

# Analysis of Design Parameters in SIL-4 Safety-Critical Computer

Hamzeh Ahangari, Bilkent University  
Yusuf İbrahim Özkök, ASELSAN Corporation  
Asil Yıldırım, ASELSAN Corporation  
Fatih Say, ASELSAN Corporation  
Funda Atik, Bilkent University  
Ozcan Ozturk, Bilkent University

Key Words: Safety, safety-critical computer system, IEC 61508, Markov modeling.

## SUMMARY & CONCLUSIONS

Nowadays, Safety-critical computers are extensively used in many civil domains like transportation including railways, avionics and automotive. We noticed that in design of some previous works, some critical safety design parameters like failure diagnostic coverage (DC) or common cause failure (CCF) ratio have not been seriously taken into account. Moreover, in some cases safety has not been compared with standard safety levels (IEC-61508 SIL1-SIL4) or even have not met them. Most often, it is not very clear that which part of the system is the Achilles' heel and how design can be improved to reach standard safety levels. Motivated by such design ambiguities, we aim to study the effect of various design parameters on safety in some prevalent safety configurations: 1oo2 and 2oo3. 1oo1 is also used as a reference. By employing Markov modeling, sensitivity of safety to each of the following critical design parameters is analyzed: failure rate of processing element, failure diagnostics coverage, common cause failures and repair rates. This study gives a deeper sense regarding influence of variation in design parameters over safety. Consequently, to meet appropriate safety integrity level, instead of improving some system parts blindly, it will be possible to make an informed decision on more relevant parameters.

## 1 INTRODUCTION

Nowadays, safety-critical computers are obligatory constituent of many electronic systems that deal with human life safety. Several areas of transportation industry like railways, avionics and automotive increasingly use such systems. In this domain, safe microcontrollers with limited processing capabilities are available in the market for mostly control purposes. However, as systems become more and more complex and versatile, having safe processors with intensive processing capabilities becomes an essential need. Although there are not any generic safe processors that can potentially answer this eminent need, a computing platform

can be architected safety in mind. To design a computer for safety-critical applications, industrial safety levels (e.g. European safety standard: IEC 61508, Table 1), must be achieved through improvement in numerous aspects of a general system.

Table 1: Safety levels according to IEC 61508 standard for high demand/continuous systems (PFH: probability of dangerous failure per hour, SIL: Safety Integrity Level).

$10^{-9}$	$\leq$	PFH of SIL4	$<$	$10^{-8}$
$10^{-8}$	$\leq$	PFH of SIL3	$<$	$10^{-7}$
$10^{-7}$	$\leq$	PFH of SIL2	$<$	$10^{-6}$
$10^{-6}$	$\leq$	PFH of SIL1	$<$	$10^{-5}$

Reliability of electronic components is the most obvious factor needed to be satisfied for building a robust system. Besides, clever system design by means of available electronic components is as important as the quality of components. Even with reliable and robust parts, safety goals may not be achieved without safety aware design process.

Prevalent design issues like perfect printed circuit board, EMC/EMI, isolation, power circuitry, fail rates of equipment etc., are examples of common quality considerations. However, in critical systems, in addition to these, some other less obvious issues have to be observed. The ratio of Common Cause Failures (CCFs), meaning the ratio of concurrent failure rate over total failure rate, has great impact on safety. The percentage of failures the system is able to detect by means of fault detection techniques has also a direct effect on safety. This is because undetected failures are potential dangers. A usually ignored factor is the frequency and the quality of system maintenance. How often and how comprehensive the system is tested (automatically or by technical personnel) to repair or replace the impaired components, can guarantee the safety by removing transient failures or refreshing worn-out parts.

As safety is a very wide subject, the main objective of this paper is to investigate the sensitivity of system safety to some

crucial design parameters. Three widespread configurations; 1oo1, 1oo2 and 2oo3; with known parameter values, are assumed as base systems. For these systems, it will be illustrated that how adjustment of any individual parameter, can contribute to safety promotion of base systems.

This paper is organized as follows: In Section II, some of the recent works are reviewed and our motivation is expressed. In Section III, definition and modeling for considered design parameters are described. In section IV, base systems and their Markov modeling are proposed. In Section V, simulation results are discussed, and finally conclusion is given in Section VI.

## 2 RELATED WORKS AND MOTIVATION

During design, concentrating on multiple aspects of design altogether for the purpose of improvement can be a complicated and difficult task. Normally, if the prototype design does not meet the requirements, it is rational to find the system's bottleneck and focus on it. In safety-related designs, by knowing the share each parameter provides to safety, the designer can decide where to put more effort to improve the outcome.

In [3], two dual-duplex and TMR synchronous systems are built using MIL-SPEC electronic parts. The effect of CCFs are not assessed there, and the effect of diagnostic coverage is not assessed for TMR system. Besides, the achieved safety level is not compared to standard levels. In microcontroller-based SIL4 software voter [1], SIL4 level is claimed to be obtained. Nevertheless, neither failure coverage nor CCFs are assessed in sufficient details. In safe computer for a train [4], not only is the safety level not compared to standard levels, but neither CCFs nor diagnostic coverage is discussed.

In the above projects, a lack of consideration of some the critical safety parameters or lack of comparison the system safety with standard safety assessment levels makes them incomplete to be considered for real safety-critical applications. This stimulated us to have an analysis over a few safety architectures also used in above works. By showing the sensitivity of safety to each of the ignored parameters, we hope to inform our readers. This can help practitioners to select most appropriate parameter to be improved to achieve safety goal. Depending on the project constraints, the most appropriate parameter can be translated to the one that leads to cheapest, fastest or easiest system modification.

