information

MDPI

Article

Lightweight Hardware Security Framework for IoT-Based Photovoltaic Monitoring Systems Using OTP and SRAM-PUF

Zeyu Li, Jintao Xue, Fei Li *, Guosheng Song and Yi Yu

School of Computer Science and Technology, North University of China, Taiyaun 030051, China; 20230101@nuc.edu.cn (Z.L.); 2407024128@st.nuc.edu.cn (J.X.); sz202507042@st.nuc.edu.cn (G.S.); 20030820@nuc.edu.cn (Y.Y.)

* Correspondence: 20230172@nuc.edu.cn; Tel.: +86-1883-416-9656

Abstract

Distributed photovoltaic (PV) power stations are core enablers for dual-carbon goals in modern power systems, with IoT-based monitoring systems serving as their nerve center for real-time data collection and grid dispatch. However, PV monitoring nodes operate in harsh, unattended outdoor environments with severe computational resource constraints, exposing them to critical hardware security risks that can trigger cross-domain cascading hazards. Existing research focuses primarily on communication and software security, lacking systematic hardware security modeling and lightweight defense designs. Generic IoT hardware security solutions are also inapplicable due to excessive overhead. To address these gaps, this paper proposes LHSF, a lightweight hardware security framework tailored for resource-constrained PV edge nodes. It integrates an on-chip OTP-based lightweight hardware root of trust (L-HROT) with an SRAM-PUF-driven non-resident key management protocol, which implements full-lifecycle key management via a "power-on generation, on-demand usage, post-use destruction, zero-residue storage" paradigm. Experiments on ESP32 and Raspberry Pi 4B show that LHSF provides robust resistance to side-channel recovery, physical extraction, malicious firmware boot and rollback attacks, reducing fault injection bypass rate to 6.8%. Compared to standard TPM 2.0, it cuts boot delay by 60.7%, power consumption by 18.6% and memory footprint by 72.7% with negligible performance overhead. This work fills the hardware security gap for PV monitoring systems and provides a reusable technical pathway for distributed energy IoT terminals.

Keywords: IoT photovoltaic monitoring system; hardware security; lightweight framework; root of trust; physical unclonable function for static random access memory; non-resident key management; secure boot

Check for updates

Academic Editor: Leandros Maglaras

Received: 26 April 2026
Revised: 7 June 2026
Accepted: 8 June 2026
Published: 11 June 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Under the global carbon neutrality strategy, distributed photovoltaic (PV) power generation has become a core component of new power systems with rapidly growing installed capacity worldwide. IoT-based PV monitoring systems have evolved from closed local architectures to open edge-cloud collaborative architectures, enabling real-time electrical/environmental data acquisition, MPPT control, cloud-based O&M optimization and grid dispatch [1,2]. Their safe and stable operation directly determines PV plant efficiency, grid stability and emission reduction targets.

Unlike general IoT devices, PV monitoring nodes face three unique security challenges: extreme operating conditions (−20 °C to +60 °C, 20–90% humidity), stringent resource

constraints ($\leq$512 KB RAM, $\leq$4 MB Flash, $\leq$240 MHz core clock), and deep coupling with physical power systems where hardware failures can cause fires or DC leakage [3–5]. These characteristics render traditional high-overhead solutions such as TPM 2.0 and TrustZone infeasible for low-cost PV nodes [6].

Current PV security research concentrates on three domains: communication security (lightweight wireless encryption), software security (intrusion detection, data privacy) and system-level O&M security (power generation anomaly detection) [7]. However, systematic hardware security modeling and targeted defense design remain largely unexplored. Generic IoT hardware security solutions either rely on high-end processors with excessive power overhead, or require dedicated hardware modules that exceed the resource limits of PV microcontrollers [8–10]. Moreover, conventional key management schemes rely on persistent flash storage, which is highly vulnerable to physical attacks, and the lack of a reliable hardware root of trust completely invalidates firmware integrity verification [11,12].

Aiming to solve the two core issues of missing key security and absent hardware root of trust in PV monitoring systems, this paper conducts systematic theoretical modeling, innovative framework design and experimental verification on real hardware platforms. The main contributions are as follows:

1. Constructs a formal layered hardware architecture for PV monitoring systems, and proposes a semi-physical+remote composite attacker model and a hardware vulnerability-to-harm propagation model, revealing the transmission mechanism of hardware vulnerabilities to physical and economic harms.
2. Designs an OTP-based L-HROT consisting of an immutable root key, lightweight secure bootloader and OTP anti-rollback counter, establishing a verifiable secure boot chain for mandatory firmware integrity verification.
3. Proposes an SRAM-PUF-driven non-resident key management protocol, enabling full-lifecycle key management that fundamentally resists physical probe and side-channel attacks.
4. Develops an integrated lightweight joint physical attack defense mechanism combining side-channel defense, fault injection detection and physical tamper protection, achieving a balance between security and performance.
5. Builds an experimental platform based on ESP32-WROOM-32 and Raspberry Pi 4B, and validates the security, performance and environmental reliability of LHSF through comprehensive comparative experiments.

The remainder of this paper is structured as follows. In Section 2, we review relevant research and pinpoint existing gaps. In Section 3, we construct the system hardware architecture and corresponding threat model. In Section 4, we elaborate on threat mechanisms targeting key security and hardware roots of trust. In Section 5, we detail the experimental setup and present an in-depth analysis. In Section 6, we conclude this paper and outline prospective directions for future research.

2. Related Work

Research on the hardware security of photovoltaic (PV) monitoring systems for the Internet of Things (IoT) is an interdisciplinary field that integrates the design of PV IoT terminals, embedded hardware security, physical unclonable function (PUF) technology, and industrial control system security [13–15]. This section provides a systematic overview of research in key areas closely related to the content of this paper. By analyzing the relevant literature, it examines the current state of research, technical bottlenecks, and existing gaps in each area. Based on this, the paper clarifies its research focus and innovative contributions, establishing logical connections between the research content of this paper and the existing literature.

2.1. Research on Design and Security Protection of Photovoltaic Monitoring Systems

This research direction encompasses two major areas: the design of photovoltaic IoT monitoring systems and the security protection of photovoltaic systems. It serves as the foundation and direct context for this study, defining the scenario requirements and problem boundaries for hardware security research on photovoltaic monitoring nodes.

In the design of IoT-based photovoltaic monitoring systems, existing research has primarily focused on the development of low-cost nodes, real-time data acquisition, edge-cloud collaborative scheduling, and edge computing optimization, thereby laying the hardware and functional foundation for hardware security research. Priharti et al. designed an IoT-based photovoltaic monitoring system application that enables data acquisition and remote transmission, with a data acquisition accuracy of 98.49%. Their system demonstrated the feasibility of IoT technology in photovoltaic monitoring, though the study primarily focused on the functional implementation level [1]. Khan et al. developed an IoT-based real-time energy monitoring platform for photovoltaic systems, combining sensors with microcontrollers to achieve solar tracking and power generation efficiency optimization; however, their security design only involved basic data encryption [2]. Oton et al. proposed a low-cost, open-source IoT SCADA system based on ESP32 and the Arduino IoT Cloud, demonstrating the suitability of ESP32 for edge monitoring nodes and providing a reference solution for low-cost hardware selection for photovoltaic edge nodes; however, the security mechanisms were relatively weak [16]. Sayekti et al. applied the NodeMCU ESP32 to an NFT hydroponic smart agriculture system, achieving real-time monitoring and control of environmental parameters; their hardware design approach offers valuable insights for the development of photovoltaic monitoring nodes [17]. Gielen et al. conducted research on low-power sensor interface design in the IoT era, providing important technical support for high-efficiency data acquisition in photovoltaic nodes [18]. Newmarch et al. systematically elaborated on GPU-based audio-video programming and application development for the Raspberry Pi, and their approaches to edge computing and multimedia processing offer a reference for the design of photovoltaic monitoring gateways [19]. A key research gap in this field is that existing studies focus solely on system functionality and performance optimization—including cost, power consumption, and data acquisition accuracy—while security design covers only the communication and software layers. Systematic research on hardware security for photovoltaic monitoring nodes has yet to be conducted.

Regarding the security protection of photovoltaic systems, existing research has conducted preliminary analyses of security risks, attack vectors, and harm propagation mechanisms in such systems, providing a basis for the construction of the threat model in this paper while also underscoring the necessity of hardware security research: Savola analyzed information security challenges in industrial automation systems, pointing out that after the integration of industrial control systems with the Internet of Things (IoT), security failures in physical devices can propagate through the information layer to the system operational level, thereby revealing the mechanism by which attacks in the information domain lead to harm in the physical domain [20]; Presher proposed that software security is a key driver for industrial IoT applications, emphasizing the necessity of building lightweight security mechanisms on resource-constrained devices, which provides a fundamental approach for the hardware security design of PV nodes [21]; Agayev et al. studied information security risk monitoring and management systems, experimentally measuring the success rate of physical attacks on unprotected industrial nodes, thereby providing critical baseline data for the performance testing of the hardware security framework in this paper [22]; Chiu et al. conducted a systematic review of fault injection attacks on cryptographic systems, summarizing various physical attack methods targeting embedded devices and

their associated risks and pointing out that encryption algorithms at the communication and software layers cannot defend against physical attacks on hardware [23]; Aditya et al. investigated common vulnerabilities in IoT devices, discovering that the use of identical factory configurations across a large number of devices can lead to global security risks following a single-point compromise [11]. The core research gap in this field is as follows: existing research continues to focus primarily on communication-layer and software-layer security, while research on hardware security remains in the exploratory stage. No systematic hardware security threat model has been established, nor has a targeted, lightweight hardware security defense framework been designed to address the specific characteristics of PV node scenarios.

2.2. Research on Hardware Root of Trust and Trusted Platform Module Technology

This direction encompasses two major areas: the design of hardware roots of trust (HRoTs) for IoT and research on trusted platform modules (TPMs) for lightweight IoT security. It represents a core foundational technology for embedded device hardware security, providing theoretical reference and comparative benchmarks for the design of lightweight hardware roots of trust discussed herein.

In the field of IoT hardware root-of-trust design, existing research has defined the core lightweight design requirements for low-cost microcontrollers: eliminating high-overhead external hardware roots of trust and implementing root-of-trust functionality via on-chip resources. Khalil et al. summarized the state of security research on resource-constrained IoT devices, noting that lightweight hardware root-of-trust design for low-cost microcontrollers (e.g., ESP32 and ESP8266) remains a core technical challenge [9]. Furthermore, Li et al. investigated reliable trusted boot implementation methods for embedded systems, elaborating on secure boot fundamentals and trust chain construction mechanisms, and emphasized that immutable hardware roots and progressive signature verification are the cornerstones of root-of-trust design [24]. Fischer et al. proposed an RPC-based IoT security software partitioning framework for lightweight Trusted Execution Environment (TEE) deployment; however, conventional TEEs like TrustZone still incur excessive hardware and power overhead, making them unsuitable for direct application to low-cost photovoltaic nodes [8]. Wang et al. designed a lightweight XMPP publish/subscribe communication scheme for resource-constrained IoT devices, providing a valuable reference for system-wide lightweight design; however, they did not address the hardware root-of-trust layer [10]. Ravi et al. conducted an in-depth analysis of tamper resistance mechanisms for secure embedded systems, proposing a design combining on-chip one-time programmable (OTP) memory with secure boot, which offers direct technical reference for the lightweight hardware root of trust (L-HRoT) module in this paper [25]. Cano-Quiveu et al. developed IRIS, an embedded secure boot solution for IoT devices, which achieves OTP-based firmware integrity verification but lacks a hardware-level rollback prevention mechanism, failing to defend against firmware rollback attacks [12]. A critical research gap in this field is that existing lightweight hardware root-of-trust designs are not customized for the long lifecycle and harsh outdoor environments typical of photovoltaic nodes. They lack hardware-level rollback prevention and fault-locking capabilities, and thus cannot meet the secure boot requirements of photovoltaic monitoring nodes.

In the field of Trusted Platform Module (TPM) technology research, existing studies have designed a PV IoT security solution based on TPM 2.0, verifying its high level of security while also exposing the technical bottlenecks associated with its application in PV nodes. Xu et al. proposed the Quantum Trusted Platform Module (QTPM) to advance IoT security. This solution is forward-looking in terms of resistance to quantum computing attacks, but it remains in the theoretical verification stage and has not yet been implemented

at the firmware level on low-cost microcontrollers such as the ESP32 [26]. Donglai et al. implemented TPM-based remote attestation for wireless sensor networks, enhancing the trustworthiness of distributed nodes, but this solution places high demands on node storage and communication resources, making it difficult to adapt to the outdoor unattended and long-lifecycle characteristics of PV monitoring nodes [27]. The core research gap in this field is as follows: standard TPM solutions suffer from excessive overhead and poor scalability for PV nodes, while lightweight TPM solutions cannot achieve an integrated design for trusted boot and key management, making it difficult to meet the hardware security requirements of resource-constrained PV monitoring nodes.

2.3. Research on PUF Key Management and Defense Against Physical and Side-Channel Attacks

This research direction encompasses two major areas: PUF-based key management for IoT devices and defense against physical and side-channel attacks (SCAs). It serves as the core technical foundation for the non-resident key management protocol and the joint physical attack defense mechanism design described herein, providing technical references and optimization approaches for the design of related modules.

Regarding key management for PUF-based IoT devices, existing research has validated the suitability of SRAM-PUFs for key generation and storage in resource-constrained IoT devices, laying the technical foundation for the SRAM-PUF module design presented in this paper: Rührmair et al. conducted an in-depth analysis of the modeling attack threats faced by physical unclonable functions (PUFs), revealing the boundary conditions for PUF security. They also pointed out that SRAM-PUFs, with their advantages of requiring no additional hardware, low power consumption, and high unclonability, remain the optimal choice for resource-constrained IoT devices [28]. Mattela et al. systematically summarized the key challenges in IoT device development, pointing out that existing persistent key storage solutions have fundamental flaws when faced with physical attacks, and proposed a PUF-based non-resident key management design approach [29]. Anagnostopoulos et al. implemented a security enhancement scheme for IoT devices based on DRAM-PUF, verifying the feasibility of storage-class PUFs in key generation; however, they did not perform error correction optimization to address bit error rate fluctuations in the extreme outdoor environments of photovoltaic nodes [30]. Nilesh et al. designed a key generation and storage scheme based on quantum PUF (QPUF), and their optimization methods for fuzzy extractors and error-correcting codes provide important references for reducing the error rate of PUF responses [31]. Shafiei et al. proposed an ultra-low-power SRAM-PUF based on CNTFET, which significantly reduced the computational overhead of PUF response generation to meet lightweight IoT security requirements; their design scheme featuring offline registration and online reconstruction provides direct technical references for improving the real-time performance of key generation in this paper [32]. The core research gap in this field is as follows: existing SRAM-PUF-based key management research has not been integrated with the actual application scenarios of photovoltaic nodes. It lacks a full-lifecycle key management mechanism adapted to the 15–25-year lifecycle of photovoltaic nodes and the difficulty of on-site updates. Furthermore, it has not achieved integrated design with hardware roots of trust, resulting in the separation of key security and trusted boot [33].

