Design of a Defense-in-Depth Interlock System for a Proton Accelerator Based on EPICS

Authors: Chen, Sirui, Pei, Ms. Shang, Guijin, Mrs. Yang, Dr. Yuhui Guo, Zhou, Detai, jiming, Mr. yang, Xiuyan, Ms. Han, Qimin, Mr. Ma, GUO, Dr. Yuhui

Date: 2025-11-18T00:00:00+00:00

Abstract

To address the stringent requirements for real-time performance, reliability, and flexible protection in the complex operational environments of proton accelerators, this study proposes a layered interlock and defense-in-depth system that integrates PLC-based hardware interlocks, EPICS-based software protection, and node attribute register mechanisms. Employing a three-tier architecture—"hardware rapid response, software flexible protection, and redundant backup defense" —the system achieves millisecond-level response at the hardware layer, dynamic logic reconfiguration at the software layer, and fail-safe closed-loop operation at the backup layer. Node attribute registers function as cross-layer interfaces that enable parameter mapping and online verification between software and hardware logic, while also supporting real-time configuration and strategy switching through EPICS. The Python-based logic modules extend EPICS' s computational capability, enabling multi-signal fusion and dynamic threshold adjustment. This approach maintains logical consistency across varying operating conditions and minimizes false triggers. An independent fail-safe backup defense chain is established through communication-monitoring and execution-feedback mechanisms to ensure safe device isolation in the event of failures at the hardware or software layers. Experimental results demonstrate that the proposed architecture significantly outperforms traditional solutions in terms of response speed, logical consistency, and fault tolerance. It exhibits strong engineering applicability for large-scale deployment, providing a reproducible layered defense pathway for next-generation accelerator control and protection systems.

Full Text

Design of a Defense-in-Depth Interlock System for a Proton Accelerator Based on EPICS

Chen SR^{1,2}, **Shang P**^{1}, **Yang GJ**^{2}, **Guo YH**^{1,†}, **Zhou DT**^{1}, **Yang JM**^{1,3}, **Han XY**^{1,2}, and **Ma QM**^{1,3}

^{1}Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou 730000, China
^{2}Northwest Normal University, Lanzhou 730070, China
^{3}University of Chinese Academy of Sciences, Beijing 100049, China

To address the stringent requirements for real-time performance, reliability, and flexible protection in the complex operational environments of proton accelerators, this study proposes a layered interlock and defense-in-depth system that integrates PLC-based hardware interlocks, EPICS-based software protection, and node attribute register mechanisms. Employing a three-tier architecture—"hardware rapid response, software flexible protection, and redundant backup defense" —the system achieves millisecond-level response at the hardware layer, dynamic logic reconfiguration at the software layer, and fail-safe closed-loop operation at the backup layer. Node attribute registers function as cross-layer interfaces that enable parameter mapping and online verification between software and hardware logic, while also supporting real-time configuration and strategy switching through EPICS. The Python-based logic modules extend EPICS' s computational capability, enabling multi-signal fusion and dynamic threshold adjustment. This approach maintains logical consistency across varying operating conditions and minimizes false triggers. An independent fail-safe backup defense chain is established through communication-monitoring and execution-feedback mechanisms to ensure safe device isolation in the event of failures at the hardware or software layers. Experimental results demonstrate that the proposed architecture significantly outperforms traditional solutions in terms of response speed, logical consistency, and fault tolerance. It exhibits strong engineering applicability for large-scale deployment, providing a reproducible layered defense pathway for next-generation accelerator control and protection systems.

Keywords: Proton Accelerator; EPICS; PROFINET; Node Attribute Register; Defense in Depth

Introduction

Proton accelerators are essential instruments in high-energy physics research, nuclear medicine, and materials modification [1–3]. The reliability of their control and protection systems directly affects both experimental stability and equipment safety. With the continuous increase in device scale and energy levels, the range of equipment managed by the system and the associated interlock logic have become increasingly complex, imposing greater demands on control real-

time performance, operational flexibility, and redundant safety mechanisms. Although traditional centralized control architectures were widely adopted in early accelerator facilities, their limitations in communication latency, scalability, and fault tolerance have become increasingly evident [4,5] as system complexity has grown. Particularly problematic are the single-point-of-failure risk introduced by centralized control nodes and the cumbersome process of recompiling and redeploying modified logic. These challenges hinder the rapid reconfiguration and dynamic adjustments required for multi-experiment operational modes.

To address these challenges, major accelerator facilities worldwide have progressively adopted distributed and hierarchical control strategies. For example, CERN' s LHC has developed a multi-level hardware interlock system [6] that leverages high-speed networks to achieve millisecond-level response times. Similarly, the U.S. SNS and Japan' s J-PARC, employing a collaborative EPICS-PLC architecture, implement layered hardware protection [7–9] and software strategies to improve system maintainability. Domestic facilities such as SSRF and CSNS have likewise introduced the "fast protection and slow interlock" concept [10,11] to balance operational safety and flexibility. Despite these advances, existing systems continue to exhibit significant shortcomings in three key areas.

