CS GROUP

FPGAs and ISO 26262

Ensuring Functional Safety in Embedded Systems for Autonomous Vehicles

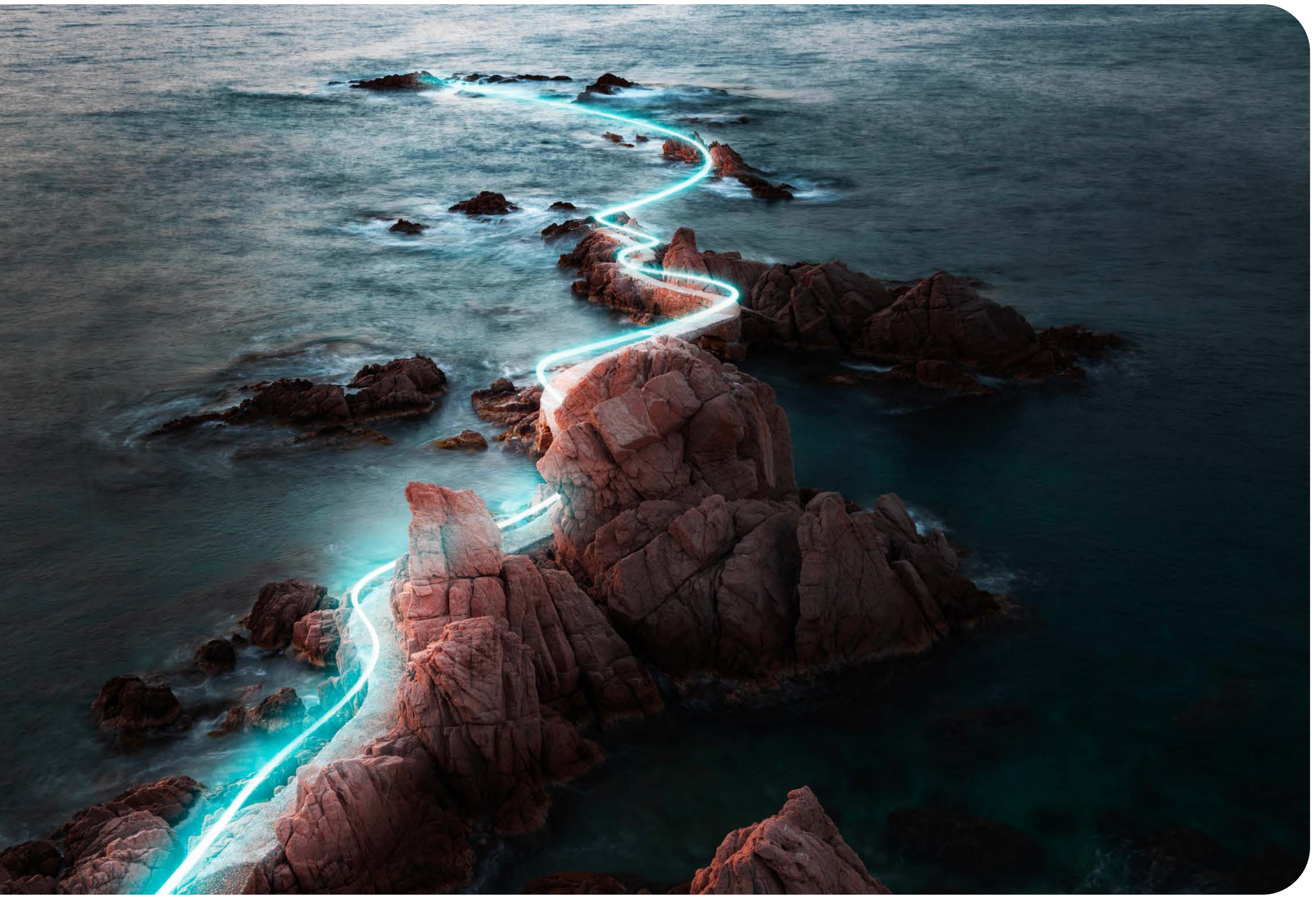
sopra steria

Since the first combustion engine vehicle produced in the 19th century with a fly wheel to start the engine and no safety consideration, the development of vehicles continued for two centuries so far, making them no longer mere modes of transportation. More and more functions and innovative features developed at the cutting-edge technologies are introduced in vehicles to provide safer, efficient and more enjoyable driving experience.

Vehicles today are evolving into intelligent, connected, highly aware and full featured machines. This increases the required speed and processing power and adds complexity to the advanced embedded systems inside the vehicle to attain the specified level of safety, performance, comfort and connectivity.

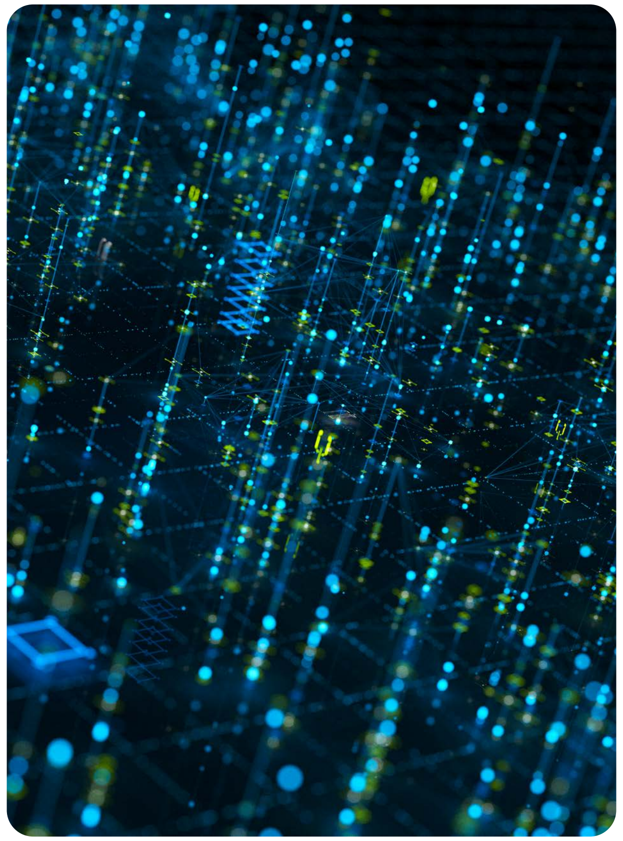
FPGAs are recognized for their capacity in parallel processing, pipelining, real-time data handling and flexibility, which makes them perfect for addressing the huge quantity of data to be processed and the complexities encountered in control systems in automotive sector.

In this white paper, CS Canada would like to provide guidance to assist system developers, hardware designers and integrators in performing Safety Analysis for safety critical FPGA-based automotive systems or items and reduce risks of errors and unnecessary iterations to pave the way for smooth and successful ISO26262 Safety certification



2. Why Choose FPGAs for Autonomous Vehicles?

Microcontrollers used to serve the embedded system market long time ago, where hardware and software combine together to implement the specified function. This made microcontrollers good candidates for applications involving sensor interfaces, control systems, output commands and communication protocols. They are utilized in vehicles for a variety of functions, including brakes system, transmission, and engine control.



However, microcontrollers are designed for sequential processing and their performance is affected by their inherent hardware-fixed architecture, Instruction Set and clock speed limiting the processing of data above a certain threshold. Although microcontrollers provide some flexibility in software customization, the hardware customization capabilities are limited.