## 3 SYSTEM'S SAFETY PARAMETERS

In this section, we review the definition and modeling of main parameters that we want to investigate their effect on safety.

### 3.1 Processing Element Failure Rate

A safety-critical computer system is composed of one or more redundant processing elements (PEs, also called channels), connected to each other by communication links. Generally, there is no extraordinary requirement regarding reliability of PEs. It means a PE is a regular processing module, built from available Commercial-Off-The-Shelf

(COTS) electronic parts including microprocessor, memory, power circuitry, etc. Components are not necessarily the kind that are specifically designed for reliability purpose. In this study, overall, we take a PE as a black box and assume a single failure rate  $\lambda_{PE}$  or simply  $\lambda$  for it.

### 3.2 Common Cause Failure (CCF)

According to IEC 61508-4 standard [7], Common Cause Failure (CCF, or dependent failure) is defined as concurrent occurrence of failures in multiple channels caused by one or more events leading to system failure. The  $\beta$  factor represents the fraction of dangerous failures that is due to CCF. Typically for a duplicated system  $\beta$  value is around a few percent. In safety standards, two  $\beta$  values are defined for detected and undetected failures ( $\beta_D$  and  $\beta$ ), while here we take a single  $\beta$  value for both. By using the extended  $\beta$  modeling and notations from [5], below formulations are supposed to be valid for all systems in this work:

- The 1oo1 configuration: There are no redundant PE, then  $\beta_{1oo1} = 0$ .
- The 1oo2 configuration: As depicted in Figure 1, the  $\beta$  of system is only related to CCFs between two PEs. Then  $\beta_{1oo2} = \beta$ .
- The 2oo3 configuration: As depicted in Figure 1, the overall  $\beta$  of 2oo3 system is related to mutual CCFs plus CCFs shared among three PEs. Note that by definition of  $\beta$ , CCF ratio between every two PEs is taken as  $\beta$  ( $0.3\beta + 0.7\beta = \beta$ ). According to modeling proposed in [5], for a typical 2oo3 system, in total  $\beta_{2oo3} = 2.4\beta$ .

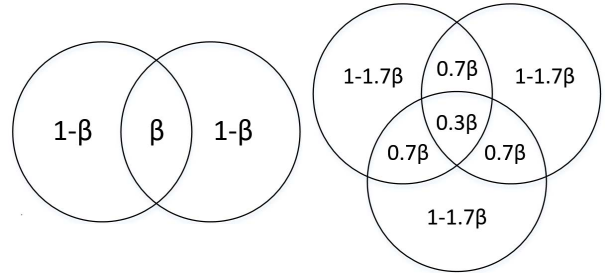


Figure 1:  $\beta$  model for duplicated and triplicated systems [5].

### 3.3 Failure Diagnostics Coverage

According to IEC 61508-4 [7], Diagnostic Coverage (DC) is defined as fraction of dangerous failures detected by automatic online test. Here we assume that failures are detected by two complementary techniques, self-testing and by comparison. Self-testing routines run upon each PE to diagnose occasional failures autonomously. Usually self-testing detects absolute majority of failures, around 90%. Second diagnostic technique is data comparison among redundant PEs to detect rest of failures. Then generally we have:

$$DC = DC_{selftest} + DC_{compare} \approx 1 \quad (1)$$

Borrowing the formulations and notations from [2], the following terms are hypothesized to describe the system's DC rate. According to the referred formulation, the total DC is

reformulated as:

$$DC = DC_{selftest} + (1 - DC_{selftest}) \cdot k \quad (2)$$

The  $k$  is the efficiency of comparison testing. Since the comparison method is more effective against independent failures (none-CCFs), it is sensible to differentiate between DC rate of CCF and independent failures. Therefore, two variants of former formula are [2]:

$$DC^i = DC_{selftest} + (1 - DC_{selftest}) \cdot k^i \quad (3)$$

$$DC^c = DC_{selftest} + (1 - DC_{selftest}) \cdot k^c \quad (4)$$

Here  $k^i$  and  $k^c$  are two constants,  $0 \leq k^i, k^c \leq 1$ , describing the efficiency of comparison for either of two classes of failures. Since comparison is less effective against CCFs, the  $k^c$  value is normally low, for example less than 0.4, while  $k^i$  can be close to one [2].

### 3.4 Test and Repair Rate

According to IEC 61508, for all systems, two forms of test (and repair) are available: proof test and online test. Their rates are denoted by  $\mu_{PT}$  and  $\mu_{OT}$  respectively. The  $\mu_{PT}$  is the frequency of periodic system maintenance by technicians.  $MTTR_{PT}$  (mean time to restoration) is defined as  $1/\mu_{PT}$ , the time interval at which a thorough system check is performed. During proof test, any undetected failures (not detected by online diagnostics) are detected and faulty parts are repaired or replaced then. Typically,  $MTTR_{PT}$  is from a few weeks to a few years.

Online (or automatic) testing is the second form of testing whereby test routines run on each channel periodically. In some situations, as soon as online test routine detects a failure, system is supervised to go into fail-safe mode to avoid dangerous output. After that, system tries to resolve the failure with immediate call for personnel intervention or even without it by doing a self commanded restart. For transient failures, a system restart can be a fast solution, while for persistent failures switching to a backup system provides a fast system recovery. In any case, online repair is supposed to last from a few minutes to a few days.