First, the hardware layer provides deterministic responses but lacks dynamic configuration capability [12]. Although PLC-based hardware interlocks can achieve millisecond-level response, their logic is typically hardcoded within the control program. This rigidity makes it difficult to dynamically adjust thresholds or interlock relationships according to different experimental modes. Reconfiguration requires system shutdown and reprogramming, thereby reducing operational flexibility. Second, the software layer provides flexibility but suffers from insufficient decoupling from the underlying hardware logic. EPICS-based software protection supports complex algorithms and strategic adjustments but typically depends on variable-based communication. This dependency introduces reliability vulnerabilities [13–15] during network interruptions, communication congestion, or software failures, thereby hindering deep coordination with hardware logic. Third, there is a lack of redundant safety loops under system failure conditions. When hardware or software protection fails due to communication anomalies, actuator malfunctions, or similar faults, traditional systems lack independent backup safeguards. This absence prevents the formation of effective layered safety defenses, resulting in protection blind spots.

To overcome the aforementioned technical bottlenecks, this study proposes a hierarchical interlock system integrating PLC-based hardware interlocks and EPICS-based software protection to enhance real-time performance, operational flexibility, and system reliability. Two key technologies are introduced, namely the node attribute register (NAR) mechanism and the Defense-in-Depth Backup Layer (DBL). The objective of this research is to establish a generalizable multi-level safety control framework that integrates rapid hardware response, flexible software configuration, and defense-in-depth redundancy, thereby forming a unified platform with multi-tier redundancy and cross-layer fault-tolerance ca-

pabilities. To achieve this goal, the system adopts a three-tier defense-in-depth architecture comprising "hardware rapid response, software flexible protection, and redundant backup defense." The bottom layer employs PLCs to execute deterministic logic judgments and achieve rapid disconnection within 10 ms. The middle layer integrates EPICS-IOC with a Python-based logic engine for multi-source signal fusion, dynamic logic computation, and strategy bypassing. The top layer establishes the DBL, which independently performs fail-safe actions in cases of communication failure or actuator non-response, thereby forming a closed-loop system-level safety mechanism. The key research focuses include: (1) the design of a real-time communication architecture based on PROFINET; (2) the engineering implementation of the node attribute register mechanism; (3) dynamic logic decoupling and strategy deployment for EPICS-based software protection modules; and (4) redundancy verification and performance evaluation of the defense-in-depth backup layer.

II. System Architecture and Technical Route

This chapter presents the overall architecture and technical approach of the layered interlock system designed for the proton accelerator. The system employs a three-tier defense-in-depth structure [16]—comprising hardware rapid interlocking, software flexible protection, and independent backup defense—to achieve end-to-end protection from signal acquisition to safety lockout. The overall architecture of the proton accelerator' s layered interlock system is illustrated in [Figure 1: see original paper].

The system consists of three functional layers. The bottom layer, the equipment-level hardware interlock layer, is centered on distributed PLC control units. This layer performs real-time monitoring and closed-loop shutdown of equipment such as vacuum systems, beam diagnostics, temperature sensors, door interlocks, fast-acting valves, power supplies, and liquid-level devices. The middle layer serves as the EPICS-based software protection layer, operating within the IOC framework. It employs Python-based logic modules to fuse and evaluate multi-source signals, implement delay control, and dynamically adjust protection strategies, thereby enabling flexible logic-driven protection. The top layer, known as the Defense-in-Depth Backup Layer (DBL), operates independently of the main control chain. This layer continuously monitors actuator feedback and communication status, issuing fail-safe lockout signals in response to hardware or software anomalies to directly disconnect power from critical components such as the ion-source high-voltage system, microwave power modules, beam-chamber supplies, and magnetic-field excitation power sources. Data mapping and parameter synchronization among the three layers are accomplished through node attribute registers, forming a layered defense framework characterized by rapid response, logical judgment, and redundant protection.

[Figure 1: see original paper] shows the system architecture diagram. The primary interlock system, composed of distributed PLC units, monitors critical equipment such as vacuum systems, beam diagnostics, temperature sen-

sors (T3/T4), door interlocks, CM4 fast-acting valves, power supplies, and liquid-level devices. It enables local closed-loop decision-making with response times of $\leq$ 10 ms, ensuring the immediate isolation of hazardous conditions during the onset of an incident [17]. The secondary interlock layer, implemented through EPICS IOCs and Python-based logic modules, primarily provides software-level protection for devices such as insertable components, valve states, LEBT_{DUMP} signals, BPM temperatures, MEBT/HEBT thermal readings, and power-supply operating status. This layer enables cross-device signal fusion and dynamic protection-policy configuration. The tertiary interlock functions as an independent Defense-in-Depth Backup Layer (DBL). By collecting actuator feedback and communication heartbeat signals, this layer directly issues fail-safe lockout commands when both software and hardware layers malfunction, thereby forming the system' s ultimate line of defense.

