Leveraging Lessons Learned from Automotive Design Reliability and Functional Safety for Aerospace and Defense Applications

Meirav Nitzan
PM Director
Synopsys, Inc
Mountain View, CA 94043
Email: meirav@synopsys.com

Ian Land
Sr. A&D Marketing Director
Synopsys, Inc
Mountain View, CA 94043
Email: land@synopsys.com

***Abstract* - The automotive industry has the scale to invest heavily in reliable, safe, and secure microelectronic design innovations, especially related to autonomous vehicles and assisted driving. Similarly, the aerospace and defense (A&D) industry has also been making similar investments, but does not operate at the same scale. The aligned needs of these industries can help benefit both.**

The lessons learned in high-reliability automotive design can be leveraged to improve the reliability, safety, and security of A&D applications, as well as the radiation tolerance of semiconductor devices.

In this paper we introduce all the aspects of reliable automotive design, its applicability to A&D, and focus on techniques used for automotive functional safety and their applicability to A&D.

Keywords – functional safety; automotive; aerospace and defense; ISO 26262; fault simulations; FMEA; FMEDA

I. INTRODUCTION

Existing tool and IP infrastructure developed for automotive functional safety (FuSa) can be leveraged in the design of complex microelectronic systems used in aerospace and defense (A&D) design. Complex systems in both automotive and A&D require consideration of performance, function, power, safety, reliability, quality and security during their design. Additionally, aerospace designs require radiation resistance whether in terrestrial aircraft, earth-orbit satellites or systems intended for use in deep space.

A&D applications have similar needs to those of the automotive industry:

- Reliable and robust design, i.e., resistance to systematic faults
- Safe and radiation-tolerant design to prevent random faults
- Secure design or resistance from targeted faults

Automotive and A&D goals map nicely to each other with the automotive FuSa goal aligning well with the A&D customer's need for high-availability electronics. Of course, a system is not safe if it is not also secure. Both customers need high-reliability to the point of zero defect, fault-tolerant operation. Finally, both have a need for high-quality, mission-critical operation.

Both automotive and A&D have critical standards that are valuable to consider:

Automotive:

- ISO 26262 – Automotive Safety
- AEC-Q100 – IC Qualification

A&D:

- DO-254 – Design Assurance for Airborne Electronic Hardware
- DO-178 – Software Considerations in Airborne Systems
- NTSS – NASA Technical Standards System
- MIL-PRF-38535 – IC Package Reliability

General:

- ISO 9001 – Quality Management

The large number of players in the automotive domain drove the need for a clear definition of FuSa that could be agreed to by all members of the supply chain. The result was the development of the ISO 26262 standard [1], which lays out a clear development lifecycle with specific work products, tracking requirements, implementation, and validation at every step of the V-shape flow (see Fig 1).

In terms of the supply chain, A&D shares the same need for a clear definition for design for functional safety. The left side of Fig. 1 shows the mapping between the automotive industry and A&D players during the product development lifecycle.

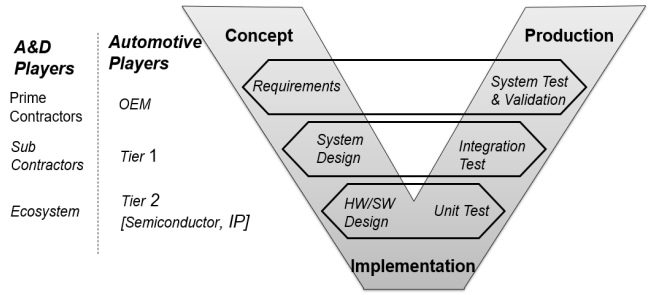

Fig. 1. V-Shaped Model for Microelectronics Development

Not surprisingly, others have addressed the similarities of the requirements between the automotive and A&D domains, such as [2], [3] and [4] and attempts were made to map the requirements in each domain in order to leverage solutions from the automotive industry. However, no detailed guidance was provided so far on quantitative safety analyses and validation.

the two domains, it becomes apparent that state-of-the-art (SOTA) automotive design risk mitigation techniques, flows and tools can be applied to A&D design, and hence, can be leveraged to allow for a better data exchange among members of the A&D supply chain. Using the design automation tools and methodologies built originally for the automotive industry can result in a more efficient development cycle.

