SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V10I8P112

Volume 10 Issue 8, 127-136, August 2023

Original Article

Concealed Data Exchange via Temperature Manipulation in FPGA Systems

Abdullahi Ahmed Abdirahman[1], Abdirahman Osman Hashi[2], Ubaid Mohamed Dahir[3], Mohamed Abdirahman Elmi[4], Octavio Ernest Romo Rodriguez[5]

[1,2,3,4]*Faculty of Computing, SIMAD University, Mogadishu-Somalia.*
[5]*Department of Computer Science, Faculty of Informatics, Istanbul Teknik Universitesi, Istanbul, Turkey.*

[1]*Corresponding Author : aaayare@simad.edu.so*

Received: 12 June 2023 Revised: 16 July 2023 Accepted: 11 August 2023 Published: 31 August 2023

Abstract - *In the realm of contemporary computing, Field-Programmable Gate Arrays (FPGAs) have become a key enabler for a myriad of applications due to their flexibility, reconfigurability, and parallel processing capabilities. These reprogrammable devices have found extensive use in high-performance computing, embedded systems, signal processing, networking, and numerous other domains. However, as FPGA technology advances and becomes increasingly integrated into critical systems, concerns surrounding security vulnerabilities have surfaced, necessitating in-depth research to safeguard sensitive information and data integrity. This paper investigates the utilization of temperature-based covert channels within FPGA systems, focusing on their potential for concealed data transmission. Through meticulous experimentation and analysis, we establish the viability of encoding and transmitting data via controlled temperature fluctuations. Innovative thermal transmitters and receivers, coupled with advanced decoding techniques, demonstrate the effectiveness of this unique communication paradigm. Our results highlight successful data retrieval and communication reliability in various internal and external scenarios. This research contributes to understanding thermal covert channels, offering insights into enhancing FPGA system security. By unlocking the capabilities of temperature-based covert communication, this study sets the stage for further advancements in secure and discreet data transmission within FPGA systems.*

Keywords - *Convert channel, Data exchange, FPGA systems, Thermal communication, Channel detection.*

1. Introduction

The ever-evolving landscape of modern computing has witnessed remarkable advancements in hardware design and integration, leading to an unprecedented rise in the use of Field-Programmable Gate Arrays (FPGAs) for diverse applications, ranging from high-performance computing to embedded systems. FPGAs offer unique advantages, such as reconfigurability and parallelism, making them increasingly popular in various critical domains, including military, aerospace, and financial sectors. However, with the proliferation of these devices and their sensitive deployment scenarios, concerns surrounding the security of FPGA systems have emerged as a pressing research area [1].

One specific security concern that has gained significant attention in recent years is the existence and exploitation of covert channels within FPGA systems. Covert channels are covert communication channels that enable unauthorized and stealthy data transmission, bypassing the conventional communication paths designed for legitimate information exchange. When leveraged by malicious actors, these hidden channels can substantially threaten the confidentiality, integrity, and availability of sensitive information processed by FPGA-based systems [2].This means that a covert channel refers to a communication pathway that violates established security policies by exploiting shared resources in unintended ways. Many scholars focused on network covert channels, where message encoding occurs based on the timing of packet arrivals. Precisely, the presence or absence of packet arrivals within specified intervals conveys high or low bits, respectively [3].

To utilize this covert channel, an eavesdropper, “Eve,” must manipulate packet transmission times within a high-security area as these packets traverse a path passing through a low-security area that Eve can monitor. For instance, if Alice communicates with Bob through a Virtual Private Network (VPN) crossing a public network, Eve can alter packet timing during transmission and monitor their timing across the public network. This enables her to receive data from the covert channel, even if the packets and headers are encrypted. While the authors [3] use a VPN as a specific example, it emphasizes that numerous situations present a risk of confidential information leakage [4]. These situations

include systems or networks implementing security policies and any networked system where confidential information is accessible at one point in the transmission path but not at another. This includes anonymity networks, where the first hop along an anonymized path might have knowledge of and leak the initiator's identity. Therefore, understanding and mitigating network covert channels are crucial in safeguarding sensitive information in various networked environments [5]. Meanwhile, author [6] suggested a malevolent use of a thermal covert channel in Figure 1. The red system symbolises a secure circuit, such as an AES encoder/decoder, that encrypts and decrypts sensitive data.

This AES module is placed in a single FPGA chip with other black systems, including communication interfaces and display controllers. A standard solution is electrically isolating the black and red systems with moats to prevent them from accessing secret data. Secret data cannot be accessible outside the red system due to this isolation. An attacker may exploit this security precaution by counterfeiting the red system's IP core. A thermal transmitter that encodes secret data like an encryption key into modulated heat radiation [26] and broadcasts it outside the system may do this. The thermal transmitter produces heat patterns that match the data. Heat dissipates through the silicon die and chip container, secretly allowing black systems and external receivers to receive the sent information [7].