In contrast, the parallel architecture of FPGAs, built based on Configurable Logic Blocks, integrated SRAM memories and interconnects allows parallel processing that enables workload distribution, low latency, and high bandwidth throughput. FPGAs are completely re-configurable offering desired flexibility to meet rapidly evolving requirements at any phase of the development life cycle. This makes FPGAs the best choice to implement high demanding real-time mission-critical applications such ADAS and autonomous driving at the highest level of safety and performance.

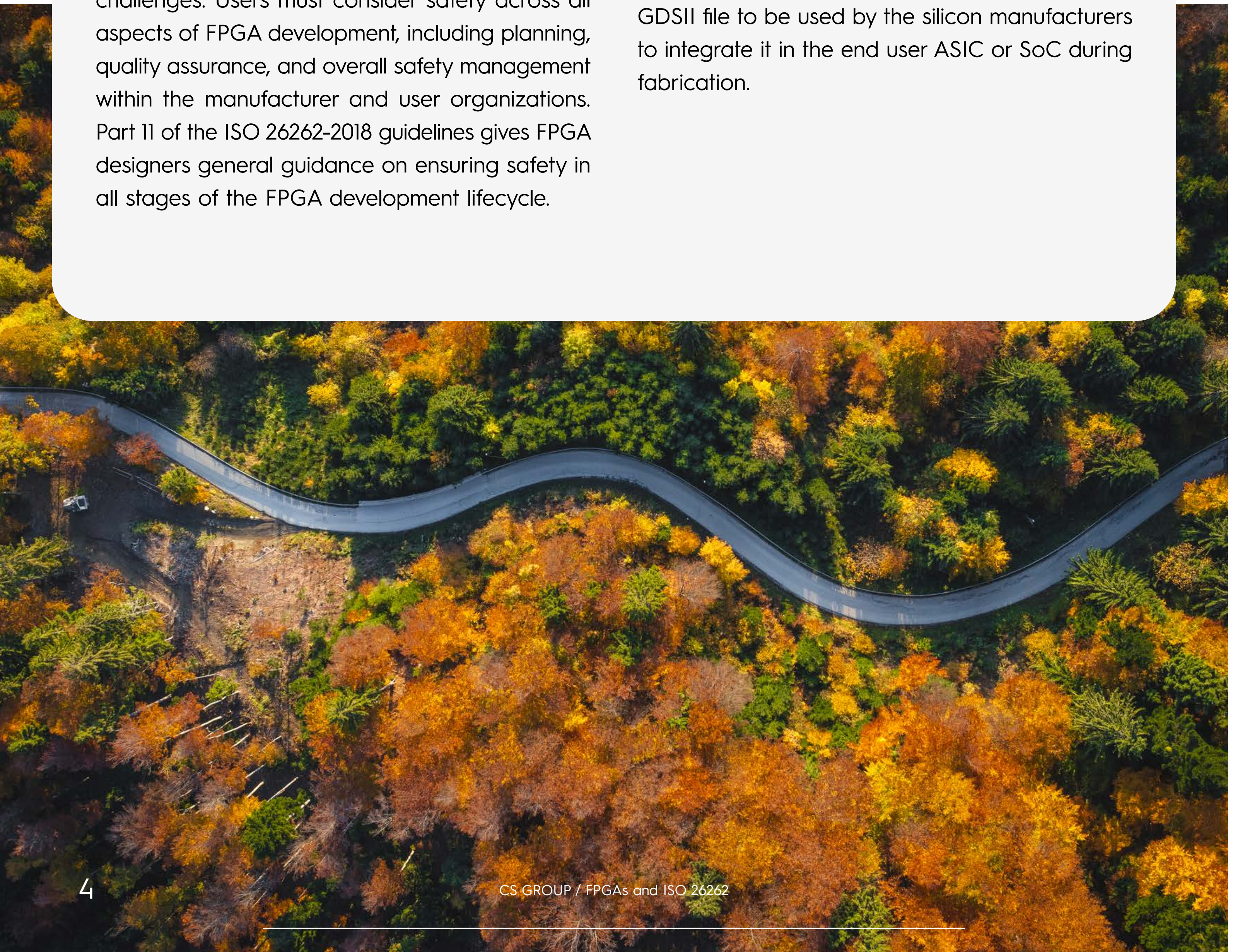
FPGAs facilitate sensors fusion and provide critical real-time processing for Advanced Driver Assistance Systems (ADAS) and autonomous driving (AD). FPGAs support cybersecurity, encryption, functional safety, infotainment and V2X communication customization.

3. Challenges of Functional Safety in FPGAs

FPGAs today have a strong heritage in high-reliability applications deployed in space and avionics in both commercial and military sectors. Their adoption is increasing in automotive applications due to their high performance and flexibility.

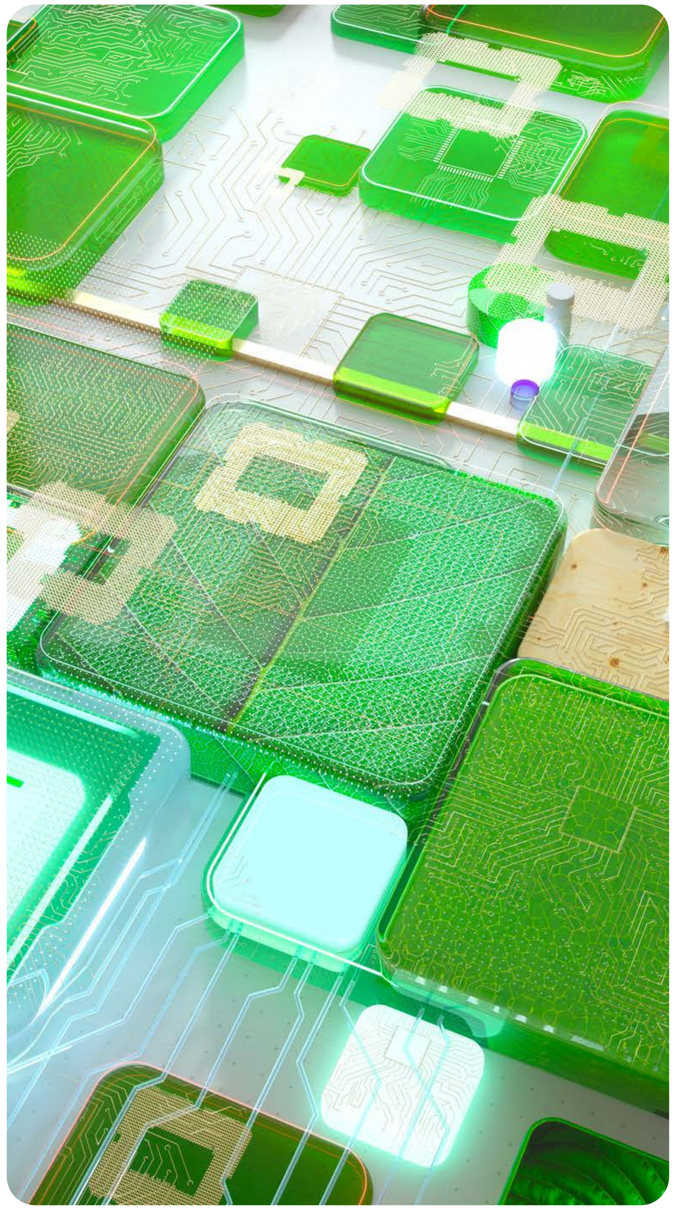
However, developing FPGA-based designs for critical automotive missions poses significant challenges. Users must consider safety across all aspects of FPGA development, including planning, quality assurance, and overall safety management within the manufacturer and user organizations. Part 11 of the ISO 26262-2018 guidelines gives FPGA designers general guidance on ensuring safety in all stages of the FPGA development lifecycle.