In this paper we focus on the concept of functional safety analysis for identifying potential for random hardware faults, as done for automotive design according to ISO 26262 and discuss how it can be leveraged in A&D design.

Section II introduces the topic of functional safety and presents the concept of design faults, while section III discusses fault modeling and failure-mode-effect diagnostic analysis (FMEDA) and the ISO 26262 safety metric calculation,

Section IV discusses how these metrics can be validated by fault injection testing and fault simulation in general.

Section V discusses lessons learned and how principles and concepts from ISO 26262 can be applied to A&D.

II. Functional Safety Overview

Safety standards (such as IEC 61508 and its derivative, ISO 26262 for automotive systems) define functional safety as "the absence of unreasonable risk due to hazards caused by malfunctioning behavior of E/E systems." [6]. Specifically, for ISO 26262, the level of risk and its mitigation is expressed as an automotive safety integrity level (ASIL) (see Table I)

From the FuSa perspective, design faults can come from two sources:

- Systematic faults – those due to bugs in software or hardware, incorrect design specifications and implementation, or incomplete verification and validation; they can also be introduced by tools involved in the design and verification processes.
- Random hardware faults – those are failures in hardware only which are either permanent (due to silicon aging or other permanent effects) or transient (soft error due to high-energy neutrons, alpha particles, etc.).

A. Systematic Faults Mitigation

The mitigation for systematic faults requires a well-defined development lifecycle process, with specific work products documenting the activities in each phase of the development. A few examples are as follows:

During the planning phase:

- Define the safety concept (including use cases, environmental conditions and safety mission).
- Create a safety plan.
- Perform a high-level failure mode and effect analysis (FMEA) and FMEDA.
- Create the safety requirements and technical safety concept (including architecture and technical safety requirements).

During the implementation phase:

- Create specifications at the SoC and module level, as well as a verification plan.
- Conduct confirmation reviews for all specifications and plans.
- Trace back all design and verification activities to the safety requirements and ensure that these requirements are fully covered.
- Manage the software development according to known standards (such as MISRA-C).
- Use SOTA design and verification tools and methodologies to avoid implementation bugs or failure to detect them.

During the integration and validation phase, run testing and validation of the device from a functional as well as reliability aspect (e.g., by following AEC-Q100).

B. Random Hardware Fault Mitigation and Measurements

Random fault mitigation requires a safety-aware architecture with safety mechanisms (SMs) added to monitor and detect the occurrence of random faults, as well as help the system reach a safe state if a fault is detected.

Complying with ISO 26262 requires fail-safe design; therefore, SMs are needed to detect the faults and bring the design to a safe state. Examples of such SMs are: dual-core lockstep (DCLS), temperature and voltage monitors, built-in self-test (BIST) or even software SMs.

Some SMs are safe-operational, i.e., they enable the design to continue functioning correctly even in the presence of a fault. Examples of such SMs are triple module redundancy

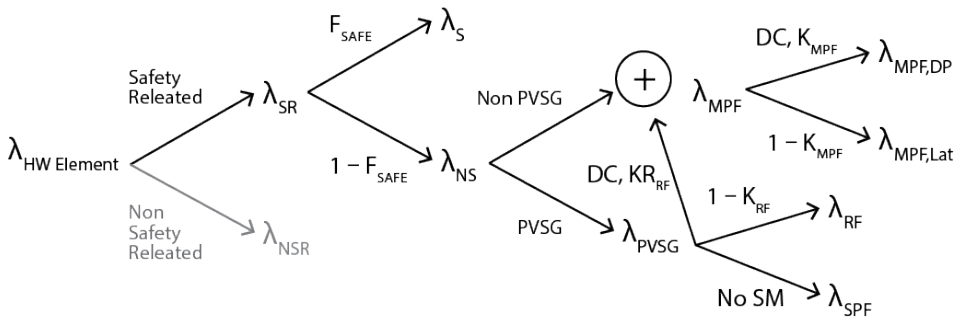

Fig. 2. Failure Rate Computation for ISO 26262 Metrics

(TMR), single error correct, double error detect (SEC-DED), and error correction code (ECC) in memories.