The system design follows three fundamental classification principles: real-time performance, logical complexity, and redundancy. For components requiring millisecond-level response and actions that can be directly verified by hardware (e.g., vacuum systems and fast-acting valves), hardware interlocks are employed. For equipment requiring multi-signal integration or operational-mode adjustment (e.g., insertable components, valve states, and BPM temperatures), EPICS- and Python-based software protection is implemented. For high-risk equipment (e.g., high-voltage power supplies and chamber power sources), both hardware and software protections are integrated to establish a "rapid shutdown plus strategic redundancy" multi-layered closed-loop defense. Hardware interlocks ensure deterministic system responses, software protections provide logical flexibility, and the backup defense layer preserves intrinsic equipment safety under extreme failure conditions. Together, these three elements constitute a complementary and progressive layered defense framework.

In terms of technical implementation, the system employs node attribute registers (NARs) as a unified interface to realize parameter mapping and state synchronization between software and hardware logic. Each physical device is abstracted into a logical node characterized by state, threshold, delay, and bypass attributes. EPICS enables online adjustment of protection parameters by modifying NARs without recompiling the PLC logic [18], thereby supporting cross-layer coordination and real-time reconfiguration. The overall data flow and control logic of the system follow a closed-loop process of "detection–judgment–execution–verification." Field signals are acquired by PLCs and first evaluated at the hardware level before being transmitted to EPICS. EPICS then performs logical computations and generates corresponding control strategies, while PLCs execute physical actions and upload feedback data. The Defense-in-Depth Backup Layer (DBL) independently verifies the execution results and issues fail-safe commands when required. [Figure 2: see original paper] illustrates the technical architecture of the hierarchical interlock system.

[Figure 2: see original paper] shows the system technology roadmap. Through this technical approach, hardware–software decoupling, inter-layer logical consis-

tency, and a closed-loop redundant protection mechanism are achieved, thereby significantly enhancing the maintainability, reliability, and intrinsic safety of the accelerator' s interlock system.

III. System Hardware Design

This chapter presents the hardware implementation scheme of the equipment layer within the hierarchical interlock system, covering the PLC control architecture, hardware interlock logic, and node attribute register (NAR) mechanism. The system is designed to provide highly reliable, real-time, and reconfigurable protection functions in complex operational environments.

The equipment layer constitutes the foundational core of the hierarchical interlock system and is responsible for acquiring physical signals from the accelerator, performing logical evaluations, and executing control actions. The system employs a Phoenix Contact AXC3050 PLC as the main control unit, forming a distributed control network through the PROFINET real-time bus and remote I/O modules to achieve deterministic response times of $\leq$ 10 ms.

A. Node Attribute Register Mechanism

Safety interlock protection represents one of the core technologies in machine protection systems. However, conventional safety interlock solutions are typically tailored to specific devices and interlock logic. Different devices, or even different operating modes of the same device [19,20], can result in variations in both the safety interlock system and its associated logic. Such systems generally lack universality and exhibit poor adaptability. When interlock nodes, dependencies, or logic are altered, system shutdown is often required to modify the corresponding logic or circuitry. This process disrupts normal machine operation and increases the workload of technical personnel. Frequent modifications can also introduce unpredictable errors, resulting in significant losses of both time and resources. Moreover, the testing process may pose risks of equipment damage and potential personal injury.

To address the aforementioned issues, and considering that particle accelerator devices share consistent primary control objectives across different operating modes or experimental goals—differing only in interlock ranges [21], thresholds, and execution actions—this study proposes a node-attribute-based configuration method, as illustrated in [Figure 4: see original paper]. By incorporating configurable and modifiable node-attribute characteristics into the input and output nodes, the variable parameters of the safety interlock are decoupled from the underlying logic. By modifying the attribute bits of these node characteristics through upper-level software, the scale, interlock relationships, and logical structure of the safety interlock system can be adjusted within predefined parameters.

[Figure 4: see original paper] shows the node attribute register schematic diagram. The implementation of the node attribute register (NAR) mechanism

adopts a bottom-up systematic design approach, comprising three stages: the hardware layer, the logic layer, and the integration layer. Each layer is responsible for signal acquisition, logic mapping, and system integration, collectively forming a closed-loop structure of "configuration–feedback–verification." This design ensures parameter synchronization and state consistency between EPICS and PLC systems.

At the hardware implementation level, the system is based on a distributed PLC control architecture designed to achieve high real-time data acquisition and cross-layer synchronization. The control network is centered on PROFINET, which enables high-speed sampling and deterministic transmission of field signals with communication cycles as short as 4 ms [22]. This configuration ensures the real-time updating and consistency verification of Node Attribute Register (NAR) parameters. Field I/O modules are functionally configured according to signal characteristics, forming a three-tier structure that encompasses status acquisition, control output, and monitoring diagnostics. These correspond respectively to the S, C, and M logical domains, enabling direct mapping and invocation of various signal types at the logical layer. During configuration, unified signal naming and address-mapping rules are established through the PLCnext Engineer software. This approach ensures a one-to-one correspondence between hardware registers and EPICS PV variables at the semantic layer, thereby achieving complete unification of hardware and software namespaces as well as data structures.

