frac* is an integer number between 1 to 15.

The "frac"binary number triggers the N-MOS transistors. To select a glitch amplitude of 0 [V], there is an additional transistor that does not have a resistor connected.
In Figure 5, when the transistor M1 is triggered, the Vout node decreases to the glitch amplitude configured with the resistor divider, and the glitch time is set by the CW FPGA.

Figura 6. Supply voltage and glitch signal

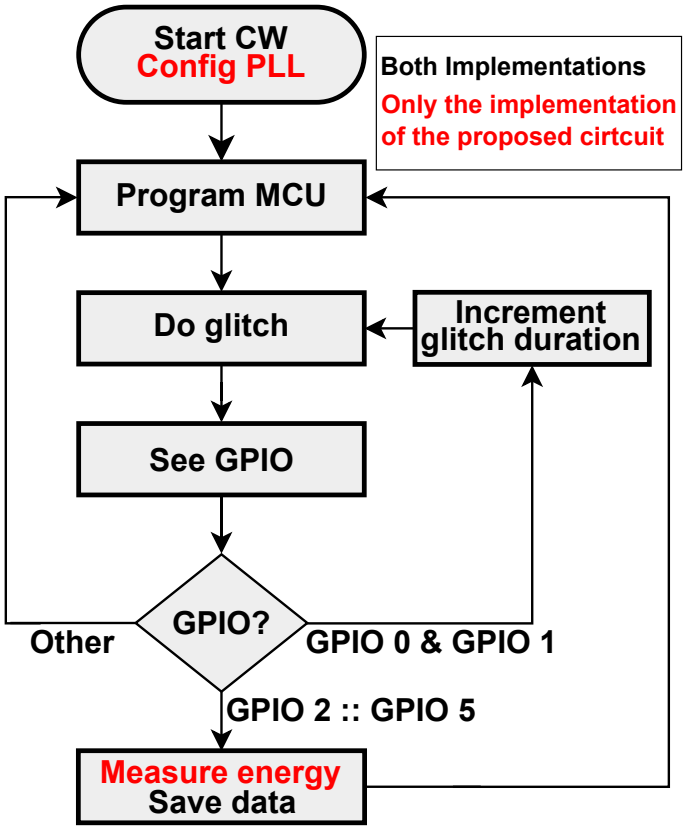


For a better understanding of the hypothesis, figure 6 illustrates the glitch trigger and the supply voltage with a two different resistor divider configuration. The first glitch attack has a VDD amplitude and leads the supply voltage going down to 0 [V]. When the glitch amplitude is less than VDD, the hypothesis suggests that the glitch duration has to be higher, so that the energy is about constant in each attack. This case is illustrated in the second glitch.

4. METHODOLOGY

In the Figure 7, there is the glitch attack methodology.

Figura 7. Methodology

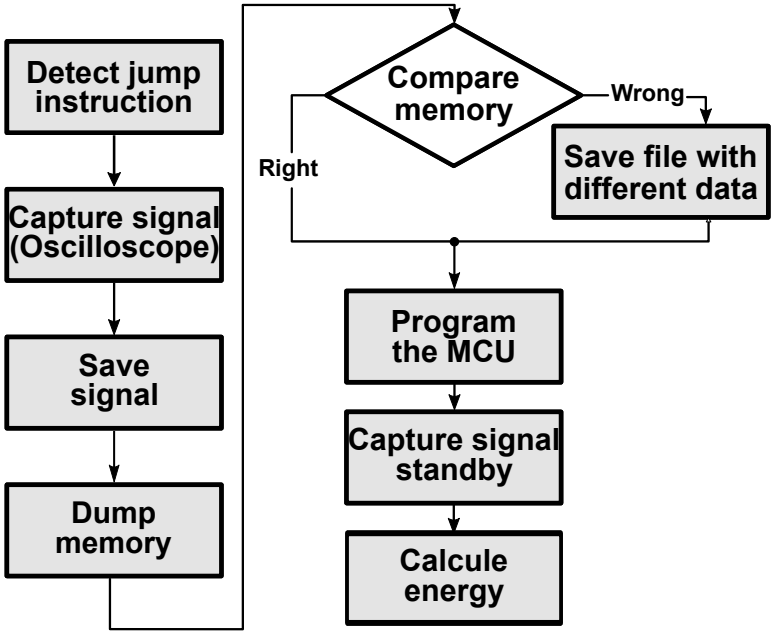


The black letters describes the glitch injection methodology for the two implementations. First, the CW is initialized and the MCU is programmed. A ready"signal goes out from the MCU. Then, a glitch is applied, and we see which GPIOs are on. If the GPIO 0 and 1 are on, the glitch duration is increased by one clock signal period in each iteration. If there is one GPIO between 2 to 5 on, it means an instruction jump occurred, and we save the glitch duration data. Otherwise, it means that the MCU reset and the process restarts with the same glitch duration.

