OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Obscure Hardware Trojan Design in 8051 Micro-controller

Sai Kumar Marri, N. Muthiah
Student researcher, UTD, USA.
Corresponding Author: Sai Kumar Marri

ABSTRACT:
Hardware Trojans represent a critical security threat to integrated circuits (ICs) by enabling malicious actors to introduce covert modifications during the design or manufacturing process. These tampered circuits can degrade performance, leak sensitive data, or cause operational failures, often activated under rare conditions to evade detection. This review synthesizes insights from multiple studies, detailing Trojan attack models, implementation strategies, detection challenges, and countermeasures. Trojan designs exploit vulnerabilities at the Register Transfer Level (RTL) or fabrication stages, using techniques such as clock edge manipulation, redundant logic, and cryptographic module weakening. Attackers may craft triggers that rely on rare events or specific input patterns to activate malicious payloads, including information leakage or functional disruption. Case studies reveal practical implementations of these Trojans in encryption engines and general-purpose processors, demonstrating their stealth and potential impact on military and commercial applications. Traditional detection methods, including functional and structural testing, often fall short in identifying such threats due to their stealthy nature. Advanced approaches, like side-channel analysis and runtime monitoring, provide some defense but remain limited by process variations and testing complexity. Countermeasures, such as designing for security, leveraging trusted foundries, and employing multi-layered verification, offer partial solutions but require further refinement to address diverse Trojan forms effectively. Emerging challenges underscore the need for research in scalable detection frameworks, proactive defense mechanisms, and global collaboration to secure the IC supply chain. As hardware dependence grows, the urgency to mitigate Trojan threats and ensure trust in critical systems intensifies.

KEY WARDS*: Hardware Trojans, Integrated Circuits (ICs), Side-Channel Analysis*

Date of Submission: 14-11-2024 Date of acceptance: 29-11-2024

I. INTRODUCTION

Hardware security has emerged as a critical area of focus in modern electronics, addressing vulnerabilities in integrated circuits (ICs) that can be exploited for malicious purposes. Traditionally, hardware was considered a trusted foundation for secure systems, but the increasing reliance on globalized supply chains, third-party intellectual property (IP), and complex manufacturing processes has exposed ICs to numerous threats, including hardware Trojans[1]. Hardware Trojans represent a particularly insidious class of attacks, involving covert modifications to ICs during design or fabrication. These modifications can degrade performance, leak sensitive data, or cause catastrophic failures, often evading traditional detection methods. The triggers for such Trojans are often rare conditions, making them challenging to activate during standard testing. For instance, stealthy Trojans embedded in cryptographic engines or general-purpose processors can compromise data integrity or disable functionality in mission-critical applications[2]. Detection of these threats is a complex challenge. Conventional testing methods, such as functional and structural tests, struggle against the stealthy nature of hardware Trojans. More advanced approaches, like side-channel analysis, runtime monitoring, and logic testing, offer some promise but face limitations due to process variations and the increasing sophistication of IC designs[3]. As these threats grow, the field of hardware security emphasizes a multi-faceted approach, combining trusted foundries, secure design methodologies, and proactive and reactive

detection strategies to ensure system integrity. Continued research in scalable frameworks and supply chain security is essential to counteract these evolving risks[4].

A microcontroller is a cost-effective computer-on-a-chip designed for specific tasks, such as reading or writing signals on ports, displaying information on LEDs, and transmitting or receiving data through a serial port. Among the various microcontroller families, the 8051 family is one of the most widely used. It remains a preferred choice for many hobbyists and professionals due to its simplicity, versatility, and extensive support community. The 8051 microcontroller is particularly well-suited for secure systems where user authentication is required for proper system operation. For instance, it can be employed in digital locking systems that grant access to a property based on a password-protected security mechanism.