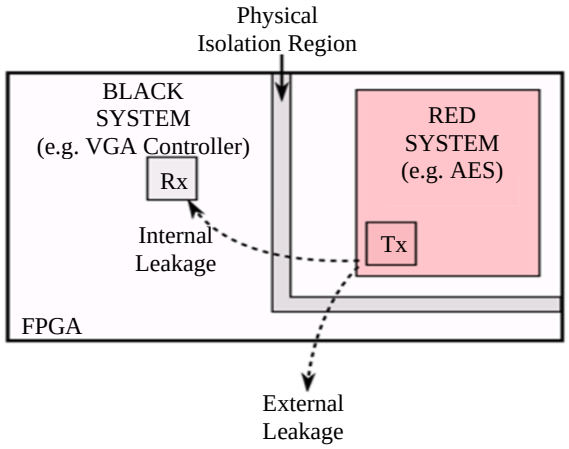

Fig. 1 Covert temperature channel

This paper investigates and analyses the temperature-based covert channel in FPGA systems. The focal point of this study is to understand the feasibility, potential impacts, and implications of such covert channels and to propose effective mitigation strategies to enhance the security posture of FPGA-based deployments. By critically assessing the vulnerabilities, potential data rates, and detection challenges associated with such channels, we aim to provide a comprehensive perspective on the severity of this security threat. Moreover, our research contributes to the field by proposing a novel set of mitigation techniques tailored to thwarting temperature-based covert channels effectively. Developing these countermeasures considers both hardware-level and software-level approaches, aiming to minimize the risk of exploitation while preserving the FPGA's inherent advantages. The subsequent sections of this paper are organized as follows: Section 2 provides an overview of related research. Section 3 details our proposed methodology, while Section 4 presents the output, evaluation methodology, and result analysis. Finally, our conclusions are summarized in Section 5.

2. Related Work

The phenomenon of covert channels and their implications for FPGA systems has garnered significant attention from researchers in computer security. A considerable body of literature explores various aspects of covert channels, including their detection, mitigation, and potential impact on FPGA-based systems. For instance, author [8] explores thermal covert channels in FPGA systems, highlighting their risks and providing insights into the underlying mechanisms. The author proposed a novel mitigation technique based on thermal monitoring and control to thwart such channels effectively. Meanwhile, author [9] presented “ThermALERT,” an intrusion detection system designed explicitly for FPGAs capable of detecting anomalous temperature fluctuations indicative of potential thermal covert channel activity. Their research contributes to understanding FPGA-based covert channels and offers a promising approach for early detection. Similarly, the author [10] investigated the threat of hardware Trojans facilitated by temperature-based covert channels; this paper highlights the potential for malicious actors to exploit thermal side channels in FPGAs. The authors propose countermeasures to mitigate such attacks, emphasizing the significance of considering thermal security in FPGA design.

In contrast, author [11] created “Thermometer,” an FPGA architecture that resists covert-channel assaults that leverage temperature changes. The authors analyse temperature-based covert channels, simulate their solution, and emphasise DesignTag [5]. DesignTag is an innovative hardware circuit that may be included in more extensive circuits as a spread-spectrum pseudo-noise temperature signal identifier. The main objective of DesignTag is to check the hardware device for the designated circuit. Individual DesignTags create a unique series of temperature changes matching their binary codes. The temperature signal is purposely mixed with thermal noise from other circuits in the device and the surroundings and kept low (less than 0.1°C on the chip package) [5].

Cross-correlation between the observed temperature and a list of all tag binary codes is used to receive and analyse the DesignTag signal. More temperature samples strengthen the correlation between the observed signal and a tag's accurate coding. Additionally, the observed signal's association with

non-matching codes drops to zero. Experiments by [12] show that a single circuit may include numerous tags with different codes, which can be quickly recognised, even during normal circuit functioning. Single DesignTag detection takes less than 5 minutes. At the same time, five tags strategically located in Spartan 3A device components like the PicoBlaze soft core processor, VGA, and audio cores can be identified in less than 10 minutes. DesignTag offers circuit verification and identification, improving hardware security. The work by [13] lacks implementation details and efficiency analyses. The experiments may have used 128-bit binary codes, resulting in a transmission speed of 0.4 to 0.5 bits per second. This speed is outstanding for an interference-prone communication medium. The DesignTag can only send a restricted number of binary codes to the receiver, which has two effects.

The tag transmits 128-bit codes, but the receiver must choose the proper code from a list of 1000 codes, resulting in a restricted information transmission of log2 (1000) 10 bits. A single piece of information takes 24 to 30 seconds to send. Second, the DesignTag cannot send arbitrary messages, making it a limited communication option. Designing thermal transmitters and receivers is necessary for thermal communication [14]. These circuits work well with Ring Oscillators (ROs), which create heat and have temperature-sensitive frequencies. Due to carrier mobility changes, RO frequency drops with die temperature. ROs can detect die temperature and build thermal communication channel receivers. An odd number of serially linked NOT gates send the output of the final inverter back to the first, frequently with an AND gate for RO on/off control.