The red letters says some aspects that were added to develop the proposed imple-

mentation. They are the measurement of the glitch signal energy, the initial configuration of the PLL, and the resistor divider.

Figura 8. Measure Energy methodology



On the other hand, Figure 8 illustrates the methodology to take the glitch signal energy.

Accordingly, once a jump instruction is detected, the voltage signal is retained and stored by the oscilloscope. Then, the MCU memory is dumped and compared with the program file. The MCU is programmed again, and the voltage signal is captured on standby. Finally, the energy is calculated with the data taken previously.

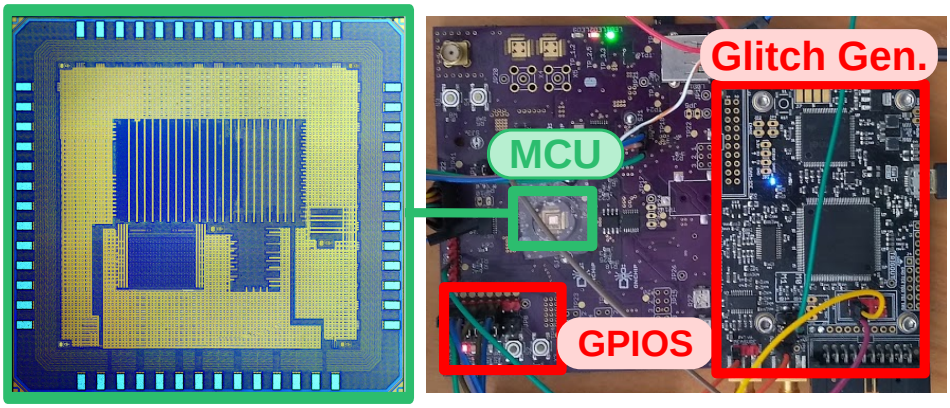
The sources codes used to implement the methodology are available in this repository.

5. RESULTS

This section has the results obtained by the CW and the proposed circuit implementations.

5.1. CW IMPLEMENTATION

Figura 9. CW Setup

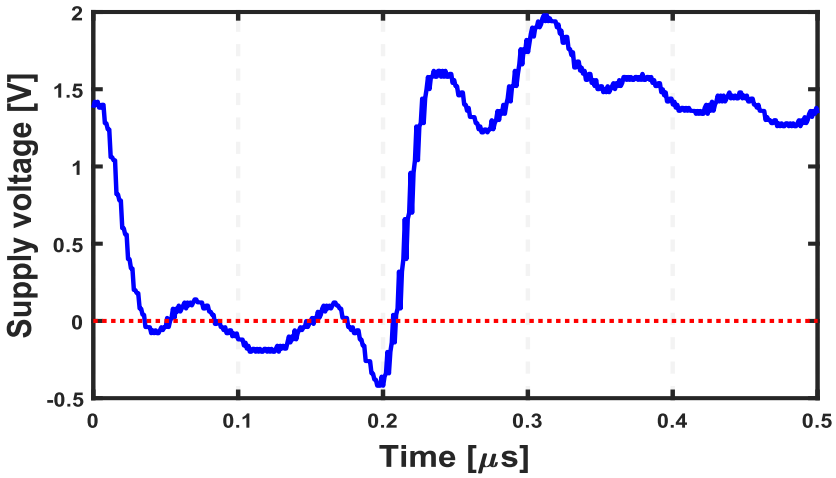


The CW implementation has three principal parts, the MCU, the glitch generation, and GPIOs. It is illustrated in the Figure 9. The glitch applied is exposed in Figure 11(a). The results about the glitch duration required to get a successful attack (jump instruction) is shown in Figure 11(b). This box diagram have 100 samples in each MCU frequency. Each box represents the data upper than 25 percent and lower than 75 percent in the histogram. The Figure suggests that glitch duration is maintained regardless of MCU frequency.