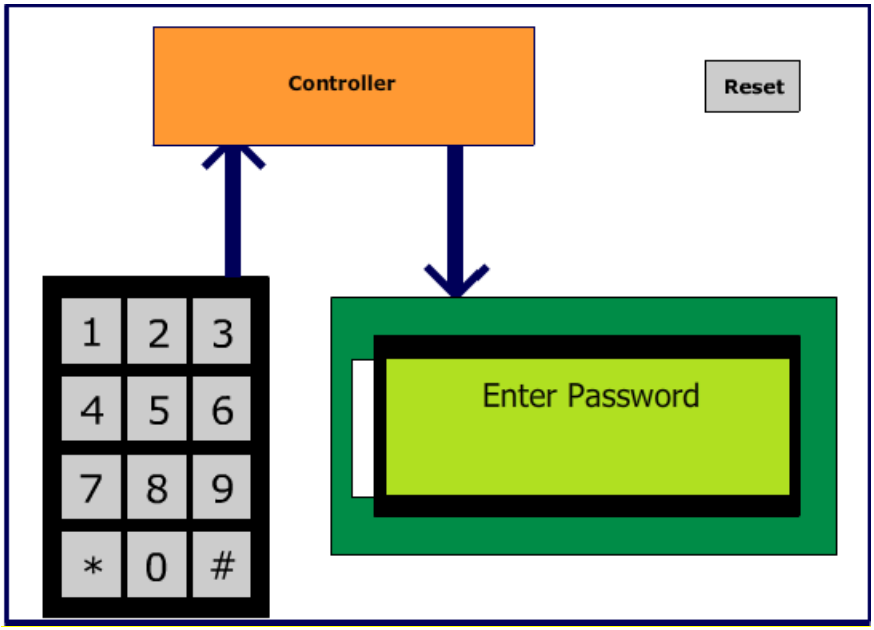


Figure 1: 8051 microcontroller based Digital Code Lock

Another application of the 8051 microcontroller is in controlling household or industrial electrical and mechanical systems. Users can remotely connect to the microcontroller over the internet using a peer-to-peer communication protocol. At the site, the microcontroller is connected to a PC with internet access. The user connects to the PC, authenticates themselves by entering a password, which the microcontroller verifies, and is then granted access if authenticated successfully. Although attackers may attempt to compromise the system at the software level and gain unauthorized access to the PC, an additional layer of security implemented at the microcontroller ensures that only authorized users can proceed. As a result, password-based security is often enforced directly at the microcontroller level for enhanced protection.

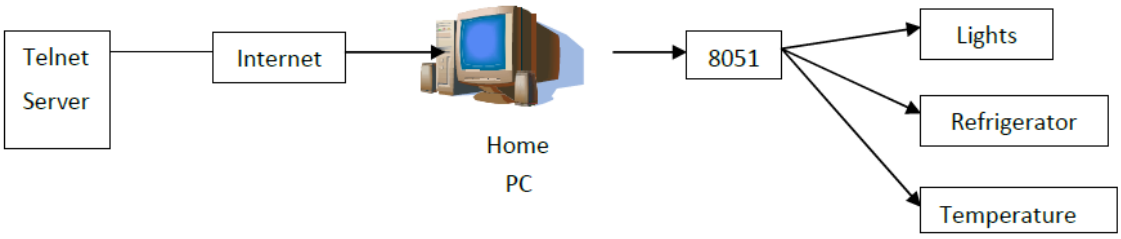


Figure 2: 8051 microcontroller based Home Automation System

II. ATTACK OVERVIEW

In our attack, we designed a Hardware Trojan implemented during the Register Transfer Level (RTL) design phase. The HDL code of the 8051 microcontroller was modified by inserting additional lines of code, creating a backdoor that enables us to bypass the secure system. To demonstrate this, we used the 8051 microcontrollers to develop a "password-based protection system." In this system, the microcontroller prompts the user to enter a password. We chose the password "MAGIC" as the correct key. If the user enters the correct password, the system grants access; otherwise, access is denied. This setup allowed us to showcase how the inserted Trojan could compromise the system's security.