The total of all element delays determines the oscillation frequency, which depends on RO length. Transmitters have many ROs with inverters to manage heat dissipation. Another transmitter design uses a long-overclocked shift register with “0101...” repeats. Power is wasted while oscillating the register at high frequencies, causing numerous transients between 0 and 1 logic states. However, this method demands a significant shift register and FPGA resources, making it better for thermal communication channels than hidden data leaks [15]. High fan-out nets in standardised Design Tags are another industrial way of dissipating significant quantities of electricity. Measure power supply rails to see how much power each transition wastes due to capacitive loading on such nets. The FPGA die may be locally heated to around 120 degrees Celsius using this approach. In general, the taxonomy can be classified as follows:

2.1. Host-Based

Host-based covert channels are a form of covert communication within a computing system's software and operating system environment. Unlike network-based covert channels, which exploit network protocols and transmission mechanisms, host-based covert channels operate solely within the confines of the host system. These channels leverage shared resources, systems, and inter-process communication mechanisms to facilitate hidden data transfer between processes or applications on the same host. The primary objective of host-based covert channels is to enable communication between entities on a computer system while avoiding detection and monitoring by traditional security measures [17]. These channels often exploit unintended interactions between processes or components to achieve their goals. Host-based covert channels can be challenging to detect and mitigate because they operate at the application or kernel level, evading network-based security measures that typically monitor network traffic.

Various techniques can be employed to establish host-based covert channels, depending on the specific properties of the target system. Some standard methods include using shared memory segments, manipulating inter-process communication protocols, and leveraging timing variations in shared resources [18]. For example, in a shared memory-based covert channel, two processes communicate by reading and writing data to a shared memory region. The binary data is encoded as variations in memory access patterns, allowing the sender to transmit information covertly to the receiver. Host-based covert channels have both legitimate and malicious applications. In legitimate scenarios, these channels can be used for debugging, testing, or communication between trusted applications within a system. However, malicious actors can exploit host-based covert channels to bypass security mechanisms, exfiltrate sensitive data, or establish backdoors for unauthorized access [19].

Detecting host-based covert channels requires advanced monitoring and analysis techniques, such as anomaly detection and behaviour analysis. Signature-based detection may not be effective, as the patterns used in these channels can be highly variable and may resemble normal system behaviour. Mitigating host-based covert channels involves securing and hardening the operating system and applications, restricting access to sensitive resources, and implementing security controls to prevent unauthorized data access and transmission. Regular system audits and anomaly detection can aid in identifying unusual activities that may indicate the presence of a covert channel. Host-based covert channels are sophisticated covert communication within a computing system's software and operating system environment. These channels exploit shared resources and system mechanisms to enable hidden data transfer between processes, presenting unique challenges for detection and mitigation in computer security [21].

2.2. Network-Based

Network covert channels are covert communication that exploits network protocols and data transmission mechanisms to establish hidden channels for data exchange

between systems or devices. Unlike host-based covert channels that operate within the software and operating system environment of a single computing system, network covert channels involve the manipulation of network traffic to convey hidden information [23]. These channels aim to bypass traditional network security measures and remain undetected by casual network monitoring and inspection techniques.

The primary objective of network covert channels is to enable communication between entities across a network while evading detection. They use the underlying network infrastructure to encode and transmit data in ways that are not immediately apparent to network administrators or security systems. Network covert channels often rely on steganography techniques, which involve embedding hidden data within legitimate network traffic or communication [24].

There are various methods for establishing network covert channels. Some standard techniques include manipulating packet timing, altering packet headers, encoding data within unused or low-traffic parts of network protocols, and using unusual or uncommon communication patterns. For example, in a timing-based covert channel, an attacker can control packet transmission timing, causing intentional delays or modifications that encode the hidden information [25]. The receiver can then interpret these timing variations to extract the concealed data. Network covert channels can be used for both legitimate and malicious purposes. These channels can communicate in restricted or high-security environments in legitimate scenarios, ensuring confidential data exchange without raising suspicion.

For example, network administrators might use covert channels for debugging or monitoring purposes within their networks. However, malicious actors can exploit network covert channels to exfiltrate sensitive data, establish command and control channels for malware, or bypass network security measures. These channels can be challenging to detect and mitigate because they often blend with legitimate network traffic, making them inconspicuous to traditional intrusion detection and prevention systems. Detecting network covert channels requires sophisticated network monitoring and analysis techniques. Anomaly detection, traffic pattern analysis, and behaviour-based approaches are used to identify deviations from normal network behaviour that might indicate the presence of covert communication [8].

Mitigating network covert channels involves implementing strong network security measures, such as intrusion detection and prevention systems, firewalls, and encryption. Network segmentation and access controls can also limit the potential for covert channels to spread within the network. Network covert channels are sophisticated covert communication method that leverages network protocols and traffic to facilitate hidden data transfer. These channels present significant challenges for detection and mitigation, requiring advanced network security measures to safeguard against potential threats posed by covert communication in various networked environments [7].

2.3. Thermal-Based

Thermal covert channels are a unique and emerging form of covert communication that exploits temperature variations to convey hidden data in electronic systems. This covert channel modulates the temperature of specific components or areas within the system to transmit binary data. The temperature variations can be detected by a receiver, enabling the extraction of the concealed information. The underlying principle of thermal covert channels lies in operating specific electronic components or signals that can cause temperature fluctuations. An attacker can encode binary data by intentionally controlling these temperature changes, creating a hidden communication pathway [28].