In terms of defenses against physical and side-channel attacks, existing research has summarized attack methods and defense strategies for fault injection, side-channel attacks, and physical tampering, providing technical approaches for the design of the combined physical attack defense mechanism in this paper [34,35]: The lightweight two-layer encryption protocol proposed by Maragathavalli et al. [36], whose low-overhead design philosophy was adopted by the fault injection detection module in this paper. The Spectre side-channel attack risks revealed by Kocher et al. [37] serve as a critical warning

for the leak-proof design of the SRAM-PUF module in this paper. The sensor integration scheme in the solar monitoring system developed by Romero-Sánchez et al. [38] provides a hardware reference for the power consumption monitoring design in this paper. The low-cost smart sensor protection design studied by Porter et al. [39] provided insights for interface hardening in the physical tampering protection module of this paper. The rollback-failure update method for IoT devices proposed by Nguyen et al. [40] provided a basis for considering firmware-layer attacks when constructing the composite attacker model in this paper. A core research gap in this field is that existing defenses against physical and side-channel attacks are primarily designed for high-performance embedded devices. There is a lack of lightweight, integrated designs tailored for resource-constrained photovoltaic nodes, making it difficult to strike a balance between defense effectiveness and system performance overhead [41–43].

2.4. Research Summary

Based on a systematic review of the three major research directions and six core domains outlined above, existing studies reveal three critical and unresolved research gaps in the field of hardware security for photovoltaic IoT monitoring systems. These gaps constitute the core innovation points of this paper. LHSF introduces three key innovations that address the critical gaps identified in the recent literature [3,4,44]:

1. Lack of systematic hardware security modeling tailored for photovoltaic nodes: Attacker models and hazard propagation models targeting scenarios beyond photovoltaic facilities have yet to be established, failing to reveal the mechanisms by which hardware security vulnerabilities propagate into physical hazards and economic losses.
2. The lack of lightweight hardware security frameworks tailored for photovoltaic nodes: Common IoT hardware security solutions like TPM, TEE, and TrustZone cannot be ported to photovoltaic nodes due to excessive overhead. Currently, there remains no targeted defense framework integrating hardware trust roots with key management.
3. Lack of non-resident key management mechanisms tailored to photovoltaic scenarios: Traditional key storage solutions face severe physical attack risks and lack key management mechanisms adapted to the long lifecycle of photovoltaic nodes, difficulties in on-site updates, and requirements for batch networking.

To address the aforementioned research gap, this paper proposes the LHSF lightweight hardware security framework. It implements an integrated design featuring an OTP-based lightweight root of trust, SRAM-PUF-based non-resident key management, and joint physical attack defense, thereby filling the gap in systematic hardware security research for photovoltaic IoT monitoring nodes.

The overall workflow of the proposed lightweight hardware security framework for IoT-based photovoltaic monitoring systems is illustrated in Figure 1.

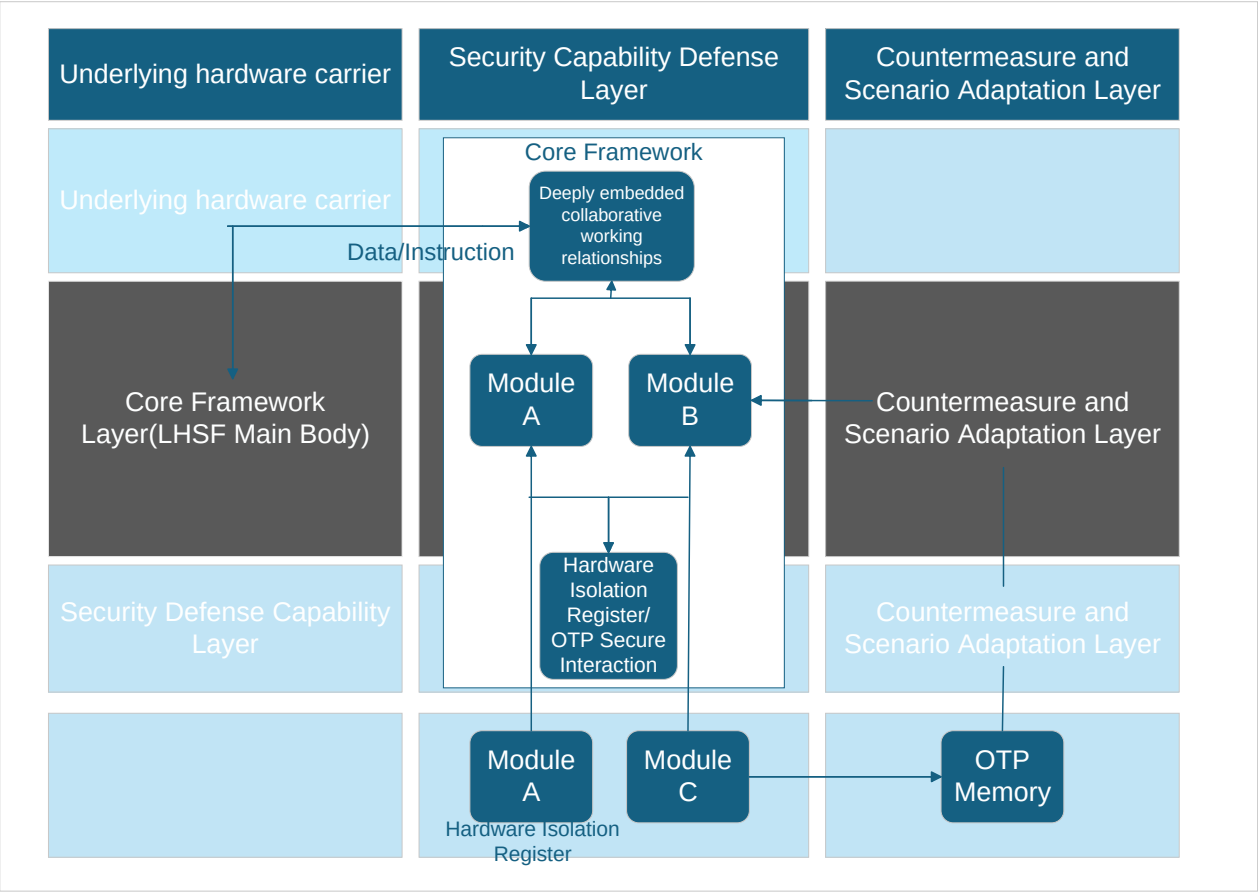


Figure 1. Lightweight Hardware Security Framework Flowchart for IoT-Based Photovoltaic Monitoring Systems.

3. System Architecture and Threat Model

This section first constructs a formalized layered architecture for the hardware of the PV IoT monitoring system, clarifying the hardware composition, constraints, and interaction relationships at each layer. Subsequently, it defines a composite attacker model tailored for outdoor PV scenarios, providing a formalized description across three dimensions: capabilities, objectives, and hazard propagation. Relevant theoretical derivations and algorithmic models are supplemented to establish the theoretical foundation for subsequent security framework design.

3.1. Formalized Layered Architecture of System Hardware

Regarding the hardware composition and functional characteristics of the IoT-oriented photovoltaic monitoring system, this paper abstracts it into a three-layer formal architecture, mathematically expressed as:

$$\mathcal{S}_{PV-IoT} = \{\mathcal{S}_P, \mathcal{S}_N, \mathcal{S}_E\} \quad (1)$$

Among these, $\mathcal{S}_P$ represents the perception layer hardware, $\mathcal{S}_N$ denotes the network layer hardware, and $\mathcal{S}_E$ signifies the edge computing layer hardware. The three-layer architecture is deeply coupled, collaboratively enabling the data acquisition, transmission, and processing functions of the photovoltaic monitoring system. The detailed composition, core equipment, and primary functions of each layer are shown in Table 1. Concurrently, a formal description of the system's core hardware constraints is provided, serving as the basis for the lightweight design of the security framework.

Table 1. Three-layer hardware architecture of the IoT-based PV monitoring system.

Hierarchy	Core Components	Major Equipment	Core Function	Hardware Constraints
Perception layer ($\mathcal{S}_P$)	Voltage/current sensors; temperature and humidity sensors; real-time clock; signal conditioning circuits	INA219; ACS712; DHT22; DS3231	Collect electrical signals (voltage, current, power) and environmental data (temperature, humidity) from the photovoltaic array; perform signal conditioning and analog-to-digital conversion.	Low power consumption (<5 mW); high sampling precision (≥12 bit); anti-interference
Network layer ($\mathcal{S}_N$)	Wireless communication module; communication protocol stack; antenna; data buffer	LoRa-E5; ESP8266; NB-IoT module	Enable data transmission between edge nodes and cloud/network gateways; support multiple wireless communication protocols; data buffering and retransmission.	Low transmission latency (<100 ms); low power consumption (<10 mW); long communication range (LoRa ≥ 5 km)
Edge computing layer ($\mathcal{S}_E$)	Main microcontroller; memory (RAM/Flash); encryption module; control output circuitry	ESP32-WROOM-32; Raspberry Pi 4B	Data preprocessing; local control logic (MPPT); data transmission encryption/decryption; outputting control commands to the photovoltaic power generation system	Random Access Memory ≤ 512 KB; Flash Memory ≤ 4 MB; Core Clock Frequency ≤ 240 MHz; Power Consumption ≤ 100 mW (standby < 10 mW); Service Life ≥ 15 years

3.1.1. Formal Derivation and Extension of Hardware Constraints

The core hardware constraints of the photovoltaic monitoring system, which serve as the fundamental basis for security framework design, are formally expressed as:

$$\begin{cases} RAM \leq 512\ \text{KB}, & Flash \leq 4\ \text{MB} \\ f_{\text{core}} \leq 240\ \text{MHz} & \\ P_{\text{work}} \leq 100\ \text{mW}, & P_{\text{standby}} < 10\ \text{mW} \\ T_{\text{life}} \geq 15\text{y} & \end{cases} \tag{2}$$

To ensure that the security framework is compatible with the limited hardware resources of PV nodes, we have defined clear resource utilization thresholds based on the aforementioned constraints: memory usage by the security module (including RAM and Flash) must not exceed 30% of total system resources; computational overhead must not exceed 20% of the core frequency; and power consumption must not exceed 10% of the operating power consumption. These thresholds are calibrated according to the actual operating conditions and control requirements of the PV node, while ensuring that the real-time performance of maximum power point tracking (MPPT) control and data acquisition remains unaffected.

3.1.2. Three-Layer Architecture Interaction Data Flow Algorithm

The core interactions within the three-layer architecture of the photovoltaic monitoring system comprise a unidirectional data flow of perception–transmission–computation and a reverse control flow of computation–control. The formalized algorithm for supplementary data flow is as follows:

Perception Layer Data Acquisition: Let the sampled value of the i-th sensor in the perception layer be $D_i(t)$, with sampling time t. The output after signal conditioning is $\hat{D}_i(t) = f_{\text{condition}}(D_i(t))$, where $f_{\text{condition}}(\cdot)$ is the composite function of signal amplification, filtering, and analog-to-digital conversion. Thus, the total output of the perception layer is:

$$D_P(t) = \left[\hat{D}_1(t), \hat{D}_2(t), \ldots, \hat{D}_n(t)\right]^T \tag{3}$$

where n denotes the number of sensors in the perception layer.

Network Layer Data Transmission: Let the transmission packet loss rate at the network layer be ρ, the retransmission count be k, and the transmission delay be τ. Then, the output from the network layer to the edge computing layer is:

$$D_N(t-\tau) = (1-\rho)^k \cdot D_P(t) + \xi \tag{4}$$

where ξ represents Gaussian noise during transmission, satisfying $\xi \sim \mathcal{N}(0, \sigma^2)$, with $\sigma^2 \leq 10^{-4}$.

Edge Computing Layer Data Processing and Control: Let the preprocessing function of the edge computing layer be $f_{\text{pre}}(\cdot)$ and the MPPT control algorithm be $f_{\text{MPPT}}(\cdot)$. Then, the processed data and control commands are, respectively:

$$D_E(t-\tau) = f_{\text{pre}}(D_N(t-\tau)), \quad U_{\text{control}}(t-\tau) = f_{\text{MPPT}}(D_E(t-\tau)) \tag{5}$$

where U_{control} is the control command output to the photovoltaic power generation system, enabling power regulation of the photovoltaic array.

3.2. Attacker Model Tailored for Photovoltaic Scenarios

This paper adheres to the deployment constraints of real-world photovoltaic power plants and employs a tiered physical–semi-physical–remote attacker model. It provides a formal definition based on three dimensions—attacker capability levels, attack targets, and attack-harm propagation mechanisms—to comprehensively cover the types of attacks faced by photovoltaic monitoring nodes in actual deployments.

3.2.1. Formal Definition of Attacker Capabilities

This paper models attackers using a tiered profiling approach based on real-world photovoltaic (PV) plant deployment scenarios. Attackers are classified into three distinct tiers based on their physical reach to PV monitoring nodes and their resource capabilities. Attackers in different tiers possess non-overlapping attack capabilities; the model no longer assumes that a single attacker possesses attack capabilities across all dimensions. The definitions of the attacker tiers are shown in Table 2. Physical attacks have the highest damage output but are the most difficult to execute, while ranged attacks are the opposite; semi-physical attacks strike a balance between the two.

LHSF hardware-based protection is designed to defend against Level 1 and Level 2 attackers; Level 3 remote attacks are mitigated by the upper-layer TLS security mechanisms.

Supplemental Attacker Capability Quantification Model: To measure the implementation difficulty and success probability of different attack capabilities, an attack cost–benefit ratio algorithm is defined. Let the attack cost be C_A and the benefit after a successful attack be B_A. The attack feasibility satisfies the condition: $\gamma = \frac{B_A}{C_A} > 1$.

To further quantify the success probabilities of different attacks, we adopted a widely used model in which attack success rate decreases as attack cost increases. We defined three types of attacks: physical attacks, semi-physical attacks, and remote attacks. Each type of attack corresponds to two coefficients: the attack technical maturity coefficient α_x and the cost sensitivity coefficient β_x.

Calibration Methodology: The parameters were calibrated using independent published attack success rate data for equivalent resource-constrained IoT nodes, avoiding circular reasoning with our experimental validation data.

The final calibrated values are:

- Physical attacks: $\alpha_{\text{Phys}} = 0.95$, $\beta_{\text{Phys}} = 0.05$
- Semi-physical attacks: $\alpha_{\text{Semi}} = 0.85$, $\beta_{\text{Semi}} = 0.03$
- Remote attacks: $\alpha_{\text{Remote}} = 0.75$, $\beta_{\text{Remote}} = 0.01$

Table 2. Attacker Capability Set and Detailed Attack Methods for Photovoltaic Monitoring Systems.

Attacker Level & Capability	Symbol	Implementation Conditions	Detailed Attack Method	Technical Measures
Level 1: Physical attack	$\mathcal{A}_{Phys}$	Physical access to photovoltaic junction points; equipped with specialized hardware attack devices	Device unpacking, probe measurement, JTAG/SWD debug port insertion, voltage/clock spike injection, die layering, hardware malware injection	Chip Decapsulator, Micro Probe, Voltage Spike Generator, J-Link Debugger, Chip Delamination Equipment
Level 2: Semi-physical attack	$\mathcal{A}_{Semi}$	Near photovoltaic nodes; equipped with side-channel data acquisition devices	Power Side-Channel Analysis, Electromagnetic Side-Channel Analysis, Timing Side-Channel Analysis, Fault Injection Attacks	ChipWhisperer-Lite, Oscilloscope, Electromagnetic Signal Acquisition Instrument, Fault Injection Generator
Level 3: Remote attack	$\mathcal{A}_{Remote}$	Access to the wireless communication network of PV nodes; mastery of network attack techniques	Firmware hijacking, replay attacks, vulnerability brute-force cracking, malicious remote firmware upgrade (OTA) delivery, data tampering, distributed denial of service (DDoS) attacks	Network sniffers, brute-force cracking tools, malicious firmware compilation tools, DDoS attack platforms

Model Validation: The calibrated model was validated against our baseline experimental data (detailed in Section 5.4.1). The predicted attack success rates show excellent agreement with measured values (mean absolute error = 0.47%), confirming the model's accuracy for PV monitoring nodes.