SM implementation also brings up some physical requirements, such as temporal or physical diversity in the redundant cores, and freedom from interference (FFI) of the redundant cores and so on.

ISO 26262 defines a quantifiable random fault analysis process that includes FMEA and FMEDA. The FMEDA is used for estimating the ISO 26262 metrics considering the SM fault coverage and determining the residual failure rate compared to the original failure rate of the design. These metrics are:

- Single-point fault metric (SPFM)
- Latent fault metric (LFM)
- Probabilistic metric for hardware failures (PMHF).

These metrics determine the target ASIL of the design, as shown in Table I.

TABLE I. ASIL METRICS FOR ISO 26262

ASIL	SPFM	LFM	PMHF
D	≥ 99%	≥ 90%	$<10^{-8}$ h^{-1} or 10 FIT
C	≥ 97%	≥ 80%	$<10^{-7}$ h^{-1} or 100 FIT
B	≥ 90%	≥ 60%	$<10^{-7}$ h^{-1} or 100 FIT

$$1-\frac{\sum_{SR,HW}\left(\lambda_{SPF}+\lambda_{RF}\right)}{\sum_{SR,HW}\lambda}=\frac{\sum_{SR,HW}\left(\lambda_{MPF}+\lambda_{S}\right)}{\sum_{SR,HW}\lambda} \quad (1)$$

$$1-\frac{\sum_{SR,HW}\left(\lambda_{MPF,L}\right)}{\sum_{SR,HW}\left(\lambda-\lambda_{SPF}-\lambda_{RF}\right)}=\frac{\sum_{SR,HW}\left(\lambda_{MPF,DP}+\lambda_{S}\right)}{\sum_{SR,HW}\left(\lambda-\lambda_{SPF}-\lambda_{RF}\right)} \quad (2)$$

The failure rate types (λ) in Eq. 1 and Eq. 2 are calculated from the base failure rate (BFR) of the design and considering the percentage of safe faults (F_{SAFE}) and the diagnostic coverage (DC) of the SM (K_{RF} for single point faults, and K_{MPF} for multi-point faults) as shown in Fig. 2.

Key parameters for failure rate computation:

- PVSG – Probability to violate the safety goal
- F_{SAFE} – Fraction of safe faults as measured by structural analysis, formal proofs, expert judgement
- DC, K_{RF} – Diagnostic coverage, residual faults as measured by fault injection simulation
- DC, K_{MPF} – Diagnostic coverage, multipoint faults as measured by fault injection simulation

III. FAULT MODELS AND FMEDA PRINCIPLES

A. Permanent and Transient Fault Models

As mentioned in Section II, two types of faults are possible in a hardware design (for example, in ICs):

- *Permanent faults*: Silicon failures such as stuck-at, open circuit, and bridging faults (see Fig. 3).
- *Transient Faults*: Soft errors (can be overwritten) such as single event transient (SET), e.g., a spike in a wire, as single-bit upset (a bit flip), or multi-bit upset (see Fig. 4).

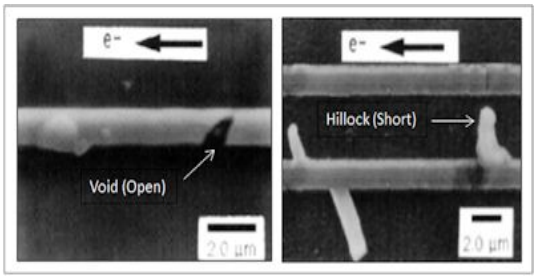

Fig. 3. Example of Permanent Silicon Faults

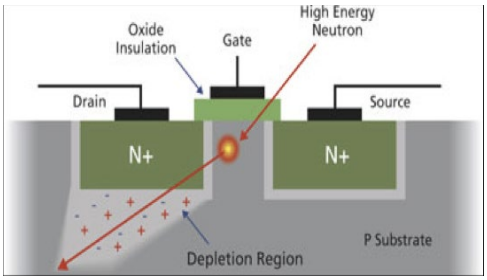

Fig. 4. Transient Faults (Soft Errors)

B. *FMEDA Throughout the Design Lifecycle*

The process of implementing the safety requirements in the design and measuring their effectiveness in detecting permanent and transient faults stretches from the design inception to the final product. Today, EDA tools, such as Synopsys Functional Safety toolset, enable a continuous development and inspection of the quality of the safety mechanisms in a design.

