Towards high-integrity redundancy role leasing

Bjarne Johansson*† ,Olof Holmgren*,
Håkan Forsberg†, Thomas Nolte†, Alessandro V. Papadopoulos†
* ABB Process Automation, Process Control Platform, Västerås, Sweden, {bjarne.johansson, olof.holmgren}@se.abb.com
† Mälardalen University, Västerås, Sweden, {hakan.forsberg, thomas.nolte, alessandro.papadopoulos}@mdu.se

***Abstract*—Control systems are often an integral part of automation solutions where high reliability is crucial due to the high cost of downtime. The risk of unplanned downtime is typically reduced with redundant solutions. Additionally, safety-critical automation functions require high-integrity controllers. Today, the prevalent redundancy solution is a standby scheme, where one active primary controller drives the process while a standby backup controller is ready to take over in case of primary failure. This redundant controller pair can consist of high-integrity controllers.**

The automation industry is trending towards Ethernet as the sole communication medium. Our work presents an initial study of a high-integrity realization of a redundancy failure detection mechanism that guarantees only one primary controller, even in the case of network partitioning between the redundant controller pair. The failure detection is a lease-based function that leases the primary role from a central lease broker. This work discusses a high-integrity realization of the primary redundancy role leasing. We deduce and present the high-integrity-related requirements and a high-level design as an initial step towards a high-integrity realization of the redundancy role leasing.

I. INTRODUCTION

High reliability is fundamental for controllers used in domains where downtime is costly or highly undesirable for other reasons. Spatial redundancy with hardware duplication forming a standby solution is the conventional redundancy pattern in the industrial controller context [1]. Standby redundancy implies that one controller is the active primary controller, and one is the passive backup controller. Only the active primary controller provides output and controls the process. The backup remains on standby and assumes the primary role when the original primary fails. Hence, a standby redundancy solution requires a state replication and failure detection mechanism; the backup must be able to detect that the primary has failed so that it can assume the primary role with the latest state of the primary. This work addresses the failure detection.

High-integrity certified controllers are typically used for safety-critical automation solutions, where failures could be catastrophic. Control system vendors usually certify against IEC 61508 [2] to a specific Safety Integrity Level (SIL) to prove that the controllers are sufficiently unlikely to fail dangerously. These certified controllers can be used to automate safety-critical functions. Safety-critical in the sense that function failure could be harmful to humans and/or the environment.

This work is funded by The Knowledge Foundation (KKS), project ARRAY, by the Swedish Research Council (VR), project PSI, and by The Swedish Foundation for Strategic Research (SSF).

Our work is the first step towards a high-integrity failure detection for controller redundancy based on the leasing version of the Network Reference Point Failure Detection (NRP FD) [3], [4]. NRP FD is a failure detection mechanism that ensures that only one of the controllers, commonly denoted as Distributed Controller Nodes (DCN), in a redundant pair, is the primary, as shown in Fig. 1a. One DCN is the primary, even in the case of network partitioning, as shown in Fig. 1b.

The NRP FD approach we base this work on is the leasing version [4], summarized next. The redundancy role is leased from the NRP, exemplified in Fig. 1c. The NRP will only grant the lease of the primary role to one DCN in the redundant pair. The primary and backup will try to obtain the lease; the backup will get the lease if the primary fails to renew it. The current lease gets precedence and can renew its lease before it expires, while the backup will only get the lease if the current lease has expired. The NRP is a device external to the redundant DCN pair, preferably a network switch, since switches are needed to create the network.