FPGAs can be discrete (FPGA) or embedded (eFPGA). An embedded FPGA (eFPGA) is a soft or hard IP core to be integrated into an ASIC or SoC, offering to users the flexibility to define the needed quantity of logic resources (LUTs, embedded memory, registers and DSP blocks) in order to reduce the cost and to allow making tradeoff between power consumption and performance with a flexible aspect ratio and number of I/Os. eFPGA hard IP core is delivered by the supplier as GDSII file to be used by the silicon manufacturers to integrate it in the end user ASIC or SoC during fabrication.



When using FPGAs or eFPGAs in safety-critical applications, the users shall define a standard development flow, starting with good planning, well controlled processes for specification and design, validation and verification processes, review checklists and reports templates. FPGAs and eFPGA may be manufactured with integrated processing core inside, DSP slices, predefined communication interfaces and some built-in safety mechanisms.

FPGA and eFPGA users must overcome the challenge of availability of IPs to be used in the design that are ISO 26262-certified and should ensure that the front-end and back-end tools used in FPGA or eFPGA development are ISO 26262-certified. They shall also ensure the tools used to verify the design and execute faults injection and monitoring are also safety certified.



Within the system architecture, The FPGA boundary shall be well defined including the interface between the FPGA and the hardware on the board as well as the interface between the FPGA and the software and between the hardware and software. System requirements shall be clearly partitioned between FPGA, hardware and software requirements to apply appropriately the semiconductor, hardware and software aspects of the ISO 26262 standard.

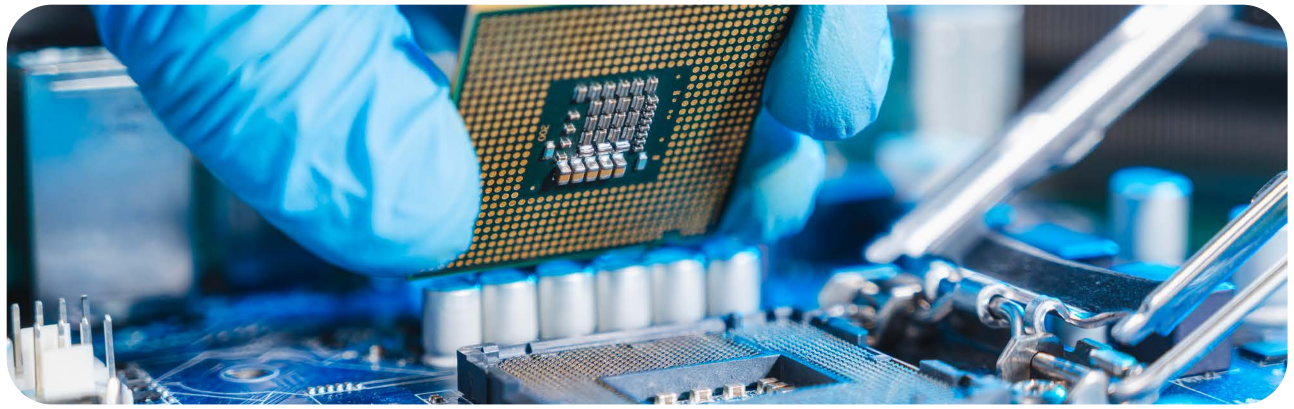
For eFPGA, the IP core is completely integrated inside the ASIC or SoC devices. The interface between eFPGA and ASIC or SoC shall be well defined.

FPGA and eFPGA logic design as well as the tools need to be qualified and certified for safety in compliance with ISO 26262 at the rigor commensurate with the safety integrity level of the implemented automotive functions.

Many tools are used in FPGA and eFPGA development flow for simulation, synthesis, placement and routing, static timing analysis, equivalence check, bitstream generation, configuration and debugging. To have confidence in the tools, the users shall ensure tools vendors have qualified the tool and followed an adequate development process appropriate to its usage in safety critical applications. If his is not the case, then the user is responsible for qualifying and certifying the tools for safety in compliance with the ISO 26262 standard. The users shall also minimize the risk of systematic faults in the developed FPGA or eFPGA due to malfunctions of the software tool.

ISO 26262 standard does not provide a specific method for Tool Qualification, and the Tool Confidence Level is determined as LOW, MEDIUM or HIGH, likely subjective. This is where the FPGA and eFPGA users need the experience of expert safety engineers to help in the evaluation and qualification of the used tool. To determine the required level of confidence (TCL), the user shall evaluate the possibility that the malfunctioning tool can introduce or fail to detect errors in the safety related FPGA functions (TI) and the confidence in preventing or detecting such errors (TD).

Performing the Tool Qualification for all the tools used in the FPGA development life cycle will be complex, time consuming and costly. FPGA manufacturers like Xilinx, Altera, Lattice, MicroSemi etc. has their own integrated EDA software tool flow. At the same time there are independent tools from many leading EDA companies. FPGA tool vendors understand the pain of functional safety certification process and the importance of the Tool Qualification, so many of the FPGA EDA tools are already TÜV SÜD or TÜV Rheinland certified.



4. Best Practices for Aligning ISO 26262 with FPGAs

The FPGA in safety critical automotive function is developed as a hardware part of a vehicle system or item (brakes, steering, engine control.) in compliance with ISO 26262. The FPGA development from safety perspectives is based on hardware safety requirements allocated from system safety requirements that are derived from the top-level safety goals of the item.

FPGAs contain fixed and non-fixed logic functions. Non-fixed functions are resources for users to use and configure them for custom functions. These functions can be simple logic gates, multiplexers, inverters, registers, memories or DSPs. Simple FPGAs usually implement frame-based configuration CRC error check for bitstream download.

More complex FPGA devices implement on top of the non-fixed logic functions, different types of fixed functions such as CPUs, memory controllers, security modules and safety mechanisms such as ECC on user memory read/write transactions, built-in CRC error detection circuitry to detect data corruption by soft errors in the configuration memory (memory scrubbing).

eFPGA IP cores on the other hand can be developed as a SEooC based on supplier assumptions of the intended functionality and use context which includes external interfaces. The validity of these assumptions is established by the end users in the context of the actual component that integrates the SEooC.

The suppliers of eFPGA IP cores shall certify their proprietary hard or soft IP cores with the associated EDA tools to allow OEM and Tiers to use them in the automotive sector. For successful certification, the suppliers shall provide evidence of compliance with ISO 26262 in planning, safety management, development and verification processes and tools qualification.

On the other hand, users shall qualify their hardware including the FPGAs and/or eFPGA IP cores in context of the item been developed for compliance with ISO 26262. FPGAs and eFPGA IP cores in the item context are part of the full hardware design that implements the functions allocated to hardware per the hardware requirements derived from the specified system requirements.

