Table of Contents

- Product Development Hardware Level
  - HW Safety Requirements
  - ASIL Decomposition
  - HW architectural metrics
  - Evaluation of violation of the safety goal due to random HW failures

- Product Development Software Level
  - SW safety requirements
  - Metrics for SW Unit Testing
  - Metrics for Software Testing
HW Safety Requirements

- Initiation of product development at the HW level (5-5)
  - Project Plan and Safety Plan are being actualized and extended
- Specification of HW Safety Requirements (5-6)
  - Requirements for safety mechanisms to detect, to tolerate, or to indicate internal and external errors
  - Include HW-metrics
    - SPF and LFM

<table>
<thead>
<tr>
<th>ASIL</th>
<th>Failure Rate</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>&lt; 10⁻⁸/h</td>
<td>&lt; 10 FIT required</td>
</tr>
<tr>
<td>C</td>
<td>&lt; 10⁻⁷/h</td>
<td>&lt; 100 FIT required</td>
</tr>
<tr>
<td>B</td>
<td>&lt; 10⁻⁷/h</td>
<td>&lt; 100 FIT advised</td>
</tr>
<tr>
<td>A</td>
<td>&lt; 10⁻⁶/h</td>
<td>&lt; 1.000 FIT informative</td>
</tr>
</tbody>
</table>

HW Safety Requirements

- HW-Design (5-7)
  - Block-Diagram: Sensors, CPU, and Actuators
  - Components: Resistors, Capacitors, ...
  - Modules (logical and technical entities): Power supply, Memory, ...
    - Traceability of the HW Safety requirements is on module-level required
    - Inherits the ASIL of the Safety Requirements it implements
      - If one module implements more than one Safety Requirement, the highest ASIL is to choose
    - Part of HW design is HW detailed design which results in schematics and layouts.
ASIL Decomposition

- To redundantly implement Safety Requirements using independent modules in order to decrease the ASIL level

HW architectural metrics (5-8)

- To check the HW design (and HW detailed design) if the Safety Requirements (HW Metrics) are fulfilled or if further improvements are necessary
- HW faults are classified into:
  - Safe faults \( \lambda_S \): do not affect the safety requirements
  - Single Point Fault \( \lambda_{SPF} \): non-tolerated faults which violate the safety requirements
  - Multiple Point Fault \( \lambda_{MPF} \): combination of independent faults which lead to a violation of the safety requirements
    - Latent (L), Dormant (D), or Perceived (P)
  - Residual Faults \( \lambda_{RF} \): fault which are not detected by any safety mechanisms and which lead to a violation of the safety requirements
HW architectural metrics (5-8)

- Single Point Fault Metric addresses Single Point Faults and Residual Faults:
  \[ \text{Single Point Fault Metric} = 1 - \frac{\sum (\lambda_{\text{MPF D+P}} + \lambda_{\text{SPF}})}{\sum \lambda_{\text{Safety related Hard elements}}} \]

<table>
<thead>
<tr>
<th>Metric</th>
<th>ASIL A</th>
<th>ASIL B</th>
<th>ASIL C</th>
<th>ASIL D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Point Faults Metric</td>
<td>Not relevant</td>
<td>&gt; 90%</td>
<td>&gt; 97%</td>
<td>&gt; 99%</td>
</tr>
<tr>
<td>Latent Faults Metric</td>
<td>Not relevant</td>
<td>&gt; 60%</td>
<td>&gt; 80%</td>
<td>&gt; 90%</td>
</tr>
</tbody>
</table>

- If this metric can not be achieved,
  - the diagnostic coverage of the safety mechanisms need to be improved or
  - better electrical components need to be used.
HW architectural metrics (5-8)

- Absolute numbers need to be considered as well:
  - A: $\lambda = 2000$ FIT, $\lambda_{SPF} + \lambda_{RF} = 20$ FIT
  - B: $\lambda = 200$ FIT, $\lambda_{SPF} + \lambda_{RF} = 20$ FIT

- Both will be considered equally safe (due to the relative metric), but the absolute availability of A = 0.99 and B = 0.90 states, that A is 10 times more reliable than B

HW architectural metrics (5-8)

- Latent Faults Metric addresses Latent Multiple Faults:

  \[
  \text{Latent Metric} = 1 - \frac{\sum (\lambda_{\text{SPF}} + \lambda_{\text{RF}})}{\sum (\lambda_{\text{SPF}} + \lambda_{\text{RF}} - \lambda_{\text{RF}})}
  \]

<table>
<thead>
<tr>
<th>Metric</th>
<th>ASIL A</th>
<th>ASIL B</th>
<th>ASIL C</th>
<th>ASIL D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Point Faults Metric</td>
<td>Not relevant</td>
<td>&gt; 90%</td>
<td>&gt; 97%</td>
<td>&gt; 99%</td>
</tr>
<tr>
<td>Latent Faults Metric</td>
<td>Not relevant</td>
<td>&gt; 60%</td>
<td>&gt; 80%</td>
<td>&gt; 90%</td>
</tr>
</tbody>
</table>

- Reliability Analysis Software — Relex
  - Relex is an integrated set of reliability analysis tools split into several modules
Evaluation of violation of the safety goal due to random HW failures (5-9)

• ISO 26262 requires to perform a diagnostic coverage analysis for any fault:
  – DC_{RF} and DC_{MPF L}
  – “Calculate the Failure Rate and improve the system until the required failure rates (according to the ASIL Level) are reached” or
  – “Analyze any remaining cause for failure and improve the safety mechanisms”

SW safety requirements

• Motor Industry Software Reliability Association
  – Used at ASIL A to C
  – Guidelines for the use of the C and C++ language in critical systems
  – MISRA-C 2004
  – MISRA-C++ 2008
  – http://www.misra.org.uk

• Software Considerations in Airborne Systems and Equipment Certification (DO-178B)
  – Used at ASIL D
Metrics for SW Unit Testing

- **Statement Coverage**
  - Call `foo(1, 1)`

- **Branch Coverage**
  - Call `foo(1, 1)` and `foo(0, 1)`

- **Modified Condition/Decision Coverage**
  - Call `foo(0, 0), foo(0, 1), foo(1, 0), foo(1, 1)`

```java
int foo (int x, int y)
{
    int z = 0;
    if (x>0 && y>0) {
        z = x;
    }
    return z;
}
```

<table>
<thead>
<tr>
<th>Method</th>
<th>ASIL A</th>
<th>ASIL B</th>
<th>ASIL C</th>
<th>ASIL D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statement Coverage C₀</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Branch Coverage C₁</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Modified Condition MC/Decision Coverage DC</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>++</td>
</tr>
</tbody>
</table>

Metrics for Software Testing

- **Function Coverage**
  - Makes sure, that a specific function gets called

- **Call Coverage**
  - Makes sure that each function gets called

<table>
<thead>
<tr>
<th>Method</th>
<th>ASIL A</th>
<th>ASIL B</th>
<th>ASIL C</th>
<th>ASIL D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function Coverage</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Call Coverage</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>++</td>
</tr>
</tbody>
</table>