At the logic layer, the core structure of the Node Attribute Register (NAR) is implemented through modular programming using the SCL language. The system abstracts each controlled device into a logical node, with each node containing fundamental attributes including State, Threshold, Delay, Bypass, and Timestamp. All nodes are declared as arrays within the global variable area of the PLC and establish a one-to-one mapping with process variable (PV) entries in the EPICS database through the PROFINET interface. Node logic adopts structured programming to perform operations including parameter reading, threshold comparison, and state updating. When field input signals exceed predefined threshold ranges, the node triggers local protective actions and writes to feedback registers containing the action results and communication status. To prevent logic drift between the software and hardware layers, a periodic consistency verification module is implemented within the PLC. This module compares the EPICS-side configuration registers with the PLC-side feedback registers and generates a verification value through CRC-based hash computation. If parameter inconsistencies are detected, the system freezes the node logic and raises an alarm flag. This design ensures the independence of logical judgments and the consistency of software and hardware data.

At the integration layer, the Node Attribute Registers (NARs) enable system-level data interaction and visualization management through the EPICS framework. The EPICS IOC functions as the core control unit in the middle layer, establishing bidirectional communication with the PLCs through the

Channel Access (CA) protocol to create real-time links for parameter distribution and status feedback. The process variable (PV) structure defined in the EPICS database corresponds strictly to the PLC register fields, following the unified naming convention "NAR:DeviceName:ParameterType" —for example, "NAR:VALVE1:THRESHOLD" or "NAR:POWER1:BYPASS." Communication adopts a dual-channel mechanism, in which the downlink channel writes EPICS configuration data to the PLC configuration registers, whereas the uplink channel reads real-time status data from the PLC feedback registers. A dual-verification logic combining timestamps with the "Config_{Valid}/Feedback_{OK}" flags is implemented to ensure data consistency and reliability during transmission. When the EPICS-side configuration is not acknowledged by the PLC, the system automatically reverts to the last stable version and records the event in the operational log.

At the interface layer, the graphical management interface developed on the Phoebus OPI platform enables the real-time display of node attributes, parameter adjustment, and visualization of verification status. The interface adopts a modular template structure that is divided into three sections: parameter display, configuration, and logging. The parameter display section presents the node' s real-time operational status and key parameters; the configuration section allows authorized users to modify thresholds, delays, and bypass flags online during operation; and the logging section displays real-time verification results and configuration change records. A color-coded dynamic-binding mechanism intuitively reflects the system status: green indicates configuration consistency, yellow denotes pending confirmation, and red signifies verification failure. Through this interface, operators can perform cross-layer parameter monitoring and logical verification within a unified environment, thereby fulfilling the design objectives of transparent hardware status, visualized logical behavior, and traceable system operations.

In summary, the implementation of the Node Attribute Register (NAR) mechanism establishes a high–real-time signal acquisition platform at the hardware layer, realizes parameter structuring and consistency verification at the logic layer, and accomplishes cross-system data mapping and visualization control at the integration layer. The collaboration among the three layers forms a complete bottom-up technology chain, endowing the system with dynamic configuration, online verification, and logical self-consistency capabilities. This provides stable data interfaces and security assurance for the subsequent design of the backup layer within the defense-in-depth architecture.

IV. Software Implementation and Defense-in-Depth Mechanism

This chapter presents the software implementation framework of the hierarchical interlock system, encompassing the structural design of the EPICS-IOC–based soft protection module, the Python-driven dynamic logic operation mechanism, and the collaborative strategy of the defense-in-depth backup layer. The soft-

ware layer serves as the core component responsible for logical determination, parameter configuration, and operational monitoring throughout the system. It functions as both the upper-level strategic extension of hardware-based rapid interlocking and the visualization hub of the entire system.

The system software is developed on the EPICS architecture and achieves modular deployment through a three-tier structure comprising the Driver Layer [23], Logic Layer, and Interface Layer. The underlying PLC communicates with the IOC layer through the TCP/IP protocol to enable high-speed data exchange. The upper EPICS layer employs the Channel Access (CA) protocol to map physical quantities to process variables (PVs), thereby enabling signal standardization and cross-layer accessibility. The Driver Layer establishes a one-to-one correspondence between PLC registers and EPICS PVs, ensuring synchronized and real-time updating of the underlying data. The Logic Layer performs logical evaluations, condition combinations, and action generation based on EPICS record types such as calc, bo, and ai. The Interface Layer manages strategic parameters such as thresholds, delays, and bypasses, supporting runtime loading and cross-platform reuse. This architecture maintains real-time performance while ensuring logical independence, parameter decoupling, and strategy reconfigurability.

A. Design of Python Soft Protection Module

To overcome the limitations of hardware interlocks in logical complexity and information representation, the system incorporates a Python- and EPICS-based soft protection module at the IOC layer. This module employs the EPICS IOC as its runtime container and Python scripts as its logic engine, thereby establishing a soft protection layer that supports multi-source data fusion and dynamic policy determination. Unlike traditional EPICS record types that rely on static expression calculations, the Python-based soft protection module is developed using the Pcaspy software package [24]. It directly accesses real-time signals from PLCs through the EPICS Channel Access interface while simultaneously retrieving data from AA historical databases and external mathematical computation systems. This enables multidimensional analysis of operational data across the time domain, frequency domain, and statistical dimensions.