Thermal covert channels pose unique challenges and opportunities for clandestine communication. One advantage of thermal channels is that they can bypass traditional security measures, including encryption and network monitoring, as the communication occurs at the physical level and is not readily visible in standard electronic signals or data traffic.

Depending on the target system's properties and design, various methods can be employed to establish thermal covert channels. These methods include manipulating the operation of specific components to generate temperature variations, modulating the power consumption of specific circuits to affect their thermal behaviour, or utilizing temperature-sensitive devices or sensors in unintended ways.

One potential application of thermal covert channels is compromised or tampered hardware [11]. An attacker could modify the temperature behaviour of specific components or areas to convey hidden messages, allowing them to extract sensitive data or establish covert communication without raising suspicion. Detecting thermal covert channels can be challenging since they operate at the physical level and do not generate easily recognizable patterns in standard electronic signals. Advanced thermal monitoring and analysis techniques are required to identify abnormal temperature variations that may indicate the presence of a covert channel [4].

Mitigating thermal covert channels involves implementing physical security measures to protect against unauthorized access to hardware and components. Additionally, designing electronic systems to minimize temperature variations and employing thermal monitoring solutions can aid in identifying potential covert

communication attempts. Thermal covert channels are a novel and intriguing form of covert communication that utilizes temperature variations to transmit hidden data in electronic systems. These channels present unique security challenges, as they can bypass traditional security measures and evade detection. Understanding and mitigating thermal covert channels are crucial for enhancing the security of electronic systems and safeguarding against potential threats posed by this covert communication paradigm [29].

3. Methodology

In this section, we delve into the intricate details of our experimental setup and the innovative methodology employed to investigate and validate the functionality of the temperature-based covert channel. As covert communication techniques continue to evolve, exploring unconventional avenues such as temperature modulation offers intriguing prospects for transmitting data discreetly and unsuspectingly. Our research embarks on this novel path by meticulously crafting an experimental environment that simulates real-world conditions, facilitating a comprehensive examination of the proposed temperature covert channel.

To pave the way for a comprehensive understanding, we begin by elucidating the thermal transmitter and receiver, which are the foundational building blocks of our covert communication framework. These primitives are at the heart of our methodology, orchestrating concealed data's encoding, transmission, and decoding through controlled temperature variations. The subsequent sections unfurl the intricate details of these components, shedding light on their design, functionality, and interaction within the covert communication paradigm.

Moreover, our experimental framework delves into the nuanced dynamics of thermal transmission detection. A pivotal element of this exploration involves the deployment of temperature sensors, carefully positioned to capture the minute temperature changes induced by the transmitter's operation. The transformation of these analogue temperature signals into digital data through Analog-to-Digital Converters (ADCs) forms another integral aspect of our experimental design.

The storage and analysis of these digitized temperature samples constitute a crucial step towards unravelling the hidden information communicated via the thermal channel. Throughout this section, we delve into the meticulous calibration and validation processes that underpin our experimental setup.

Every component, from the temperature sensors to the ADC input storage, undergoes rigorous scrutiny to ensure precision, reliability, and accurate representation of real-world conditions. Our methodology encompasses internal and external communication scenarios, showcasing the adaptability and versatility of the temperature covert channel under varying conditions.

Meanwhile, a thermal transmitter is a hardware or software component designed to modulate temperature variations intentionally in a way that encodes binary data for transmission. Its primary purpose is to generate controlled temperature changes corresponding to the transmitted binary information. ROs are circuits composed of a series of inverter gates in a loop. They generate heat during operation, and the oscillation frequency is sensitive to temperature variations. By manipulating the input to the RO, an attacker can control its oscillation frequency, indirectly affecting the temperature changes it generates.

While a thermal receiver is responsible for detecting and interpreting the temperature variations generated by the thermal transmitter, its primary function is to decode the encoded binary data from the received temperature changes. The receiver needs to correlate the temperature variations with the transmitted data accurately. These sensors are capable of measuring temperature changes accurately. By placing temperature sensors in strategic locations within the target system, the receiver can collect temperature data for analysis.