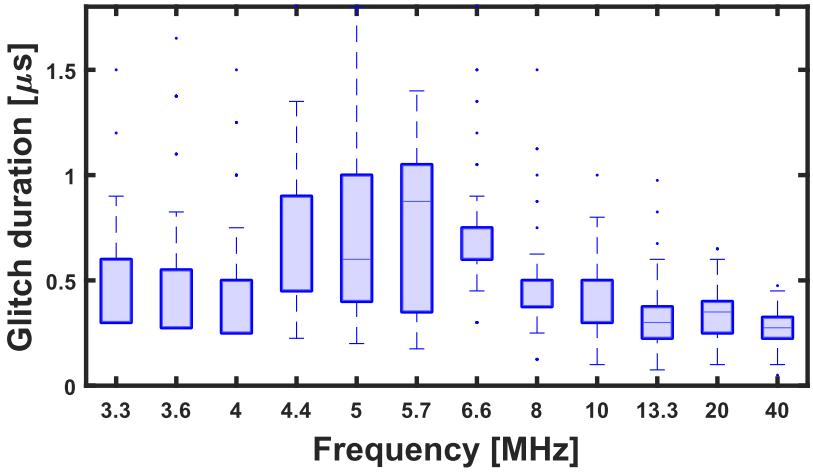
5.2. PROPOSED CIRCUIT IMPLEMENTATION

The setup for the proposed circuit is in Figure 11. The applied glitches at different amplitudes are illustrated in Figure 12. The results are shown in Figure 13, which expose the glitch signal energy required to get a successful attack. This box diagram

Figura 10. CW implementation results



(a) Supply voltage wave with 1.2 V glitch attack amplitude



(b) Results using CW setup. Glitch duration required to get a jump instruction. There are 100 samples in each frequency

have 100 samples in each MCU frequency at different glitch amplitudes. The energy medians of each amplitude are very close in each frequency. It suggest that the energy is about constant. This reinforces the hypothesis of the doctoral thesis but should be investigated more deeply.

When a power glitch attack is applied, there are 3 options: anything happens, the MCU is reset or an instruction jump occurs. If there are high number of resets more time is spent to achieve a successful attack. The number of resets behavior is illus-

