Compliance driven Integrated circuit development based on ISO26262
Outline

- ISO26262 life cycle overview
  - Item development
  - Component development

- Compliance driven development process
  - V model with safety extension
  - Safe requirement management
  - HW risk assessment and safety analysis
  - Requirement driven verification
  - Tool qualification

- Questions?
ISO 26262 Life cycle Overview

Concept phase

Risk based approach

V Model based development (system, HW & SW)

Safety Management during Development

Supporting Processes

1. Vocabulary

2. Management of functional safety

3. Concept phase

4. Initiation of development

5. Hazard analysis and risk assessment

6. V model based development

7. Production and operation management

8. Supporting processes

Item development life cycle

Component development life cycle

Safety case

Structured documentation with evidence collection

Requirement management

Configuration and Change management

Verification and Validation

Item

System

Component

Element

MCU

Sensor IC

Safety engineering policy

Project functional safety plan

Technical safety requirement specification

V model based development

Confirmation measures & Evidence collection

Safety element in context or out of context

Risk assessment & Safety analysis

ISO 26262 life cycle: Item Development

**Item: System or array of systems to implement a function at the vehicle level**

**Hazard analysis** and **risk assessment** techniques to assign safety goals and safety integrity levels

Safety Goal: only applies to a car with a malfunctioning Item as it is in Context. All other component/elements depend on the **Interface** through which they are integrated. Handled through a Development Interface Agreement (**DIA**)

**Fault detection, mitigation & transition to safe state** at the **item** level is the key architectural requirements (safety integrity). Subsystem/component level diagnostic features helps a lot to achieve this at item level at reasonable cost

**Structured documentation** based on confirmation measures and **evidence collection** is the way to measure the effectiveness of a development approach
Components/elements depend on the **Interface** through which they are integrated at item level. This is handled through a Development Interface Agreement (DIA).

For generic component development standard recommends a **safety element out of context methodology** (e.g. MCU).

Safety requirement through risk assessment & safety analysis.

**Requirement management with traceability**

**Verification and Validation** at every stage and are **driven by methods**

Structured documentation with evidence collection.

**MCU: Micro controller unit**
V Model with Safety Extension

- Safety requirements based on “safety element out of context”.
- Formal Requirement management

Safety strategy & diagnostic features based on allocated safety requirements from system

Safety analysis & safety metrics (FMEA, FMEDA)

Change management - Impact on functional safety

Safety IP’s (diagnostic sensors) and safety monitoring

Safety Design

Safety Architecture

Functional Safety Requirement

Safety verification

Safety validation

Safety assessment

Formal review/audits

Safety manual

- Proven in use
- Mission profile
- Risk based
- TPE (stress & characterization)
- Establish traceability
- Compliance

- Requirement driven
- Risk based (FMEA/FMECA)
- Fault injection testing
- Structural (formal review, audit, code walkthrough)
- Establish traceability

FME(C)A: Failure mode and effects (criticality)analysis
FMEDA: Failure mode and effects & diagnostic analysis
TPE: Test and production, IP: Intellectual property
Component level Safety Goals and Requirements

- A Component always interacts through an **interface** through which it connected to the rest of the system

- Direct assignment of safety goals to a component not easy. However...

- Derived safety requirements can be allocated to a component & an **interface** through which it interact with rest of system

- The interactions with other systems and components can be handled through
  - Interface control document (technical interface)
  - Development interface agreement (process interface)
  - Safety manual (assumptions on use)
Component level (SEooC) Interface Agreement

**SEooC element/component**

- Safety element Development
  - Assumptions on Safety Requirements
  - Development
    - Safety work products for safety case
    - Change assumptions
    - Standard DIA
    - Custom DIA

**System**

- Safety component development
- Definition of Requirements
- Development of other elements
- Check validity of assumptions
  - OK
    - Integrate
  - N
    - Change request
    - Safety work products for safety case
    - Modify requirements

SEooC: Safety Element out Of Context
(Safe) Component level Requirement Management

- Safety requirements are allocated to a component through
  - Allocated from next higher level system
  - Through risk assessment and safety analysis

- Formal requirement management with traceability is mandatory
  - All requirements can be tracked to a design, verification and validation item
  - No design element in repository without an assigned requirement
  - Any PR raised can be tracked to a corresponding requirement ID
  - Impact on existing requirements can be identified for CR

- Formal process to track
  - Customer requirements
  - Allocated requirements from system or through risk assessment

PR: Problem report, CR: Change request, ID: Identification Number
HW Risk Assessment and Safety Analysis

Safety requirement met?