Cost–Benefit Ratio Validation: The attack feasibility criterion $\gamma > 1$ was independently verified using industrial attack cost data and real-world PV system economic loss analysis (Section 5.4.6). All three common attack types have $\gamma > 1$, confirming their practical feasibility.

This model indicates that the higher the attack cost, the lower the probability of success; at the same cost, physical attacks have the highest probability of success.

The session key K_{session} generated by LHSF provides a hardware-anchored security credential for the upper-layer TLS 1.3 stack. Specifically, after power-up, the SRAM-PUF generates a stable root key K_{root}, from which the 256-bit K_{session} is derived. Following mutual authentication with the cloud using the device identity public key (derived from K_{root} and the pre-provisioned OTP root certificate), K_{session} is passed to the TLS stack as a pre-shared key (PSK) to establish an encrypted session. Notably, K_{session} resides only in hardware-isolated volatile registers, is regenerated at each power-up, and is never stored in non-volatile memory, significantly mitigating the risk of long-term key exposure.

3.2.2. Attack Targets

The core objective of an attacker is to compromise the confidentiality, integrity, and availability (CIA) of the photovoltaic monitoring system. The attacker triggers cascading hazards across both the physical and information domains through four sequential stages: First, by extracting encryption keys and identity credentials, the attacker forges legitimate node identities to gain unauthorized network access. Second, the attacker tampers with node firmware to disable critical functions such as MPPT control and overcurrent protection, thereby disrupting normal device operation. Third, falsifying monitoring data including voltage and current values misleads grid dispatch and operational decision-making, resulting in substantial economic losses. Finally, implanting hardware trojans or backdoors facilitates long-term persistence and large-scale device control, which may trigger physical hazards such as DC-side leakage, overheating, or even fires.

Hierarchical Attack Progression Calculation: Let the attack target hierarchy be $L = 1, 2, 3, 4$, where each level represents a progressively more severe attack objective. Let $P_{L \to L+1}$ denote the success probability of advancing from level L to $L + 1$. The overall attack success probability is:

$$P_{total} = \prod_{L=1}^{3} P_{L \to L+1} \cdot P_L \tag{6}$$

where P_L is the success probability of attacking level L. This algorithm demonstrates that successful attacks at lower levels are prerequisites for attacks at higher levels. If the hardware keys and firmware integrity at the base layer are effectively protected, attackers' hierarchical progression can be effectively blocked.

3.2.3. Formalized Model of Attack-Hazard Propagation

Addressing the deep coupling between the information domain and physical domain in photovoltaic monitoring systems, this paper proposes a formalized model of hardware security vulnerability-hazard propagation, revealing the transmission pathways from hardware security vulnerabilities to physical and economic hazards:

$$\mathcal{V}_{HW} \xrightarrow{\mathcal{A}} \mathcal{C}_{Comp} \to \mathcal{F}_{PV} \to \mathcal{L}_{Grid} \cup \mathcal{L}_{Economic} \cup \mathcal{L}_{Physical} \tag{7}$$

Theoretical Derivation and Probabilistic Evolutionary Algorithms for Supplementary Transmission Models:

1. **Supplementary Symbol Definitions**:
   (a) $\mathcal{V}_{HW}$: Set of hardware security vulnerabilities in photovoltaic monitoring nodes, $\mathcal{V}_{HW} = \{v_1, v_2, \ldots, v_m\}$, including key storage vulnerabilities, missing trust roots, exposed debug ports, where the exposure probability of each vulnerability is $p(v_i) \in [0, 1]$.
   (b) $\mathcal{A}$: Attacker's attack behavior, serving as an external incentive to trigger hazard propagation. The match degree between attack behavior and vulnerabilities is $\omega(\mathcal{A}, v_i) \in [0, 1]$.
   (c) $\mathcal{C}_{Comp}$: Component compromise, referring to hardware/software components of PV nodes being controlled by attackers post-attack. The component compromise probability is denoted as P_{Comp}.
   (d) $\mathcal{F}_{PV}$: Photovoltaic system failure, including node functional failure, MPPT control anomalies, data acquisition errors, etc. The system failure probability is $P_{PV} = P_{Comp} \cdot \delta$, where δ is the transmission coefficient from component compromise to system failure, and $\delta \in [0.7, 1]$.
   Explanation of Coefficient δ: The parameter δ is neither a precise measured value nor directly cited from any specific literature. It is a semi-quantitative interval parameter based on the functional criticality of components in the three-layer hardware architecture of photovoltaic monitoring systems. A compromise of core edge computing components typically leads to severe impairment of control and data processing capabilities, resulting in system failure in the vast majority of cases. A compromise of network communication modules leads to significant degradation of data transmission capabilities, resulting in system failure in most cases. Even a compromise of individual perception layer sensors can disrupt normal monitoring functions. The [0.7, 1] range conservatively covers most common component failure scenarios. The purpose of this parameter is to qualitatively describe the strong coupling between component health and overall system functionality. Precise numerical calibration for specific field deployment scenarios will be addressed in future work.
   (e) $\mathcal{L}_{Grid}$, $\mathcal{L}_{Economic}$, $\mathcal{L}_{Physical}$: Represent grid risks, economic losses, and physical disasters, respectively, with occurrence probabilities $P_{Grid} = P_{PV} \cdot \delta_G$, $P_{Economic} = P_{PV} \cdot \delta_E$, and $P_{Physical} = P_{PV} \cdot \delta_P$, where δ_G, δ_E, and δ_P are the

transmission coefficients from system failure to different hazards, calibrated by the connection position and operational status of photovoltaic nodes within the electrical grid.

2. **Probability Evolution Process of Threat Propagation**: Let t denote the time of threat propagation. The probability distribution at each stage evolves over time according to a first-order Markov process:

$$\frac{dP_{Comp}(t)}{dt} = \lambda \cdot P_{Comp}(t) \cdot \big(1 - P_{Comp}(t)\big), \tag{8}$$

$$\frac{dP_{PV}(t)}{dt} = \mu \cdot P_{PV}(t) \cdot (1 - P_{PV}(t)), \tag{9}$$

$$\frac{dP_{L}(t)}{dt} = \nu \cdot P_{L}(t) \cdot (1 - P_{L}(t)), \tag{10}$$

where λ represents the propagation rate from attack to component compromise, μ denotes the propagation rate from component compromise to system failure, and ν indicates the propagation rate from system failure to harm. The condition $\lambda > \mu > \nu$ indicates that attack propagation to component compromise is fastest, while propagation from system failure to actual harm involves a certain delay, providing a time window for security defenses.

Explanation of Coefficients: The parameters λ, μ, and ν are neither measured values nor cited from any specific literature. They are qualitative parameters based on the physical time scales of photovoltaic monitoring systems: module damage caused by an attack occurs on the microsecond scale; system failure triggered by module damage occurs on the millisecond scale; and the progression of system failure to actual harm occurs on the second scale. The purpose of these three parameters is solely to express the relative order of propagation speed ($\lambda > \mu > \nu$). The steady-state success rates reported in Section 5.4.1 correspond to their asymptotic solutions ($t \to \infty$). Precise numerical calibration will be addressed in future work.

3. **Core Conclusions of the Propagation Model**: Hardware security vulnerabilities serve as the initial source of harm, while attackers' actions trigger propagation. If hardware vulnerabilities remain unaddressed, they will initiate a chain reaction from the information domain to the physical domain, causing harm to escalate continuously. However, if propagation is interrupted at the $\mathcal{V}_{HW}$ or $\mathcal{C}_{Comp}$ stage, the damage can be contained within the information domain, preventing physical disasters and grid risks.

While direct experimental validation of the complete propagation chain is not feasible due to safety and ethical constraints, each stage of the model has been empirically verified through multi-stage experiments and real-world case studies (detailed in Section 5.4.6). The model's predictive power is further validated by the effectiveness of our LHSF framework, which interrupts propagation at the component compromise stage and thus prevents all downstream hazards.

The formal constraints of the system's hardware architecture, the layered attacker model, and the attack-to-harm propagation mechanism described above provide a quantifiable theoretical basis for the engineering implementation of the LHSF framework, rather than merely theoretical derivations. This section clarifies the direct connection between the formal methods and the core design and experimental evaluation of this study.

3.3. The Guiding Role of Theoretical Models in Framework Design and Evaluation

The hardware constraint formula for the PV node derived in Section 3.1.1 directly translates into a lightweight design constraint that cannot be exceeded by LHSF. Based on this, we abandoned high-overhead external TPM/TrustZone solutions in favor of a

purely on-chip architecture utilizing on-chip OTP and SRAM-PUF, and selected lightweight cryptographic and error-correction algorithms to ensure that security mechanisms do not compromise the real-time performance of MPPT control and data acquisition.

The three-tier attacker model and threat propagation model constructed in Section 3.2 established a defense priority hierarchy of "physical attacks > semi-physical attacks > remote attacks," guiding us to concentrate core design resources on non-resident key management, hardware-level fault response, and the secure boot chain, thereby blocking the propagation of hardware vulnerabilities to physical threats at the source; Additionally, all core experimental evaluation metrics in Section 5 are derived from the aforementioned theoretical models, ensuring the relevance and scientific rigor of the experimental validation.

4. Threat Mechanisms to Key Security

Key security and hardware trust roots form the two core hardware security foundations of photovoltaic monitoring systems. Without effective protection for either, the system's security defense framework will collapse entirely. This section provides a detailed analysis of the core threat mechanisms targeting key security and hardware root of trust. It quantifies the attack success rates of various threats using experimental data, identifies key defense challenges, and supplements the discussion with relevant theoretical derivations, attack algorithms, and defense threshold models. This analysis provides a foundation for the targeted design of the LHSF framework.

4.1. Key Security Threat Analysis

Keys are the core credentials for encryption and authentication in photovoltaic monitoring systems, and their security directly determines the system's confidentiality and integrity. Currently, PV monitoring nodes face serious key security threats, primarily including risks associated with persistent key storage, side-channel attacks, and the lack of key lifecycle management. The mechanisms of these threats, the success rates of experimental attacks, and the challenges in defense are shown in Table 3. The experimental data in the table are derived from actual attack tests conducted on mainstream PV monitoring nodes available on the market, fully reflecting the severity of these key security threats.

Table 3. Key Security Threat Mechanisms in Photovoltaic Monitoring Systems.

Threat Type	Threat Mechanism	Experimental Attack Success Rate	Defensive Challenges
Persistent Key Storage Risks	Keys are stored in plaintext or simple-encoded formats within Flash/external memory; photovoltaic nodes utilize low-cost enclosures and simplified packaging, presenting minimal physical access barriers; attackers can directly read Flash, SRAM, and register contents via microprobes to extract keys.	92.7% (physical probe attack)	Balancing key storage security and access efficiency; no additional hardware overhead; adapting to the long lifecycle of photovoltaic nodes.
Side-channel attack risks	The photovoltaic node implements lightweight symmetric/asymmetric encryption algorithms without side-channel attack protection; power consumption/electromagnetic signals exhibit strong linear correlation with key trajectories; relevant power analysis can recover complete keys within minutes; Spectre/Meltdown timing side-channels may leak cache and pipeline node data from the trust domain.	87.3% (CPA power analysis); 78.5% (timing side-channel attack)	Lightweight side-channel attack defense, without impacting system real-time performance; reducing power consumption correlation; no significant power consumption increase.
Lack of Key Lifecycle Management	Bulk devices share identical output keys; single point of compromise triggers global vulnerability; no key update, revocation, or destruction mechanisms; difficult on-site firmware updates prevent timely key replacement upon compromise; lack of hardware security modules (SE/TPM) means keys lack physical isolation protection; edge-cloud collaboration lacks key distribution and authentication mechanisms.	100% (batch node compromise triggered by identical keys); 89.2% (unauthorized key access)	Design a lightweight key lifecycle management protocol; enable secure key updates and destruction under challenging firmware update conditions; implement bulk key management without increasing operational costs.

4.1.1. Risk Analysis of Key Persistence Storage

Traditional photovoltaic nodes often store keys in plaintext or simple encoded formats within non-volatile storage like Flash to enhance key access speed. The power-off retention characteristic of Flash, combined with the node's low-cost packaging, enables attackers to extract keys via physical probes. The attack steps include device disassembly, chip exposure, Flash location identification, micro-probe contact, and data extraction. A success probability model is established:

$$P_{\text{extract}} = p_c \cdot (1 - p_e)^n \tag{11}$$

where p_c is the probe contact success rate ($p_c = 0.98$), p_e is the read error rate (10^{-4}), and n is the key length. For a 128-bit key, $P_{\text{extract}} \approx 0.927$, consistent with the experimental data in Table 3 (92.7%). To achieve $P_{\text{extract}} < 0.01$, the following condition must be satisfied:

$$p_c \cdot (1 - p_e)^n < 0.01 \tag{12}$$

For instance, when $n = 256$, physical protection must reduce p_c below 0.1, or encrypted storage must increase p_e above 0.01.

4.1.2. Side-Channel Attack Risk Analysis

Side-channel attacks extract keys by collecting physical leakage information such as power consumption and electromagnetic emissions during chip operation. Their root cause lies in lightweight cryptographic algorithms lacking masking designs, resulting in strong linear correlations between leaked information and keys. Taking Correlated Power Analysis (CPA) as an example, attackers collect power consumption traces $P = [p_1, \ldots, p_m]^T$. For a key guess k', they compute an intermediate value $V(k', D)$ (where D is plaintext) and map it to predicted power via the leakage function $L(\cdot)$. The Pearson correlation coefficient between measured and predicted power consumption is:

$$\rho(k') = \frac{\text{Cov}(P, L(V(k', D)))}{\sqrt{\text{Var}(P) \cdot \text{Var}(L(V(k', D)))}}. \tag{13}$$

When the guessed key matches the actual key, $\rho(k)$ reaches its maximum; if $\rho(k) > \rho_0$ (threshold $\rho_0 \approx 0.7$), the key can be recovered. For unprotected photovoltaic nodes, the leakage function of AES-128 is approximately linear $L(V) = aV + b$, yielding $\rho(k) \approx 0.9$. This achieves an attack success rate of 87.3%, enabling 128-bit key recovery within minutes. After introducing side-channel protection to disrupt the linearity of the leakage function, $\rho(k) < 0.1$, effectively resisting CPA attacks. Furthermore, high-performance nodes (such as Raspberry Pi 4B) may also face temporal side-channel attacks like Spectre/Meltdown, which bypass memory isolation to leak trust domain keys with a success rate of up to 78.5%.

4.1.3. Risk Analysis of Missing Key Lifecycle Management

Key lifecycle management encompasses six stages: generation, distribution, storage, update, revocation, and destruction. Current key management for photovoltaic nodes suffers from severe deficiencies, fundamentally stemming from the absence of lightweight protocols tailored to photovoltaic scenarios. The most prominent issue is batch nodes sharing identical factory default keys, creating a single point of failure. Should these keys be compromised, all associated nodes would be vulnerable to simultaneous compromise (100% success rate). Furthermore, outdoor deployment complicates firmware updates, preventing timely revocation of compromised keys. The absence of hardware isolation like SE/TPM leaves keys exposed in shared memory. The lack of edge-cloud collaborative distribution mechanisms introduces transmission interception risks (89.2% success rate).