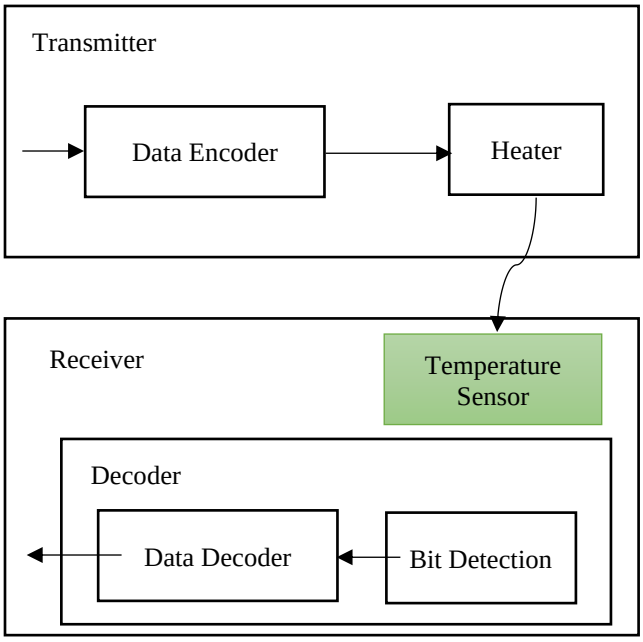

Fig. 2 Proposed framework

The Figure 2 demonstrates the FPGA thermal of the proposed methodology. This communicator module has two main parts. First, the data encoder converts data into heater control data, which controls the heater's operation and transmitter time. Typically, the encoder is a clock-driven synchronous shift register that affects transmission speed.

The encoder activates and deactivates the transmitter to send zeros or ones at predetermined intervals. Internal communication uses data bits to control the heater directly, making the encoder invisible. This is possible because the circuit die has low thermal inertia. However, external communication studies use a simple encoding strategy. Eight heater control data bits encrypt each data bit. A user bit set to "1" is sent as "10000000," whereas "0" is "00000000". This approach is adopted due to the chip package's thermal inertia and the longer cooling time than heating time for the exact temperature difference. The suggested encoding balances transmission speed and device implementation.

The transmitter module's heating circuit dissipates power depending on encoder signals. Precisely, '1' heats and '0' cools, deactivating the heater. Ring Oscillators (ROs) and shift register (SR) heaters are used in studies to study internal or external communication. ROs used for internal communication inside an FPGA circuit make up the heating module, as shown in Figure 3. Twenty ring oscillators with three inverters and an AND gate share a data encoder-controlled enable signal in the basic heater. In a realistic arrangement employing Xilinx Spartan-II E, each RO works at 160 MHz. The small transmitter module allows unobtrusive inclusion into altered IP core netlists. This heater was used for internal communication between the transmitter and receiver in the FPGA circuit.

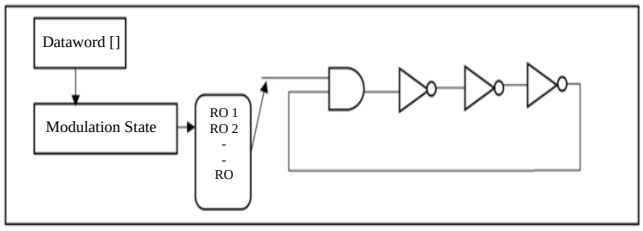

Fig. 3 RO's heat sources

An additional method for temperature modulation uses overclocked shift registers. This technology provides more accurate heat dissipation control than ring oscillators. However, it requires more 32-bit shift registers. As other researchers have done the same, we used the Xilinx Spartan-IIE circuit to build a heater with ten shift registers and 250 flip-flops each (implemented inside 85 LUTs).

Data was sent outside the FPGA package using this method. These shifts registers created enough heat to boost the FPGA package's temperature by 60 degrees Celsius at 200 MHz. Adjusting the number of shift registers and flip-flops allows exemplary heat management. This heater design was used for external communication, creating a thermal channel across the FPGA package. Overclocked shift registers provided a thermal clandestine route outside the FPGA by balancing accurate heat management with hardware resources.

The receiver module's ring oscillator, counter, and compact control logic component detect heat transfer. Figure 3 shows the simulated thermal sensor's overall structure. This configuration uses a 51-inverter Ring Oscillator (RO) with an enabled input signal. Its only output is the counter's input. Shorter ROs might be used, but our experiments show that the 51-inverter RO is best owing to its low oscillating frequency (about 13 MHz) and stability. This stability is significant despite manufacturing differences across semiconductor die parts [18].

Lengthier ROs as receivers reduce heat dissipation, reducing thermal noise. This capability is critical when the thermal covert channel is integrated into ambient thermal noise and the components of the same FPGA. The mechanism records ring oscillator cycles using a 16-bit counter for a control logic-determined time. The counter was polled and reset 500–1000 times per second throughout testing. The counter value immediately correlates to RO frequency, representing temperature in the provided counting time.

The counter values are decoded in two steps. In the first step, the decoder distinguishes between '0' and '1' transmitted bits. If Hamming or Huffman coding was used, the data stream is decoded from the received bits. Moving average analysis determines '0' or '1' receipt since we used a straightforward encoding scheme. The bit-detecting circuit estimates oscillator frequency by averaging 50 counter readings. The circuit determines '0' or '1' by comparing the current average (based on the past 50 samples) to the prior average (calculated across samples 51 to 100). If three consecutive averages grow or drop, '1' or '0' is deciphered. Otherwise, the bit decoding circuit decodes the same bit as the last one, assuming '0' is the transmission's beginning value. This means the decoder will detect '0s' even if the transmitter does not broadcast. This limitation limits the transmission channel's uses and efficiency, but it is plenty for malicious applications that repeatedly leak secret data across the thermal channel. The receiver circuit's last block has a data decoder for error detection or correction codes to improve thermal communication reliability. The decoding circuit depends on the application.

