In Sync We Trust: The Power of Synchronous Resets in High-Reliability FPGA Architectures

Dongil Lee [a*], Han Gyu Kim [a]

[a] *Korea Hydro & Nuclear Power Co., Ltd Central Research Institute.*

**Corresponding author: diturtle@khnp.co.kr*

1. INTRODUCTION

Instrumentation and Control (I&C) systems in nuclear power plants require the highest levels of safety and reliability. Recently, the application of Field-Programmable Gate Arrays (FPGAs) has accelerated due to facility upgrades in operating plants, obsolescence of legacy components, and the suitability of FPGAs for high-mix, low-volume production in new digital systems. Unlike software-based sequential execution, the hardware-based parallel processing nature of FPGAs provides critical advantages in ensuring deterministic performance and rapid response times. However, in hardware logic design, the selection of the reset topology goes beyond simple initialization—it profoundly impacts the reliability of the entire system. Focusing on the metastability perspective in asynchronous versus synchronous reset signal processing, this paper comparatively analyzes reset design methodologies between the commercial and nuclear industries. Furthermore, it proposes essential reset strategies for the nuclear sector based on NUREG/CR-7006 and IEC 62566 technical standards [1,2].

2. Comparison of Reset Design Perspectives: Commercial vs. Nuclear Industries

Resets, the core control signals that initialize a system or recover it to a safe state during abnormal conditions, are broadly classified into asynchronous and synchronous resets depending on the design approach [2].

Table 1. Comparison of Reset Design Perspectives

	Commercial Industry	Nuclear Industry
Reset Preference	Prefers asynchronous resets. Minimizing logic delay.	Synchronous resets or strictly controlled synchronized asynchronous resets
Error Tolerance	Malfunctions caused by glitches can be accommodated via system reboots or watchdogs.	Metastability during reset de-assertion poses a potential safety risk

3. Metastability and Its Correlation with Resets

Metastability occurs when the input data of a flip-flop changes during the unstable window(setup/hold time) near the active edge of the clock. Consequently, the output state fails to stabilize to a valid '0' or '1' logic level, hovering instead at an indeterminate voltage level for an unpredictable duration [4].

When utilizing a purely asynchronous reset, the assertion of the reset signal immediately initializes the flip-flops regardless of the clock, causing no logical issues. The critical vulnerability arises during the de-assertion of the reset. If the release of an asynchronous reset occurs too close to the clock's active edge, it violates the recovery and removal times. In large systems, this causes some flip-flops to resume normal operation while others miss the clock edge and remain in the reset state. This asynchronous state inconsistency directly induces metastability, resulting in broken Finite State Machines (FSMs) or fatal system malfunctions.

4. Design Requirements from Technical Standards (NUREG/CR-7006 & IEC 62566)

International standards and regulatory guidelines for safety-grade nuclear FPGA designs strongly emphasize synchronous design practices to achieve high reliability.

4.1. NUREG/CR-7006

The latest digital I&C review guidelines from the U.S. Nuclear Regulatory Commission (NRC), including NUREG/CR-7006, mandate the adoption of Synchronous Design Practices for complex logic circuits. The guidelines warn that asynchronous resets are highly susceptible to external Electromagnetic Interference / Radio Frequency Interference (EMI/RFI) noise and radiation environments, increasing the probability of spurious resets caused by glitches. Therefore, they demand a robust reset architecture where all timing paths are deterministic and clearly verifiable through Static Timing Analysis (STA).

4.2. IEC 62566

IEC 62566 from the International Electrotechnical Commission is a foundational standard for developing HDL-Programmed Devices (HPDs) that perform Category A safety functions in nuclear power plants. This standard explicitly requires the elimination of potential defects starting from the HDL coding phase and mandates STA with 100% timing coverage. The inherent uncertainty of the de-assertion timing in purely asynchronous resets makes it nearly impossible to fully verify via STA. Consequently, complying with IEC 62566 necessitates implementing verifiable synchronous reset circuits that proactively eliminate race conditions and metastability.

Table 2. Comparison of Reset Design by Standard

	IEC 62566	NUREG/CR-7006
Core Philosophy	Determinism and ease of verification.	Defense against physical faults.
Preferred Reset	Pure Synchronous Reset.	Asynchronous Assertion, Synchronous De-assertion
Risks/Drawbacks	A critical dependency: if the clock stops, the reset cannot be executed.	Special time constraints and rigorous verification are required.

5. Optimal Reset Strategy for the Nuclear Industry

Synthesizing the regulatory requirements and the critical need to prevent metastability, the following reset topologies are recommended for nuclear FPGA designs.

5.1. Fully Synchronous Reset

The reset is asserted and de-asserted synchronously with the clock. It naturally filters out external glitch noise unless the glitch perfectly aligns with the clock edge, offering high immunity and permitting flawless STA verification. However, due to the dependency that the reset will not function if the clock signal fails, it must be implemented alongside an independent fail-safe design.

5.2. Asynchronous Assertion, Synchronous De-assertion (Limited Application)

While this architecture enables immediate hardware initialization regardless of clock status, it is more appropriately considered for control systems where availability and uninterrupted recovery are prioritized, rather than for top-tier safety systems (e.g., PPS, ESF-CCS). Architecturally, the assertion is executed asynchronously to ensure immediate responsiveness, but when the reset signal is de-asserted, it must pass through a Reset Synchronizer circuit (a chained series of flip-flops) so that it is fully synchronized with the clock before release. This design secures immediate safety responsiveness while fundamentally eliminating the metastability problem. If such a reset is utilized, rigorous verification must prove that the reset architecture is problem-free.

6. Conclusions

In FPGA designs applied to essential nuclear safety systems, the reset is not merely an initialization signal; it is the ultimate safeguard that transitions the system to a safe state in the presence of unforeseen faults. Subtle timing violations during the reset de-assertion process and the resulting metastability can cause asynchronous state inconsistencies within massive systems, leading directly to catastrophic loss of control.

Based on an analysis of regulatory and technical standards such as NUREG/CR-7006 and IEC 62566, the optimal solution for high-reliability FPGA architectures in top-tier safety systems is the standardization of the fully synchronous reset structure. While an asynchronous approach might seem advantageous for handling extremely rare clock failures, such faults are more appropriately mitigated at the system level—using external hardware watchdogs, processor redundancy, and multi-channel designs—rather than introducing persistent glitch vulnerabilities and verification complexities into the FPGA.

However, if a synchronized reset is not used, rigorous verification must prove that the reset architecture works without a single exception. Thus, adopting strictly controlled synchronized resets is deemed the proper way to guarantee nuclear I&C system safety and meet regulatory standards.

REFERENCES

[1] NUREG/CR-7006, Review Guidelines on Software Languages for Use in Nuclear Power Plant Safety Systems. Office of Nuclear Regulatory Research, 2010.

[2] IEC 62566, Nuclear power plants - Instrumentation and control important to safety - Development of HDL-programmed integrated circuits for systems performing category A functions, 2014.

[3] Cummings, C. E., & Mills, D. Synchronous Resets? Asynchronous Resets? I am so confused! How will I ever know which to use? SNUG (Synopsys Users Group) San Jose, 2002

[4] Wakerly, J. F., Digital Design: Principles and Practices (4th ed.). Pearson Prentice Hall. (Focusing on Metastability and Synchronizer Design), 2005