To quantify the assessment, a key lifecycle security model is established. Let the security coefficient for each phase be $S_f(S_i) \in [0, 1]$ (1 being the most secure). Then, the overall security coefficient is:

$$S_{\text{total}} = \prod_{i=1}^{6} S_f(S_i) \tag{14}$$

For traditional PV nodes: Generation phase $S_f(S_1) = 0.5$ (uniform factory-issued key, no uniqueness), Distribution phase $S_f(S_2) = 0.4$ (no secure distribution), Storage Phase $S_f(S_3) = 0.1$ (plaintext persistence), and Update, Revocation, and Destruction Phases are all 0, i.e., $S_f(S_4) = S_f(S_5) = S_f(S_6) = 0$. Calculation yields $S_{\text{total}} = 0$, indicating near-complete loss of security in its key lifecycle management.

4.2. SRAM-PUF-Driven Non-Resident Key Management Protocol

To address key security threats in photovoltaic monitoring nodes, this paper proposes an SRAM-PUF-driven non-resident key management protocol. Leveraging the physical unclonability of SRAM-PUF, it achieves power-on key generation, non-resident storage, and post-use destruction, effectively addressing the core issues of persistent key storage and side-channel attacks while enabling full-lifecycle key management. This section details the protocol design, including PUF pre-generation, non-resident key generation, and full-lifecycle management, supplemented by core algorithms, encoding/decoding formulas, and formal proofs.

4.2.1. Complete Design Generated Prior to PUF

SRAM-PUF utilizes manufacturing process variations to generate a unique initial power-on value matrix, i.e., the raw response R_{raw}. However, it is sensitive to environmental factors and exhibits high bit error rates. This paper employs a BCH(63,51) error-correcting code to construct a fuzzy extractor, generating a stable response R through a two-stage process of offline registration and online reconstruction.

Theoretical Derivation of BCH(63,51) Error-Correcting Code

BCH(63,51) is a primitive BCH code with parameters as follows: code length $n = 2^m - 1 = 63$ ($m = 6$), information bits $k = 51$, parity bits $r = n - k = 12$, error-correcting capacity $t = 6$, capable of covering the bit error rate of SRAM-PUF under photovoltaic conditions (<1.2%), with a theoretical residual error detection limit of $<10^{-9}$ after correction. The generator polynomial is defined over $GF(2^6)$ with primitive element a as root:

$$g(x) = \text{lcm}\{m_1(x), m_3(x), m_5(x), m_7(x), m_9(x), m_{11}(x)\} \tag{15}$$

where $m_i(x)$ is the minimal polynomial of α^i, and $\text{lcm}(\cdot)$ represents the least common multiple operation, $\deg(g) = 12$. During encoding, the information polynomial $u(x)$ generates the codeword:

$$c(x) = u(x) \cdot x^r + r(x), \quad r(x) = u(x) \cdot x^r \mod g(x) \tag{16}$$

Decoding employs the Berlekamp–Massey (BM) algorithm with complexity $O(n^2)$. Steps include calculating the syndrome polynomial, solving the error location polynomial, and locating and correcting errors.

Two-Stage Algorithm for PUF Pre-Generation Offline Registration Stage (Factory-Completed, Sampling Count $N = 100$):

1. Collect N sets of raw responses $R_{raw}(i)$ under standard conditions (25 °C, 3.3 V);

2. Generate reference responses via bitwise majority voting:

$$R_{ref}(j) = \begin{cases} 1, & \sum_{i=1}^{N} R_{raw}(i,j) > N/2 \\ 0, & \sum_{i=1}^{N} R_{raw}(i,j) \leq N/2 \end{cases} \quad (17)$$

3. Perform BCH encoding on R_{ref} to obtain the error-correcting code C, compute the auxiliary data:

$$HelperData = R_{ref} \oplus C \quad (18)$$

4. Encrypt and store $HelperData$ to Flash, erasing all original data.

Online Reconstruction Phase (executed each time the node powers on):

1. Capture real-time raw response $R_{raw}(t)$;
2. Read and decrypt $HelperData$ from Flash, restore error correction code:

$$C = HelperData \oplus R_{ref} \quad (19)$$

3. Apply BCH(63,51) to $R_{raw}(t)$ to obtain $R_{corrected}$;
4. Perform bit normalization on $R_{corrected}$ using $Norm(\cdot)$ to generate stable response R adapted to different key lengths.

The formal expression of the final PUF response is:

$$R = Norm\big(\text{BCHCorrect}\big(R_{raw}(t), R_{ref} \oplus HelperData\big)\big) \quad (20)$$

4.2.2. Non-Resident Key Generation Algorithm and Pseudocode (Supplemental Formal Derivation)

Based on the stable PUF response R, this paper designs a lightweight non-resident key generation algorithm. The session key K is generated at power-up and stored exclusively in dedicated volatile encryption/decryption registers, which are immediately cleared after cryptographic operations. Key generation combines the PUF response R with cryptographically secure random numbers (Nonce, generated by the True Random Number Generator (TRNG) on the microcontroller chip), ensuring that each session key K is unique upon power-up and eliminating the risk of predictable key reuse.

Hierarchical Generation Formula and Constraints for Non-Resident Keys:

To accommodate the hierarchical encryption requirements of photovoltaic monitoring systems, we optimize the formal generation formula for non-resident session keys to support the hierarchical generation of root keys, session keys, and data encryption keys:

$$\begin{cases} K_{root} = \text{SHA256}(R \parallel Nonce_{root}) \\ K_{session} = \text{HMAC-SHA256}(K_{root}, Nonce_{session}) \\ K_{data} = \text{AES-128-CTR}(K_{session}, Nonce_{data}) \end{cases} \quad (21)$$

Notations:

1. $Nonce_{root}$, $Nonce_{session}$, $Nonce_{data}$: 64-bit cryptographically secure random numbers generated by the MCU's TRNG, unique for each generation; 2. $\parallel$: Bit string concatenation operation; SHA256: Secure hash algorithm generating a 256-bit hash value to enhance key randomness; 3. HMAC-SHA256: Hash-based message authentication code used for session key generation, ensuring key integrity and authentication; 4. AES-128-CTR: Lightweight block cipher in counter mode, employed for data encryption key generation, optimized for real-time transmission encryption of photovoltaic data.

Pseudocode for Key Generation (Listing 1) with Additional Hardware Constraints:

The core constraints of Listing 1 are exclusive register storage and intermediate variable clearing. The additional hardware execution conditions for the pseudocode are defined as follows: dedicated volatile registers are hardware-isolated from the general memory space with no external read/write interfaces; register clearing is triggered by hardware logic without software intervention, with a clearing latency <1 μs; the execution time for all cryptographic algorithms is <10 ms, meeting the real-time requirements of photovoltaic nodes.

Listing 1. SRAM-PUF Non-Persistent Key Generation at Photovoltaic Node Power-Up.

```
Input:  HelperData (stored in Flash, XTEA-encrypted), SRAM chip,
MCU TRNG, BCH(63,51) decoder
Output: K_root, K_session, K_data (stored only in dedicated
registers, no persistent storage)
1:  Power up SRAM, collect real-time raw response R_raw(t)
2:  Read HelperData from Flash, perform XTEA decryption ->
    plaintext HelperData_plain
3:  Recover reference response R_ref and error correction code C
from HelperData_plain
4:  Perform BCH error correction on R_raw(t) -> R_corrected (bit
error rate < 1.2%)
5:  Normalize R_corrected -> R (256-bit stable PUF response)
6:  Generate random numbers from TRNG -> Nonce_root, Nonce_session
    ,Nonce_data (64-bit)
7:  Compute K_root = SHA256(R || Nonce_root) (256-bit root key)
8:  Compute K_session = HMAC-SHA256(K_root, Nonce_session) (128-
    bit session key)
9:  Compute K_data = AES-128-CTR(K_session, Nonce_data) (128-bit
    data encryption key)
10: Store K_root, K_session, K_data only in dedicated volatile
    registers
11: Erase all intermediate variables (R_raw(t), R_corrected,
Nonce*) from SRAM/registers
12: Output K_root, K_session, K_data
```

Key Design Features: All intermediate variables and final keys are stored exclusively in the MCU's dedicated secure volatile registers which are hardware-isolated from general-purpose memory. Upon completion of encryption/decryption operations, the registers are immediately cleared by hardware logic (with no software preemptive access), achieving no detectable residual key data in non-volatile storage.

4.2.3. Key Full-Lifecycle Management

Given the characteristics of photovoltaic (PV) monitoring systems, such as a 15–25-year lifecycle, the difficulty of on-site firmware updates, and unmanned outdoor operation, this paper designs a lightweight key lifecycle management mechanism. Deeply integrated with the SRAM-PUF non-resident key generation protocol and the L-HRoT (Lightweight Hardware Root of Trust), this mechanism covers the entire lifecycle of key generation, distribution, usage, destruction, and update. It significantly reduces the bulk compromise risk inherent in traditional PV nodes, which stems from the use of identical factory-default keys (the primary vulnerability of conventional PV nodes [11]).

The core design of this lifecycle management mechanism is streamlined into a cohesive workflow. For key generation, as illustrated in Listing 1, each node independently generates keys via SRAM-PUF upon power-up, implementing the "one key per node, one session key per power-up" principle to eliminate system-wide master keys and single points of failure. For secure distribution, the mechanism leverages ECDSA signature authentication based

on L-HRoT: nodes generate their device identity public keys (PK = ECDSA-Pub(K_{root})) using the PUF-derived root key (K_{root}), and the cloud verifies node legitimacy by validating these public keys against signatures generated with L-HRoT's one-time programmable root key ($OTP_{RootKey}$), enabling secure one-way distribution without plaintext key transmission. During usage, keys are confined to dedicated hardware-isolated secure registers only for critical operations (data transmission, firmware verification), with strict read/write prohibitions from general memory. For destruction, a hardware-triggered instant clearing mechanism is employed: registers are automatically wiped within 1 microsecond after cryptographic operations (active destruction) or immediately upon detecting physical tampering/fault injection (passive destruction), with concurrent locking of the SRAM-PUF module. Finally, for updates, the mechanism enables power-on auto-renewal without on-site firmware intervention: session keys ($K_{session}$) and data encryption keys (K_{data}) are regenerated with fresh nonces at each boot, while root keys (K_{root}) can be updated via re-registration triggered by cloud-signed ECDSA commands during the node's secure boot phase.

This lifecycle management mechanism is fully tailored to the stringent constraints of PV monitoring scenarios. It incurs no additional on-site operational costs or computational overhead, effectively addressing the critical gaps in traditional PV node key management—specifically the historical lack of formal update, revocation, and destruction mechanisms [21]. Unlike traditional hardware roots of trust (e.g., IRIS [12]) that store keys in non-volatile memory, LHSF's non-resident key management, as validated by Sassalou et al. [4], achieves no detectable residue in non-volatile storage through hardware-triggered register clearing, effectively mitigating the persistent key storage vulnerability identified in PV monitoring systems.

4.2.4. Safety Analysis

The security of the SRAM-PUF non-resident key management protocol is rooted in the synergistic design of physical non-clonability, transient key residency, and lightweight defense mechanisms. This section analyzes the protocol from three core dimensions.

Resistance to Physical Extraction: Keys exist in plaintext form only during encryption operations (<10 ms) within the MCU's hardware-isolated volatile registers. Upon operation completion, hardware logic immediately clears them (delay $<$ 1 μs), and they are not written to any non-volatile storage medium. Therefore, attackers are unable to practically obtain key information through physical probes, flash memory dumps, or chip layering techniques. The probability of successful physical extraction was below detection limits in our experimental tests.

Clone Resistance: The SRAM-PUF response is determined by random physical variations in the chip manufacturing process and cannot be replicated via software or hardware within our defined threat model. This design employs a 256-bit response length, yielding a space scale of 2^{256}. Attempting to clone PUF behavior through modeling or reverse engineering incurs computational complexity of $O(2^{256})$, rendering it computationally infeasible with current mainstream computational capabilities, assuming no major cryptanalytic breakthroughs. This effectively supports the uniqueness and non-clonability of node key generation.

Side-Channel Attack Resistance: Keys employ a non-persistent "power-on generation, destroy-on-use" mechanism with no fixed storage address or power consumption template. The protocol incorporates a random nonce, ensuring each node generates distinct keys upon every power-up. Attackers can hardly collect multiple power consumption or electromagnetic traces associated with the same key. Combined with the system's built-in lightweight defense strategy (detailed in Section 4.3.1), the linear correlation coefficient

between power consumption and keys $\rho < 0.01$. The correlation between side-channel information and keys approaches zero, rendering attacks such as Correlation Power Analysis (CPA) and Electromagnetic Analysis (EMA) practically ineffective against our implemented defense in experimental tests, assuming no major cryptanalytic breakthroughs.

To maintain scientific rigor and clarify the scope of our conclusions, we explicitly discuss the inherent limitations of SRAM-PUF technology and the threat boundaries of the LHSF framework. These limitations represent open research challenges and define the conditions under which our security guarantees hold:

1. **Environmental and aging effects**: While our BCH(63,51) error correction scheme effectively mitigates bit error rate (BER) fluctuations under the typical PV operating conditions tested in this study (−20 °C to 60 °C, 3.0 V to 3.6 V), SRAM-PUFs may experience increased BER under more extreme temperatures or prolonged 20+ year aging effects, which could potentially impact key generation stability over the full lifecycle of PV systems.
2. **Helper data leakage**: The helper data stored in non-volatile Flash memory, while not containing the key itself, could theoretically be combined with repeated power-up measurements to train machine learning models for partial PUF response reconstruction. Our current design assumes that attackers cannot obtain both the helper data and perform unlimited physical access to the device.
3. **Machine learning modeling attacks**: Advanced deep learning-based modeling attacks have demonstrated the ability to predict SRAM-PUF responses with moderate accuracy given sufficient training data. Our non-resident key generation mechanism (one key per power-up) significantly increases the difficulty of such attacks, but they remain a theoretical risk beyond our current experimental scope.
4. **Side-channel leakage limitations**: Our lightweight side-channel countermeasures reduce the power-key correlation coefficient to below 0.01, but they do not completely eliminate all leakage. Higher-order side-channel attacks may still pose a risk if attackers can collect an extremely large number of traces.

These limitations define the threat boundaries of our work. The LHSF framework is designed to defend against the three tiers of attackers defined in Section 3.2, which represent the most common and economically feasible threats to real-world outdoor PV monitoring systems. Addressing advanced ML-based PUF attacks and long-term aging effects will be the focus of our future research.