## 4 BASE SYSTEMS

### 4.1 Assumptions

In this study we make the following assumptions. All channels (PEs) are asynchronous and identical (homogenous). Channels are connected to each other by in-system links, whereby software voting and comparison mechanisms work (Figure 2). In this work, our focus is on processing elements (PEs). I/O ports and communication links are assumed to be black-channel, by which safety is not effected. This assumption can be realized by obeying standards of safe communication over unsafe mediums (see EN-50159). The system is a continuous/high demand safety system, meaning it continuously runs safety functions. Specifically, the safety function of the system is to detect and prevent to outcome any erroneous calculation on PEs.

### 4.2 Default Parameters

For all parameters which we intend to investigate, we assign some default values to define a default base system for every configuration (Table 2). After that, in the simulation phase we sweep each parameter around default value and illustrate how the system safety changes. In this way the sensitivity of system safety, with respect to that parameter and around its default value is revealed.

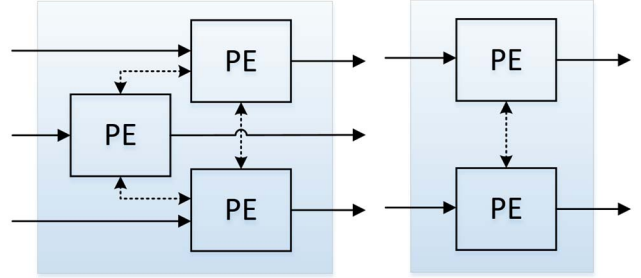


Figure 2: Architecture of triplicated and duplicated systems.

Table 2: Default values for safety parameters

parameter	meaning	default value
$\lambda_{PE}$	PE failure rate	1.0E-05 /hour
$DC_{selftest}$	Diagnostics coverage of self-testing [2]	0.90
$k^i$	Comparison efficiency for independent failures [2]	0.90
$k^c$	Comparison efficiency for CCFs [2]	0.40
$\beta = \beta_D$	CCF ratio between each two PEs	0.02
$\mu_{OT}$	Online repair rate	1 /hour
$\mu_{PT}$	Proof test and repair rate	0.00001 /hour ( $\approx 1$ per year)

### 4.3 Markov Models

In this section, the behavior of all safety systems are modeled by Markov chain. Calculations of RAMS (Reliability, Availability, Maintainability and Safety) measures are according to guidelines suggested in ISA-TR84.00.02[8], IEC 61165[9] and [6]. States are divided into up (operational) and down (none-operational) states. In up states the system is able to correctly run safety functions. Up state is either all-OK initial state or any state with tolerable failures. Down states are those in which system is not able to correctly run safety functions, either intentionally as in the fail-safe state(s) or unintentionally as in unsafe (hazardous) state.

As far as it is detected, a dangerous failure may either be tolerated (like first detected failure in 2oo3 system) or the system goes to fail-safe state. On the other hand, if the failure is left undetected, system may inadvertently tolerate it (like first undetected failure in 1oo2 or 2oo3 systems) or go to

unsafe state.

PFH (probability of dangerous failure per hour) is defined as rate of entering into an unsafe state. For calculation, repairs from down states toward up states are removed, as in the reliability calculation. However, from total failure rate  $\lambda$ , only the hazardous part  $\lambda_H$ , should be considered. The required formulation for decomposing  $\lambda$  into  $\lambda_H$  and  $\lambda_S$  is proposed in [11] ( $P_H$ : Probability of being in hazardous state,  $P_S$ : Probability of being in fail-safe state,  $P_{HS} = P_H + P_S$ ):

$$\lambda_H(t) = \frac{P'_{HS}(t)}{1 - P_{HS}(t)} \cdot \frac{P_H(t)}{P_{HS}(t)} \quad (5)$$

Transition of Markov over time is described below:

$$\mathbf{P}'_{n \times 1}(t) = \mathbf{A}_{n \times n} \cdot \mathbf{P}_{n \times 1}(t) \quad (6)$$

where  $\mathbf{P}(t)$  is vector of state probabilities,  $n$  is number of Markov states and  $\mathbf{A}$  is transition matrix. Abbreviations used in Markov chains are listed in Table 3.

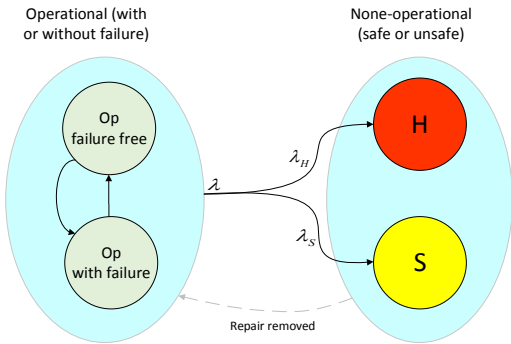


Figure 3: hazardous and safe failure rate.

Table 3: IEC 61508 standard abbreviations

D	Dangerous failure
DD	Dangerous detected failure
DU	Dangerous undetected failure
CCF	Common cause failure (dependent failure)
DC	Diagnostic coverage

**1oo1 Configuration:** Single system has no CCF, so all failures are independent. If failure is detected, next state is safe-failure, otherwise it is unsafe (Figure 4).