Figura 11. Final Setup

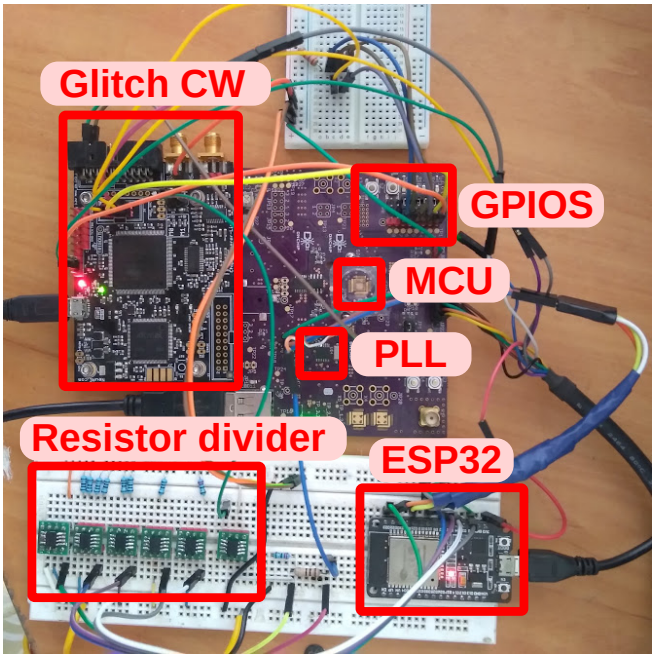


Figura 12. Supply voltage during glitch attack at different amplitudes

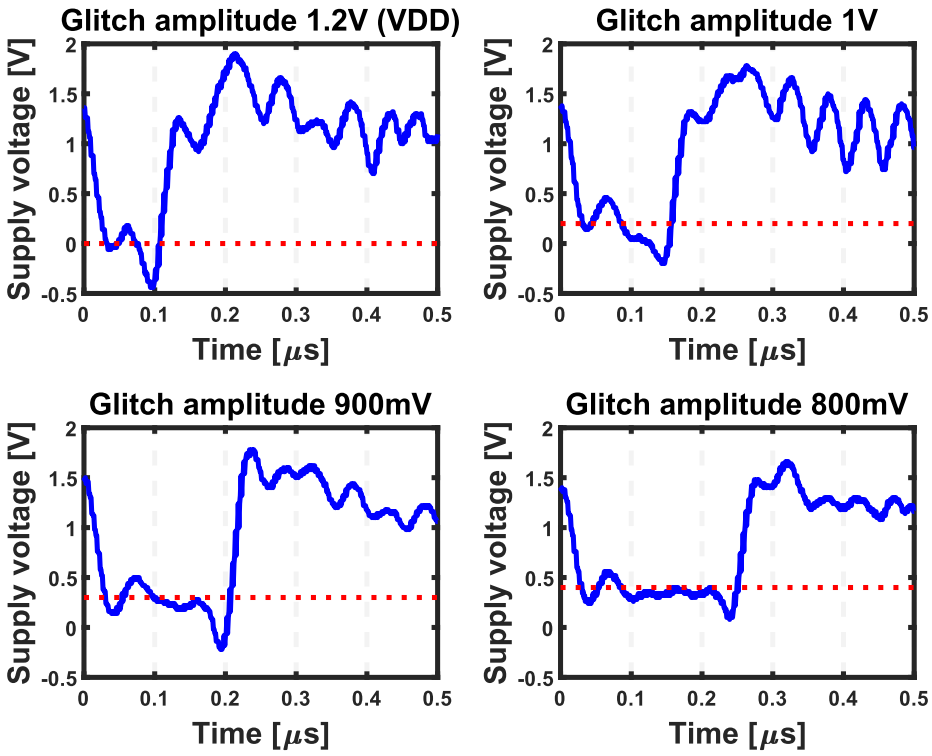


Figura 13. Results with the proposed circuit. Glitch wave energy required to get a jump instruction. There are 100 samples in each frequency and glitch amplitude

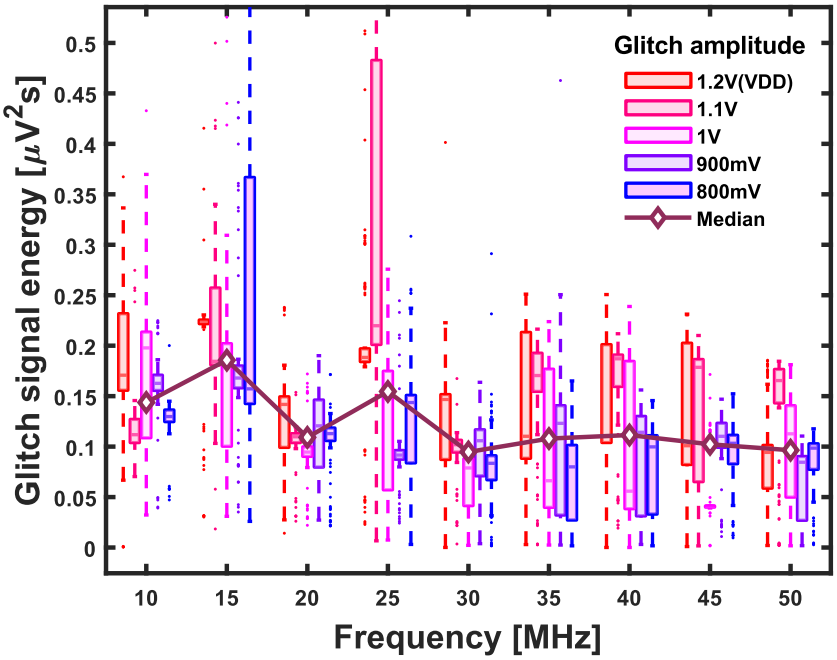
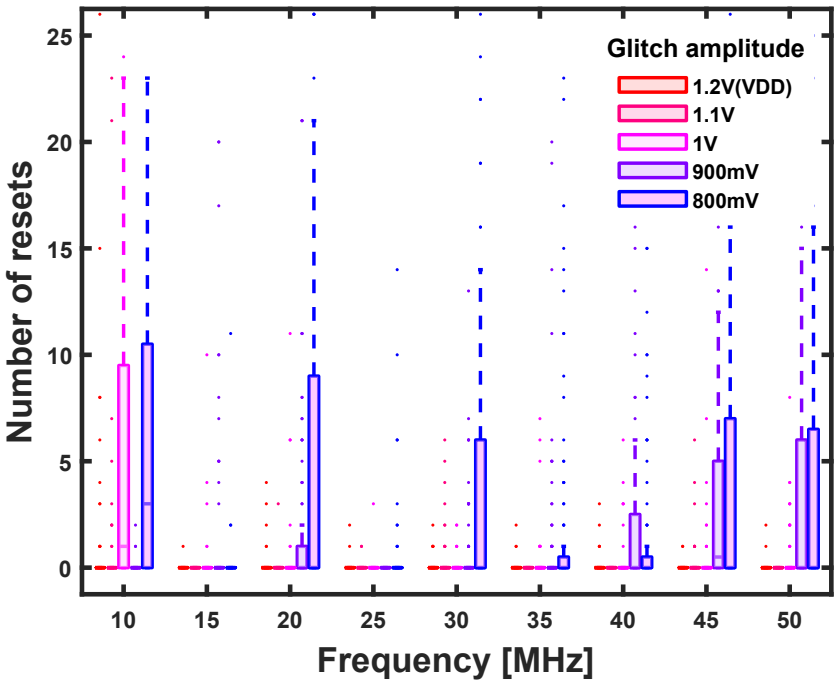