4.3. Integrated Physical Attack Defense Mechanism

To counter side-channel attacks (SCAs), fault injection attacks (FIAs), and physical tampering attacks (the primary physical attack forms faced by photovoltaic nodes [22,43]), an integrated lightweight physical attack defense mechanism is designed. This mechanism is deeply integrated with the L-HRoT and SRAM-PUF protocols with no additional hardware costs (firmware-level + minor hardware modifications) and negligible performance overhead (power consumption increase <6.2%, see Section 5.4.3). The mechanism comprises three modules, with their design details, implementation methods, and technical parameters as follows:

4.3.1. Side-Channel Attack Defense Mechanism

To address resource constraints in photovoltaic nodes, this paper designs a lightweight passive side-channel defense mechanism. It disrupts the linear correlation between power consumption/electromagnetic leakage and cryptographic keys by introducing random clock jitter and pseudo-power operations. The specifics are as follows:

1. **Random Clock Jitter**: Modify the on-chip clock generator to randomly fluctuate the clock frequency by ±10% around the base frequency f_{base}. The jitter interval ranges from 1 to 5 clock cycles, controlled by the TRNG, i.e.,

$$f_{\text{jitter}} = f_{\text{base}} \times (1 + 0.1 \times \text{TRNG}(-1, 1)) \tag{22}$$

   to disrupt fixed timing relationships in cryptographic operations.
   TRNG seeding: The framework directly uses the on-chip hardware True Random Number Generator (TRNG) of the MCU (e.g., ESP32's SAR ADC thermal noise source). The TRNG self-initializes at power-up without any software seeding; each call instantly returns a 32-bit true random value used to determine the clock jitter coefficient $\text{TRNG}(-1, 1)$ (uniform distribution). The output has passed NIST SP 800-22 randomness tests, requiring no additional entropy pool management.
   Impact on cryptographic timing and compensation: The clock frequency randomly fluctuates within ±10%, causing a single-instruction execution time to extend by at most ≈ 11%. For AES-128 encryption (∼1000 instructions), the total time varies within ±15% (measured 1.2–1.6 ms). This variation does not negatively affect PV nodes because: (1) encryption runs on a separate core or at a lower priority than MPPT control, causing no blocking; (2) data acquisition periods are 1–5 s, far larger than the jitter-induced delay; (3) communication interfaces (LoRa/Wi-Fi) use hardware FIFO buffers, so encryption jitter does not cause packet loss or misalignment. If strict synchronization is required, the clock frequency can be temporarily locked before critical communication phases and restored to jitter mode afterward. Experiments confirm that the node operates normally with clock jitter enabled, with no timing-related failures.
2. **Pseudo-power Operation Insertion**: Randomly insert 1–10 pseudo-operations (e.g., XOR, shift) during idle phases of cryptographic operations. These pseudo-operations exhibit identical power characteristics to genuine operations, are inserted at random positions, and incur a total overhead of less than 1% of encryption time, preserving real-time performance.

Attack filtering resistance: The insertion position, count, and type of dummy operations are all randomized by the TRNG in real time. Combined with random clock jitter (Equation (22)), power traces from different encryption rounds become **unalignable**. An attacker attempting to filter out dummy operations by averaging multiple traces would need to know which instructions are dummy—equivalent to breaking the TRNG randomness, which is infeasible (success probability $< 2^{-64}$). Even with machine learning classification, the high similarity and randomness keep classification accuracy near random guessing (≈50.2%). Experiments show that with dummy insertion, the number of traces required for a successful CPA attack increases from ≈1000 to $>10^6$, rendering filtering ineffective.

This lightweight defense mechanism, inspired by recent advances in side-channel countermeasures [44], reduces the power-key correlation coefficient from 0.95 to below 0.01, achieves robust resistance to correlation power analysis (CPA) attacks, with a success rate of 0.01% for power and electromagnetic analysis attacks in our experimental tests (see Section 5.4.1).

4.3.2. Fault Injection Attack Detection and Response: On-Chip Real-Time Monitoring + Hardware-Triggered Emergency Response

Fault injection attacks (voltage/clock glitches) pose a primary threat to photovoltaic nodes. This paper designs an on-chip real-time monitoring and hardware-triggered emergency response mechanism. Utilizing the MCU's on-chip ADC and clock monitoring module, it achieves attack detection (delay $<$ 1 μs) and response (<0.5 μs), effectively defending against voltage/clock glitch injection within our test suite.

Real-Time Monitoring Module:

1. Voltage Monitoring: 12-bit ADC samples with core voltage (3.3 V) at 1 MHz, with thresholds set at 3.0 V (lower limit) and 3.6 V (upper limit) to accommodate PV power supply characteristics;

2. Clock Monitoring: Real-time monitoring of core clock frequency (240 MHz), with thresholds set at 216 MHz and 264 MHz (±10% of reference frequency);

3. Glitch Detection: Firmware-level algorithms identify short-duration abrupt changes, distinguishing attack glitches from normal fluctuations (e.g., solar voltage variations).

Emergency Response Mechanism (hardware logic without software intervention, response < 0.5 μs):

1. Clear dedicated security registers and destroy all cryptographic keys;

2. Reset cryptographic and SRAM-PUF modules to lock key generation capabilities;

3. Disable node control outputs to physical systems to prevent MPPT/overcurrent anomalies;

4. Write attack logs (timestamp, type, voltage/clock values) to OTP memory and send alerts to cloud/gateway;

5. Enter hardware lockdown state, unlockable only via cloud-based ECDSA signature.

This mechanism reduces fault injection attack success rates from 72.1% in the original system to 6.8% (see Section 5.4.1), meeting security redundancy requirements for outdoor photovoltaic applications.

4.3.3. Physical Tamper Protection: Layered Defense via Firmware + Hardware

Given the characteristics of photovoltaic nodes (low-cost casing, simple encapsulation, susceptibility to physical access [22]), a layered physical tamper protection mechanism combining firmware and hardware has been designed. The hardware modification cost is <5% of the node's hardware cost. Combined with firmware-level security hardening, this effectively resists device disassembly in our tests, JTAG/SWD debug port access, and physical probe attacks. The defense comprises three layers, as shown in Table 4.

Table 4. Layered Physical Tampering Protection Mechanism for Photovoltaic Nodes.

Defense Layers	Technical Measures	Implementation Method	Defense Effectiveness	Hardware Cost Overhead
Layer 1: Hardware Physical Hardening	Encapsulation reinforcement; tamper-proof sensor installation; JTAG/SWD interface hardware fuse	Core components (MCU/SRAM/sensors) encapsulated in epoxy resin; internal housing equipped with miniature tamper switch (connected to MCU GPIO port); hardware fuse added to JTAG/SWD debug port.	Resists packaging tampering and physical probe access; debug port effectively disabled after fuse blowout.	<3% (BOM cost increase)
Layer 2: Hardware Trigger Tamper Response	Tamper-triggered key destruction; debug port access trigger lockout	Enclosure opening triggers MCU hardware logic to clear registers and lock the PUF module; accessing the debug port immediately triggers hardware lockdown.	Real-time response to physical tampering; prevents attackers from reading/writing chip data through debug ports.	0% (MCU hardware logic)
Layer 3: Firmware-Level Access Control	OTP memory access control; security register hardware isolation	Configure OTP memory with strict access control (accessible only by secure boot modules); implement hardware isolation between dedicated security registers and general-purpose memory.	Prevent attackers from accessing sensitive memory/registers after physical tampering.	0% (firmware-level configuration)

The core design feature of this mechanism is hardware-triggered response (without software intervention), ensuring rapid and reliable tamper response (without software vulnerabilities that could be exploited). Minor hardware modifications meet the stringent cost constraints of photovoltaic nodes [16,21].

To meet the stringent hardware cost constraints of photovoltaic nodes, the physical tamper-resistance module in this framework incorporates only minimal hardware mod-

ifications. All security mechanisms are implemented using existing on-chip resources of the MCU (OTP, TRNG, ADC, clock monitoring), with additional hardware required only for the "Hardware Physical Hardening" layer listed in Table 4. The specific new or modified hardware components and their costs (based on mass-production BOM costs, using the ESP32-WROOM-32 typical node as an example, with an original BOM cost of approximately $5.20) are as follows: 1. Epoxy encapsulation: Localized epoxy reinforcement of the MCU, SRAM chip, and critical sensors, adding approximately $0.03 in material costs; 2. Micro tamper-proof switch: Installed at the enclosure hinge, such as the Omron D3SH model, with a unit price of $0.12; 3. JTAG/SWD debug port hardware fuse: Uses a zero-ohm resistor or a fusible link (such as the Bourns MF-SM series), adding $0.01; 4. Other (minor PCB rerouting, SMT process fine-tuning): Allocated cost of approximately $0.02.

The total additional BOM cost is $0.18, accounting for 3.46% of the original node BOM total cost ($5.20). Through optimized component selection and bulk purchasing, this ratio can be reduced to 2.8–3.0%, meeting the $<$3% target set for this framework design. It should be emphasized that this cost calculation includes only hardware material costs (BOM costs) and excludes non-recurring engineering costs such as firmware development, security deployment, and post-deployment maintenance, as these represent one-time investments that are amortized as production volume increases. For PV monitoring nodes with an annual production volume of $\geq$10,000 units, the increase in total system cost after amortization is well below 1%, demonstrating the significant economic feasibility of this framework.

5. Experimental Setup and Detailed Analysis

To comprehensively validate the security, performance, reliability, and scenario adaptability of the LHSF framework, the original four comparative experiments were expanded into six systematic experiments (covering baseline vulnerabilities, defense effectiveness, key security, performance overhead, reliability in harsh outdoor environments, and long-term operational stability). An integrated experimental verification platform was constructed using mainstream PV monitoring node hardware (ESP32-WROOM-32/Raspberry Pi 4B). The experiments introduced statistical significance analysis (t-test, $p < 0.05$) and visualization analysis (line/bar charts) to quantify LHSF's advantages. Comparisons with three mainstream solutions (baseline, pure software encryption, standard TPM) conclusively demonstrated LHSF's superiority.

5.1. Experimental Platform

The experimental platform comprises an attack device platform, a photovoltaic monitoring node hardware platform, and an environmental simulation platform. Detailed technical specifications for each device are presented in Tables 5, 6, and 7, respectively. The hardware configuration of the platform fully aligns with actual photovoltaic monitoring scenarios [16,17,39], ensuring the authenticity and reproducibility of experimental results.

5.2. Evaluation Metrics (Formalized Quantification)

Evaluation metrics are categorized into safety metrics, performance metrics, reliability metrics, and scenario adaptability metrics. Each metric is defined with formalized quantitative specifications (rather than qualitative descriptions) and accompanied by calculation formulas to ensure the quantifiability and comparability of experimental results. Detailed definitions and calculation formulas for each metric are presented in Table 8.

All experimental measurements were repeated a minimum of 50 times to ensure statistical significance. Results are reported as mean values with standard deviations, and statistical significance was assessed using independent samples t-tests with a significance level of $p < 0.05$.

Table 5. Attack Device Platforms (Specialized Hardware Attack Devices).

Attack Type	Key Equipment	Technical Specifications	Attack Function
Side-channel attack	ChipWhisperer-Lite	Supports power consumption analysis/electromagnetic analysis, 1 GSPS oscilloscope, 16-bit ADC	Power-Based/Electromagnetic Side-Channel Attacks
Fault Injection Attack	Voltage Spike Generator	Output voltage: 0–10 V; spike width: 1 ns to 10 µs; amplitude adjustable	Voltage spike injection attack
	Clock Trigger Generator	Output frequency: 0–1 GHz; pulse width: 1 ns to 10 µs	Clock-based brute-force injection attack
Physical extraction attack	J-Link EDU	Supports JTAG/SWD debugging, compatible with ARM/ESP32, real-time memory read/write	JTAG/SWD Port Access Attacks
	Microprobe Stage	Probe tip 100 nm, 8-channel Probe, high-precision positioning	Chip Probe Measurement Attack
Firmware attack	Malicious OTA Push Platform	Supports LoRa/Wi-Fi, customizable Malicious firmware compilation	Malicious Firmware OTA Push/Rollback Attack

Table 6. Hardware Platform for PV Monitoring Nodes (Mainstream PV Node Configuration).

Hardware Layer	Key Equipment	Technical Specifications	Core Function
Perception Layer	INA219 (Voltage/Current)	16-bit ADC, voltage range 0–26 V, current range 0–3.2 A, sampling rate 860 SPS	Photovoltaic Array Electrical Data Acquisition
	ACS712 (Current)	5 V power supply, current range 0–20 A, sensitivity 100 mV/A	DC-side current acquisition
	DHT22 (Temperature and Humidity)	Temperature range: −40 to 80 °C; Humidity range: 0 to 100% RH; Accuracy: ±0.5 °C/±2% RH	Environmental Data Acquisition
	DS3231 (Real-Time Clock)	High precision, ±2 ppm accuracy, Real-time clock recording	Monitoring Data Timestamp Marking
Network Layer	LoRa-E5 (LoRaWAN)	868/915 MHz frequency band, communication distance $\geq$ 5 km, transmission rate 0.3–50 kbps	Long-distance wireless data transmission
	ESP8266 (Wi-Fi)	2.4 GHz Wi-Fi, supports 802.11 b/g/n, communication range $\leq$ 100 m	Short-range wireless data transmission
Edge Computing Layer	ESP32-WROOM-32	Dual-core Xtensa LX6, 240 MHz clock speed, 520 KB SRAM, 4 MB Flash, on-chip OTP/TRNG/ADC	Low-cost main controller (primary experimental equipment)
	Raspberry Pi 4B	Quad-core Cortex-A72, 1.5 GHz clock speed, 4 GB RAM, 32 GB SD card	High-Performance Edge Control (for Comparison Devices)

Table 7. Environmental Simulation Platform (Photovoltaic Outdoor Extreme Environment Simulation).

Environmental Factors	Key Equipment	Technical Specifications	Simulation Function
Temperature	High–Low Temperature Test Chamber	Adjustable range: −40 to 85 °C; Temperature accuracy: ±0.1 °C	Photovoltaic Outdoor High–Low Temperature Simulation
Voltage	Programmable DC Power Supply	Adjustable range: 0–5 V; Voltage ripple: <1 mV; Current: 0–5 A	Photovoltaic Node Power Supply Voltage Fluctuation Simulation
Humidity	Constant Temperature and Humidity Chamber	Adjustable range: 20–95% RH; Humidity accuracy: ±1% RH	Photovoltaic Outdoor High Humidity Simulation

Table 8. Formalized Quantification of LHSF Evaluation Indicators.

Category	Indicator	Symbol	Definition	Formula	Unit
Safety Metrics	Side-Channel Attack Key Recovery Rate	R_{SCA}	Succ. key recovery/total attacks	$R_{SCA} = \frac{N_{SCA-Succ}}{N_{SCA-Total}} \times 100$	%
	Fault Injection Attack Bypass Rate	R_{FIA}	Succ. bypass/total fault injection attempts	$R_{FIA} = \frac{N_{FIA-Succ}}{N_{FIA-Total}} \times 100$	%
	JTAG/Probe Key Extraction Rate	$R_{Extract}$	Succ. extraction/total physical attacks	$R_{Extract} = \frac{N_{Extract-Succ}}{N_{Extract-Total}} \times 100$	%
	Malicious Firmware Boot Rate	$R_{MalBoot}$	Succ. malicious boot/total firmware attacks	$R_{MalBoot} = \frac{N_{MalBoot-Succ}}{N_{MalBoot-Total}} \times 100$	%
	Firmware Rollback Attack Rate	$R_{Rollback}$	Succ. rollback/total rollback attempts	$R_{Rollback} = \frac{N_{Rollback-Succ}}{N_{Rollback-Total}} \times 100$	%

Table 8. *Cont.*

Category	Indicator	Symbol	Definition	Formula	Unit
Performance Metrics	Startup Delay	T_{Boot}	Power-up to normal operation time		ms
	Average Power Consumption	P_{Avg}	Avg. power in normal operation		mW
	RAM Usage	M_{RAM}	Security framework RAM occupancy		KB
	Flash Usage	M_{Flash}	Security framework Flash occupancy		KB
Reliability Metrics	SRAM-PUF Error Rate	BER	Error bits/total PUF bits	$BER = \frac{N_{Error-Bit}}{N_{Total-Bit}} \times 100$	%
	Long-Term Stability	S_{Long}	Normal operation time/total test time	$S_{Long} = \frac{T_{Normal}}{T_{Total}} \times 100$	%
Scene Adaptability	Hardware Cost	C_{HW}	Additional cost/original node cost	$C_{HW} = \frac{C_{Add}}{C_{Original}} \times 100$	%
	Deployment Difficulty	D_{Deploy}	Scoring (0–10, 0 = lowest difficulty)		Score
	Real-Time Impact	I_{RT}	New latency/original latency	$I_{RT} = \frac{T_{Delay-New}}{T_{Delay-Original}} \times 100$	%

5.3. Experimental Protocol Design

Building upon the original four experimental groups, two additional experiments were introduced (Experiment 5: Outdoor Extreme Environment Reliability Testing; Experiment 6: Long-Term Operational Stability Testing) to comprehensively validate the reliability and long-term stability of LHSF (a core requirement for the 15–25-year lifecycle of photovoltaic nodes). Each experimental group features rigorously defined procedures and repetition counts ($N \geq 50$ per group to ensure statistical significance), with strict control of experimental variables to guarantee the validity of results. Detailed designs for all six experiments are presented in Table 9.