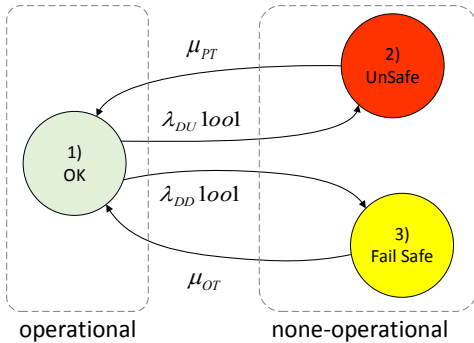


Figure 4: Markov model of 1oo1 system.

Transition terms for 1oo1 system are

$$(k^i = 0, DC^i = DC_{selftest}):$$

$$\lambda_{DD}1oo1 = DC^i \lambda_{PE}$$

$$\lambda_{DU}1oo1 = (1 - DC^i) \lambda_{PE}$$

$$\lambda_D1oo1 = \lambda_{DD}1oo1 + \lambda_{DU}1oo1 = \lambda_{PE}$$

Transition matrix for illustrated Markov is as follows:

$$\mathbf{A} = \begin{bmatrix} -\lambda_D1oo1 & \mu_{PT} & \mu_{OT} \\ \lambda_{DU}1oo1 & -\mu_{PT} & 0 \\ \lambda_{DD}1oo1 & 0 & -\mu_{OT} \end{bmatrix}$$

**1oo2 Configuration:** According to IEC 61508-6, 1oo2 system consists of two parallel channels which can both run the safety function. One dangerous-failure free channel is sufficient to provide system safety. Single DU failure is tolerable (hardware fault tolerance =1), but no DD is tolerated. 1oo2 system has high safety against DU failures and low availability against safe failures (Figure 5). Transition terms for 1oo2 system are:

$$\lambda^i_{DU}1oo2 = 2[(1 - DC^i) \cdot (1 - \beta_{1oo2})] \lambda_{PE}$$

$$\lambda^c_{DU}1oo2 = [(1 - DC^c) \cdot \beta_{1oo2}] \lambda_{PE}$$

$$\lambda_{DD}1oo2 = \lambda^i_{DD}1oo2 + \lambda^c_{DD}1oo2 = (2 \cdot DC^i \cdot (1 - \beta_{1oo2}) + DC^c \cdot \beta_{1oo2}) \lambda_{PE}$$

$$\lambda_D1oo2 = \lambda_{DD}1oo2 + \lambda_{DU}1oo2 = \lambda^i_{DD}1oo2 + \lambda^c_{DD}1oo2 + \lambda^i_{DU}1oo2 + \lambda^c_{DU}1oo2$$

Transition matrix for illustrated Markov is as follows:

$$\mathbf{A} = \begin{bmatrix} -\lambda_D1oo2 & \mu_{PT} & \mu_{OT} & 0 & \mu_{PT} \\ \lambda^i_{DU}1oo2 & -(\lambda_{DD}1oo2 + \lambda_{DU}1oo2 + \mu_{PT}) & 0 & \mu_{OT} & 0 \\ \lambda_{DD}1oo2 & 0 & -\mu_{OT} & 0 & 0 \\ 0 & \lambda_{DD}1oo2 & 0 & -\mu_{OT} & 0 \\ \lambda^c_{DU}1oo2 & \lambda_{DU}1oo1 & 0 & 0 & -\mu_{PT} \end{bmatrix}$$

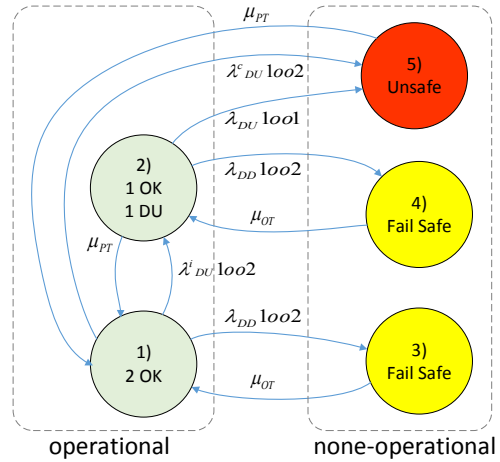


Figure 5: Markov model of 1oo2 system.

**2oo3 Configuration:** Similar to 1oo2, 2oo3 is a safe configuration and is capable of tolerating one DU failure (hardware fault tolerance=1). However, it has also high availability due to being able to tolerate single DD failure, similar to 2oo2 system (Figure 6). 2oo3 has benefits of both 1oo2 and 2oo2 (2oo2 is not described here). Due to more

vulnerable channels (since the vulnerability of the system increases with the increasing number of channels), 2oo3 is not as safe as 1oo2 and not as available as 2oo2.

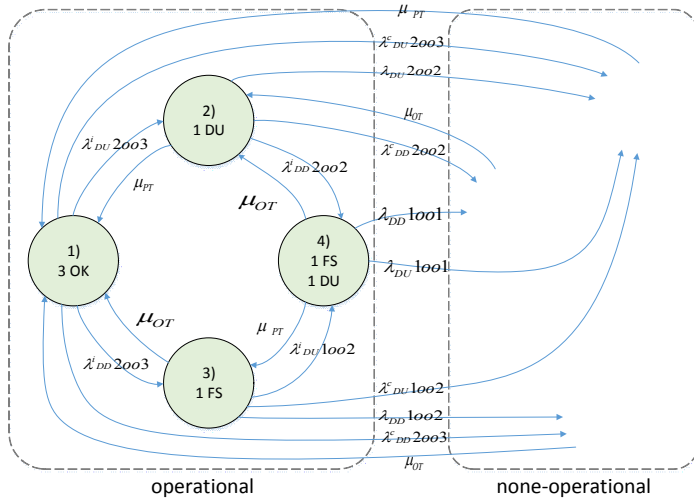


Figure 6: Markov model of 2oo3 system.