At the architecture exploration phase, only high-level failure modes are considered. Addressing their mitigation yields requirements for safety mechanisms based on design assumptions only. Once the design specification has solidified, and the RTL design starts, static analysis technologies can be applied to estimate the effectiveness of the safety mechanisms – their diagnostic coverage (DC), which is translated to K_{RF} and K_{MPF} in the FMEDA computation. These tools can also potentially propose additional low-level safety mechanisms such as implementing TMR on vulnerable flip-flops.

As the RTL design matures, and is properly verified, the testbench can be used for *fault injection,* validating the assumptions made on the DC of the SMs, as well as measuring the percentage of safe faults (F_{SAFE}). The FMEDA metric is then updated with the measured safeness or fault coverage at that level.

Next, BIST insertion and design synthesis can occur, accommodating the physical requirements for the hardware SMs. After the netlist is ready, fault simulation can be performed again, this time on a much more accurate design representation, and the FMEDA metric is updated again with the final measurements of safe and detected faults.

Lastly, ISO 26262 compliance requires official reports, or work products, on the FMEDA analysis, as well as a safety manual describing all the implemented and assumed safety mechanisms. These reports can be automatically generated within the Synopsys Functional Safety Manager cockpit.

C. *Failure Rate and ISO 26262 Metric Considerations*

Calculating a design's SPFM and LFM based on the FMEA enables designers to differentiate between those failure modes which violates the safety goals, and those that do not. These metrics further assist in determining the *effective* failure rate of the design, which is lower than the nominal failure rate calculated assuming all design parts have an equal probability of failure and causing a safety goal violation.

Specifically, for the soft error rate (SER), the concept of design vulnerability addresses the types of transient faults which do not impact mission safety. Athavale has presented in [4] a formula which reflects a derating of the effective soft error rate by calculating the vulnerability factors for a design (Eq. 3):

$$SER^{DERATED} = \sum_{UCs} F_{UC}(V, f_{clk}) \times \sum_{\substack{Circuits \\ /Nodes}} SER^{NOM} \times TVF \times AVF \times PVF \quad (3)$$

Where:

- AVF = Architectural vulnerability factor, which is a function of micro-architecture and workload
- TVF = Timing vulnerability factor, which is a function of clocking, circuit behavior and workload
- PVF = Program vulnerability factor, which is a function of the final user observable program output

This kind of analysis is very relevant to A&D applications, dealing mainly with SER analysis.

The next section discusses how measurements of safe faults and diagnostic coverage of the SM can be performed in EDA tools, and deliver accurate results at both in RTL and gate-level netlists.

IV. Fault Injection Testing

With ever increasing complexity of hardware components in a safety system, the challenge is to address the huge fault universe and fault coverage measurements in an efficient way, with fast convergence during debug cycles. Potentially, every transistor and every connectivity in the SoC could fail; therefore, the total number of faults in a typical SoC could be in the billions.

In principle fault injection testing take the design as-is (a good machine) and compares its behavior to the same design with a fault injected (a faulty machine). The comparison is done in either static analysis (Fig. 6) or dynamic simulation (Fig. 7).

Fault injection testing has many applications, including:

to discover defects in the silicon manufacturing process.

- *FuSa metric* – for measuring the percentage of safe faults, as well as the diagnostic coverage of the SM.
- *Soft error vulnerability* – for measuring the non-dangerous transient faults for SER de-rating.
- *Security vulnerability* – for measuring the effect of malicious fault attack on the silicon.

For FuSa, the goal of fault injection testing is to validate the K_{RF}, K_{MPF} and F_{SAFE} metrics for every failure mode in the FMEDA, and thereby, validate the design metric calculation.

The challenge of identifying faults with a complex IC design can be met by a SOTA toolset such as the Synopsys Unified Fault Campaign — a UFC solution which includes the technologies and methodologies described in the rest of this section, starting with fault reduction.

A. *Fault Reduction*

Fault reduction is performed by both static and formal analyses.