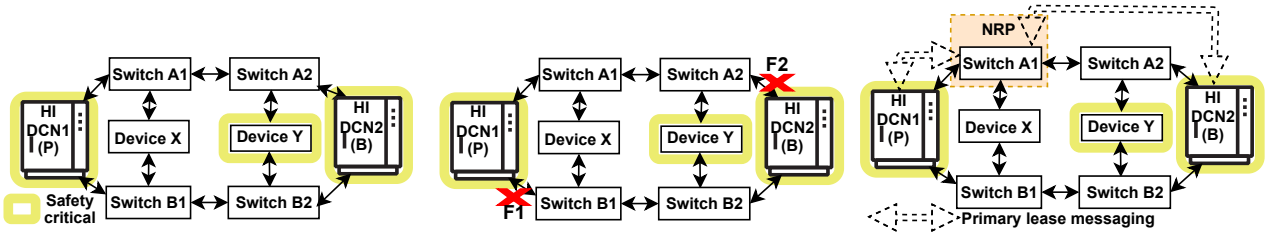
In this work, we take the first investigative steps toward realizing a high-integrity version of the redundancy role leasing. We attack the problem using a functional analysis to deduce requirements that impact the design. With these requirements in mind, we present a design that meets them.

Our contribution is the initial analysis leading to the proposed design for a high-integrity version of redundancy role leasing.

The paper's outline is as follows: Sec. II presents the background and related work, and Sec. III presents the analysis that deduces the high-integrity design requirements. Sec. IV describes the design, and Sec. V concludes the paper with a discussion and future work.

II. BACKGROUND AND RELATED WORK

Today, a redundant DCN pair is typically connected with point-to-point, often redundant, links. Breakage of all these links, causing a pair partitioning, can lead to a non-deterministic dual primary situation [5], [6]. The automation domain and control systems are transitioning into a network-centric era, where the network, rather than the controller, is the system's center [7]. In this context, NRP FD mitigates the dual primary problem as a consequence of network partitioning [3]. NRP FD prioritizes consistency over availability when a tradeoff between the two is necessary, as described by



(a) OK. (b) Network failure causing partitioning. (c) NRP FD with leasing.

Fig. 1: High-integrity network-based remote redundancy deployment. Fig. (a) shows a fault free network. Fig. (b) shows a failure situation where the redundant pair cannot communicate. Fig. (c) shows the NRP and the leasing of the primary role.

the Consistency, Availability, and Partition Tolerance (CAP) theorem [8]. A strengthened version of NRP FD proposes that the redundancy role is leased from the NRP, and this is the algorithm on which we base this work [4].

The research related to reliability and redundancy is vast, ranging from dynamic allocation of resources to the mitigation of single points of failure in voting redundancy solutions, as well as comprehensive surveys [9], [10], [11]. And these are just a few examples.

Regarding safety, communication, and Ethernet in particular, Huang et al. investigate a safe communication approach on top of Ethernet [12]. Peserico et al. discuss and explore adaptations of protocols for safety-critical applications to the Industrial Internet of Things (IIoT) and wireless communication methods [13].

Our work differs from those mentioned above as we address a high-integrity implementation of a specific function: the redundancy role leasing mechanism.

III. Architecture and Hazard Analysis

A. Architecture

The high-level architecture is shown in Fig. 1c. The High-Integrity (HI) DCN internally features a dual-channel architecture with diverse hardware and software. In this paper, we focus solely on the interaction with the NRP and the leasing of the primary role, specifically the Lessee-Lessor interaction, as shown in Fig. 2.

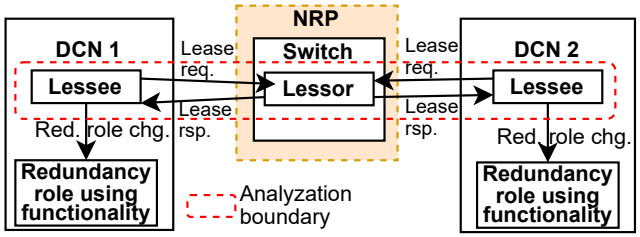


Fig. 2: The system components and analyzation boundaries.

B. Function analysis

As described in Sec. I, the system function to realize and analyze is the primary role leasing. At all times, there should be at most one primary, i.e. only one primary role lease active. The analysis aims to identify the failure modes that lead to Dual Primary and deduce HI Function Requirements (HIFR) to reduce the probability of the function failing so that a Dual Primary situation can occur.