Transition terms for 2oo3 system are:

$$\lambda_{DU}^{i2oo3} = 3 \cdot [(1 - DC^i) \cdot (1 - 1.7\beta)] \lambda_{PE}$$

$$\lambda_{DU}^c2oo3 = [(1 - DC^c) 2.4\beta] \lambda_{PE}$$

$$\lambda_{DD}2oo3 = \lambda_{DD}^i2oo3 + \lambda_{DD}^c2oo3 = (3 \cdot DC^i \cdot (1 - 1.7\beta) + DC^c \cdot 2.4\beta) \lambda_{PE}$$

$$\lambda_D2oo3 = \lambda_{DD}2oo3 + \lambda_{DU}2oo3 = \lambda_{DD}^i2oo3 + \lambda_{DD}^c2oo3 + \lambda_{DU}^i2oo3 + \lambda_{DU}^c2oo3$$

Transition matrix for illustrated Markov is as follows:

$$A = \begin{bmatrix} -\lambda_D2oo3 & \mu_{PT} & \mu_{OT} & 0 & 0 & \mu_{OT} & \mu_{PT} \\ \lambda_{DU}^i2oo3 & -(\lambda_D2oo2 + \mu_{PT}) & 0 & \mu_{OT} & \mu_{OT} & 0 & 0 \\ \lambda_{DD}^i2oo3 & 0 & -(\lambda_D1oo2 + \mu_{OT}) & \mu_{PT} & 0 & 0 & 0 \\ 0 & \lambda_{DD}^i2oo2 & \lambda_{DU}^i1oo2 & -(\mu_{OT} + \mu_{PT}) & 0 & 0 & 0 \\ 0 & \lambda_{DD}^c2oo2 & 0 & \lambda_{DD}1oo1 & -\mu_{OT} & 0 & 0 \\ \lambda_{DD}^c2oo3 & 0 & \lambda_{DD}1oo2 & 0 & 0 & -\mu_{OT} & 0 \\ \lambda_{DU}^c2oo3 & \lambda_{DU}2oo2 & \lambda_{DU}1oo2 & \lambda_{DU}1oo1 & 0 & 0 & -\mu_{PT} \end{bmatrix}$$

## 5 SIMULATION RESULTS

In this section, we show the effect of variation in each of aforementioned parameters around defined default value, over system safety (PFH). SIL1-SIL4 safety levels are plotted by horizontal lines to show relative safety position. Mathematically, the following simulations show the partial derivations,  $\partial PFH / \partial p$ , where p is any of individual safety parameters. In initial point 1oo1 system has marginally achieved SIL2, while both 1oo2 and 2oo3 are in SIL3 region. As expected and explained before, generally 1oo2 is a safer configuration than 2oo3 (but 2oo3 is better at availability).

### 5.1 Parameter $\lambda_{PE}$ :

Figure 7 shows how safety is affected by processing element's failure rate.  $\lambda_{PE}$  represents equivalent failure rate

of whole PE, including all of internal components like processor, memory, power units, etc. The linear logarithmic plots with slope of one ( $m=1$ ) describes a linear function, meaning:  $PFH = A \cdot \lambda_{PE}$ . Linearity implies that by just knowing the line slope and a single  $(\lambda_{PE}, PFH)$  point, without solving complicated Markov model every time, system safety can be tuned. It is obvious both from formula and figure that an order of magnitude ( $\times 10$ ) change in  $\lambda_{PE}$  results in one safety level shift.

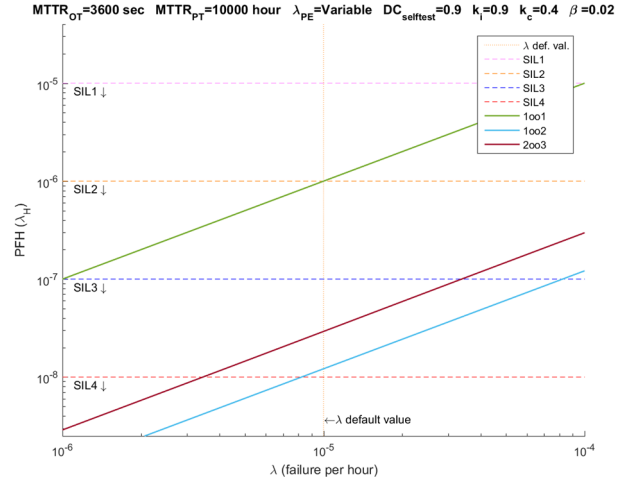


Figure 7: Effect of  $\lambda_{PE}$  variation over safety.

### 5.2 Parameter $\beta$ :

$\beta$  is the indicator of isolation among channels. Figure 8 depicts how safety is affected by  $\beta$ . 1oo1 system is obviously indifferent to  $\beta$ . An observation here is how  $\beta$  does not influence the relative superiority of 1oo2 and 2oo3 systems. It is a well-known fact that CCF is a bottleneck of safety-critical systems and it strongly affects safety. It is observable in the figure that for current parameter values it is extremely difficult to get SIL4 through  $\beta$  improvement. Because by questionnaire  $\beta$  estimation method in IEC 61508-6,  $\beta$  can hardly be estimated below 0.01.