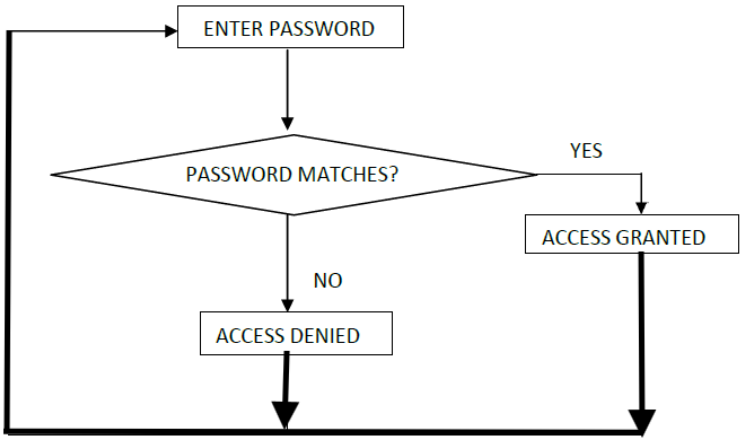


Figure 3: Password-bad protection system

The 8051 system can be targeted to gain unauthorized access to password-based systems. Among the instructions in the 8051 instruction set, the only one capable of performing a comparison operation is the CJNE instruction.

Syntax: CJNE operand1, operand2, relative address

Function: Compare and Jump If Not Equal

The CJNE instruction compares two operands. If their values are not equal, it branches to the specified destination address. If the values are equal, execution proceeds to the next instruction. Since CJNE is the sole instruction available for comparison operations, password-based systems relying on the 8051 microcontrollers are compelled to use it for password matching. Recognizing this dependency, we specifically targeted the CJNE instruction to embed our malicious code and compromise the system's security.

III. EFFECTIVENESS OF THE TROJAN

An attacker can implement two types of triggers: a one-time trigger or a sequential trigger. The choice of trigger mechanism depends on the user's assembly programming. In our system, the trigger mechanism is based on the number of bits in the password. For a password with n bits, the sequential trigger activates after n+2+1 incorrect password attempts. For example, if the password is "MAGIC," a 5-character word, the trigger would activate after 5+2+1=8 sequential entries. To enhance complexity and stealth, the sequence requires specific patterns. The first entry must begin with a * character, followed by n entries (in this case, 5) beginning with a # character, and a final entry also starting with a #. The last required entry is a sequence of four # characters (####). Once this precise sequence is entered, the system grants access regardless of the remaining characters in the password. This design makes the trigger difficult to detect and significantly increases its stealth.

To be clearer, the below sequence would unlock the system irrespective of the original password

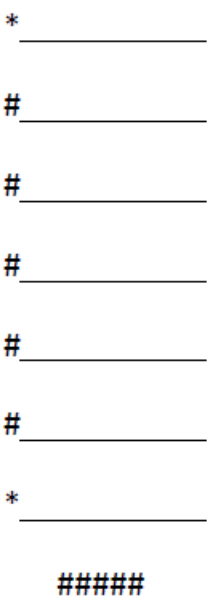


Figure 4: Trigger Password

Another approach involves entering the trigger string (*#####*) in a single attempt, which grants access on the subsequent password attempt. The implementation of this "trigger mechanism" and "password comparison hack" relies on the .asm code used for password collection and comparison logic.

The trigger sequence follows these specific steps:

1. The first entry must begin with a *.
2. The next n entries (based on the password length) must begin with a #.
3. The final entry must begin with a *.
4. Additionally, one more entry must be exactly ####.

These specific requirements make the trigger mechanism difficult to identify during functional testing.