[Figure 5: see original paper] illustrates the soft protection flowchart. This module leverages Python' s numerical computation libraries (NumPy, SciPy, pandas, etc.) to execute advanced computational tasks [25–27], including trend fitting, threshold prediction, derivative analysis, and anomaly detection. Through custom logic functions, the system enables nonlinear condition evaluation, dynamic threshold adjustment, and multi-condition joint decision-making, thereby overcoming the expressive limitations inherent in EPICS' s native record types. During operation, dynamic logic computations are performed by the module based on real-time parameters provided by the Node Attribute Register (NAR). When the computation results satisfy the predefined action conditions, output signals are written to the EPICS process variables (PVs), thereby triggering the corre-

sponding interlock responses or alarm events through upper-level logic.

Unlike traditional hardware interlocks that output only binary "active/inactive" results, the soft protection module outputs multidimensional status elements, including status codes, bypass flags, delay countdowns, trigger conditions, and timestamps. This design ensures interpretable status information and traceable system operations. It allows the soft protection layer to function not only as a logical redundancy mechanism for critical equipment but also as an independent module capable of handling complex fault determination and predictive computation tasks beyond hardware capabilities, thereby enhancing overall system safety and intelligence.

The soft protection module adopts a parameterized configuration architecture, enabling more flexible policy adjustment and operational mode switching. Users can modify key parameters such as thresholds, delays, and bypass settings online through the Phoebus OPI interface. These modifications are instantly loaded by the IOC and propagated to active processes via Channel Access, taking effect without the need for system recompilation or restart. This design enables hot logic updates and runtime reconfigurability, thereby significantly improving operational efficiency and system maintainability.

[Figure 6: see original paper] shows the soft protection logic execution flow. Simultaneously, the Phoebus interface integrates three visualization components —real-time waveform display, alarm logging, and logical linkage diagrams. The system displays the verification results of each process variable (PV) using color-coded status indicators: green signifies configuration consistency, yellow indicates pending confirmation, and red denotes verification failure. By utilizing time-series curves, alarm trigger records, and dynamic threshold trend analyses, operators can conduct cross-level system monitoring and historical state tracing. This mechanism achieves full-chain transparency from parameter configuration to logic verification, thereby transforming the interface from a simple monitoring tool into a decision-support resource.

B. Defense-in-Depth Backup Layer Collaboration

The Defense Backup Layer (DBL) constitutes the highest level of protection within the system' s security architecture. It is designed to independently perform security isolation and fault control, even in situations where both hardware-based rapid interlock mechanisms and software protection logic fail or exhibit abnormal responses. Through dedicated execution units and communication-monitoring mechanisms, this layer performs redundant verification of lower-level logic and provides system-level fault-tolerance protection, thereby forming the ultimate line of defense that integrates hardware, software, and policy-based safeguards into a three-tiered, interconnected architecture.

The design of the Defense Backup Layer (DBL) adheres to the principles of logical independence, data redundancy, and fail-safe operation [28]. The system achieves rapid response at the hardware layer, performs logical evaluations at the

software layer, and maintains control-chain integrity through periodic self-checks and link monitoring within the backup layer. Upon detecting communication anomalies, interlock logic card malfunctions, or prolonged unresponsiveness of critical nodes, the DBL automatically assumes control authority and executes safety actions. These actions include shutting down high-voltage power supplies, cutting off fast-acting valves, and closing vacuum isolation channels, thereby establishing a physical safety loop.

[Figure 7: see original paper] illustrates the Defense Backup Layer (DBL) principle. The core logic of the backup layer comprises two components: the Communication Watchdog and the Execution Feedback Unit. The Communication Watchdog periodically monitors the communication status between EPICS and PLC systems, as well as the update cycles of the node attribute registers. Upon detecting consecutive timeouts (e.g., exceeding three communication cycles) or CRC checksum anomalies, it triggers a Communication Failure Flag. The Execution Feedback Unit continuously monitors the actual operational states of field devices and compares them with the outputs of the software protection layer. When the feedback signals contradict the logical outputs (e.g., when a valve should be closed but its position signal remains open), the system automatically enters forced-safe mode. The backup layer subsequently issues a direct physical disconnection command.

The logical implementation of the Defense Backup Layer (DBL) is based on the EPICS framework but operates within an independent IOC process that is isolated from the main control IOC. This process accesses lower-level device states exclusively through restricted channels and does not participate in normal logical operations, thereby ensuring operational independence. The system employs a Heartbeat Process Variable (PV) monitoring mechanism, based on PV subscriptions, to dynamically verify the response cycles of each logical layer [29]. Upon detecting a halted heartbeat from any IOC or a suspension of the soft protection logic, the DBL immediately executes predefined safety actions and records the anomaly for subsequent analysis.

To achieve multi-level coordination, a policy-synergy mechanism has been established between the Defense Backup Layer (DBL) and the soft protection layer. During logical operation, the soft protection layer synchronizes critical state variables—such as thresholds, action flags, and bypass states—with the DBL' s monitoring cache. When the system enters a pre-alert state, the DBL executes the corresponding defense strategies based on the most recent valid configuration, thereby achieving temporal–spatial decoupling between soft-logic determination and hard-action execution. This mechanism not only prevents protection gaps resulting from upper-layer failures but also enables the backup layer to execute response actions that are more targeted and explainable.