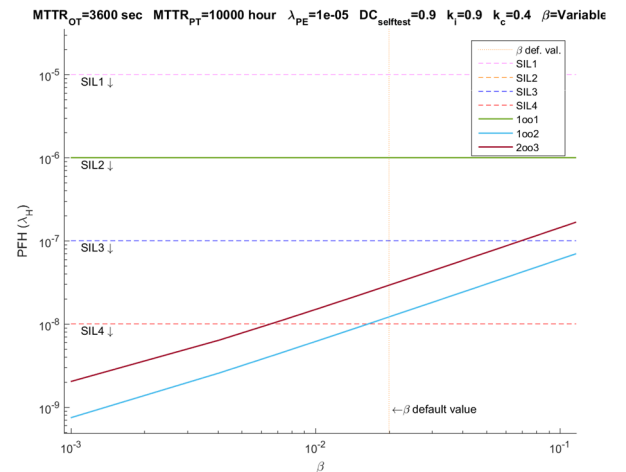


Figure 8: Effect of  $\beta$  variation over safety.

Similar to  $\lambda_{PE}$ , plots are linear in logy-logx plane with slope of one ( $m=1$ ). This describes a linear function in y-x plane, meaning  $PFH = A \cdot \beta$ . Actually, this conclusion is only valid when  $\beta$  is not very closed to zero, seemingly for  $\beta > 0.001$ . The only argument behind this linearity is dominance of CCFs over safety. Linearity makes tuning of safety by  $\beta$  parameter easy without need to solve complicated mathematic systems.

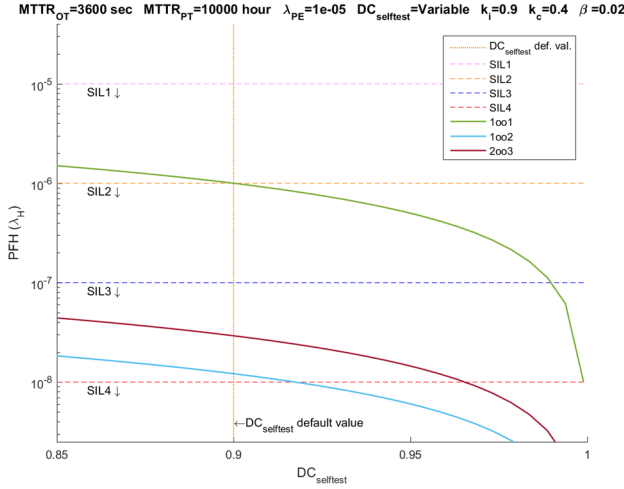


Figure 9: Effect of  $DC_{self\ test}$  variation over safety.

### 5.3 Parameter $DC_{self\ test}$ :

According to formulas in section III.C, self-testing is assumed to be equally effective for both CCFs and non-CCFs. Hence, what is shown in Figure 9 is understandable, where the variation of this parameter does not affect the superiority of systems over each other. In 1oo2 system, for achieving SIL4 the  $DC_{self\ test}$  has to be increased 1-2%, while for 2oo3 it is more difficult, where 6-7% of improvement is required (default of  $DC_{self\ test} = 0.9$ ).

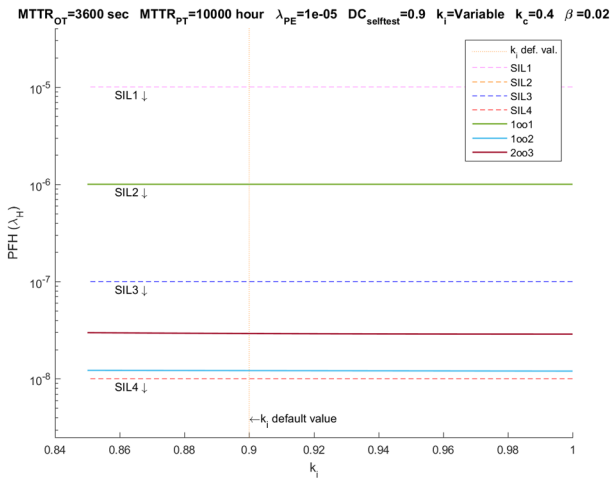


Figure 10: Effect of  $k^i$  variation over safety.

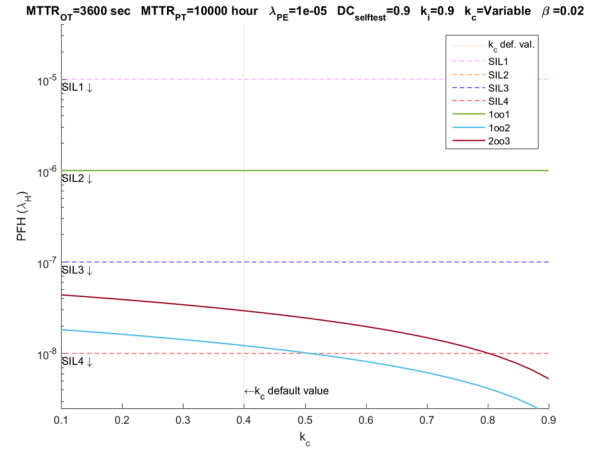


Figure 11: Effect of  $k^c$  variation over safety.

### 5.4 Parameter $k^i$ :

$k^i$  is a constant which specifies the efficiency of comparison among PEs for detecting independent failures. Comparison is expected to be very efficient against non-CCFs (obviously in 1oo1  $k^i=0$ ).