A static analysis considers the design ports which indicate an occurrence of a failure mode (such as a bus data signals, for a failure mode of a data corruption), and checks all possible faults in their cone of influence (COI) as shown in Fig. 5. This technique shows which faults are not contributing to the failure mode, and hence can be considered as safe (unless they belong to another failure mode).

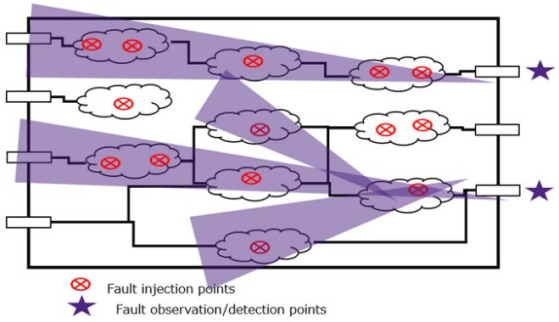

Fig. 5. Detecting Dangerous Faults by Looking at the Cone of Influence of the Observed Points

Formal techniques deploy controllability and observability analyses to determine if a fault has the ability to propagate to observable points given all possible stimuli. Faults which do not have any viable stimuli to cause it to propagate to observable points is, therefore, a safe fault as shown in Fig. 6.

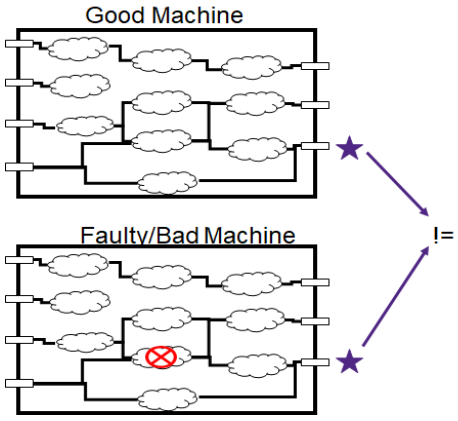

Fig. 6. Formal Observability Analysis to Determine Safeness of Faults

B. *Fault Simulation*

In principle, fault simulation (see Fig. 7) uses the regular functional verification testbench, runs the design as-is once (good machine), and then reruns the simulation, with one fault inserted in the design, either permanent or transient (faulty machine). The fault simulation then monitors certain signals in the faulty machine and compares their values with those recorded during the good machine run. If there are differences, the fault changes status to observed, or detected, according to the changed signal.

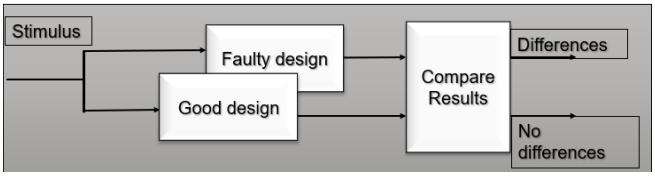

Fig. 7. Fault Simulation and Detection Principles

Naturally, considering today's design complexity, even addressing only parts of the design which are relevant to a specific failure mode, requires simulating a huge number of faults. In order to complete such a task in a reasonable amount of time, it is recommended to use a concurrent fault simulation technology.

Similarly, when the challenge is to simulate faults requiring many cycles, or when using a heavy software stack, *fault emulation* technology may become the solution.

C. *Fault Simulation Considerations for Transient Faults (SEUs)*

Transient fault modeling to address EM effects is normally performed on memories and other sequential elements. These faults are determined not only by the location (e.g., on a certain flip-flop or a memory bit), but also by the specific simulation cycle they occurred in.

When measuring the effects and diagnostic coverage of transient faults during fault simulation, it is important to specify a window of time in which a transient fault is injected at every clock cycle.

As this fault space becomes too large to manage, expert judgement should be made regarding the sample size of the overall possible faults.

D. *Architectural Vulnerability Analysis for Transient Faults*

Using static tools for vulnerability analysis of sequential elements can yield two outcomes:

- Calculate the safeness metric of a design part
- Suggest candidate flip-flops for hardening or TMRing to reduce their vulnerability

Fig. 8 is an example of a static analysis, using statistics to calculate probabilistic vulnerability of a flip-flop, based on its observability.