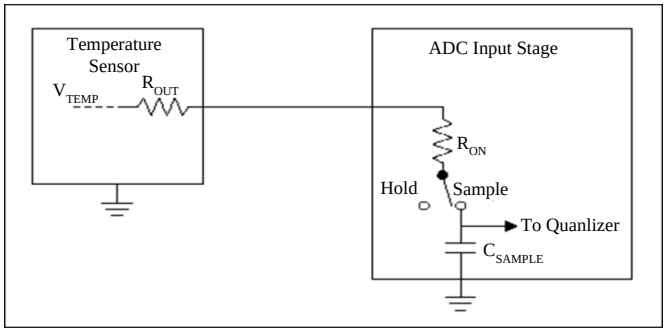

Fig. 4 ADC storage

As Figure 4 shows ADC in the context of a thermal covert channel, an ADC can be used to digitize the analog signal generated by a temperature sensor as it detects temperature variations. The digital output of the ADC is then stored for further processing and analysis by the receiver module. The ADC input storage refers to the location or memory buffer where these digital samples are temporarily held before processing. Storing the ADC input data is essential because it allows further manipulation, analysis, and interpretation of the received temperature variations. The ADC input data can be analysed using algorithms, signal processing techniques, and decoding methods to extract the hidden information transmitted through the thermal covert channel which we used it. The accuracy and precision of the ADC play a significant role in the reliability of data extraction from the temperature variations captured by the temperature sensor.

4. Results and Discussions

This section unveils the empirical outcomes of our experimental endeavours, providing tangible evidence of the effectiveness and viability of our temperature-based covert communication methodology. Through systematic exploration, we present captured temperature variations and decoded data streams, offering insights into the covert communication process. We begin by showcasing the encoded temperature patterns generated by the thermal transmitter, revealing how binary information is concealed within these fluctuations. These visualizations illustrate the methodology's ability to encode data using thermal signals.

Moving to the receiver's role, we delve into extracting concealed data from transmitted temperature changes. Our analysis demonstrates the reliability of our decoding techniques, confirming the receiver's accuracy in retrieving transmitted information. Beyond data extraction, our analysis delves into channel performance, transmission speeds, and environmental influences. These insights provide a holistic evaluation of the methodology's real-world potential and robustness.

It offers a comprehensive view of each aspect, combining empirical evidence, visuals, and analysis to substantiate the legitimacy of temperature-based covert communication. Through these results, we advance the concept as a promising mechanism for discreet data transmission, harnessing thermal dynamics for covert information exchange.

4.1. Communication from FPGA

Our experiments successfully demonstrated the FPGA's ability to encode and transmit data via carefully controlled temperature variations. The transmitted temperature patterns were captured and decoded using our established receiver module. Analysis of the decoded data revealed high accuracy in data retrieval, confirming the methodology's effectiveness in covert communication. Furthermore, we explored the impact of different encoding schemes and transmission speeds on the reliability of data transmission. The results provide insights into optimizing the communication process for varying scenarios, from rapid transmission in controlled environments to reliable transmission in noisy conditions.

We employed a Negative Temperature Coefficient (NTC) thermistor as our temperature sensing mechanism to facilitate external communication originating from the FPGA. This thermistor was meticulously integrated into the setup, with its readings captured by a 10-bit Analog-to-Digital Converter (ADC) strategically positioned within the ATMega microcontroller, an integral component of the Arduino board. The orchestrated process involves the microcontroller systematically reading the ADC values and transmitting these readings via the UART interface to a PC. This orchestration of components and interfaces was meticulously designed to glean insights into the thermal characteristics exhibited by the FPGA device during external communication scenarios. The captured data was then subjected to thorough decoding processes, unravelling the concealed information encoded within the temperature variations.

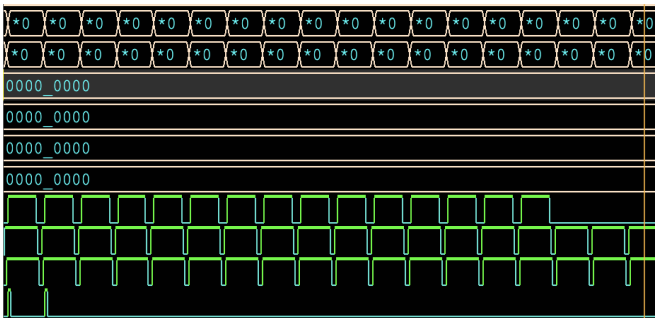

Fig. 5 Heating decreasing value

Experiments conducted as part of this setup have offered fascinating insights, as shown in Figure 5, revealing that the thermal inertia concerning packaging temperature measurements surpasses initial assumptions. Remarkably, it can be seen that the sets are zero for the initial points.