In Figure 10, there is a weird observation stating that  $k^i$  has almost no sensible (very small) influence on safety of 1oo2 and 2oo3 systems. One explanation for this finding, is the absolute dominance of CCFs in both systems. More precisely, any CCF immediately brings both systems into unsafe state. However, two consecutive undetected independent failures have to occur to cause same situation which is far less probable. This is translated to one order of magnitude less influence of independent failures rather than CCFs over safety. As a result, both systems seems to be rather insensitive to  $k^i$ .

One possible incorrect conclusion from this observation is to give up comparison for independent failures. But the fallacy is the point that whether a failure is dependent or not is not decidable at runtime. As we will see,  $k^c$  still has considerable effect on safety. Therefore, comparison cannot be ignored. Still a beneficial conclusion can be taken by some modification to previous one. As we will see,  $k^c$  is usually as low as 0.1-0.4. Therefore, if a relaxed comparison mechanism leads to  $k^i$  value as low as  $k^c$ , it is completely acceptable. Because it is enough to just have a logical value for  $k^c$ .

### 5.5 Parameter $k^c$ :

$k^c$  is a constant which specifies the efficiency of comparison among PEs for detecting CCFs. By definition of CCFs, comparison is not expected to be very efficient against them (again, obviously in 1oo1  $k^c=0$ ). Since in both 1oo2 and 2oo3, CCFs are by far more influential than non-CCFs (because one non-CCF is tolerable in both), then improving  $k^c$  has a sensible effect over safety.

### 5.6 Parameter $\mu_{OT}$ :

Fast repair which is invoked after online failure detection is either employed when system is in fail-safe state or while a detected failure is being tolerated (like in 2oo3, provided that

partial-repair is available). Effect of repair rate in earlier case is obviously zero (remember repairs from down states toward up states are removed in PFH calculation). In later case (only in 2oo3), because the repair does not reduce the number of undetected failures, the effect is negligible. In practice, this parameter is for adjusting availability.

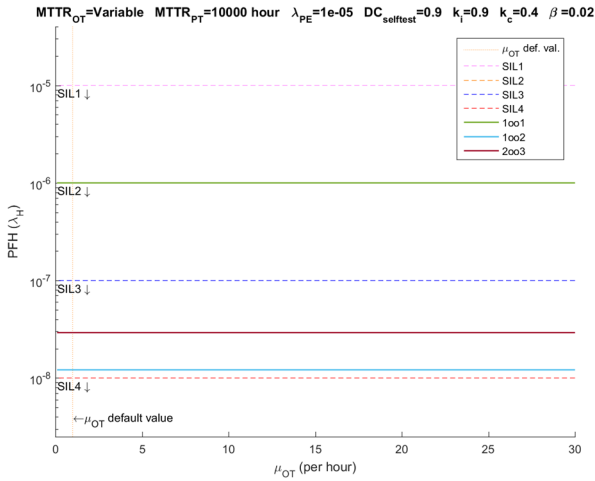


Figure 12: Effect of fast (online) repair rate variation over safety.

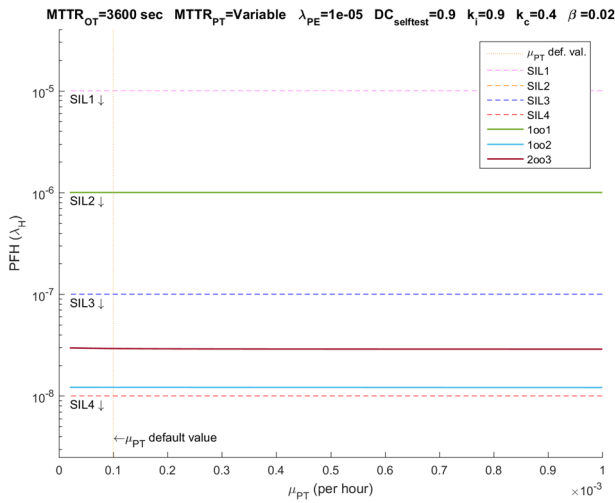


Figure 13: Effect of proof-test rate variation over safety.

### 5.7 Parameter $\mu_{PT}$ :

Proof test and repair occurs periodically in long time intervals to remove undetected failures. It is either employed when system is in unsafe state or while an undetected failure is being tolerated (in both 1oo2 and 2oo3). Its effect in earlier case is obviously zero. Similar to online testing, repairs from down states are removed. In later case, although number of undetected failures are reduced, but due to dominance of CCF rate, such improvement is not sensed in term of safety.

## 6 CONCLUSION

In this work, we analyzed the sensitivity of system safety to design parameters in two basic multi-channel safe

configurations, 1oo2 and 2oo3. 1oo1 system is also held as reference. All systems have been modeled by Markov chain. By this analysis, we aimed to clarify that in a specific design, at which point of SIL safety level the safe configuration supports. Hence it will be clear that how the current safety level differs from the desired safety level (SIL levels) by every individual safety parameter. By knowing this, instead of blindly improving an unsafe system, designers can make an informed decision to select the most appropriate parameter to improve. By simulation we showed there is a linear relation between safety (PFH) and two parameters:  $\lambda_{PE}$  and  $\beta$ , where the later linearity expresses the dominance of CCF over system safety. Moreover, we concluded that only the parameters which have any effect on CCF rate, are appropriate candidates to enhance safety level. These include  $\lambda_{PE}$ ,  $\beta$ ,  $DC_{self\ test}$  and  $k^c$ . On the other hand,  $k^i$  (efficiency of comparison in detecting independent failures) and both types of repair (online and proof test) have no sensible effect on system safety level.