The execution outputs of the Defense Backup Layer (DBL) adopt a fail-safe design, in which all safety-action outputs are implemented through dual-channel redundant relays. When the primary channel signal fails, the backup channel automatically assumes the output function. The output status is simultaneously

verified through closed-loop feedback signals to ensure that the action outcome aligns with the commanded state. The system design equips the backup layer with an independent power supply and a separate grounding path, thereby guaranteeing autonomous operational capability even in the event of complete upper-level control failure.

Through the aforementioned mechanism, the backup layer within the defense-in-depth architecture provides dual fault-tolerance protection for both the hardware interlock and software protection layers, thereby establishing a multi-level redundancy framework that extends from logical anomaly detection to physical security isolation. The proposed "policy coordination–link self-check–independent execution" mechanism enables the accelerator system to demonstrate greater security autonomy and enhanced robustness under complex operating conditions, while providing data support and logical interfaces for subsequent fault diagnosis and risk prediction.

V. System Testing and Results Analysis

To validate the effectiveness and reliability of the hierarchical interlock and defense-in-depth system, two types of tests were conducted in an experimental environment: functional verification and performance verification. Functional verification primarily assessed the logical correctness and coordination consistency among the hardware interlocks, the EPICS-based soft protection layer, and the backup layer within the defense-in-depth system. Performance verification quantitatively evaluated the system' s overall real-time performance and robustness using metrics such as response latency, redundant switching, and fault-tolerant recovery. The test platform consisted of field PLC cabinets, IOC hosts, independent backup control units, and Phoebus operator terminals. All test signals were introduced through real equipment simulators and field sensors.

A. Functional Validation: Multi-Layer Interlock Logic Consistency Testing

The objective of the functional verification is to confirm that the three-tier interlock system maintains logical consistency and safety responses under various operational scenarios. The testing process is conducted at three levels: (1) Hardware-Level Verification: Fault-injection tests were performed on critical equipment signals, including vacuum, fast-acting valves, power supplies, and temperature sensors. The PLC program monitored input changes and executed lockout commands within ≤ 10 ms, thereby verifying the deterministic response capability of the field-level rapid interlocks. (2) Software Protection Layer Verification: Signal-combination and threshold-debouncing experiments were conducted using Python-based dynamic logic modules on the EPICS IOCs. When equipment thresholds approached critical states, the system dynamically adjusted action determinations based on real-time changes, thereby enabling multi-condition fusion judgment and bypass-strategy switching. Experimental results demonstrated that under high-frequency disturbance conditions, the

false-trigger rate of the software protection module outputs was less than 0.05%, which is significantly lower than that of traditional fixed-logic solutions. (3) Backup Layer Validation: The autonomous takeover capability of the Defense-in-Depth Backup Layer (DBL) was verified under conditions of communication interruption and logic-card latch-up. When the EPICS-layer IOCs were manually suspended, the DBL detected lost heartbeats within approximately 300 ms and executed fail-safe outputs, thereby automatically disconnecting the ion-source high voltage and fast-valve control to achieve final physical isolation.

During redundancy and fault-tolerance testing, the system simulated scenarios such as communication interruptions, threshold-configuration errors, and partial I/O module failures. The results show that when the primary communication link was lost, the Defense Backup Layer (DBL) assumed control within 0.2 seconds and maintained the equipment in a safe state. When the Node Attribute Register (NAR) was misconfigured, the attribute-verification mechanism of the EPICS layer promptly blocked command issuance, thereby preventing erroneous actions. When certain PLC modules failed, the system maintained stable operation through bypass strategies and redundant signals from the DBL. [Figure 9: see original paper] illustrates the fault-tolerance recovery time distribution across different failure scenarios, thereby evaluating the self-recovery capability and fail-safe triggering characteristics of the hierarchical interlock and defense-in-depth system under complex failure conditions.

The test results demonstrate that the three-tier interlock system operates correctly under various operating conditions. The strategy outputs of the software protection layer were consistent with the execution results of the hardware layer, while the backup layer safely assumed control when the upper layers failed. These results validate the effective implementation of logical consistency and multi-level redundancy within the system.

B. Performance Validation: Real-time, Redundancy, and Fault Tolerance Testing

Performance verification evaluates the system' s operational characteristics from three perspectives: (1) In real-time testing, response delays across different layers were measured using oscilloscopes and timestamp-recording modules. The results indicate an average response time of 8.6 ms for the hardware interlocks, 162 ms for the EPICS-based software protection layer, and approximately 300 ms for the backup layer' s fail-safe output under trigger conditions. The overall system response satisfies the accelerator control system' s safety requirement of less than 200 ms, exhibiting stable latency distribution across all layers and jitter below 2 ms. [Figure 8: see original paper] provides a comparison chart of response delays across layers.

