Developing intelligent automotive systems with functional safety

Optimised, efficient SoC technology powering innovation in automotive
Developing intelligent automotive systems with functional safety

- Automotive markets trends
- Technical challenges
- Functional safety

James Scobie
Senior Product Manager
Arm, Cambridge
Embedded & Automotive LoB

Ann Keffer
Product Management Director
Cadence Design Systems, San Jose
System and Verification Group
The most complex piece of electronics you will own
Increasing complexity in functional safety markets

**Automotive**
- Autonomous driving

**Transportation**
- Train control systems

**Industrial**
- Factory automation

**Avionics**
- Flight systems

**Healthcare**
- Robotic surgery

**Consumer**
- Domestic robots
Automotive semiconductor growth

Source: IHS Markit, 2017
Autonomous vehicles

- **Level 0**: No automation
  - Driver performs part or all DDT
  - OEDR- driver
  - ODD unlimited
  - Fallback- driver

- **Level 1**: Driver assistance

- **Level 2**: Partial automation
  - ADS performs entire DDS (when engaged)
  - OEDR- ADS
  - ODD limited
  - Fallback-user

- **Level 3**: Conditional automation
  - Fallback- ADS

- **Level 4**: High automation

- **Level 5**: Full automation
  - ODD unlimited
“Almost 80% of automotive innovation comes from electronics (semiconductors) and software”

– Audi at CES Asia
The foundation for autonomous systems

Autonomous system

- Gather environment information from sensors
- Filter, interpret, & understand sensor data
- Safely choose actions
- Initiate actions

Sense | Perceive | Decide | Actuate
What are the challenges?

Complex and demanding compute requirements

Increasing need for security

Rising functional safety requirement
Arm® Cortex® processors offer a range of choices
Complex and demanding compute requirements

**Highest performance**
- Sophisticated virtual memory support for rich OS
- Advanced programmer’s model
- Software-managed interrupts
- Multi-core and multi-cluster
- Arm TrustZone® technology support

**Cortex-A**

**Cortex-M**

**Cortex-R**

**Real-time processing performance**
- Hard real-time deterministic
- Software-managed interrupts
- Fast interrupts
- Multi-core
- Hardware virtualization (in Armv8-R)

**Smallest area and lowest power profile**
- Standardized memory map, optimized for RTOS
- Simple programmer’s model
- Hardware-managed interrupts and lowest latency
- TrustZone technology in Armv8-M

*Size of bubble indicates increasing system and software complexity*
Flexible solutions need a range of capabilities
Heterogeneous compute requirements

Mix of IP and solution

• Compute capability to meet the requirements
  • Within the constrained power window

• Accelerators
  • Diverse components designed for specific tasks

• System IP
  • Interconnect system IP delivering coherency and the quality of service required for lowest memory bandwidth

• Software
  • Increasing system efficiency with optimized software

• Subsystems
  • Efficient integration
Arm: the foundation for autonomous systems

Autonomous system

- Gather environment information from sensors
- Filter, interpret & understand sensor data
- Safely choose actions
- Initiate actions

Sense | Perceive | Decide | Actuate

Arm Cortex-M | Arm Cortex-A | Arm Mali™ GPU and ML | Cortex-M

Arm Cortex-R
Autonomous vehicle security challenges

**ADAS / Autonomous Vehicle Controls Systems**
- Spoofed Hardware Identity
- Compromised ECU via SW Injection

**Connected Vehicle Services**
- Mobile Device Malicious Application Synchronization

**Firmware**
- Firmware Rollback
- Malicious Firmware Update

**Mobile applications**
- Malicious Mobile Applications Synchronization

**Vehicle Communication Busses**
- Injection Attack on Vehicle Communication Busses
- Data Capture / Sniffing Communication Busses

**Wireless Communications**
- MITM – Man in the Middle Attacks

**Integrated Vehicle Security**
- Integrated Attack on Keystore or KMS
- Weak Random Number Generation

**Connected Vehicle Services**
- Code Bugs or Non Secure Code Attack
- Download Attacks

Note: These characterizations are loose, subsystems may exist in multiple categories.
Framework to secure 1 trillion devices...