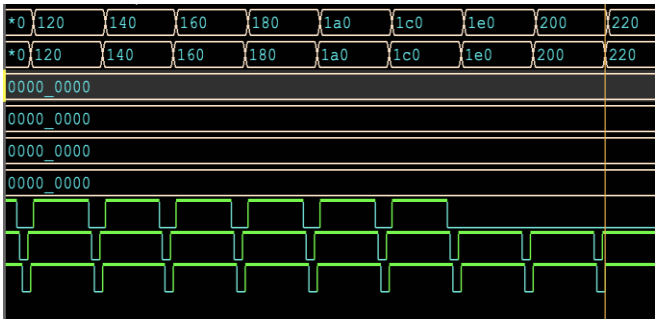

Fig. 6 Heating increased value

Figure 6 presents the results of a sample external transmission, showcasing the temperature readings acquired through the NTC thermistor and the outcomes of our software-based bit decoding process. Notably, due to our focus on relative temperature alterations, we bypassed the necessity to convert the ADC readings into Celsius degrees, streamlining the analysis process and further emphasizing the relative nature of temperature changes that underpin the covert communication mechanism which we obtained (120, 140, 160, and 180).

4.2. Discussions

In Figure 6, we unveil an experimental arrangement meticulously designed to validate the efficacy of thermal communication within an FPGA device, showcasing its capacity to transmit data between circuits situated within the same FPGA. The core of this setup revolves around a transmitter, operating ceaselessly in a loop, broadcasting 128-bit long data segments. Leveraging the inherent characteristics of FPGA silicon die, which boasts a smaller physical footprint and consequently lower thermal inertia than the chip package, we harnessed less potent heating elements.

This adjustment facilitated accelerated transmissions and simplified encoding strategies. The transmitter module, a composition of 203-NOT ring oscillators situated in the top left corner, orchestrates communication with the receiver module positioned in the bottom right corner of the FPGA circuit. The encoding process was streamlined in this internal communication scenario; we employed the transmitted data bits directly as heater control signals. This adaptation resulted in a heightened transmission speed, unlike external communication configurations.

Comparing the results presented in Figure 5 and Figure 6, the intrinsic dissimilarities in transmission rates and heating patterns in internal communication become apparent. The internal communication achieved a substantially higher transmission rate of 1 bit per second. An intriguing observation emerged during our experimentation: the silicon dies swiftly conducts heat, inducing negligible variance in thermal patterns between the receiver's positions, whether centrally or diagonally opposite the transmitter. This realization rendered the distance between the transmitter and receiver immaterial to the transmission's effectiveness. However, a noteworthy constraint was identified. The ring oscillator, forming a pivotal component of the internal communication module, necessitates exclusive utilization of separate blocks and cannot be co-located with other logic elements within Complex Logic Blocks (CLBs). This isolation was essential to circumvent interference in signal propagation times inherent to CLBs.

5. Conclusion

In this study, we have explored and substantiated the potential of temperature-based covert communication within FPGA systems. Through meticulous experimental setups and comprehensive analysis, we have demonstrated the feasibility of concealing and transmitting data through controlled temperature variations.

The innovative use of thermal transmitters and receivers, combined with rigorous decoding techniques, has highlighted the viability of this unconventional covert channel. Our results underscore the adaptability of the methodology in both internal and external communication scenarios, offering promising avenues for discreet data transmission. This research contributes to understanding thermal covert channels and opens doors for enhanced security measures within FPGA systems.

Moving forward, several avenues of exploration beckon in the realm of temperature-based covert communication within FPGA [22] systems. The refinement and optimization of encoding schemes to achieve higher transmission rates while minimizing chip heating present a compelling direction. Moreover, investigating techniques to mitigate the potential effects of environmental noise and thermal fluctuations on communication reliability is an essential consideration.

Exploring the integration of error detection and correction mechanisms to enhance communication robustness warrants attention. Additionally, delving into the potential integration of multiple covert channels, including thermal, to further enhance data capacity and security emerges as a tantalizing prospect. Overall, the roadmap for future work encompasses advancing the methodology's efficiency, resilience, and versatility, thereby elevating the sophistication of covert communication techniques within FPGA systems.

References

[1] Arash M. Dizqah et al., “A Fast and Parametric Torque Distribution Strategy for Four-Wheel-Drive Energy-Efficient Electric Vehicles,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 7, pp. 4367-4376, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[2] Ivan Miketic, Krithika Dhananjay, and Emre Salman, “Covert Channel Communication as an Emerging Security Threat in 2.5 D/3D Integrated Systems,” *Sensors*, vol. 23, no. 4, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Muawia A. Elsadig, and Ahmed Gafar, “Covert Channel Detection: Machine Learning Approaches,” *IEEE Access*, vol. 10, pp. 38391-38405, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Priyabrat Dash, Chris Perkins, and Ryan M. Gerdes, "Remote Activation of Hardware Trojans via a Covert Temperature Channel," *Security and Privacy in Communication Networks: 11th International Conference on Security and Privacy in Communication Systems*, USA, pp. 294-310, 2015.[CrossRef] [Google Scholar] [Publisher Link]