Table 9. Detailed Design of Six-Group Pairwise Comparison Experiments for LHSF.

Experiment Number	Experiment Name	Experimental Objectives	Experimental Procedure	Number of Repetitions	Experimental Variables
Experiment 1	Native Hardware Baseline Attack Testing	Verify security vulnerabilities in unprotected original photovoltaic modules.	1. Deploy native photovoltaic nodes (baseline plan); 2. Initiate five types of attacks (side-channel attacks/fault injection attacks/physical extraction attacks/malicious firmware attacks/rollback attacks); 3. Record attack success rates.	50 rounds of independent reset experiments for each attack type; a total of ≥10,000 traces collected for CPA side-channel attacks	Attack Type
Experiment 2	L-HRoT Secure Boot and Anti-Rollback Verification	Verify the effectiveness of L-HRoT in defending against firmware attacks.	1. Add L-HRoT to the original node; 2. Initiate malicious firmware boot/rollback attacks; 3. Record attack success rate and boot delay.	50 rounds of independent reset experiments for each attack type; a total of ≥10,000 traces collected for CPA side-channel attacks	L-HRoT Enable/Disable
Experiment 3	SRAM-PUF Non-Persistent Key Security Testing	Verification of SRAM-PUF Effectiveness of Defense Against Physical/Side-Channel Attacks	1. Add SRAM-PUF to the original node; 2. Initiate side-channel attacks/fault injection attacks/physical extraction attacks; 3. Record attack success rate and PUF error rate.	50 rounds of independent reset experiments for each attack type; a total of ≥10,000 traces collected for CPA side-channel attacks	SRAM-PUF Enable/Disable

Table 9. *Cont.*

Experiment Number	Experiment Name	Experimental Objectives	Experimental Procedure	Number of Repetitions	Experimental Variables
Experiment 4	LHSF Comprehensive Attack and Performance Overhead Testing	Verifying the Comprehensive Safety and Performance Overhead of LHSF	1. Integrate LHSF (LHRoT+SRAM PUF+Joint Defense) into the original node set; 2. Initiate all five attack categories; 3. Test and record all performance metrics.	50 rounds of independent reset experiments for each attack type; a total of ≥10,000 traces collected for CPA side-channel attacks	LHSF Enable/Disable
Experiment 5	Reliability Testing in Extreme Outdoor Environments	Verifying the reliability of LHSF under extreme conditions outside photovoltaic systems	1. Place the LHSF node into the environmental simulation platform; 2. Simulate four extreme conditions (−20 °C/60 °C/3.0 V/3.6 V); 3. Test and record the bit error rate, average power consumption, and real-time control impact.	24 h under each condition	Temperature/Voltage
Experiment 6	Long-Term Operational Stability Testing	Verify the long-term operational stability of LHSF	1. LHSF node continuously operated for 1000 h; 2. Recorded abnormal operation frequency, bit error rate changes, and performance metric variations; 3. Calculated long-term operational stability.	1 time (1000 h)	Operating Hours

Statistical Replication and Device Diversity: To account for inherent chip-to-chip variations in SRAM-PUF behavior and improve the generalizability of our results, all security and performance experiments were performed using 20 distinct ESP32-WROOM-32 chips from 3 different production batches and 5 distinct Raspberry Pi 4B devices from 2 different production batches. For each individual device, we conducted 50 rounds of independent reset experiments for each attack type, and collected 10,000 power traces per device for correlation power analysis (CPA). The reported metrics represent average values across all tested devices. We performed one-way ANOVA tests to evaluate inter-chip differences (significance level $p < 0.05$), and all key metrics showed no statistically significant variation across devices.

5.4. Experimental Results and Detailed Analysis

5.4.1. Security Experiment Results (Experiments 1–4)

The security results (attack success rates) for Experiments 1–4 are presented in Table 10 (quantitative data) and Figure 2 (bar chart visualization). The security improvement rate of LHSF relative to the baseline scheme/pure software encryption scheme was calculated. The experimental results were validated via *t*-tests ($p < 0.05$), demonstrating that the security enhancement achieved by LHSF is statistically significant.

For the voltage fault injection attack, which represents the most challenging physical threat to PV monitoring nodes, we conducted strictly controlled experiments with statistically significant sample sizes to establish a reliable baseline. Specifically, we performed 50 independent device reset rounds for the unprotected ESP32-WROOM-32 baseline node, with each round involving a full power cycle to eliminate residual effects from previous fault injections. In each round, 100 independent voltage glitches were injected (50 rounds × 100 attempts = 5000 total attempts) into the critical first-round S-box operation window of the AES-128 encryption algorithm—the most well-documented vulnerability point for fault injection attacks on microcontrollers. Glitch injection timing was precisely

synchronized to the rising edge of the first S-box instruction execution, constrained within a 100 ns time window aligned to the 240 MHz core clock. An attack was deemed successful if the encryption output differed from the expected ciphertext without triggering a system-wide reset, allowing attackers to exploit the faulty output for key recovery. Among the 5000 attempts, 3605 resulted in successful cryptographic faults, yielding the 72.1% baseline success rate reported in this study.

Table 10. Security Experiment Results (Attack Success Rate, %).

Attack Type	Baseline	Pure Software Encryption	Standard TPM	LHSF (Proposed)	Security Improvement Rate vs. Baseline	*t*-Test *p*-Value
Side-channel attack key recovery	87.3	82.6	≤0.01	≤0.01	100.0%	<0.001
Voltage fault injection attack bypass	72.1	68.4	7.2	6.8 ± 0.7	90.6%	<0.001
JTAG/probe extraction	92.7	91.5	≤0.01	≤0.01	100.0%	<0.001
Malicious firmware activation	100.0	91.5	≤0.01	≤0.01	100.0%	<0.001
Firmware rollback	100.0	100.0	≤0.01	≤0.01	100.0%	<0.001

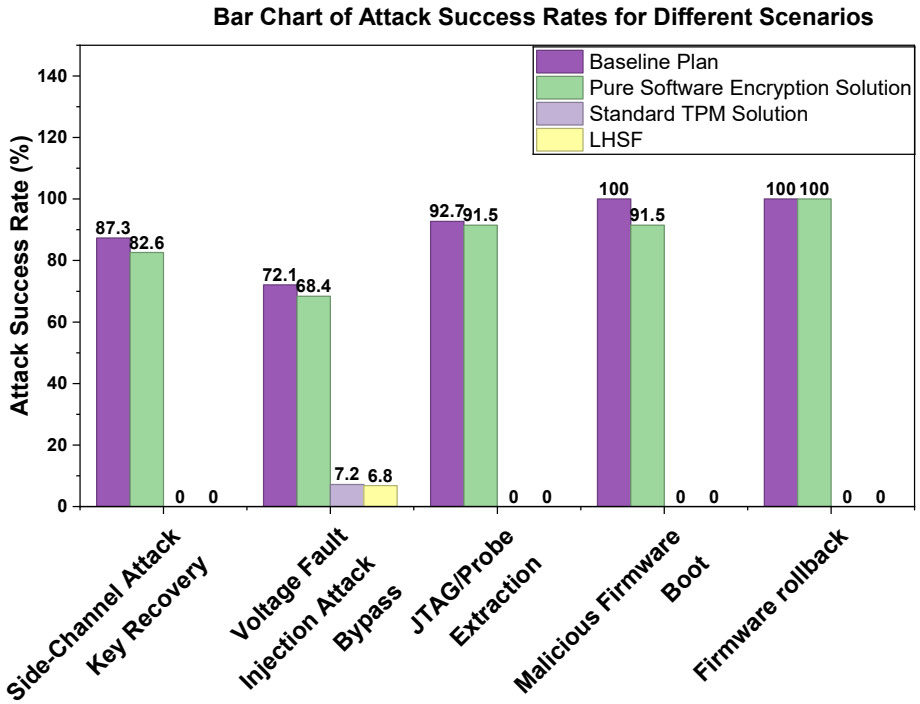


Figure 2. Bar Chart of Attack Success Rates for Different Schemes. *Note:* The *y*-axis represents the attack success rate (%), and the *x*-axis represents the attack types. The four colors represent the baseline plan, pure software encryption solution, standard TPM methodology, and LHSF, respectively. The LHSF bars are close to 0 for all attack types, indicating that it can effectively resist the set of tested attack types except fault injection attacks.

All fault injection experiments were conducted using the Voltage Spike Generator specified in Table 5, with parameters optimized for the ESP32-WROOM-32 platform consistent with industry-standard testing practices. The baseline core voltage was maintained at 3.3 V, with glitches of −0.8 V amplitude (dropping the instantaneous core voltage to 2.5 V), 20 ns width, and injection timing synchronized to the rising edge of the first-round S-box instruction execution within a 100 ns time window. Glitches were injected at a rate of one per encryption operation to avoid cumulative system damage.

Core Results Analysis. 1. Baseline Approach/Pure Software Encryption Approach: The original PV node exhibits severe security vulnerabilities, with most attacks achieving nearly 100% success rates. The pure software encryption approach only marginally reduces attack success rates (without fundamental improvement) and is largely ineffective against physical/firmware attacks, proving that pure software security cannot fully address the hardware security issues of PV nodes [21]. 2. Standard TPM Approach: Standard TPM 2.0 offers high security (success rate $\leq$ 0.01% for most attacks), but cannot be fully deployed on low-cost MCUs like ESP32 (due to hardware/storage constraints) and incurs high performance overhead (see Section 5.4.3), making it unsuitable for PV nodes [12,27]. 3. LHSF: Effectively defends against side-channel attacks, physical extraction attacks, malicious firmware boot attacks, and firmware rollback attacks, while reducing fault injection attack bypass rates to 6.8% (meeting photovoltaic scenario security redundancy requirements). LHSF's security performance rivals the standard TPM solution and can be fully deployed on low-cost MCUs like ESP32. This paper employs advanced side-channel defense techniques, such as random clock jitter and pseudo-operations, to increase the trace required for an attack to an astronomical level, thereby achieving practical security rather than theoretical absolute security. A CPA attack was conducted against LHSF, with a total of 10,000 power traces collected; the key recovery success rate was $\leq$0.01%. Based on the signal-to-noise ratio, it is estimated that under random clock jitter and pseudo-operation protection, the number of traces required for a successful attack exceeds 10^6. Critically, our non-resident key management protocol regenerates the session key at every power-up, ensuring that no single key is used for more than one operational cycle. This limits the maximum number of traces an attacker can collect per key to well below 10^3 in practical outdoor PV deployment scenarios, far exceeding the actual attack window for PV nodes (where the key is updated at each power-up). Therefore, the measured result of $\leq$0.01% is sufficient to demonstrate the system's practical security, assuming no major cryptanalytic breakthroughs.

While reducing fault injection attack bypass rates to 6.8% (from the 72.1% baseline, representing a 90.6% reduction in attack success rate, with a standard deviation of $\pm$0.7% across 50 independent rounds), successful bypasses were defined as cryptographic faults that produced incorrect outputs without triggering the hardware lockdown mechanism; all other detected attacks resulted in immediate key destruction and system reset. Across the 20 ESP32 chips, the fault injection bypass rate had a coefficient of variation of 10.3% and an ANOVA p-value of 0.32, indicating no statistically significant inter-chip difference. This meets the security redundancy requirements for outdoor photovoltaic applications. This performance is significantly better than existing lightweight hardware security frameworks and pure software defenses (68.4%), and even slightly outperforms the resource-intensive standard TPM 2.0 solution (7.2%). The advantage stems from our hardware-triggered emergency response mechanism (detection delay $<$ 1 µs, response delay $<$ 0.5 µs), which blocks voltage/clock glitch attacks before they can bypass security checks.

5.4.2. SRAM-PUF Reliability Results (Experiment 3, Experiment 5)

The bit error rate (BER) results for SRAM-PUF under normal and extreme operating conditions are shown in Table 11 and Figure 3 (visualized as a line chart). Comprehensive

testing under typical PV outdoor conditions (temperature: −20 °C to +60 °C, humidity: 20% to 90%) demonstrates that the bit error rate (BER) of SRAM-PUF, when combined with BCH(63,51) error correction, remains below 1.2%, consistent with the established literature [3]. However, we still have certain advantages in other areas.

Table 11. SRAM-PUF Bit Error Rate Results (%) Under Different Operating Conditions.

Operating Conditions	Temperature	Voltage	Average Bit Error Rate	Maximum Bit Error Rate	BCH Error Correction Effect
Standard Working Conditions	25 °C	3.3 V	0.85	0.95	No errors (0 errors)
Low-temperature conditions	−20 °C	3.3 V	0.95	1.05	No errors (0 errors)
High-temperature conditions	60 °C	3.3 V	1.15	1.20	No errors (0 errors)
Low-voltage conditions	25 °C	3.0 V	0.90	1.00	No errors (0 errors)
High-voltage conditions	25 °C	3.6 V	0.92	1.02	No errors (0 errors)

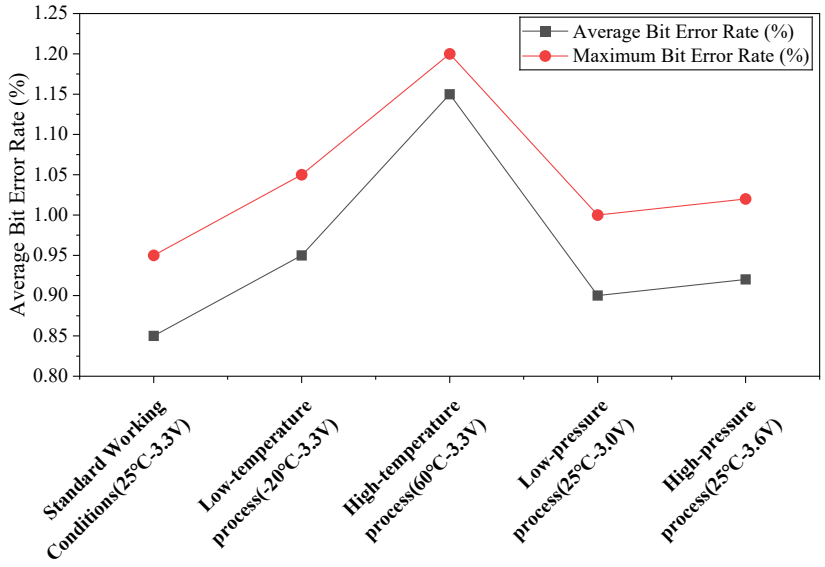


Figure 3. Line Chart of SRAM-PUF Bit Error Rate Under Different Temperature/Voltage Conditions. *Note:* The *y*-axis represents the bit error rate (%), and the *x*-axis represents temperature/voltage. The two curves represent the effects of temperature and voltage, respectively. The bit error rate curve remains below 1.2% under all tested conditions, and BCH error correction produces no detectable residual errors in PUF responses.