Fig. 2 shows the system and analyzation boundaries. This work covers the components within the boundaries; wider analysis boundaries are future work. This analysis includes the Lessee, the Lessor, and their message exchange. The Lessee is responsible for providing the current redundancy role to the other functionality on the DCN; however, that is left for future work.

The Lessee and Lessor each have specific subfunctions. The Lessee is responsible for continuously renewing or attempting to obtain the primary role lease from the Lessor. As mentioned, the Lessee should inform the other components/sub-functions in the DCN about the redundancy role, though this is outside the scope of this work.

The Lessor should grant leases to the requesting Lessee so that only one Lessee obtains the primary role.

The function involving both the Lessee and the Lessor is the Primary Role Lease, divided into the following subfunctions: (i) Lease request, (ii) Lessor brokerage, and (iii) Lessee state management.

Table I presents the analysis. No Primary indicates a safe state, which is undesirable but a deterministic controlled state. Dual Primary, on the other hand, might lead to a non-deterministic system state, which is unwanted in a safety-critical context. The output from the analysis is the HIFR listed in Table I. We address those in the following section, Sec. IV.

IV. Design

The analysis in Sec. III, summarized in Table I, indicates that a fail-silent function is a controlled and deterministic state; it leads to No Primary. What is critical is the integrity of the function that can cause Dual Primary. Specifically, we need the design to cover these three requirements: (HIFR 1) Error detection means in messaging, (HIFR 2) Diverse SW and HW and cross comparison before grant, and (HIFR 3) Diverse hardware and software, with comparison on expiration.

The design decisions for each requirement are further described in the subsections below.

TABLE I: Sub-functions, failure mode, and corresponding high-integrity function requirements.

Subfunction	Failure mode	Consequence	HI Function Req. (HIFR)
Lease request	Request not reaching Lessor.	No primary.	-
	Request acknowledgement not reaching Lessee.	No primary.	-
	Lease rejection message corrupted and interpreted as lease grant.	Dual primary.	HIFR 1: Error detection mean in messaging.
Lessor brokerage	False negative grant - lease request wrongly rejected.	No primary.	-
	False positive grant - multiple leases granted to different lessors.	Dual primary.	HIFR 2: Diverse SW and HW and cross comparison before grant.
Lessee state management.	Granted lease wrongly believed to be not granted.	No primary.	-
	Not granted lease wrongly believed to be granted.	Dual primary.	HIFR 3: Diverse HW and SW, with compare on expiration.

A. HIFR 1: Error detection mean in messaging

The communication link used is wired Gigabit Ethernet according to IEEE 802.3ab, which specifies a Bit Error Rate (BER) smaller than 10^{-10}. One full-size frame per millisecond and a BER of 10^{-10} result in an hourly frame loss of 4.3 frames/hour. Although missing frames do not lead to a Dual Primary situation, missing frames decrease reliability. Hence, a retransmission mechanism is needed.

What is critical is wrongfully interpreting a rejected lease as accepted due to message corruption. The Ethernet frame has a Frame Check Sequence (FCS) for integrity purposes, capable of detecting faults within a Hamming distance of four. The Mean Time To False Packet Acceptance (MTTFPA) is 60 billion years [14]. Adding an additional Cyclic Redundancy Check (CRC) to the exchanged message will further decrease the probability.

Fig. 3 gives an example of message layout. One channel populates the data fields, and the other calculates and populates the CRC independently. Another alternative would be to use a safety protocol, such as PROFISafe [15].

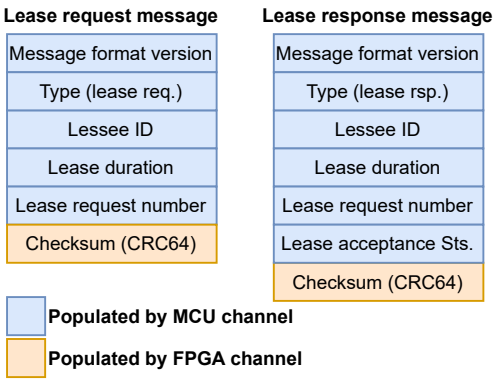


Fig. 3: Example message layout. Each field is eight-byte - optimization is future work.