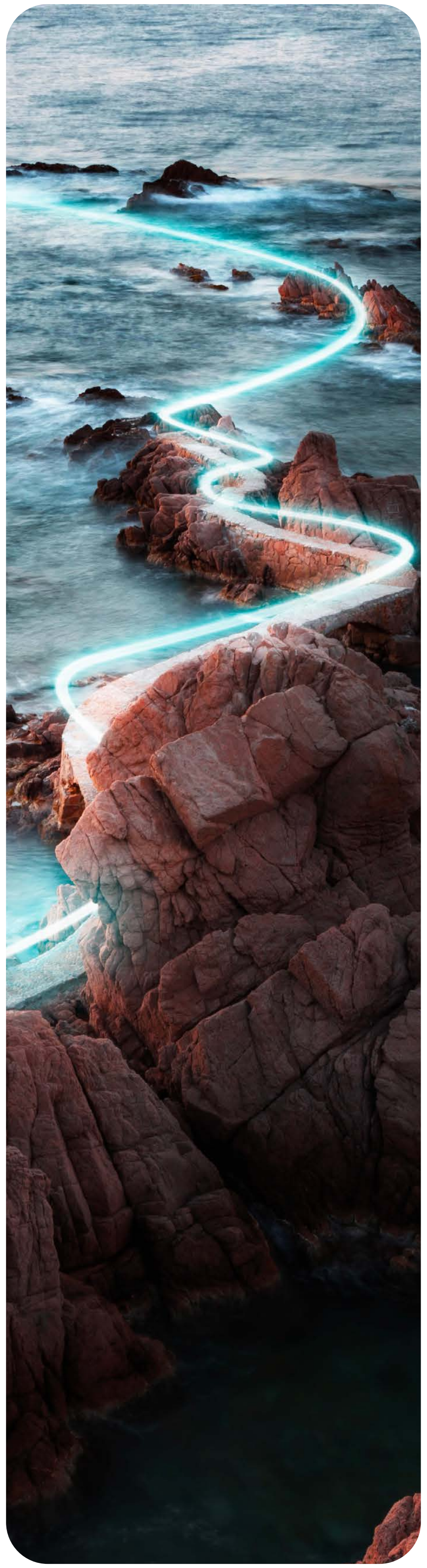
Both FPGAs and eFPGAs suppliers and end users are responsible for the management of functional safety when these devices are to be used in the automotive sector. They need to adapt the management of functional safety to the appropriate level by functional safety experts. For example, hazard analysis and risk assessment is not applicable in the safety plan at FPGA level. Functional safety audit needs to be managed by FPGA or eFPGA IP core suppliers at the device level.

FPGA and eFPGA IP core manufacturers and suppliers, in contrast to users, do not have any responsibility during the concept phase, unless they engage in the integration of the devices in the items So, the functional safety concept is not applicable at FPGA level.

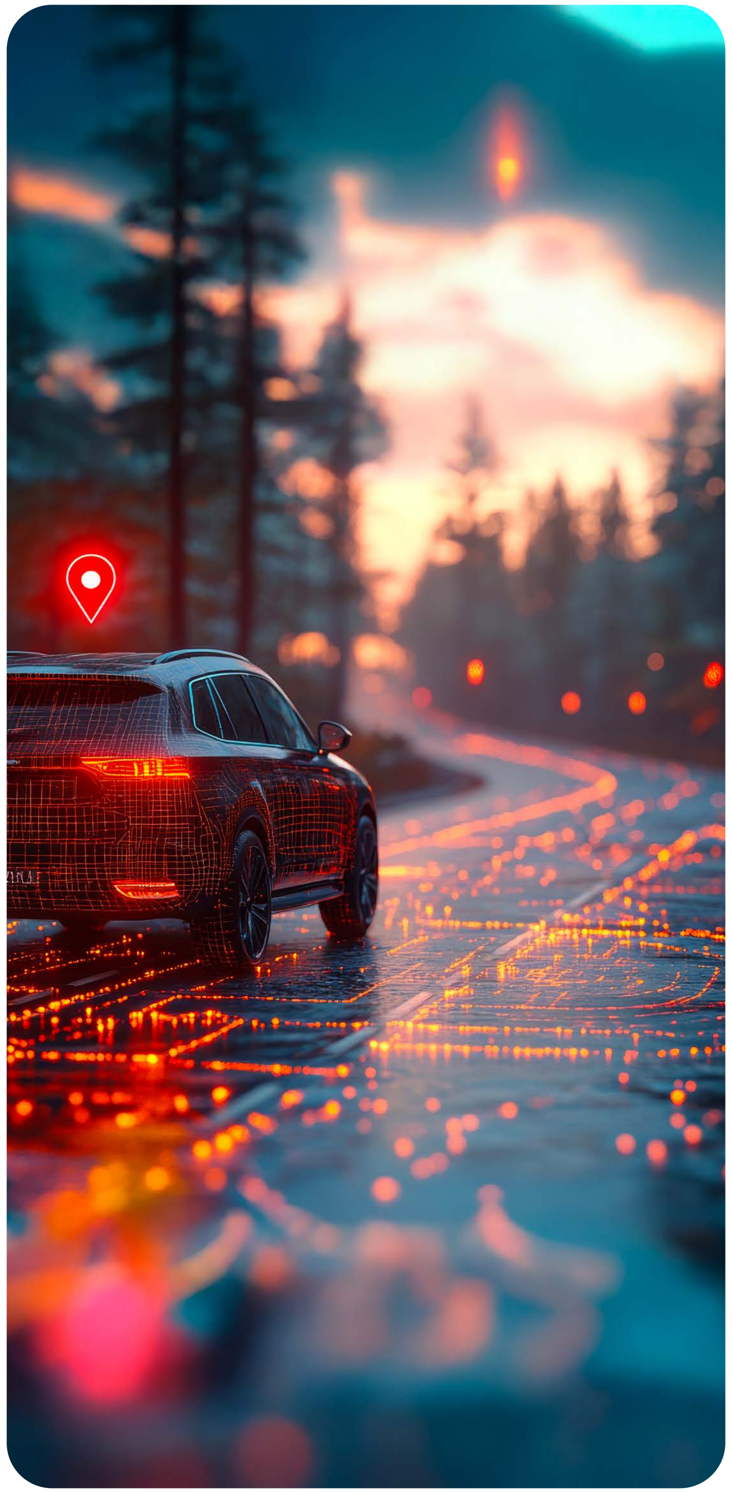
Product development at the system level can be partially or fully in scope depending on the integrated functions that support the technical safety concept and dedicated hardware safety measures. The end user must ensure all the functions brought as IPs, off the shelf or open source plus dedicated safety measures are ISO 26262 certified, otherwise he shall qualify them. The user, when integrating these IPs, must validate the assumptions of use, and the constraints and limits specified in the safety manual of the IPs.

Product development at the hardware level is fully in scope for the FPGA and eFPGA users. For the FPGA manufacturers and eFPGA suppliers, the scope is applicable according to their contribution to the overall safety concept. For example, if the FPGAs or eFPGAs contain built-in safety mechanisms, the diagnostic coverage shall be communicated to the users. If there is no contribution to safety concept, the manufacturers and suppliers are at least responsible to provide base failure rate, failure modes and failure modes distribution with reference or exemplary computation of hardware architectural metrics.

Product development at the software level is not relevant for the FPGA manufacturers, eFPGA suppliers, and users when the development flow is based on an HDL language (VHDL, Verilog). However, if the development flow uses a high-level language (systemC, OpenCL, C-to-HDL) or a model-based approach, then the development of the product at the software level (part 6) of the standard is applicable.