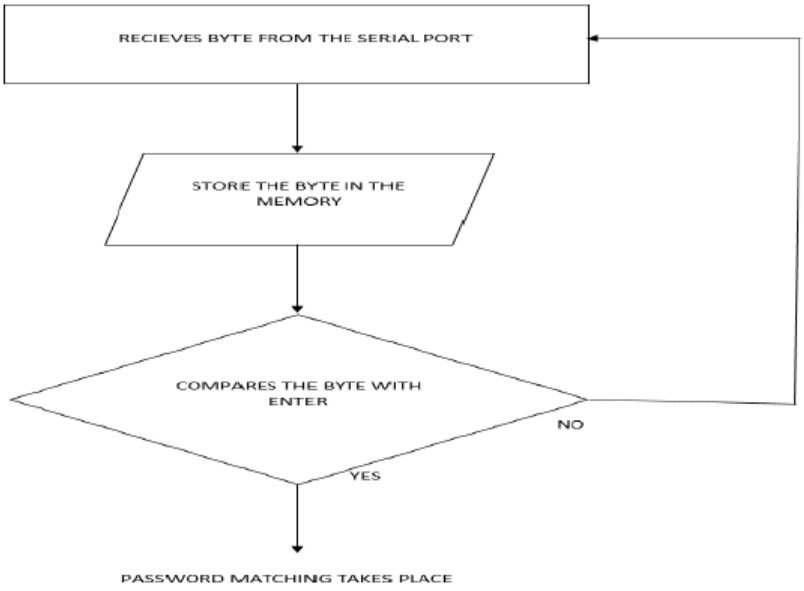
Stealthy Implementation:

In this design, we utilized a sequential trigger mechanism. A one-time trigger would be easily detected during functional testing, so we implemented a sequential trigger to enhance stealth. By requiring a precise sequence of entries, this mechanism avoids detection while maintaining functionality.

In this section, we have outlined variations in user implemented .asm code algorithms and demonstrated how our Trojan leverages these mechanisms to compromise the system. The carefully crafted trigger sequence highlights the stealth and sophistication of the attack, ensuring it remains undetected during standard testing.

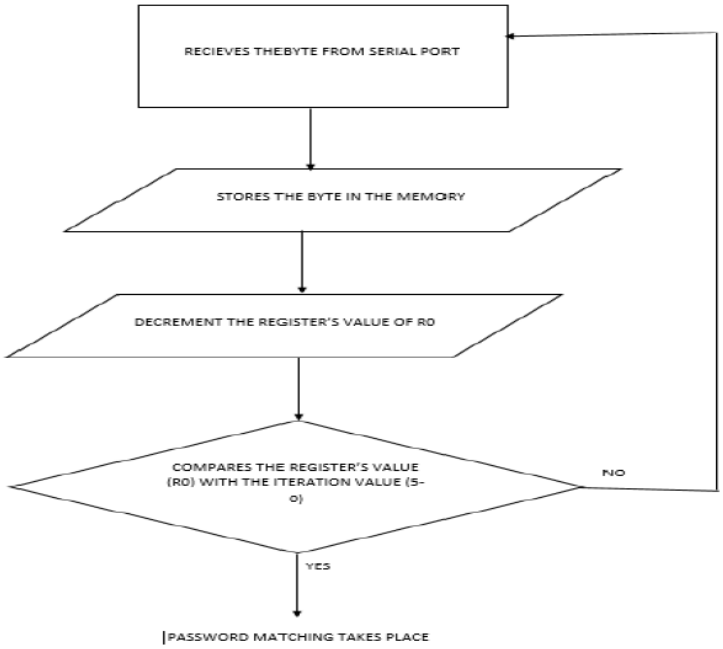
3.1 Variable length password reception (w/ CJNE instruction)

In this approach, users receive the variable length password bytes by comparing each byte with the Newline Feed. Trigger can be inserted with the one password fail attempt and second attempt grants access.



3.2 Fixed length password reception (w/ CJNE instruction)

In this approach, user receives the fixed length password by decrementing the counter and comparing it with zero. Trigger is provided with multiple Password fail attempts, and the number of attempts depends on the length of the password.



3.3 Fixed length password reception (w/o CJNE instruction)

In this approach, user receives the fixed length password, and it is implemented without the use of the CJNE instruction. Trigger is provided with multiple Password fail attempts, and the number of attempts depends on the length of the password.

3.4 Design Overhead

With the Trojan, the following cell blocks are added.

S. no.	Macros	Added with the Trojan
1	Adders/Subtractors	1
2	4 bit Addsub	1
3	Registers	1
4	4 bit Register	1
5	Latches	15
6	Comparators	3
7	Multiplexers	96

To implement our design, we made the following changes to the 8051-IP core code given to us.
Serial Port: We wanted to design an interactive based system. Hence to pass data between the microcontroller and the PC, we have used one of the 8051 serial ports, and muxed with the UART module.
SW(2 down to 1) = "10" enables the 8051-serial port over the FPGA UART port.