Functional and performance tests have demonstrated that the proposed layered interlock and defense-in-depth system for the proton accelerator outperforms existing solutions in logical consistency, response speed, and fault tolerance. The

hardware layer ensures deterministic and rapid response; the software protection layer enables flexible and reconfigurable logical expansion; and the backup layer provides an independent fail-safe protection loop. The system' s multi-level redundancy mechanism significantly reduces the probability of false triggers and system failure rates, thereby establishing a reproducible engineering framework for the safe control of large-scale accelerator facilities.

VI. Conclusion

This paper proposes and implements a hierarchical interlock system for proton accelerators, integrating PLC-based hardware interlocks and EPICS-based software protection to provide an innovative solution to challenges of real-time performance, safety, and flexibility in complex experimental environments. By introducing a three-tier interlock protection architecture, the system achieves coordinated operation among rapid hardware-level response, flexible software-level configuration, and a defense-in-depth backup layer. This approach enhances the system' s maintainability and scalability while ensuring the safe operation of the accelerator. Notably, the introduction of the node-attribute register mechanism enables more efficient online switching and expansion of device logic, thereby significantly enhancing the system' s adaptability. Furthermore, the soft-protection module, leveraging Python' s computational capabilities, processes complex multi-source data and provides stronger redundancy protection than traditional hardware interlocks. The defense-in-depth backup layer ensures equipment safety during system failures, thereby enhancing the system' s fault tolerance and overall reliability.

Although the proposed system has achieved significant progress in accelerator safety protection, further optimization remains necessary as accelerator facilities continue to expand in scale and complexity. Future research will focus on enhancing the system' s level of intelligence by exploring the application of artificial intelligence (AI) and machine learning (ML) algorithms to enable predictive analysis of equipment status and early fault warning. Additionally, with the advancement of cloud computing and virtualization technologies, cross-platform compatibility and deployment flexibility will become key research priorities, facilitating broader system applications and upgrades across diverse experimental scenarios. Finally, deep integration between software and hardware layers represents a crucial direction for future system development. By more tightly coupling hardware redundancy mechanisms with software logic, the system' s safety and real-time responsiveness can be further enhanced. Through these optimizations and extensions, the system will be better equipped to meet the safety protection requirements of future high-energy physics experiments, nuclear medicine applications, and industrial systems.

VII. Bibliography

[1] Wang R, Qian C, Guo YH, Zhang P, Ma JD. Automatic spectrum recognition

system for charge state analysis in electron cyclotron resonance ion sources. *Nuclear Science and Techniques*, 34(11): 178 (2023).

[2] Yuan C, Zhang W, Ma T, Yue M, Wang PP. Design and implementation of accelerator control monitoring system. *Nuclear Science and Techniques*, 34(4): 56 (2023).

[3] Zhao K, Chen L, Lv N, Zhou LD, He SY, Ruan JL, Wang H, Ouyang XP. Comprehensive study of pulse shape discrimination in a Ga-doped zinc oxide scintillating detector. *Nuclear Science and Techniques*, 36(3): 37 (2025).

[4] Gu YL, Yang F, Guo YY, Yan Z, Huang AJ, Hou J. Insights into the effects of oxygen content regulation on the microstructure and mechanical properties of in situ ODS 304L stainless steel processed by laser powder bed fusion. *Nuclear Science and Techniques*, 36(6): 1–18 (2025).

[5] Liu WP, Guo B, An Z, Cui BQ, Fang X, Fu CB, Gao BS, He JJ, Jiang YC, Lv C, et al. Recent progress in nuclear astrophysics research and its astrophysical implications at the China Institute of Atomic Energy. *Nuclear Science and Techniques*, 35(12): 217 (2024).

[6] Zhou LY, Zha H, Shi JR, Qiu JQ, Wang CJ, Han YS, Chen HB. A non-invasive diagnostic method of cavity detuning based on a convolutional neural network. *Nuclear Science and Techniques*, 33(7): 94 (2022).

[7] Zhang H, Li JZ, Hou R, An S, Xu SQ, Liu YC, Zhang PJ, Song J, Zhang YL. Design and development of an ACCT for the Shanghai advanced proton therapy facility. *Nuclear Science and Techniques*, 33(10): 126 (2022).

[8] Deng C, Wang SJ, Hu Q, Tang YH, Li PC, Xie B, Yang JB, Tuo XG, Wang QB. Deep learning-based compressed sampling reconstruction algorithm for digitizing intensive neutron ToF signals. *Nuclear Science and Techniques*, 36(7): 112 (2025).

[9] Fu QB, Zhang Y, Wang YC, Huang TC, Zhu HY, Deng XW. Systematic analysis and modeling of the FLASH sparing effect as a function of dose and dose rate. *Nuclear Science and Techniques*, 35(10): 171 (2024).

[10] Chen JH, Guo FK, Ma YG, Shen CP, Shou QY, Wang Q, Wu JJ, Zou BS. Production of exotic hadrons in pp and nuclear collisions. *Nuclear Science and Techniques*, 36(4): 55 (2025).

[11] Yu YB, Liu GF, Xu W, Li C, Li WM, Xuan K. Research on tune feedback of the Hefei Light Source II based on machine learning. *Nuclear Science and Techniques*, 33(3): 28 (2022).