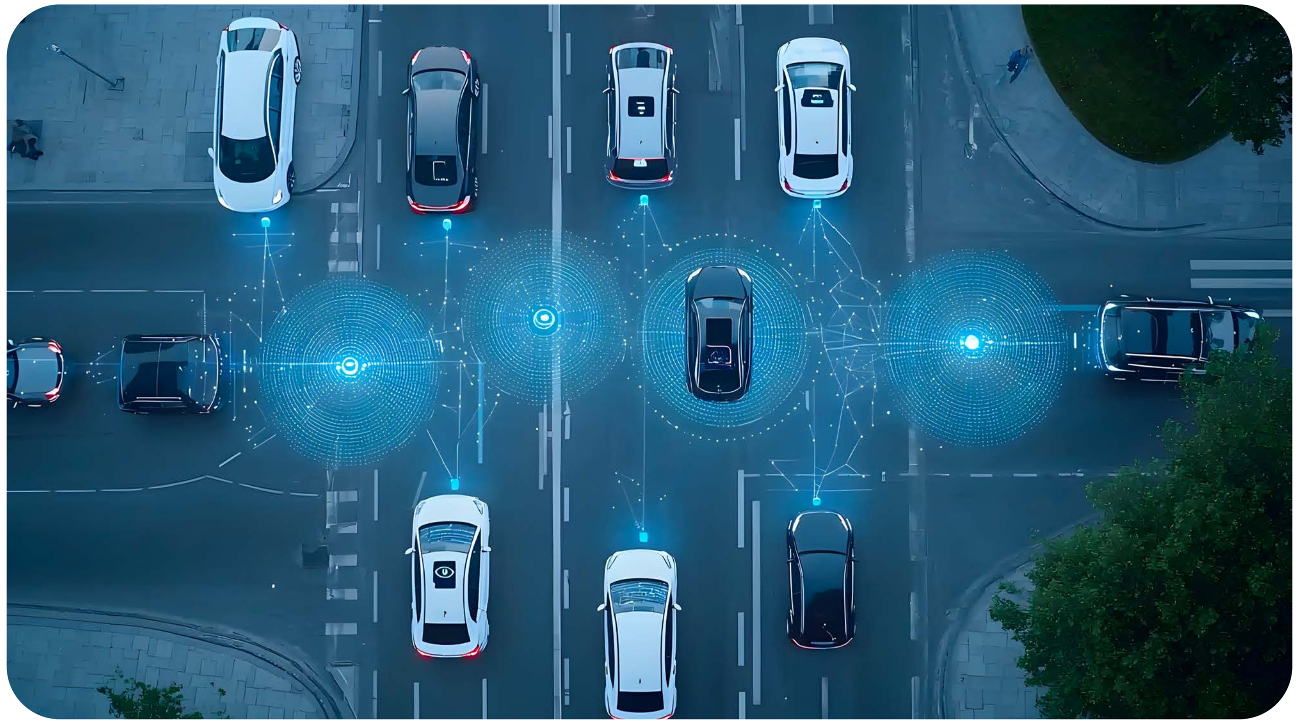
Production and operation requirements of ISO 26262 standard are applicable for FPGA manufacturers and to some extent with adaptation to the eFPGA suppliers. It is also applicable for users when they are involved in the production of the hardware integrating the FPGA or eFPGA. Manufacturers, suppliers and users shall all identify reasonably foreseeable process failures and their effect on functional safety and implement appropriate measure to address these issues before production. They shall all implement a field monitoring process when the FPGAs or eFPGAs are in operation. Decommissioning instructions are typically not applicable.



In addition to comply with ISO 26262, it is crucial to consider the Safety of the Intended Functionality (SOTIF), especially for ADAS and autonomous driving features. SOTIF complements ISO 26262 by addressing safety concerns related to the unintended behavior of these systems, which may arise from different factors such sensor limitations, environmental changes, or user misuse.

FPGA designers should implement measures to ensure that their systems not only prevent electronic malfunctions but also handle scenarios where the system might behave unintentionally due to external factors such environment abnormal conditions. This involves rigorous testing and validation processes that go beyond traditional functional safety assessments, ensuring that the system operates safely under all foreseeable conditions.

5. Safety Analysis with FPGAs



Failures in FPGAs are systematic or random. Random failures could be permanent or transient. Permanent failures are irreversible changes in operation that may manifest as stuck at low/high logic value, open/short or a single event hard error (SHE) causing permanent damage from a single radiation event. Transient failures are momentary voltage excursions or upset (soft errors) caused by a single energetic particle or event. Transient failures may occur as Single Event Transient or Upset (SET/SEU), Single or Multiple Bit Upset (SBU/MBU) or Multiple Cell Upset (MCU). Failures may occur in any element inside the FPGA or eFPGA. The failure may occur in the configuration or user memories, in the fixed function IP or the CLB, in the digital or analog I/Os and in signal routing wires and switches.

Transient faults are considered when they are relevant due to the operating frequency and the semiconductor front end technology and the materials on top of the die surface including the package. Transient faults can be addressed by a quantitative or qualitative approach. In quantitative approach, the FPGA or eFPGA user specifies a dedicated target for Single-Point Fault Metric (SPFM) and verifies if the design meets that target. In qualitative approach, the user elaborates a rationale based on analysis and verification of the effectiveness of the safety mechanisms implemented to cover the transient faults.

During development, qualitative and quantitative safety analyses are performed at the appropriate level of abstraction. Qualitative analysis identifies failure modes of the FPGA or eFPGA including the dependent failure analysis and consists of techniques or measures to detect or avoid systematic failures and reduce risk of violation of the safety goals.

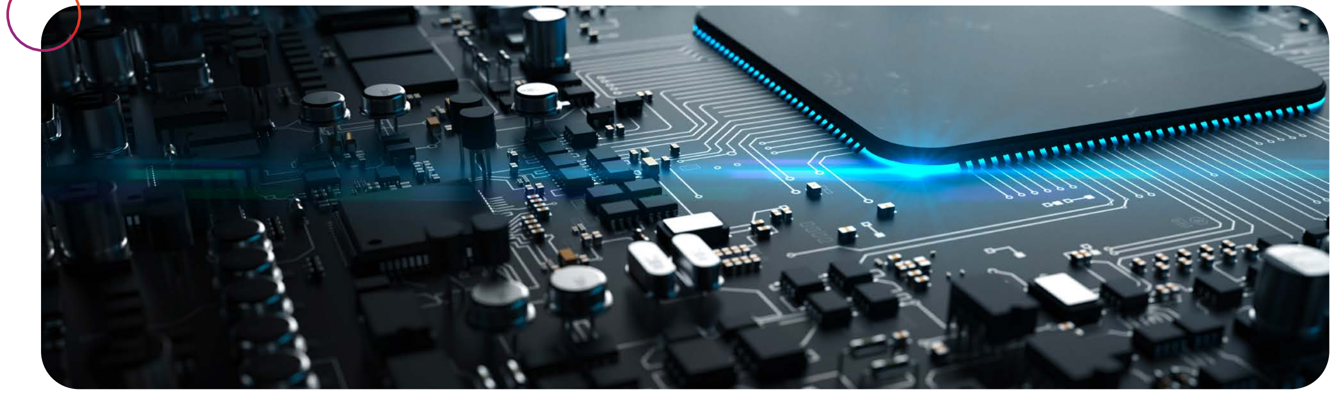
FPGAs and eFPGAs shall be developed based on standardized development process that provides evidence of sufficient measures for avoidance of systematic failures. FPGA manufacturers, eFPGA IP cores suppliers, and users can instantiate certified 3rd party soft-cores and hard-cores, use checklists and field data from similar FPGA technology and must document their design, tests, tools and verification results.

To achieve compliance with ISO 26262 requirements during the development of FPGAs or eFPGAs, the standard specifies different techniques and measures to apply at different design phases from the design entry phase to the production.

To meet hardware architectural metrics: Single-Point Fault Metric (SPFM), Latent- Fault Metric and Probabilistic Metric for random Hardware Failure PMHF, FPGA or eFPGA targets for diagnostic coverage of relevant failures can be derived from the targets at the item level or by Evaluating Each Cause (EEC) of safety goal violation. To provide evidence of meeting the targets, Manufacturers, Suppliers and users need to perform the quantitative analysis.