## REFERENCES

- [1] M. Idirin, X. Aizpurua, A. Villaro, J. Legarda and J. Melendez, "Implementation details and safety analysis of a microcontroller-based SIL-4 software voter", IEEE Trans. Ind. Electron., vol. 58, no. 3, pp. 822-829, 2011.
- [2] Hokstad, P. "Probability of failure on demand (PFD)-the formulas of IEC 61508 with focus on the 1oo2d voting." ESREL 2005, Gdansk, Polen (2005).
- [3] H. K. Kim, H. T. Lee and K. S. Lee, "The design and analysis of AVTMR (all voting triple modular redundancy) and dual-duplex system", Reliability Eng. Syst. Safety, vol. 88, no. 3, pp. 291-300, 2005.
- [4] X. Chen, G. Zhou, Y. Yang and H. Huang, "A newly developed safety critical computer system for China metro", IEEE Trans. Intell. Transp. Syst., vol. 14, no. 2, pp. 709-719, 2013.
- [5] P. Hokstad and K. Corneliussen, "Loss of safety assessment and the IEC 61508 standard", Reliability Engineering and System Safety, vol. 83, pp. 111-120, 2004
- [6] D.J. Smith, "Reliability, Maintainability and Risk: Practical Methods for Engineers", Eighth Edition. Butterworth-Heinemann, Elsevier, pp. 436, 2011
- [7] IEC 61508, Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems.
- [8] ISA-TR84.00.02-2002. Safety Instrumented Functions (SIF), Safety Integrity Level (SIL), Evaluation Techniques. Instrumentation Society of America (ISA), 2002.
- [9] IEC 61165, Application of Markov techniques, International Electrotechnical Commission, 2006.
- [10] "PD CLC/TR 50126-2:2007." Railway Application: The Specification and Demonstration of Dependability-Reliability, Availability, Maintainability and Safety (RAMS)Part 2.
- [11] Rástočný, Karol, and Juraj Ilavský. "Quantification of the

safety level of a safety-critical control system." (2010).

### BIOGRAPHIES

Hamzeh Ahangari  
Department of Computer Engineering,  
Bilkent University, Main Campus  
06800 Bilkent, Ankara, Turkey.

e-mail: [hamzeh@bilkent.edu.tr](mailto:hamzeh@bilkent.edu.tr)

Hamzeh Ahangari received BSc degree in Computer Hardware from Sharif University of Technology, Iran and MSc degree in Computer Architecture from University of Tehran, Iran. He is currently PhD student in computer engineering at Bilkent University, Turkey. His research interests are reliability, safety, reconfigurable architectures and high performance computing.

Yusuf İbrahim Özkök,  
ASELSAN Corporation  
P.K. 1, 06172  
Yenimahalle / Ankara, Turkey

e-mail: [yozkok@aselsan.com.tr](mailto:yozkok@aselsan.com.tr)

Yusuf İbrahim ÖZKÖK is employed as Lead Design Engineer at ASELSAN Defense System Technologies Division. He has been involving design of mission critical and safety critical embedded systems for about 15 years. He has BSc degree from Istanbul Technical University on Electronical and Communication Engineering and MSc degree on EE from METU.

Asil YILDIRIM  
ASELSAN Corporation  
P.K. 1, 06172  
Yenimahalle / Ankara, Turkey

e-mail: [asily@aselsan.com.tr](mailto:asily@aselsan.com.tr)

Asil YILDIRIM is a Senior Software Engineer at ASELSAN Defense System Technologies Division. He is involved in safety critical embedded systems as embedded software engineer. He has BS and MS degrees in Electrical and Electronics Engineering from Middle East Technical University.

Fatih Say  
ASELSAN Corporation  
P.K. 1, 06172  
Yenimahalle / Ankara, Turkey

e-mail: [fsay@aselsan.com.tr](mailto:fsay@aselsan.com.tr)

Fatih Say is the Manager of Computer Systems Department at Aselsan Inc. He holds a BS, MS and Doctorate of Philosophy in Electrical Electronics Engineering from Middle East Technical University, Turkey. His current research interests include mission critical embedded computer systems and high performance computing systems.

Funda Atik  
Department of Computer Engineering  
Bilkent University, Main Campus  
06800 Bilkent, Ankara, Turkey.

e-mail: [funda.atik@bilkent.edu.tr](mailto:funda.atik@bilkent.edu.tr)

Funda Atik was born in 1991 in Tekirdag, Turkey. In 2014, she received the B.S. degree in the field of Computer Engineering from Bilkent University, Turkey. She is currently M.S. student at Bilkent University and her supervisor is Dr. Ozcan Ozturk. Her research interests include parallel computing, GPUs and accelerators, and computer architecture.

Prof. Ozcan Ozturk  
Department of Computer Engineering  
Bilkent University, Main Campus  
06800 Bilkent, Ankara, Turkey.

e-mail: [ozturk@cs.bilkent.edu](mailto:ozturk@cs.bilkent.edu)

Ozcan Ozturk has been on the faculty at Bilkent since 2008 where he currently is an Associate Professor in the Department of Computer Engineering. His research interests are in the areas of cloud computing, GPU computing, manycore accelerators, on-chip multiprocessing, computer architecture, heterogeneous architectures, and compiler optimizations. Prior to joining Bilkent, he worked at Intel, Marvell, and NEC.