B. HIFR 2 - Diverse SW and HW and cross comparison before grant

This requirement addresses the issue of wrongfully granting a lease due to faults in the lease brokerage functionality in the Lessor. Such faults could lead to double-granted leases and a Dual Primary situation as a consequence.

The first such potential fault is a logic error in the software. Therefore, the lease brokerage implementation should be diverse, preferably with different algorithms. The most suitable approach remains to be determined. As exemplified below, one way could be to implement the leasing logic in two diverse channels: an FPGA channel and an MCU channel.

The second concern is that the perception of time is critical. If the Lessor's perception of elapsed time is wrong, it could grant a new lease too early, leading to a Dual Primary situation. Hence, two independent clock sources are needed.

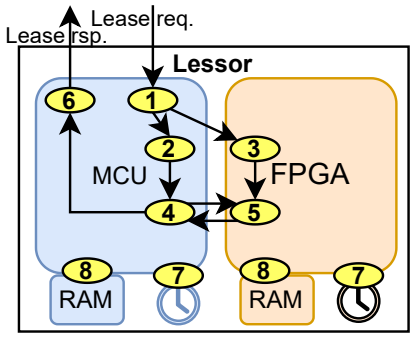
Fig. 4a exemplifies the internal design and data flow, starting with the reception of a lease request (1). The MCU distributes the lease request message to the FPGA channel, and both channels process the request to determine if a lease should be granted (2) (3). The channels cross-compare the results (4) (5). If the results are equal and both channels conclude that a lease should be granted, a positive response is sent, and information about the granted lease is stored in each channel's RAM. Likewise, if both channels agree that the lease request should be rejected, a negative response is sent (6).

Finally, in cases where the channels do not agree, no response is sent. In other words, the Lessor implements fail-silent semantics.

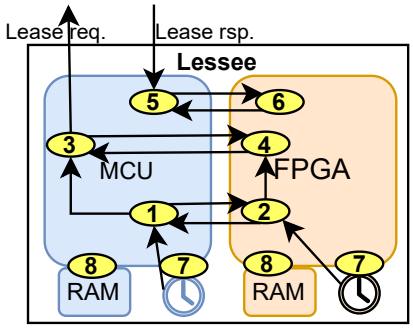
Additionally, various diagnostics and error correction mechanisms are implemented. The RAM memory has error-correcting capabilities and is tested regularly (8). Also, since the notion of elapsed time is important, self-diagnostics and cross-comparisons of elapsed time are conducted regularly (7).

C. HIFR 3: Diverse HW and SW, with compare on expiration

HIFR 3 addresses the probability of a Lessee's incorrect perception of the lease state. This incorrect state can result



(a) Lessor design and lease handling flow.



(b) Lessee design and lease handling flow.

Fig. 4: Lessor and Lessee internals and lease handling.

from a logic error in the lease handling, undetected corruption of state data, or a wrong perception of elapsed time. The probability of these errors can be reduced with diverse hardware and software. Again, we exemplify the design using a diverse HW and SW architecture consisting of an MCU and an FPGA, as shown in Fig. 4b.

Every millisecond, each channel checks if the lease has expired and cross-compares their results (1) (2). In case of a discrepancy, the Lessee enters a safe state. The safe state for the Lessee ensures that it does not indicate that it is primary, i.e., that it does not have the lease. If the lease has expired or is about to expire (determining the suitable renewal period is future work), the Lessee requests a lease. One channel populates the data field in the message, and the other channel populates the checksum (3) (4). Before sending the request, a check is made to ensure that both channels agree; if not, the safe state is entered. When the reply arrives, both channels check the reply, update the state of the lease accordingly, and cross-compare the result (5) (6).

Each channel stores information about the lease in RAM, which is ECC-protected, and a memory test is conducted at regular intervals (7). As with the Lessor, the perception of time is critical, so both channels continuously cross-compare their perception of elapsed time (8).