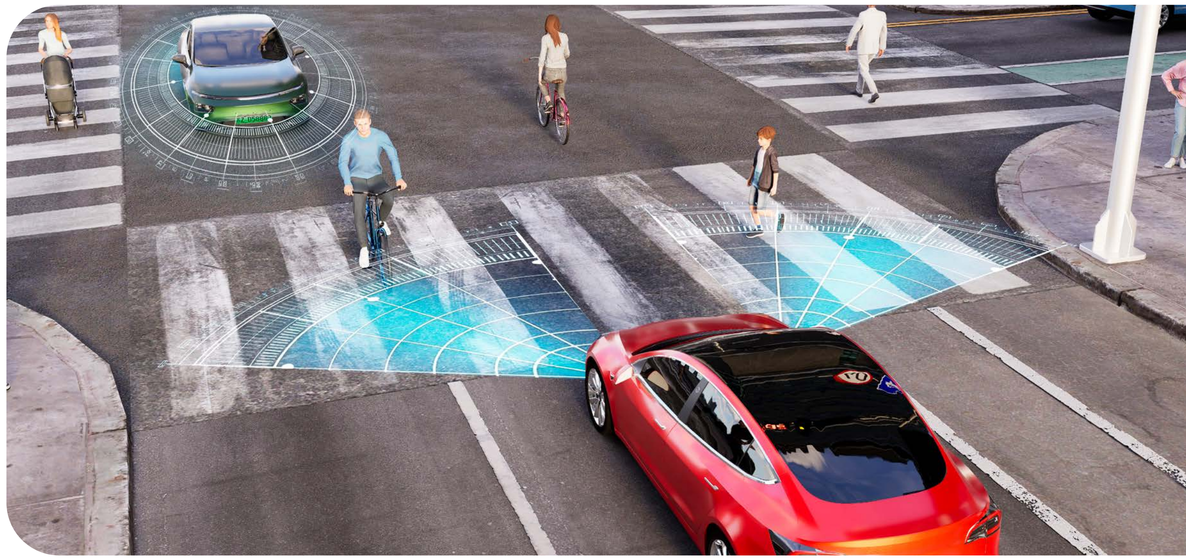
Quantitative analysis focuses on random hardware failures, both permanent and transient. It is accomplished by calculating the die failure rate and distributing it to the identified failure modes. The calculation is based on the base failure rate (raw) provided by the manufacturer of the FPGA or eFPGA silicon die. If this is not yet available for the targeted silicon technology, a preliminary base failure rate from similar technology node, or from reliability handbooks may be used by the eFPGA IP core suppliers with notification in the safety manual. The calculation will be updated later by the user with the true base failure rate specific to the technology node. The base failure rate shall be provided with all the assumptions made and supporting rationale to allow the end user integrator to evaluate and possibly harmonize failure rates for different components from different suppliers

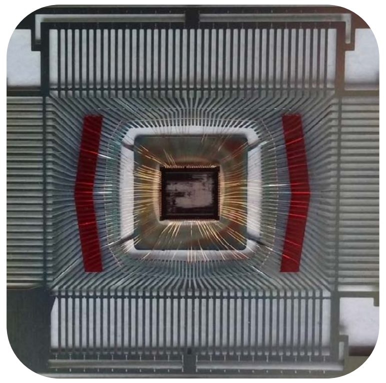
When the base failure rate from the silicon manufacturer is not yet available, users can use any model for reliability prediction such as IEC TR 62380, Siemens 29500, FEDES, to make preliminary calculation of the base failure rate on condition to be consistent and not mix between different models.



When calculating FPGA or eFPGA die failure rate, attention must be made for the configuration memory as the number of transistors and the failure rate of them may be different than the rest of the resources. Failure of unused resources must also be analyzed for their effects on the user design by a dependent failure analysis

The failure rate of the FPGA or eFPGA silicon die must be distributed to the components (CLB, Registers, Memories, Muxes, LUTs, I/O pads). Users can extract the failure rate per mm2 and multiply it by the part or subpart area related to each failure mode. The failure rate per mm2 is extracted by dividing the FPGA die failure rate by the die area of the component. Users can use a different method, based on base failure rates multiplied by the estimated number of equivalent gates or transistors for each part. The die area, the area of parts and sub-parts and the number of gates may be extracted from the synthesis and placement reports.



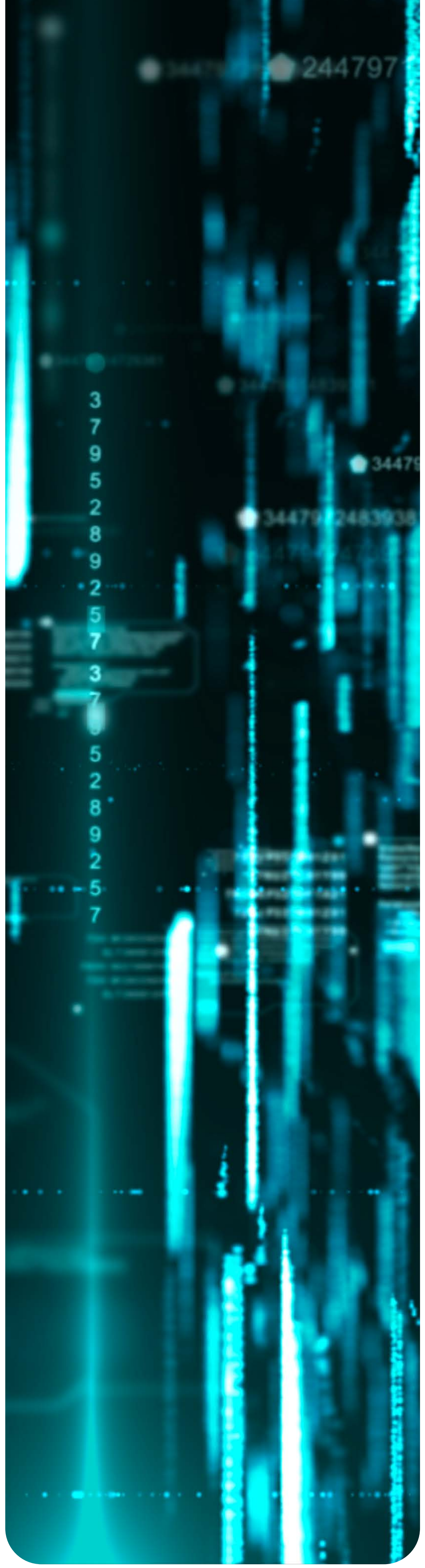


When the failure rate of the package is available, the failure rate per pin is extracted by dividing the failure rate of the package by the total number of pins. This allows distribution of the failure rate of pins that are safety-related per failure mode.

Quantitative analysis uses the functional information derived from the requirements and design description, the RTL code, the structural information from the synthesized gate level netlist and layout information in the final stage and is repeated during design development based on the latest information. Quantitative analysis also uses the information from verification of the diagnostic coverage and expert judgement supported by rationale and evaluation of the effectiveness of the system level measures.

Quantitative analysis must include Transient failures depending on the impact of the faults and when relevant to the FPGA front end technology and the materials on top of the die's surface including the package and the apparatus enclosure. The Manufacturer shall provide the base failure rate for soft errors with the conditions in which it has been computed or measured including the de-rating factor.

Estimated FPGA failure rate must be distributed to the identified failure modes. FPGA and eFPGA users can use fault models of memory elements and failure modes of digital components described in ISO 26262 standard. Fault models depend on the memory architecture and technology and failure mode of digital components are characterized based on their functional specification.