- Not met
- met

Top-Down analysis

Selection of safety mechanisms

Bottom up analysis

Quantitative Safety evaluation

Fault Tree

FMEA

Diagnostic Coverage

Fault metrics

Component models

Update Architecture Assumptions

Detailed Design

Reliability data

Reliability data (FIT)

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Functional Architecture

Safety Requirements

Fault metrics

Component models

Update Architecture Assumptions

Detailed Design

Reliability data

Reliability data (FIT)

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault Tree

FMEA

Diagnostic Coverage

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics

FME(C)A: Failure mode and effects (criticality)analysis
FIT: Failure in time

Fault metrics
Requirements and Risk driven Verification

**Requirement driven verification** (functional, safety, interface, environmental etc.)

- **Requirement traceability matrix**
  - Consistency & completeness between requirements and design components including their interfaces
  - Consistency & completeness between requirements and verification specifications

**Risk driven verification** (FMEA, FTA, FMEDA)

- Verification strategy to include critical items from risk assessment
- Safety analysis reports
- Safety verification report

Recommended methods from Standards is used to ensure compliance

**Action plan in case of safety anomalies detected**

FME(C)A: Failure mode and effects (criticality)analysis
FMEDA: Failure mode effect and diagnostic analysis
FTA: Fault tree analysis
Tool Qualification

- Why tool qualification
  - IC design involves many translation process, and tool in general has the capacity to introduce error during various translation process
  - e.g. RTL-> Synth netlist -> post layout

- Verification tools may fail to detect errors in the hardware items

- Tool qualification makes sure that tool correctly functions (improve confidence in tool function)

- The library components and different views used during the design translation process also need to be of mature quality and qualified one

- Recommended practice is to deploy a validated reference design flow (RDF) at organizational level
Tool Qualification Flow

- **Requirements**
- **Tool list**
  - Review and select tools
- **Tool chain analysis**
- **Safety analysis**
  - Based on ASIL
  - Determine TCL
  - Ti review
  - TD review
  - Determine TCL
- **Determine qualification plan and method**
- **Review the process of selected methods**
- **Execute qualification plan**
- **Review results & solve problems**
  - Find work around
- **Create action plan and Document results**
- **Tool classification and analysis report**
- **Tool qualification plan**
- **Tool qualification report**
- **Tool safety manual**

- **Use cases**
  - Characteristic functions required
- **Tool characteristics required**
  - Supporting tools
    - Requirement management, configuration management etc...
  - Development (Design) tools
    - Based on reference IC design flow
  - Development (verification) tools
    - IC, System validation
  - Development (validation) tools
  - Test and product engineering tools
- **Reference work flows and methodologies**

- **Action for risk mitigation**
  - Tool safety manual

**Tool Characteristics required**

- **Functions required**
- **Test and product engineering tools**
- **Development (verification) tools**
- **Development (validation) tools**
  - IC, System validation
- **Supporting tools**
  - Requirement management, configuration management etc...

**Reference work flows and methodologies**

- **Tool Characteristic functions required**
- **Use cases**
Conclusion

- Component development always need to be taken through a Development Interface Agreement both from process & product point of view.
- Direct assignment of a safety goal at a component level is not always possible.
  - Risk assessment and safety analysis at component level shall be used as a key instrument in deriving the safety requirements
  - Risk assessment shall be performed based on assumed safety context
- Requirement driven and risk oriented verification shall be employed as a verification approach
  - ISO 26262 standard provides a set of recommended methods to achieve the verification goals
- Tool confidence and tool qualification is a critical item from the ISO 26262 standard
  - Recommendation is to use a validated reference design flow at organization level as a reference
QUESTIONS ??
Impact on Methodologies & Tools

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement management</td>
<td>- Vertical traceability up-to <strong>design</strong> &amp; Horizontal traceability (DOORS)</td>
</tr>
<tr>
<td></td>
<td>- Traceability shall support <strong>impact analysis</strong> (DOORS)</td>
</tr>
<tr>
<td></td>
<td>- Method driven (Risk analysis) approach for <strong>safety</strong> requirement capture</td>
</tr>
<tr>
<td>Verification &amp; Validation</td>
<td>- Requirement driven</td>
</tr>
<tr>
<td></td>
<td>- Fault injection</td>
</tr>
<tr>
<td></td>
<td>- Proven in use argument generation</td>
</tr>
<tr>
<td></td>
<td>- Verification in every phase of development (follow V model)</td>
</tr>
<tr>
<td>Configuration management</td>
<td>According to ISO/TS 16949, ISO 10007 and ISO/IEC 12207</td>
</tr>
<tr>
<td></td>
<td>All safety work products under configuration control</td>
</tr>
<tr>
<td>Change management</td>
<td>Procedure driven approach</td>
</tr>
<tr>
<td></td>
<td>Evaluation of change request through Impact analysis (use DOORS)</td>
</tr>
<tr>
<td>Confirmation measures</td>
<td>Review: Results of a safety work product</td>
</tr>
<tr>
<td></td>
<td>Audit: Process and procedures</td>
</tr>
<tr>
<td></td>
<td>Assessment: Achievement of functional safety against safety plan</td>
</tr>
<tr>
<td>Tools</td>
<td>SW tool qualification based on tool impact on safety work products and detectability</td>
</tr>
<tr>
<td>Documentation</td>
<td>Structured documentation on all the above into safety case with argumentation</td>
</tr>
</tbody>
</table>
Safety work flow (Safety component or element)

1. Safety engineering policy
2. Project functional safety plan
3. Technical safety requirement specification
4. V model based development
5. Confirmation measures & Evidence collection
6. Organization level
   - Process, procedures, templates
7. Detailed Safety activities
8. Safety element out of context
9. Risk assessment & Safety analysis
10. Requirement management
11. Configuration and change management
12. Safety verification and validation
13. Quality assurance

Safety case