V. Conclusion and Future Work

By conducting an analysis of the redundancy role leasing function, we identified requirements impacting the design. Based on these requirements, we presented a dual-channel, high-integrity design. From the requirements and the presentation of the design, we identified critical functionality and challenges in realizing those in a high-integrity fashion. That needs to be addressed in future works.

In addition, there are many more areas for further exploration. For example, one could dive deeper into the hardware design and concretize the implementation further, especially regarding the real-time aspects, since the solution requires accurate time perception of the lease and needs to be performant with minimal impact on the control loop execution. Additionally, finding suitable, diverse algorithms for diverse implementations is an appropriate area for future work.

References

[1] A. Simion and C. Bira, "A review of redundancy in plc-based systems," *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies XI*, vol. 12493, pp. 269–276, 2023.

[2] "IEC 61508 functional safety standard." https://webstore.iec.ch/publication/5515. Accessed: 2024-05-14.

[3] B. Johansson, M. Rågberger, A. V. Papadopoulos, and T. Nolte, "Consistency before availability: Network reference point based failure detection for controller redundancy," in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–8, IEEE, 2023.

[4] B. Johansson, B. Pourvatan, Z. Moezkarimi, A. Papadopoulos, and M. Sirjani, "Formal verification of consistency for systems with redundant controllers," in *6th Workshop on Models for Formal Analysis of Real Systems, MARS 2024, Luxembourg City, 6 April 2024*, vol. 399, pp. 169–191, Open Publishing Association, 2024.

[5] Siemens, "Siemens system manual s7-1500r/h redundant system." https://cache.industry.siemens.com/dl/files/833/109754833/att_965668/v3/s71500rh_manual_en-US_en-US.pdf, 2024. Accessed: 2024-05-16.

[6] PACSys, "Pacsystems™ rx3i hot standby cpu redundancy." https://emerson-mas.my.site.com/communities/en_US/Documentation/PACSystems-Hot-Standby-CPU-Redundancy-Users-Manual, 2023. Accessed: 2024-05-16.

[7] B. Leander, B. Johansson, T. Lindström, O. Holmgren, T. Nolte, and A. V. Papadopoulos, "Dependability and security aspects of network-centric control," in *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–8, 2023.

[8] S. Gilbert and N. Lynch, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services," *Acm Sigact News*, vol. 33, no. 2, pp. 51–59, 2002.

[9] A. Ballesteros, M. Barranco, J. Proenza, L. Almeida, F. Pozo, and P. Palmer-Rodríguez, "An infrastructure for enabling dynamic fault tolerance in highly-reliable adaptive distributed embedded systems based on switched ethernet," *Sensors*, vol. 22, no. 18, p. 7099, 2022.

[10] P. Ulbrich, M. Hoffmann, R. Kapitza, D. Lohmann, W. Schroder-Preikschat, and R. Schmid, "Eliminating single points of failure in software-based redundancy," in *2012 Ninth European Dependable Computing Conference*, pp. 49–60, IEEE, 2012.

[11] E. Djambazova and R. Andreev, "Redundancy management in dependable distributed real-time systems," in *j. Problems of Engineering Cybernetics and Robotics*, vol. 79, pp. 37–54, Prof. Marin Drinov Publishing House of Bulgarian Academy of Sciences, 2023.

[12] Z. Huang, X. Jiang, L. Chen, and D. Fan, "Research on safe communication architecture for real-time ethernet distributed control system," *IEEE Access*, vol. 7, pp. 89821–89832, 2019.

[13] G. Peserico, A. Morato, F. Tramarin, and S. Vitturi, "Functional safety networks and protocols in the industrial internet of things era," *Sensors*, vol. 21, no. 18, p. 6073, 2021.

[14] R. Walker, B. Amrutur, T. Knotts, and R. Dugan, "64b/66b coding update," *Presentation at IEEE*, 2000.

[15] "IEC 61784-3-3:2021 industrial communication networks - profiles - part 3-3: Functional safety fieldbuses." https://webstore.iec.ch/publication/68894. Accessed: 2024-05-23.