Quantitative analysis is augmented by Dependent failure analysis to be performed as the FPGA or eFPGA are implemented in a single physical component. The analysis shall cover the absence of both cascading and common cause failures to confirm independence. Independence justifies ASIL decomposition of a safety function while absence of cascading failures justifies the coexistence of functions with different or no assigned ASIL. Absence of cascading failure confirms freedom from interference between implemented functions.

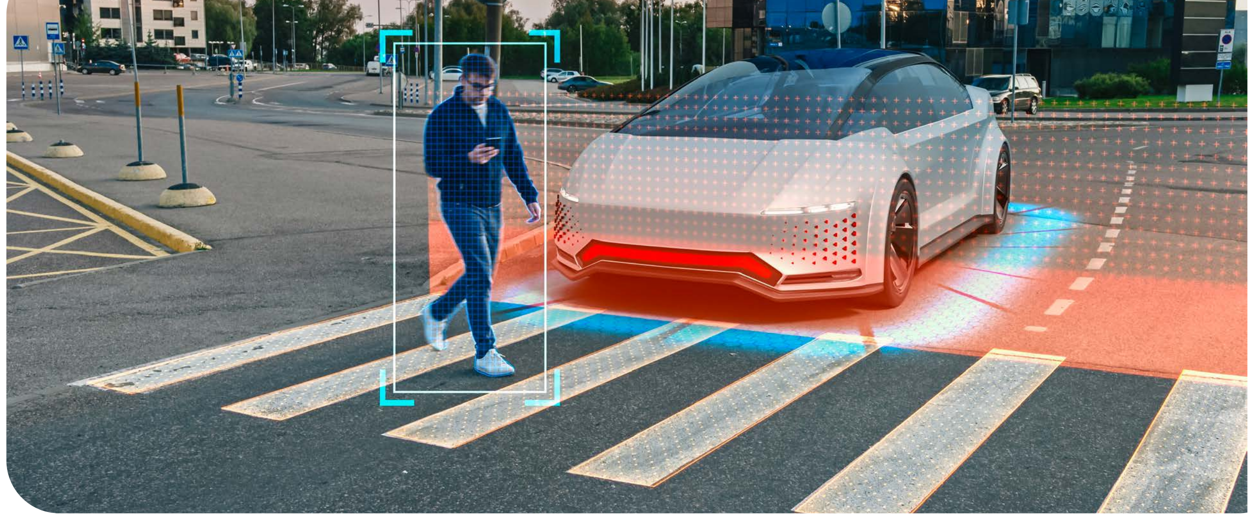
Users develop their FPGA or eFPGA designs using proprietary EDA or CAD tools delivered by the manufacturers or suppliers and compatible with their FPGA structure technology. The CAD tool is installed with supporting libraries specific to each family member of the FPGA node technology. CAD tool is used for FPGA back-end processes flow: placing, routing, timing analysis, bitstream generation, configuration, and on-chip debugging and work in conjunction with the same supplier's or third-party's front-end simulation and synthesis tools to provide a complete design environment.

The tool shall be certified to be pre-qualified for use in the development of FPGAs for the automotive sector in compliance with ISO 26262 and up to the required ASIL. If this is not the case, then the end users shall qualify the CAD software tool and evaluate the adequacy of the tool development process. The tool qualification shall be commensurate to the evaluated tool confidence level.

The tool confidence level shall be defined based on the tool's used functions and properties. This is required to minimize the risks of systematic faults in the developed FPGA due to the malfunction of the tool leading to erroneous outputs. To determine the confidence level, two criteria shall be evaluated:

- The tool impact (TI) assessed by the possibility of introducing or failing to detect errors in the developed FPGA due to malfunctioning tool that causes erroneous output
- The tool error detection (TD) assessed by confidence in preventing or detecting such errors in the tool outputs

Based on the tool impact and the tool error detection, the tool confidence class (TCL) is defined. The standard ISO 26262 provides one table to determine the tool confidence level.



The tool shall be qualified based on the determined confidence level. The standard specifies two tables for qualification methods depending on the determined confidence level. The are four alternative methods defined in each table but the certification authority such TUV SUD requires at least two methods to be used for qualification: usually Evaluation of the tool development process and Validation of the software tool. Increased confidence from use, is a method that may not be well controlled especially for relatively new tools that are not wide used, or tools developed by small vendors, as there are no warranties that all the failures are reported by the users and documented and published by the vendors with planned fixes. There are also concerns that not all the systematic faults are detected due to particularity or simplicity of the designs and the development environment.

Method 1c

Validation of the software tool, shall provide evidence that the tool complies with specified requirements to its purpose and usage. The validation shall include fault injection to assess the error detection and the reaction of the tool to anomalous operating conditions.

Method 1b

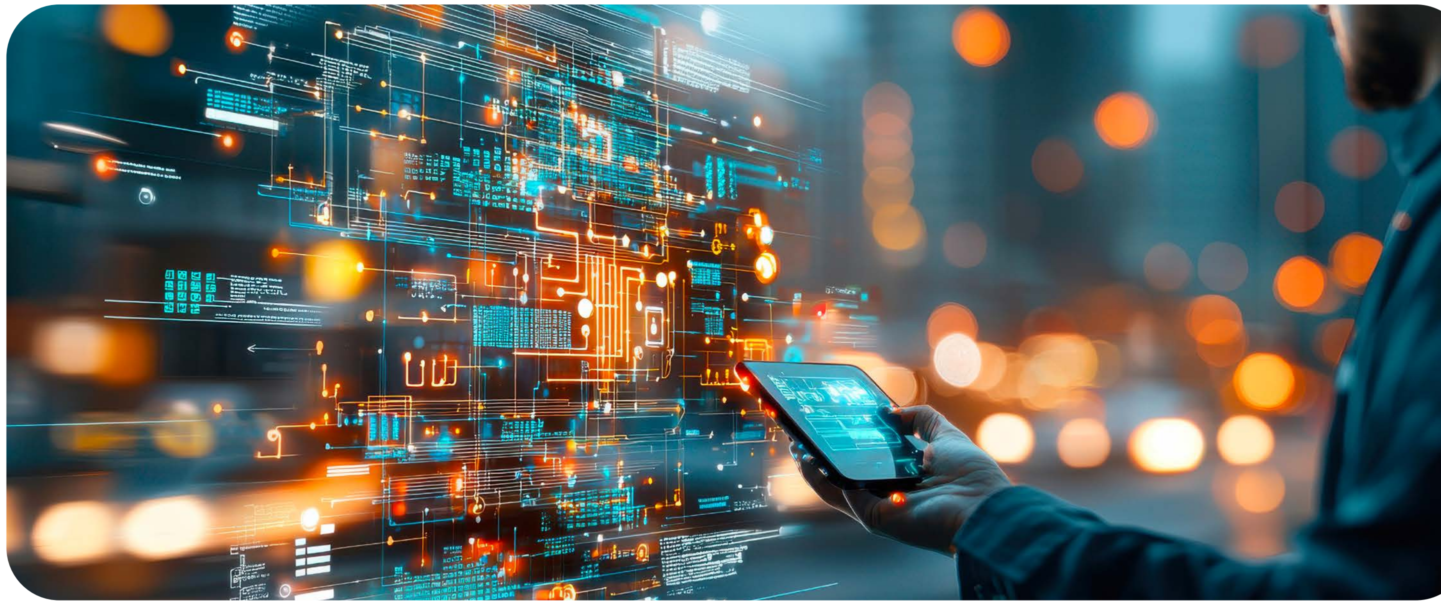
Evaluation of the tool development process shall be based on an appropriate national or international standard and provide evidence that a suitable software development process has been applied.