[12] Zhang S, Meng C, Zhou ZS, He X, Zhang JR, Iqbal M, Zhang ZD, Bai BW, Chi YL. Design of 10 MeV electron linear accelerator for space environment simulation. *Nuclear Science and Techniques*, 35(10): 177 (2024).

[13] Fang WC, Huang XX, Tan JH, Wang CP, Xiao CC, Lu YX, Zhang Y, Yang YQ, Xu YM, Gong HY, et al. Proton linac-based therapy facility for ultra-

high dose rate (FLASH) treatment. *Nuclear Science and Techniques*, 32(4): 34 (2021).

[14] Wang JC, Ren J, Jiang W, Ruan XC, Liu YY, Yang HL, Xu KZ, Pan XY, Sun Q, Bao J, et al. In-beam gamma rays of CSNS Back-n characterized by black resonance filter. *Nuclear Science and Techniques*, 35(10): 164 (2024).

[15] Qin B, Liu X, Chen QS, Li D, Han WJ, Tan P, Zhang ZQ, Zhou C, Chen AT, Liao YC, et al. Design and development of the beamline for a proton therapy system. *Nuclear Science and Techniques*, 32(12): 138 (2021).

[16] Wang SY, Song YT, Feng HS, Li S, Cao HL, Zhang J, Huang OW, Li Z. Design of a personnel safety interlock system for proton therapy. *Nuclear Science and Techniques*, 32(4): 39 (2021).

[17] Liu Y, Zhu TF, Luo Z, Ouyang XP. 3D robust anisotropic diffusion filtering algorithm for sparse view neutron computed tomography 3D image reconstruction. *Nuclear Science and Techniques*, 35(3): 50 (2024).

[18] Gu YL, Yang F, Guo YY, Yan Z, Huang AJ, Hou J. Insights into the effects of oxygen content regulation on the microstructure and mechanical properties of in situ ODS 304 L stainless steel processed by laser powder bed fusion. *Nuclear Science and Techniques*, 36(6): 1–18 (2025).

[19] Zheng PS, Shi FR, Dutt S, Zhang YL, Zhang YS, Wang W, Li GS, Wang SC, Yang HR, He JQ, et al. Study of true coincidence summing effects on FEP efficiency of HPGe detectors during decay measurements at HIRFL. *Nuclear Science and Techniques*, 36(5): 74 (2025).

[20] Yang LJ, Peng JY, Qiu F, He Y, Ma JY, Xue ZH, Jiang TC, Zhu ZL, Chen Q, Xu CY, et al. Classification of superconducting radio-frequency cavity faults of CAFE2 using machine learning. *Nuclear Science and Techniques*, 36(6): 104 (2025).

[21] Mingtao K, Yuliang Z, Dapeng J, Yongcheng H, Mingchuan Z, Peng Z, Xuan W, Fengqin G, Lin W. The machine protection system for CSNS. *Radiation Detection Technology and Methods*, 5(2): 273–279 (2021).

[22] Jin H, Choi Y. Development of fast protection system and slow interlock system in the RAON accelerator. *Journal of the Korean Physical Society*, 76(7): 601–607 (2020).

[23] Liu S, Wei YX, Lu YR, Wang Z, Han MY, Wei TH, Zheng PF. Design of PLC and EPICS based control system for a deuteron RFQ. *Journal of Instrumentation*, 17(06): T06002 (2022).

[24] Xia Y, Wang Q, Zhao J, Feng L, Guo E, Yang T, Wang Y, Li F, Guo Z, He Q, et al. Design and implementation of EPICS on the laser accelerator: CLAPA-I control system upgrade. *IEEE Transactions on Nuclear Science*, 71(1): 18–30 (2023).

[25] Nicklaus DJ, Hanlet P, King C, McArthur D, Neswold R. Controls at the Fermilab PIP-II Superconducting Linac. arXiv:2401.15160 (2024).

[26] Tian RX, Wu JX, Li ZX, Gu KW, Su JJ, Ni FF, Wei Y, Xie HM, Li LL, Zhang Y, et al. Design of beam position monitoring interlocking protection system. *Proc. IBIC2024*, pp. 110–113 (JACoW Publishing, Geneva, Switzerland).

[27] Colinet A, Romera I, Bolton S, Guasch-Martinez J, Martin C, Uythoven J, Secondo R. JACOW: Testing aspects of the CERN beam interlock system prior to installation in the accelerator. *JACoW IPAC2024*, THPG59 (2024).

[28] Zhao LL, Yang Z, Guo Y, Zhang J, Chen J, Wang X, Zhang X. Personal Safety Interlock System Based on Siemens Safety PLC. *People*, 6(8): 9–10 (2025).

[29] Jena SS, Shrotriya S, Patel NR, Shiju A, Pande M, Joshi G. Interlock protection and monitoring system for SSA. Technical Report (2024).

[30] Sato KC, Kimura T, Yamada S, Kamikubota N, Yamamoto N. The software-based machine protection system using EPICS in J-PARC MR. In: *ICALEPCS' 19: International Conference on Accelerator and Large Experimental Physics Control Systems*, New York, USA, 05–11 October 2019. JACoW Publishing, pp. 1418–1420 (2020).

Note: Figure translations are in progress. See original paper for figures.