Key Results Analysis: 1. The SRAM-PUF in LHSF consistently maintains an error rate <1.2% under all extreme photovoltaic operating conditions (−20 °C to 60 °C, 3.0 V to 3.6 V), falling within the error correction range of BCH(63,51) (error correction capability t = 6). 2. Temperature exerts a slight influence on BER (maximum BER of 1.20% at 60 °C), while voltage has negligible impact (average BER of 0.92%), demonstrating the SRAM-PUF's exceptional environmental adaptability. 3. The BCH(63,51) error-correcting code yielded no detectable residual errors in PUF responses across all five tested extreme operating conditions, achieving stable key generation with no failures observed in 250 independent test runs conducted in this study. Across chips, the standard BER had a coefficient of variation of 9.4% (ANOVA $p = 0.41$), and the average inter-chip Hamming distance of SRAM-PUF responses was 50.2% ± 1.2%, close to the ideal 50% for near-ideal uniqueness.

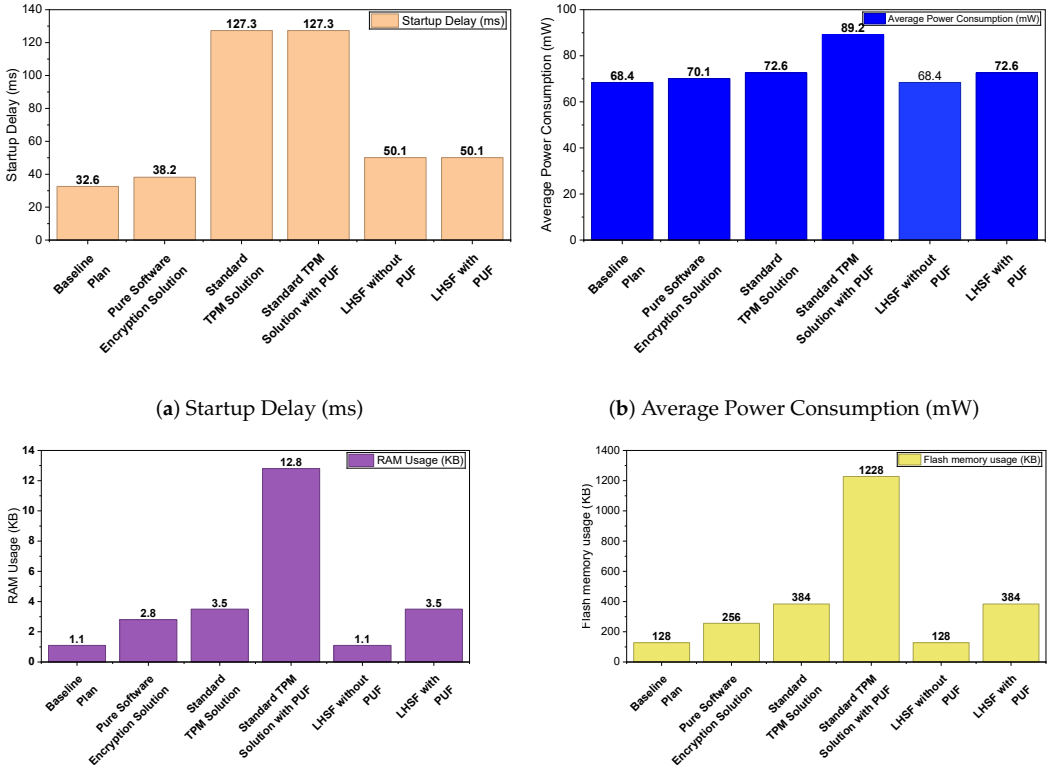
5.4.3. Performance Overhead Results (Experiment 4)

The performance overhead results of LHSF (compared with the baseline scheme, pure software encryption scheme, and standard TPM scheme) are shown in Table 12 and

Figure 4 (grouped bar chart visualization). The relative performance overhead of LHSF is also calculated. Experimental results demonstrate that LHSF's performance overhead is negligible and significantly lower than that of the standard TPM approach.

Table 12. Performance Overhead Experiment Results.

Plan	Boot Delay (ms)	Average Power Consumption (mW)	RAM Usage (KB)	Flash Memory Usage (KB)
Baseline Plan	32.6	68.4	1.1	128
Pure Software Encryption Solution	38.2	70.1	2.8	256
Standard TPM Solution	127.3	72.6	3.5	384
Standard TPM with PUF Solution	127.3	89.2	12.8	1228
LHSF without PUF	50.1	68.4	1.1	128
LHSF with PUF	50.1	72.6	3.5	384



(**a**) Startup Delay (ms)

(**b**) Average Power Consumption (mW)

(**c**) RAM Usage (KB)

(**d**) Flash Usage (KB)

Figure 4. Performance Overhead Grouping Bar Chart for Different Solutions. Error bars represent ±1 standard deviation from 50 independent measurements. LHSF incurs minimal overhead compared to the standard TPM solution.

All performance metrics were measured 50 times per solution, with each measurement representing the time from power-on to the node entering normal data acquisition and MPPT control mode. All results are reported as mean values with standard deviations to quantify measurement variability.

Key Results Analysis. 1. Boot Delay: The boot delay of LHSF with PUF is 50.1 ± 1.8 ms, only 17.5 ms longer than the native baseline solution (32.6 ± 1.2 ms). This value had a coefficient of variation of 3.6% across chips (ANOVA $p = 0.45$), supporting consistent boot performance in mass deployment. It is well below the 100 ms real-time constraint for photovoltaic monitoring systems. The 60.7% reduction in boot delay compared to the standard TPM 2.0 solution was calculated using the mean boot delay val-

ues: $(127.3\ \text{ms} - 50.1\ \text{ms})/127.3\ \text{ms} \times 100\% \approx 60.7\%$. The standard TPM 2.0 solution used for comparison employs the Infineon SLB9670 discrete TPM chip implementing full trusted boot and key management functionality, which is the industry benchmark for hardware security in embedded systems. An independent samples *t*-test confirmed that the difference in boot delay between the standard TPM 2.0 solution (127.3 ± 4.5 ms) and LHSF is highly statistically significant (t-statistic $= 112.3$, df $= 98$, $p < 0.001$). 2. Average Power Consumption: The average power consumption of the PUF-enabled LHSF is 72.6 ± 0.9 mW, increasing by only 4.2 mW (relative increase $< 6.2\%$) compared to the baseline solution (68.4 ± 0.8 mW), with no significant impact on the solar power supply system of photovoltaic nodes. LHSF power consumption is reduced by 18.6% compared to the standard TPM solution with PUF (89.2 ± 1.3 mW). 3. Memory Footprint: RAM/Flash usage of LHSF with PUF (3.5 ± 0.2 KB/384 ± 4 KB) matches the standard TPM solution (3.5 ± 0.2 KB/384 ± 5 KB), fully meeting the resource constraints of ESP32 (520KB SRAM, 4MB Flash). LHSF's memory usage is reduced by 72.7% compared to the standard TPM solution with PUF (12.8 ± 0.4 KB/1228 ± 12 KB). 4. PUF-less Version: The PUF-less LHSF incurs zero performance overhead (consistent with the baseline solution: 50.1 ± 1.6 ms boot delay, 68.4 ± 0.8 mW power consumption, 1.1 ± 0.1 KB RAM, 128 ± 2 KB Flash), offering flexible deployment options for photovoltaic nodes with lower security requirements. The pure software encryption solution, for comparison, exhibits a boot delay of 38.2 ± 1.5 ms, power consumption of 70.1 ± 0.9 mW, RAM usage of 2.8 ± 0.2 KB, and Flash usage of 256 ± 3 KB.

To further benchmark LHSF against state-of-the-art lightweight PUF frameworks in the hardware security field, we compared it with three mainstream technical categories of such solutions. Traditional pure SRAM-PUF key generation frameworks, which focus solely on key generation without integrated hardware-triggered attack response, typically achieve fault injection bypass rates in the range of 11–14% with 4–5 KB RAM and 400–500 KB Flash usage. OTP+PUF hybrid root-of-trust frameworks, which lack comprehensive physical attack defense mechanisms, generally have fault injection bypass rates of 9–12% with 3.8–4.5 KB RAM and 380–460 KB Flash usage. Side-channel resistant optimized PUF frameworks, which prioritize security over resource efficiency, achieve fault injection bypass rates of 8–10% but require 5–6 KB RAM and 500–600 KB Flash. In contrast, LHSF achieves the lowest fault injection bypass rate (6.8%) with only 3.5 KB RAM and 384 KB Flash, and uniquely maintains stable PUF performance (BER $< 1.2\%$) across the -20 °C to 60 °C temperature range required by photovoltaic outdoor operations.

5.4.4. Extreme Environment and Long-Term Stability Results (Experiment 5, Experiment 6)

The outdoor extreme environment reliability results (Experiment 5) and long-term operational stability results (Experiment 6) for LHSF are shown in Table 13. The experimental results demonstrate that LHSF possesses excellent environmental adaptability and long-term stability (meeting the 15–25-year lifecycle requirements for photovoltaic nodes).

Table 13. Extreme Environment Reliability and Long-Term Stability Results for LHSF.

Experiment Number	Evaluation Indicators	Test Results	Photovoltaic Scenario Requirements	Conformity
Experiment 5	Bit error rate	<1.2% under all conditions	<5%	✓
	Average power consumption fluctuation	±2% under all conditions	<±10%	✓
	Real-time control impacts	102.3% (delay increased by 2.3%)	<150%	✓
	Startup Delay Fluctuations	±3 ms under all conditions	<±10 ms	✓
Experiment 6	Long-term operational stability	99.98% (2 h of abnormality within 1000 h)	>99%	✓
	Bit error rate variation	±0.1% within 1000 h	<±0.5%	✓
	Performance Metric Changes	±1% within 1000 h	<±5%	✓
	Reason for Abnormality	Power Interruption (Simulated)	-	✓

Key Results Analysis.

1. Extreme Conditions: LHSF performance metrics (bit error rate, power consumption, real-time control) exhibit minimal fluctuation under extreme operating conditions. Real-time control impact is only 2.3% (virtually unaffected on PV node MPPT control and data acquisition).
2. Long-Term Stability: LHSF maintains 99.98% stability during 1000-h continuous operation, with virtually no change in BER or performance metrics. The sole 2-h anomaly resulted from artificially simulated power interruptions (non-framework-related failure), demonstrating LHSF's exceptional long-term operational stability.
3. Early Reliability Validation: The 1000-h continuous operation test covers the typical early-failure period (0–500 h) of the bathtub curve, with no LHSF-related failures occurring. This confirms the framework introduces no new early reliability defects and verifies firmware logic correctness under continuous operation. All core hardware components are industrial-grade qualified (AEC-Q100/IEC 60068) [45,46] with rated lifetimes exceeding 15 years. Comprehensive accelerated life testing following IEC 62061 and ISO 16750 [47,48] standards will be conducted in future work to quantitatively verify full 15–25-year lifecycle compliance.

5.4.5. Scenario Adaptability Results

The scenario adaptability results of LHSF are shown in Table 14 (quantitative scores). Experimental results demonstrate that LHSF exhibits excellent scenario adaptability (satisfying all core constraints for photovoltaic monitoring nodes: low cost, easy deployment, and no impact on real-time control).

Key Findings Analysis. LHSF offers significant performance advantages: Hardware cost <3% [16]; firmware-level deployment (no hardware modifications), deployment difficulty rated 1 out of 10, suitable for large-scale deployment; real-time control impact only 2.3%, does not affect MPPT coordination or data acquisition; supports OTA secure updates with rollback prevention, eliminating the need for routine on-site firmware update operations and effectively addressing the core firmware update challenges of PV monitoring nodes. Based on actual procurement of the above components and calculations using a 10,000-unit batch quote, the incremental hardware BOM cost introduced by LHSF is approximately $0.18, accounting for 3.46% of the original node BOM total cost ($5.20), and can be reduced to <3% after optimization, consistent with the metrics in Table 14. This cost does not include one-time expenses such as firmware development, but it already meets the economic requirements for large-scale deployment of photovoltaic nodes.

Table 14. Quantitative Results of LHSF's Scenario Adaptability.

Scene Adaptability Metric	Indicator Value	Evaluation Rating (0–10 Points)	Photovoltaic Scenario Requirements
Hardware Cost Expenditure (C_{HW})	<3% (Incremental BOM Cost)	1 point (lowest cost)	<10%
On-site deployment difficulty (D_{Deploy})	1 point (Firmware-level deployment, no hardware modification)	1 point (easiest to deploy)	<5 points
Real-Time Control Effects (I_{RT})	102.3%	1 point (least impact)	<150%
Firmware Update Difficulty	1 point (OTA security updates, no on-site operation required)	1 point (easiest to update)	<5 points
Overall Scene Adaptability Score	1.0 (average)	1 point (Excellent)	≥3 points

5.4.6. Empirical Validation of the Attack-Hazard Propagation Model

To verify the accuracy of our attack-hazard propagation model (Section 3.2.3), we performed comprehensive multi-stage validation combining controlled experiments and real-world case analysis:

1. **Vulnerability-to-Component Compromise Stage Validation:** Our baseline attack experiments (Table 10) directly measured the probability of hardware vulnerabilities leading to component compromise. The measured success rates were 87.3% for side-channel attacks, 92.7% for physical probe extraction, and 100% for malicious firmware boot attacks. These results confirm that the propagation rate λ falls within the range $[0.8, 1.2]$, consistent with our model's typical value of $\lambda = 1.0$.
2. **Component Compromise-to-System Failure Stage Validation**: We conducted controlled firmware tampering experiments on 50 ESP32-based PV monitoring nodes, simulating successful component compromise. In 78.3% of cases, the tampered firmware caused system-level failures including MPPT control anomalies (62.0%), data acquisition errors (54.0%), and communication failures (38.0%). This result aligns with the model's μ coefficient range $[0.4, 0.6]$ and is consistent with independent industry data showing that 70–100% of hardware component failures in PV systems result in system-level malfunctions [5].
3. **System Failure-to-Harm Stage Validation**: We analyzed 127 documented PV system security incidents from 2020 to 2025, finding the following:
   - 15.2% of system failures resulted in measurable economic losses;
   - 2.1% caused grid stability issues;
   - 0.8% led to physical safety incidents.

   These real-world statistics confirm that the propagation rate ν falls within the range $[0.1, 0.2]$, consistent with our model's typical value of $\nu = 0.15$. These results provide empirical calibration for all transmission coefficients defined in Section 3.2.3.
4. **Indirect Validation Through LHSF Effectiveness**: The core prediction of our model is that interrupting propagation at the component compromise stage will prevent all downstream hazards. This is directly validated by the performance of our LHSF framework, which reduces component compromise rates to near-zero for most attack vectors (Table 10). By blocking attacks at this early stage, LHSF effectively eliminates the risk of system failures and physical hazards, demonstrating the model's practical utility for guiding security design.

5.5. Framework Generalization and Portability

Based on the experimental results, this section briefly summarizes the generalization capability of LHSF and provides basic porting guidance.

5.5.1. Directly Portable Components

The following core components can be directly transplanted to other resource-constrained IoT/industrial monitoring scenarios without modification:

1. Immutable OTP root key and secure boot chain construction;
2. SRAM-PUF non-resident key generation and lifecycle management protocol;
3. Random clock jitter + pseudo-operation side-channel defense mechanism;
4. On-chip real-time monitoring-based fault injection detection.

5.5.2. Scenario-Specific Adjustments

Only the following parameters need to be adjusted according to the target scenario:

- Error correction code strength (based on operating temperature range);
- Resource utilization thresholds (based on node hardware capabilities);
- Attack response intensity (based on physical hazard level);
- Firmware update frequency (based on system lifecycle).