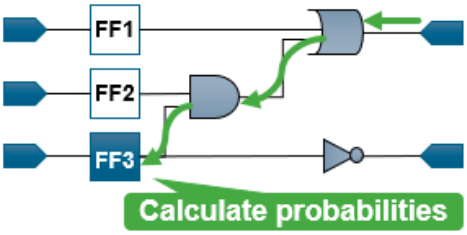

Fig. 8. FF Vulnerability by Observability Statistical Analysis

A formal analysis proposed by Arm Research [7] defines a methodology for reducing potential faults by identifying conditional vulnerability by determining when transient faults are dangerous or safe, based on an enable signal which is asserted or de-asserted accordingly.

V. Relevance to A&D

As a reminder, this document started with an overview of the idea to apply automotive design reliability and safety principles to A&D. Section II provided a technical overview of functional safety. Section III looked at fault models and FMEDA. Section IV discussed fault injection testing, a subset of which is a unified fault campaign.

The overview showed relevant standards that apply to automotive, A&D and both application areas. There are excellent references where experts have mapped automotive standards to A&D, such as [4] where Athavale has aligned ISO26262 with DO-254.

In the functional safety overview, a disciplined process for functional safety, which includes mitigation in both the planning and the implementation phase, is suggested that applies not only to automotive, but also to A&D systems. Further mitigations in the form of safety mechanisms are suggested, including safety mechanisms. A&D is already familiar with and using some of these safety mechanisms such as error correction and triple module redundancy.

While the ASIL (the level of risk and its mitigation strategies) is intended for ISO 26262 compliance, it is still relevant to not only aircraft safety, but also radiation effects.

Identifying and addressing the relevant failure modes, along with methodical analysis of a design to identify safe faults ensures a fault-tolerant design that is efficient and reliable. Implementation of safety and reliability mechanisms through the RTL synthesis process ensures effectiveness at fault detection. Many of the SMs mentioned previously, such as TMRs, are well known and used in A&D designs.

Fault injection testing, along with static analysis for detecting safe faults, can be used to ensure an efficient, safe, and secure design, with protection efficiently targeting key critical functionality, rather than time intensive analysis and modifications of the whole design.

The focus for this paper is *random faults*, which can be applied to automotive functional safety, A&D design assurance, and radiation-hardened ICs. Although it is not in the scope of this paper, safety mechanisms and fault injection technology can also address *targeted faults*, hardware failures that occur when an adversary is attempting to alter the functionality of the device.

FMEA and FMEDA are methodologies to analyze potential failures in systems. Thus, both automotive and A&D systems can use these methodologies to analyze system reliability. This document explains the use of electronic design automation tools and methods, such as beneficial code insertion (safety mechanisms), static analysis, formal verification, and fault simulation to mitigate random failures to lower risk and improve the FMEDA metrics.

VI. Conclusion

While at first glance, automotive and A&D systems may have little in common, the concepts and methodologies defined in ISO 26262 regarding function safety do have direct application. Moreover, existing tools and infrastructure developed for automotive functional safety (FuSa) can be leveraged in the design of complex microelectronic systems used in A&D.

About Synopsys

Founded in 1986 in North Carolina, USA, Synopsys is now among the “Top 15” largest software companies in the world and a world leader in the areas of Electronic Design Automation (EDA), Technology Computer Aided Design (TCAD), and Software Quality, Integrity and Security (APPSEC) tools and services. Headquartered in Mountain View, California, Synopsys employs over 14,000 engineering and support staff around the world.

References

[1] "ISO 26262 Road vehicles - Function Safety," ed: International Organization for Standardization, 2018.

[2] A. Schwierz and H. Forsberg, "Design assurance evaluation of microcontrollers for safety critical avionics," 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)

[3] M. Yadav, D. Shankar and T. Jose, "Functional Safety for Braking System through ISO 26262, Operating System Security and DO 254," 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)

[4] J. Athavale, R. Mariani, 2019 NEPP ETW: Functional Safety for Multi-Core Processor Based Avionics Systems (nasa.gov)

[5] ISO 26262:2018, part 5 Annex C

[6] ISO 26262-1 clause 3.67

[7] Venu, Balaji & Gilday, David & Logan, Angus & Jeyapaul, Reiley & Özer, Emre & Narang, Anuraag & Johar, Kauser & Lyberis, Spyros & Hughes, Zemian. (2021). PACE: AVF estimation using formal methods.