Figura 14. Number of resets occurred before having a successful glitch power attack.



trated in the Figure 14. It indicates that while decreasing the glitch amplitude, the number of resets before achieving an instruction jump, increases.

6. CONCLUDING REMARKS

The power glitch attacks can jump instructions allowing the attacker to skip, for instance, the password of an operating system or a validation in a home automation system connected to IoT. In this work, it is demonstrated that a glitch attack can be done with a low budget and expertise using the CW or any other tool that allows varying the glitch parameters. The proposed circuit allows to change the glitch amplitude and duration. The evidence from this study suggests the glitch signal energy is about constant in each attack, regardless of the MCU core frequency. This finding is useful for future studies that try to discover the consequences at hardware level of glitch attacks in MCUs, therefore, to explain the instruction jump.

7. FUTURE WORK

During the research of papers that talk about side-channel attacks, most of them just say how to do a glitch attack to violate the security but there is not a rigorous study that explains what causes the jump instruction inside the MCU. It is because the microprocessors have been with closed source architecture, for that reason, in this work, the MCU used, is with open-source architecture (RISC-V) facilitating that future works could recreate the power glitch attack and study it more deeply to find ways to counter them. For future work, Turpial simulation needs to be done to compare our results and finding the instruction jump cause. On the other hand, it is assumed that injecting power glitch attacks, the fault should be of voltage, therefore, historically voltage regulators have been made. So, looking from the energy point of view, the possibilities of countermeasure these attacks become larger. This will be investigated in a doctoral project of the OnChip research group.

BIBLIOGRAPHY

ALIOTO, M. y col. “Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variations”. En: *IEEE Transactions on Circuits and Systems I: Regular Papers* 61.2 (2013), págs. 429-442 (vid. pág. 11).

BOZZATO, C., R. FOCARDI y F. PALMARINI. “Shaping the Glitch: Optimizing Voltage Fault Injection Attacks”. En: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2019.2 (2019), págs. 199-224. DOI: 10.13154/tches.v2019.i2.199-224 (vid. págs. 7, 10, 13).

DURAN, C. y col. “A 32-bit RISC-V AXI4-lite bus-based microcontroller with 10-bit SAR ADC”. En: *2016 IEEE 7th Latin American Symposium on Circuits Systems (LASCAS)*. 2016, págs. 315-318. DOI: 10.1109/LASCAS.2016.7451073 (vid. pág. 12).

Inc., NewAE Technology. *NEWAE HARDWARE PRODUCT DOCUMENTATION*. 2020. URL: https://rtfm.newae.com/ (vid. pág. 13).

IQBAL, W. y col. “An In-Depth Analysis of IoT Security Requirements, Challenges and their Countermeasures via Software Defined Security”. En: *IEEE Internet of Things Journal* (2020), págs. 1-1 (vid. pág. 7).

KAUR, S., B. SINGH y H. KAUR. “Analytical Classifications of Side Channel Attacks, Glitch Attacks and Fault Injection Techniques: Their Countermeasures”. En: *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*. IEEE. 2020, págs. 144-151 (vid. págs. 10, 11).

NURADHA, F. R. y col. “Attack on AES Encryption Microcontroller Devices With Correlation Power Analysis”. En: *2019 International Symposium on Electronics and Smart Devices (ISESD)*. IEEE. 2019, págs. 1-4 (vid. pág. 11).

YUCE, B., P. SCHAUMONT y M. WITTEMAN. “Fault attacks on secure embedded software: Threats, design, and evaluation”. En: *Journal of Hardware and Systems Security* 2.2 (2018), págs. 111-130 (vid. págs. 10, 11).