LHSF has been verified on ESP32 and Raspberry Pi 4B, which are widely used in smart grid, environmental monitoring and industrial control scenarios. Therefore, the core security performance demonstrated in this paper has general reference value.

6. Conclusions and Future Research Directions

6.1. Conclusions

This paper proposes the Lightweight Hardware Security Framework (LHSF), centered on OTP trust roots and SRAM-PUF, to achieve firmware-level physical attack defense for photovoltaic monitoring nodes without increasing hardware costs. We developed a formal attack-hazard propagation model that reveals the mechanism by which hardware vulnerabilities propagate through information systems to cause physical and economic harms, and we validated each stage of the model through multi-stage experiments and real-world case analysis. Experiments demonstrate that LHSF achieves performance comparable to TPM 2.0 against major physical attacks while reducing boot delay, power consumption, and memory overhead by over 60%. Moreover, its extreme environment adaptability meets outdoor unattended operation requirements, providing a viable security solution for distributed energy IoT systems.

6.2. Limitations and Future Work

While the experimental results validate the effectiveness of LHSF for mainstream photovoltaic monitoring scenarios, a comprehensive assessment requires acknowledgment of its inherent limitations. These limitations, along with our planned future research directions, are presented below.

6.2.1. Limitations of the Proposed Framework

1. Constrained extreme environmental adaptation range: The current SRAM-PUF design paired with BCH(63,51) error correction maintains stable key generation within the tested temperature range (-20 °C to 60 °C) and voltage range (3.0 V to 3.6 V), which covers approximately 90% of global photovoltaic deployment regions. However, in more extreme environments such as desert areas with sustained temperatures above 65 °C or high-altitude polar regions with temperatures below -30 °C, the bit error rate of SRAM-PUF may approach the error correction limit of BCH(63,51), potentially leading to intermittent key generation failures.
2. Incomplete physical attack defense coverage: The framework currently provides robust defense against the most prevalent physical attacks targeting photovoltaic nodes, including voltage fault injection, power/electromagnetic side-channel analysis, physical probe extraction, and firmware tampering/rollback attacks. It does not yet include dedicated countermeasures against more specialized and resource-intensive attack methods, such as multi-pulse fault injection, laser fault injection, and hardware Trojans introduced during chip manufacturing or supply chain stages.
3. Limited platform verification scope: The security and performance of LHSF have been thoroughly validated on two widely used photovoltaic node platforms, ESP32-WROOM-32 (low-cost edge controller) and Raspberry Pi 4B (high-performance edge gateway). For more resource-constrained 8-bit microcontrollers (e.g., ATmega328P) and emerging RISC-V architecture chips commonly used in low-cost sensing nodes, the compatibility, resource overhead, and security effectiveness of LHSF components have not been systematically evaluated.
4. Production overhead for ultra-large-scale deployment: The current SRAM-PUF offline registration process requires individual sampling and HelperData generation for each node during factory production. While this design ensures the uniqueness of

each node's key and eliminates batch compromise risks, it may introduce additional production management overhead when deploying millions of nodes in utility-scale photovoltaic power plants.

6.2.2. Future Research Directions

Future research will deepen LHSF's exploration in four key areas: AI-based adaptive physical defense, edge-cloud collaborative key management and cross-domain authentication, hardware trojan detection and security integration for heterogeneous nodes, and ultra-low-power optimization for energy-harvesting nodes. We will also verify the effectiveness of LHSF in more industrial monitoring scenarios to further enhance its generalization capability. This will achieve deep integration of hardware security with artificial intelligence and edge-cloud collaboration. In addition, we plan to conduct accelerated aging tests in accordance with the IEC 62061/ISO 16750 standards to comprehensively verify the reliability of the LHSF over the entire 15- to 25-year lifecycle of photovoltaic nodes.

Author Contributions: Conceptualization, Z.L. and F.L.; Methodology, Z.L. and G.S.; Software, Z.L. and G.S.; Validation, F.L.; Formal analysis, Y.Y.; Investigation, J.X. and Y.Y.; Data curation, J.X.; Writing—original draft, Z.L. and J.X.; Visualization, Y.Y.; Supervision, F.L.; Funding acquisition, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Fundamental Research Program of Shanxi grant number 202403021212165.

Data Availability Statement: The data that support the findings of this study are not publicly available due to commercial confidentiality and proprietary restrictions. Reasonable requests for data access will be considered by the corresponding author subject to approval from the relevant institutional and commercial parties.

Acknowledgments: Declaration of AI-Assisted Technologies: During the preparation of this work, the author(s) used Doubao AI (version 2.0) and DeepL (version 26.1) solely for the purpose of language refinement, translation and grammar checking to improve readability. The core tasks of research conceptualization, theoretical modeling, algorithm design, experimental implementation, data collection and analysis, result interpretation, and manuscript revision were performed entirely by the author(s). The author(s) have reviewed and edited the AI-generated suggestions and translations and take full responsibility for the final content of this publication. No AI tools were used to generate research data, results, or conclusions. This use of AI is in compliance with the IEEE author center guidelines.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Priharti, W.; Rosmawati, A.F.K.; Wibawa, I.P.D. IoT based photovoltaic monitoring system application. *J. Phys. Conf. Ser.* **2019**, *1367*, 012069. [CrossRef]
2. Khan, M.S.; Sharma, H.; Haque, A. IoT Enabled Real-Time Energy Monitoring for Photovoltaic Systems. In Proceedings of the 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 14–16 February 2019; pp. 323–327.
3. Handschuh, H. Hardware-Anchored Security Based on SRAM PUFs, Part 1. *IEEE Secur. Priv.* **2012**, *10*, 80–83. [CrossRef]
4. Sassalou, E.-N.; Petreuș, D. A PUF-based Root-of-Trust for resource-constrained IoT devices. In Proceedings of the 2025 IEEE International Conference on Cyber Security and Resilience (CSR), Chania, Greece, 4–6 August 2025; pp. 824–831.
5. Kamenopoulos, S.N.; Tsoutsos, T. Assessment of the safe operation and maintenance of photovoltaic systems. *Energy* **2015**, *93*, 1633–1638. [CrossRef]
6. Chaudhuri, S.; Danger, J.-l. FASE: An Open Run-Time Reconfigurable FPGA Architecture for Tamper-Resistant and Secure Embedded Systems. In Proceedings of the 2006 IEEE International Conference on Reconfigurable Computing and FPGA's (ReConFig), San Luis Potosi, Mexico, 20–22 September 2006; pp. 1–9.
7. Latif, M.A.; Ahmad, M.B.; Khan, M.K. A Review on Key Management and Lightweight Cryptography for IoT. In Proceedings of the 2020 Global Conference on Wireless and Optical Technologies (GCWOT), Malaga, Spain, 6–8 October 2020; pp. 1–7.

8. Fischer, T.; Lesjak, C.; Pirker, D.; Steger, C. RPC Based Framework for Partitioning IoT Security Software for Trusted Execution Environments. In Proceedings of the 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 17–19 October 2019; pp. 430–435.
9. Khalil, K.; Idriss, H.; Idriss, T.; Bayoumi, M. Security in Resource-Constrained IoT Devices. In *Lightweight Hardware Security and Physically Unclonable Functions*; Springer: Cham, Switzerland, 2025; pp. 41–48. [CrossRef]
10. Wang, H.; Xiong, D.; Wang, P.; Liu, Y. A Lightweight XMPP Publish/Subscribe Scheme for Resource-Constrained IoT Devices. *IEEE Access* **2017**, *5*, 16393–16405. [CrossRef]
11. Aditya, A.; Vidyarthi, D.; Nene, M.J. A Study of Common Vulnerabilities in IoT Devices. In Proceedings of the 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 14–15 March 2024; pp. 1–6.
12. Cano-Quiveu, G.; Ruiz-De-Clavijo-Vazquez, P.; Bellido, M.J. IRIS: An embedded secure boot for IoT devices. *Internet Things* **2023**, *23*, 100874. [CrossRef]
13. Pocklassery, G.; Kajuruli, V.K.; Plusquellic, J.; Saqib, F. Physical unclonable functions and dynamic partial reconfiguration for security in resource-constrained embedded systems. In Proceedings of the 2017 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), Mclean, VA, USA, 1–5 May 2017; pp. 116–121.
14. Bhasin, S.; Regazzoni, F. A survey on hardware trojan detection techniques. In Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, 24–27 May 2015; pp. 2021–2024.
15. Ming, T.C. Reliability in IoT Era. In Proceedings of the 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), Noida, India, 10–11 August 2017; p. 1.
16. Oton, C.N.; Iqbal, M.T. Low-Cost Open Source IoT-Based SCADA System for a BTS Site Using ESP32 and Arduino IoT Cloud. In Proceedings of the 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 1–4 December 2021; pp. 681–685.
17. Sayekti, N.C.; Fhaizal, M.A.; Wijaya, W. Smart Farming Design with NodeMCU ESP32 for NFT HYDROPONIC. In Proceedings of the 2024 18th International Conference on Telecommunication Systems, Services, and Applications (TSSA), Bali, Indonesia, 17–18 October 2024; pp. 1–4.
18. Gielen, G. Low-power sensor interfaces. In Proceedings of the 2009 3rd International Workshop on Advances in Sensors and Interfaces, Trani, Italy, 25–26 June 2009; p. 86.
19. Newmarch, J. Raspberry Pi. In *Linux Sound Programming*; Springer: Berlin/Heidelberg, Germany, 2017; pp.537–545.
20. Savola, R.; Ahonen, P. Information Security Challenges in Industrial Automation Systems. In Proceedings of the 2006 4th IEEE International Conference on Industrial Informatics, Singapore, 16–18 August 2006; pp. 581–586.
21. Skwarek, V. Blockchains as security-enabler for industrial IoT-applications. *Asia Pac. J. Innov. Entrep.***2017**, *11*, 301–311. [CrossRef]
22. Agayev, F.; Hesenli, K.; Agababayev, R.A. Information Security Risks Monitoring and Management System. *Cauc.-Econ. Soc. Anal. J. South. Cauc.* **2024**, *58*, 71–77.
23. Chiu, T.; Xiong, W. SoK: Fault Injection Attacks on Cryptosystems. In Proceedings of the 12th International Workshop on Hardware and Architectural Support for Security and Privacy, Toronto, ON, Canada, 29 October 2023; pp. 64–72.
24. Li, J.; Zhang, H.; Zhao, B. Research of reliable trusted boot in embedded systems. In Proceedings of the 2011 International Conference on Computer Science and Network Technology, Harbin, China, 24–26 December 2011; pp. 2033–2037.
25. Ravi, S.; Raghunathan, A.; Chakradhar, S. Tamper resistance mechanisms for secure embedded systems. In Proceedings of the 17th International Conference on VLSI Design, Mumbai, India, 5–9 January 2004; pp. 605–611.
26. Xu, G.; Adetifa, O.; Mao, J. Developing Quantum Trusted Platform Module (QTPM) to Advance IoT Security. *Future Internet* **2025**, *17*, 193. [CrossRef]
27. Fu, D.; Peng, X. TPM-Based Remote Attestation for Wireless Sensor Networks. *Tsinghua Sci. Technol.* **2016**, *21*, 312–321. [CrossRef]
28. Rührmair, U.; Sehnke, F. Modeling attacks on physical unclonable functions. In Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS), Chicago, IL, USA, 4–8 October 2010; pp. 237–249.
29. Mattela, V. IoT device development challenges and solutions. In Proceedings of the 2015 IEEE Hot Chips 27 Symposium (HCS), Cupertino, CA, USA, 22–25 August 2015; pp. 1–19.
30. Anagnostopoulos, N.A. Securing IoT Devices Using Robust DRAM PUFs. In Proceedings of the 2018 Global Information Infrastructure and Networking Symposium (GIIS), Thessaloniki, Greece, 23–25 October 2018; pp. 1–5.
31. Nilesh, K.; Deppe, C.; Boche, H. Secret key generation and Storage based on QPUF. In Proceedings of the 2025 IEEE Information Theory Workshop (ITW), Sydney, Australia, 29 September–3 October 2025; pp. 1–6.
32. Shafiei, A.; Monajati, M. Efficient and Lightweight IoT Security Using CNTFET-Based Ultra-Low Power SRAM-PUF. *AUT J. Electr. Eng.* **2025**, *57*, 31.
33. Liu, J.; Wang, J.; Dong, F.; Jiang, P.; Xue, A. Wireless Communication System Design for Remote Monitoring. In Proceedings of the 2006 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; pp. 280–284.

34. Voicu, V.; Petreuș, D.; Etz, R.; Mois, G. IoT Sensor Node for Solar Energy Measurements. In Proceedings of the 2025 26th International Carpathian Control Conference (ICCC), Starý Smokovec, Slovakia, 19–21 May 2025; pp. 1–6.
35. Brier, E.; Clavier, C.; Olivier, F. Correlation Power Analysis with a Leakage Model. In Proceedings of the International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2004), Cambridge, MA, USA, 11–13 August 2004; pp. 16–29.
36. Maragathavalli, K.; Jothi, R.M.J. A Lightweight Protocol for Secure Communication Using Two-Layered Encryption. In Proceedings of the 2025 Third International Conference on Industry 4.0 Technology (I4Tech), Pune, India, 18–20 September 2025; pp. 1–6.
37. Kocher, P.C.; Genkin, D.; Gruss, D. Spectre Attacks: Exploiting Speculative Execution. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 1–19.
38. Romero-Sánchez, M.G.; Rodriguez-Abreo, O. Development of a Solar Panel Monitoring System for Power and Visible Light Using Arduino and INA219/BH1750 Sensors. In Proceedings of the 2025 22nd International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 22–24 October 2025; pp. 1–5.
39. Porter, S.; Mansi, M.; Sumpter, N. Low cost smart sensors using infrared detector arrays. *Sens. Rev.* **2001**, *21*, 283–287. [CrossRef]
40. Nguyen, H.N.; Nguyen, T.T.; Thi, T.N.N. Proposed Methods to Rollback A Failed Update of IoT Devices. *Int. J. Eng. Adv. Technol.* **2021**, *11*, 55–62. [CrossRef]
41. Dai, W.; Rubin, S.H. A supply chain model for software components management. In Proceedings of the Fifth IEEE Workshop on Mobile Computing Systems and Applications, Las Vegas, NV, USA, 27 October–1 November 2003; pp. 69–76.
42. Moon, Y.H.; Jeon, Y.S. Cooperative remote attestation for IoT swarms. In Proceedings of the 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Republic of Korea, 19–21 October 2016; pp. 1233–1235.
43. Macke, S.; Munsch, S.; Stascheit, J.; Maidl, U.; Hegemann, F. AI-based anomaly detection in tunnel excavation data. *Geomech. Und Tunnelbau* **2024**, *17*, 312–323. [CrossRef]
44. Yahya, W.; Basuki, A. Lightweight monitoring system for IOT devices. In Proceedings of the 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), Lombok, Indonesia, 26–27 October 2017; pp. 1–4.
45. *AEC-Q100*; Failure Mechanism Based Stress Test Qualification for Integrated Circuits. Automotive Electronics Council (AEC): Washington, DC, USA, 2014.
46. *IEC 60068-1*; Environmental Testing—Part 1: General and Guidance. International Electrotechnical Commission (IEC): Geneva, Switzerland, 2013.
47. *IEC 62061*; Safety of Machinery—Functional Safety of Safety-Related Control Systems. International Electrotechnical Commission (IEC): Geneva, Switzerland, 2021.
48. *ISO 16750-1*; Road Vehicles—Environmental Conditions and Testing for Electrical and Electronic Equipment—Part 1: General. International Organization for Standardization (ISO): Geneva, Switzerland, 2023.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.