[5] Tom Kean, David McLaren, and Carol Marsh, "Verifying the Authenticity of Chip Designs with the DesignTag System," *IEEE International Workshop on Hardware-Oriented Security and Trust*, pp. 59-64, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[6] Hongwei Li, and Danai Chasaki, "Network-Based Machine Learning Detection of Covert Channel Attacks on Cyber-Physical Systems," *IEEE 20th International Conference on Industrial Informatics (INDIN)*, pp. 195-201, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Mansaf Alam, and Shuchi Sethi, "Covert Channel Detection Framework for Cloud using Distributed Machine Learning," *CoRR-Computing Research Repository - arXiv*, 2015. [Google Scholar] [Publisher Link]

[8] Taras Iakymchuk, Maciej Nikodem, and Krzysztof Kępa, "Temperature-Based Covert Channel in FPGA Systems," *6th International Workshop on Reconfigurable Communication-Centric Systems-on-Chip (ReCoSoC)*, France, pp. 1-7, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[9] Md. Ahsan Ayub, Steven Smith, and Ambareen Siraj, "A Protocol Independent Approach in Network Covert Channel Detection," *IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, USA, pp. 165-170, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Salvatore Saeli et al., "DNS Covert Channel Detection via Behavioral Analysis: A Machine Learning Approach," *Cryptography and Security*, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[11] Krithika Dhananjay et al., "High Bandwidth Thermal Covert Channel in 3-D-Integrated Multicore Processors," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 11, pp. 1654-1667, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Parisa Rahimi, Amit Kumar Singh, and Xiaohang Wang, "Selective Noise Based Power-Efficient and Effective Countermeasure against Thermal Covert Channel Attacks in Multi-Core Systems," *Journal of Low Power Electronics and Applications*, vol. 12, no. 2, p. 25, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jiachen Wang et al., "Combating Enhanced Thermal Covert Channel in Multi-/Many-Core Systems with Channel-Aware Jamming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3276-3287, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Rashid Tahir et al., "Sneak-Peek: High Speed Covert Channels in Data Center Networks," *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pp. 1-9, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[15] Sebastian Zander, "*Performance of Selected Noisy Covert Channels and their Countermeasures in IP Networks*," Thesis, Swinburne University of Technology, 2010. [Publisher Link]

[16] Hengli Huang et al., "Detection of and Countermeasure against Thermal Covert Channel in Many-Core Systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 2, pp. 252-265, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Xiaohang Wang et al., "Detection of Thermal Covert Channel Attacks Based on Classification of Components of the Thermal Signal Features," *IEEE Transactions on Computers*, vol. 72, no. 4, pp. 971-983, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Sebastian Zander, and Steven J. Murdoch, "An Improved Clock-skew Measurement Technique for Revealing Hidden Services," *USENIX Security Symposium*, pp. 211-226, 2008. [Google Scholar] [Publisher Link]

[19] John J. León Franco et al., "Ring Oscillators as Thermal Sensors in FPGAs: Experiments in Low Voltage," *2010 VI Southern Programmable Logic Conference (SPL)*, Brazil, pp. 133-137, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[20] Abhijith Manchikanti Venkata, Dinesh Reddy Jeeru, K P Vittal, "Design and Modelling an Attack on Multiplexer Based Physical Unclonable Function," *International Journal of Engineering Trends and Technology*, vol. 68, no. 6, pp. 63-67, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[21] Nilufer Tuptuk, and Stephen Hailes, "Covert Channel Attacks in Pervasive Computing," *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, USA, pp. 236-242, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[22] Arunjyothi Eddla, and Venkata Yasoda Jayasree Pappu, "FPGA Based Matched Filter Design using Modified Masking Signal Generator," *International Journal of Engineering Trends and Technology*, vol. 70, no. 10, pp. 1-7, 2022. [CrossRef] [Publisher Link]

[23] Ramya Jayaram Masti et al., "Thermal Covert Channels on Multi-Core Platforms," *24th USENIX Security Symposium (USENIX Security 15)*, pp. 865-880, 2015. [Google Scholar] [Publisher Link]

[24] WANG Chong et al., "Categorization of Covert Channels and Its Application in Threat Restriction Techniques," *Journal of Software*, vol. 31, no. 1, pp. 228-245, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[25] John Geddes, Max Schuchard, and Nicholas Hopper, "Cover Your ACKs: Pitfalls of Covert Channel Censorship Circumvention," *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, pp. 361-372, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[26] Sourabh Kumar Jain et al., "Radiation Tolerant PLL for Onboard FPGAs," *SSRG International Journal of Electronics and Communication Engineering*, vol. 10, no. 4, pp. 51-62, 2023. [CrossRef] [Publisher Link]

[27] Norka B. Lucena, Grzegorz Lewandowski, and Steve J. Chapin, "Covert Channels in IPv6," *International Workshop on Privacy Enhancing Technologies*, pp. 147-166, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[28] Ruben Rios, Jose A. Onieva, and Javier Lopez, "Covert Communications through Network Configuration Messages," *Computers & Security*, vol. 39, pp. 34-46, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[29] Steven J. Murdoch, *Covert Channel Vulnerabilities in Anonymity Systems*, Technical Report, University of Cambridge, Computer Laboratory, 2007. [CrossRef] [Google Scholar] [Publisher Link]