6. Case Study or Practical Example

Cs Canada gives an example here to illustrate at high level the certification of an embedded eFPGA IP core targeting a new silicon technology node and supplied as a hard IP with the associated CAD environment tool. The certified eFPGA IP core is to be embedded in an SoC or ASIC in the end user product where it it is assumed to execute critical functions with ASIL D safety goals.

The supplier must certify the eFPGA IP as an ASIL D ready SEooC, to be used by OEMs, Tiers and sub-Tiers in their items or systems for automotive sector. As explained in section 3 of this paper, the supplier will have to certify the following:

- **eFPGA IP core Development Process**
- **eFPGA CAD software tool**
- **eFPGA HW IP core**



6.1. eFPGA IP core Development Process

From supplier perspective, the final user application at the vehicle level is unknown, so this is a product independent evaluation of the Quality Management System and applied processes to develop the IP core. The work products required to be delivered for certification are expected to be covered in terms of defined inputs and outputs to process steps, templates and checklists, work instructions or similar as applicable.

The eFPGA supplier shall provide the evidence that it is establishing and maintaining an appropriate Management System which meets the requirements of the applicable standard parts in ISO 26262.

The supplier shall make available for certification the following main documents:

- Organization-specific rules and processes for functional safety
- Evidence of competence management
- Evidence of quality management system
- Safety Analysis documents (Guidelines, processes, checklists, reports Templates) for DFA, FMEA, FMEDA and any other applicable analysis
- Hardware Specification (Requirements guidelines, specification process, requirements checklists)
- Hardware Development (plan, procedures and processes, checklists, report templates)
- Hardware Verification (plan, processes, specification, checklists, report templates)
- Configuration and Change Management (plans, Guidelines, processes, reports templates)
- Safety manual and Safety Case (processes)
- Identified safety anomaly reports
- Impact analysis at element level (in case of upgrade)
- Tool qualification (plans, evaluation and qualification processes, guidelines, reports templates)

6.2. eFPGA CAD Software tool

The tool shall be certified to be pre-qualified for use in the development of items or elements for the automotive sector in compliance with ISO 26262 and up to ASIL D.

The FPGA supplier or potentially the end user if the tool is not certified, shall evaluate the tool development process and shall evaluate the CAD tool as described in section 5 of this paper. The supplier or user shall generate the following reports:

- Report of the CAD tool criteria evaluation
- Report of the CAD tool validation
- Tool Functional Safety Development Process
- Safety manual

Tool criteria evaluation report shall contain the following:

- Tool Use Cases from user or Assumed Use Cases (AoU) from the supplier
- Tool Functional Safety Requirements Specification (SRS)
- Traceability between Use Cases and SRS
- Tool FMEA or HAZOP analysis and tool confidence level determination
- Traceability between SRS and FMEA or HAZOP
- Tool abnormal conditions impact analysis

Tool validation report shall contain the following:

- Tool Software Safety Requirements (HLR/LLR)
- Tool Test procedures
- Traceability between SRS and HLR/LLR and Test procedures
- Tool Test results
- Traceability between Test procedures and test results

Tool Functional Safety Development Process shall include the following:

- Functional Safety Management during the development of the tool
- Functional Safety activities during the development of the tool

6.3. eFPGA Hardware IP Core

eFPGA Hardware IP core is a generic programmable logic defined for each user (customer) specific requirements. The users specify their logic resource need: CLB, LUT, Registers, logic and clock functions, Memories, Arithmetic and DSP functions, boundary pin cells, JTAG controller functions and any other fixed function, and the supplier configure the IP core to meet each user particular requirements. The supplier helps the user in the definition of their logic resources need by doing synthesis to their source code.

Although the safety-related functionality will be implemented by the end user, the supplier included as usual some built in functionality such as Encoder/Decoder for Error Correction Code (ECC) for the internal memories, Parity check, Bitstream Cyclic Redundancy Check (CRC), Encryption/Decryption algorithms That have the potential to be used as safety mechanisms.

This eFPGA IP core will be certified in relation to applicable ISO 26262 requirements for programmable logic devices. The targeted safety integrity level is ASIL D for system integrity and up to ASIL D capability for random hardware faults depending on the supplier Assumptions of Use (AoU) on usage and integration at ASIC or SoC and system level.

The supplier shall make available for certification the following main documents

- Safety Analysis (DFA, FMEA, FMEDA and any other applicable analysis)
- Hardware Specification (Requirements Specification, requirements checklists)
- Hardware Development (plan, procedures and processes, checklist, report)
- Hardware Verification (plan, processes, specification, checklist and report)
- Configuration and Change Management (plans, Guideline and reports)
- Safety manual
- Safety Case
- Identified safety anomaly reports
- Impact analysis at the FPGA level (in case of upgrade)

7. Conclusion

FPGAs have emerged as a critical component in the development of autonomous vehicles, offering unparalleled flexibility, parallel processing capabilities, and real-time data handling. Their adoption in safety-critical automotive applications is driven by their ability to meet the stringent performance and safety requirements mandated by standards such as ISO 26262.

Key Takeaways:

1

FPGA Advantages

FPGAs excel in parallel processing, low latency, and high bandwidth throughput, making them ideal for implementing real-time, mission-critical applications like Advanced Driver Assistance Systems (ADAS) and autonomous driving.

2

Challenges in Functional Safety

Ensuring functional safety in FPGA-based systems involves rigorous planning, quality assurance, and safety management. This includes qualifying tools, IPs, and development processes to meet ISO 26262 standards.

3

Best Practices

Aligning FPGA development with ISO 26262 requires a standardized development process, clear definition of system requirements, and thorough safety analysis. Tools and IPs must be certified, and systematic faults must be minimized through robust verification and validation processes.

4

Safety Analysis

Both qualitative and quantitative safety analyses are essential. Quantitative analysis focuses on random hardware failures, while qualitative analysis identifies failure modes and implements measures to detect or avoid systematic failures.

Concrete Example

Consider an autonomous vehicle's collision avoidance system, a safety-critical application requiring real-time processing and high reliability. An FPGA can be used to integrate sensor data from cameras, LiDAR, and radar, performing sensor fusion and real-time decision-making. The FPGA's parallel processing capabilities allow it to handle multiple data streams simultaneously, ensuring low latency and high accuracy in detecting and responding to potential collisions.

To comply with ISO 26262, the development process would involve:

Defining Safety Goals

Establishing top-level safety goals, such as avoiding collisions with other vehicles or pedestrians.

Tool Qualification

Ensuring that all tools used in the development process are qualified and certified for safety-critical applications.

Verification and Validation

Implementing rigorous verification and validation processes to ensure the FPGA meets the specified safety and performance requirements.

Hardware Safety Requirements

Deriving hardware safety requirements from system-level safety goals, ensuring the FPGA meets these requirements.

Safety Analysis

Performing both qualitative and quantitative safety analyses to identify and mitigate potential failure modes, including transient and permanent faults.

By following these best practices and leveraging the technical advantages of FPGAs, automotive manufacturers can achieve functional safety compliance and develop robust, reliable systems for autonomous vehicles. CS Canada is committed to guiding and supporting FPGA manufacturers, eFPGA core IP suppliers, and end users in navigating the complexities of ISO 26262 certification, ensuring the successful integration of FPGAs in safety-critical automotive applications.