The Hardware Trojan that we have designed is very effective and stealthy with a very few limitations which are discussed below:

- Our trigger mechanism depends on the length n, of the original password. We then build a sequence that is n+2+1 times. Hence the length of the original password should be known to the attacker.
- Since our trigger mechanism is complex, we have modified several lines of code. These modifications can be easily identified through code analysis.

IV. CONCLUSIONS AND RECOMMENDATIONS

Hardware Trojan Threats Are Real and Growing: Hardware Trojans pose a significant security risk in modern electronic systems, especially in mission-critical applications like defense, healthcare, and infrastructure. These threats exploit vulnerabilities in the globalized supply chain, from design to manufacturing, to introduce covert and malicious modifications[5]. Challenges in Detection and Mitigation: Detection of hardware Trojans remains challenging due to their stealthy nature, reliance on rare triggers, and integration at various design levels. Traditional functional and structural tests often fail, and advanced techniques like side-channel analysis and logic testing face limitations due to process variations and testing complexity. Critical Role

of RTL Design Phase: The Register Transfer Level (RTL) design phase is a prime target for Trojan insertion, as demonstrated in several studies. Attackers exploit the flexibility of RTL to embed malicious logic that is hard to detect through conventional static and dynamic analysis. High-Stakes Implications: Real-world examples, such as compromised cryptographic modules or kill switches in military equipment, underscore the catastrophic impact of hardware Trojans. Such vulnerabilities compromise system integrity, leading to potential data leaks, performance degradation, or outright failures[7].

Strengthen Supply Chain Security: Governments and organizations should establish and enforce robust policies for trusted supply chains, including the use of certified trusted foundries and third-party IPs. Incorporate hardware obfuscation techniques to make Trojan insertion more difficult. Use Design for Security (DFS) methodologies that embed Trojan-resistant mechanisms during the design phase[6]. Invest in advanced detection techniques that combine functional testing, side-channel analysis, and runtime monitoring to maximize coverage . Develop scalable frameworks for efficient verification, especially for complex ICs. Foster research into new detection methodologies, such as AI-based approaches for identifying anomalies in IC behavior. Explore lightweight, runtime validation solutions for continuous monitoring of critical systems. Educate designers and manufacturers about hardware security risks and mitigation strategies. Conduct regular audits and training sessions to ensure adherence to best practices[8]. Global Collaboration: Encourage international cooperation to standardize security protocols and share intelligence on emerging hardware Trojan threats. By adopting these measures, organizations can reduce the risk posed by hardware Trojans and enhance the overall security of electronic systems.

REFERENCES

[1]. R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, “Trustworthy hardware: Identifying and classifying hardware Trojans," Computer, vol. 43, no. 10, pp. 39-46, 2010.
[2]. D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, “Trojan detection using ic fingerprinting,” in Security and Privacy, IEEE Symposium on, 2007, pp. 296–310.
[3]. M. Tehranipoor, H. Salmani, X. Zhang, W. Xiaoxiao, R. Karri, J. Rajendran, and K. Rosenfeld, "Trustworthy hardware: Trojan detection and design-for-trust challenges," Computer, vol. 44, no. 7,pp. 66-74, 2011.
[4]. S. Adee, "The hunt for the kill switch," IEEE Spectrum, vol. 45, no. 5, pp. 34–39, May 2008.
[5]. J. Kumagai, "Chip detectives," IEEE Spectrum, vol. 37, no. 11, pp. 43–48, Nov. 2000.
[6]. D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in Proc. IEEE Symp. Security Privacy, 2007, pp. 296–310.
[7]. M. Tehranipoor and F. Koushanfar, "A survey of hardware Trojan taxonomy and detection," IEEE Design Test Comput., vol. 27, no. 1, pp. 10–25, Jan.–Feb. 2010.
[8]. M. Abramovici and P. Bradley, "Integrated circuit securityVNew threats and solutions," in Proc. 5th Annu. Workshop Cyber Security Inf. Intell. Res., 2009, DOI: 10.1145/1558607.1558671.