Platform Security Architecture

- **Analyse**
  - Threat models and security analyses

- **Architect**
  - Firmware architecture & hardware specifications

- **Implement**
  - Source code & hardware IP

PSA documents

Enabling products & contributions
Threat models and security analyses example

System description
- Autonomous vehicle

Assets
- Performance or infotainment data to be protected in integrity and confidentiality

Threats
- Remote software injection, physical, or replay attack

Security objectives
- Strong Crypto

Security requirements
- Hardware-based key store
Functional safety controls risks of hazards

Rising functional safety requirement

“Absence of unreasonable risk due to hazards caused by malfunctions”
Functional safety (FuSa) essential for automotive applications

- Processors suitable for use in FuSa systems
- Physical IP suitable for use in FuSa systems
- Certified software run-time components
- Software safety package for Arm Compiler 6
- Software test libraries (STLs) to verify a running system
- Cortex-R5 Safety certificate just received

Functional safety is an essential technology for automotive
Safety island concept

Combine “safety island” with application processors

- Optimised real-time capability for actuation
- Integrate checker functions into SoC
- Sits on independent power and clock rails to reduce common cause failures
- Manages overall safety for SoC
- Enables both high compute with high safety integrity
- Reduces BOM cost and footprint
Arm functional safety package

Safety manual
- Design and verification process
- Fault detection and control
- Verification summary

FMEA report
- Evidence of safety analysis on the Arm IP
- Aids partners with their own SoC level FMEA

Development Interface Report
- Interworking relationship
- Replaces conventional DIA
- Ambiguity avoidance
ISO 26262 defines
- Processes to follow
- Hardware/software performance to achieve
- Safety documentation to produce
- Software tools compliance process
<table>
<thead>
<tr>
<th>ID</th>
<th>PART</th>
<th>SUBPART</th>
<th>Failure Mode</th>
<th>#Gates</th>
<th>#Flops</th>
<th>λp</th>
<th>Sp %</th>
<th>λpd</th>
<th>λps</th>
<th>λpd %</th>
<th>λt</th>
<th>St %</th>
<th>λtd</th>
<th>λts</th>
<th>λtd %</th>
<th>DCP</th>
<th>SMp</th>
<th>DCl</th>
<th>SMt</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BUS</td>
<td>ITF</td>
<td>Wrong Data Transaction caused by a fault in the AHB interface</td>
<td>836</td>
<td>23</td>
<td>0.010</td>
<td>0.26</td>
<td>0.007447</td>
<td>0.00252</td>
<td>100.00%</td>
<td>0.039099</td>
<td>40%</td>
<td>0.023459</td>
<td>0.015639</td>
<td>100.00%</td>
<td>30%</td>
<td>E2E</td>
<td>30%</td>
<td>E2E</td>
</tr>
<tr>
<td>2</td>
<td>DECODER</td>
<td></td>
<td>Incorrect Instruction Flow caused by a fault in the decode logic</td>
<td>326</td>
<td>9</td>
<td>0.004</td>
<td>0.01</td>
<td>0.003885</td>
<td>0.00000</td>
<td>100.00%</td>
<td>0.015298</td>
<td>15%</td>
<td>0.013003</td>
<td>0.002225</td>
<td>100.00%</td>
<td>60%</td>
<td>CTRL FLOW, WD</td>
<td>60%</td>
<td>CTRL FLOW, WD</td>
</tr>
<tr>
<td>3</td>
<td>VIC</td>
<td></td>
<td>Un-intended execution/not executed interrupt request</td>
<td>141</td>
<td>4</td>
<td>0.002</td>
<td>0.26</td>
<td>0.001256</td>
<td>0.00044</td>
<td>100.00%</td>
<td>0.006793</td>
<td>40%</td>
<td>0.004076</td>
<td>0.002717</td>
<td>100.00%</td>
<td>60%</td>
<td>INT MONITOR</td>
<td>60%</td>
<td>INT MONITOR</td>
</tr>
<tr>
<td>4</td>
<td>CPU</td>
<td></td>
<td>Corrupt data or value caused by a fault in the register bank shadow</td>
<td>7465</td>
<td>206</td>
<td>0.018</td>
<td>0.01</td>
<td>0.017841</td>
<td>0.00018</td>
<td>20.13%</td>
<td>0.069709</td>
<td>15%</td>
<td>0.059252</td>
<td>0.010456</td>
<td>19.81%</td>
<td>60%</td>
<td>PARITY</td>
<td>60%</td>
<td>PARITY</td>
</tr>
<tr>
<td>5</td>
<td>CPU</td>
<td></td>
<td>Incorrect Instruction Result caused by a fault in the multiple</td>
<td>0.009</td>
<td>0.01</td>
<td>0.009996</td>
<td>0.00009</td>
<td>10.15%</td>
<td>0.035685</td>
<td>15%</td>
<td>0.030332</td>
<td>0.005353</td>
<td>10.15%</td>
<td>90%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>CPU</td>
<td></td>
<td>Incorrect Instruction Result caused by a fault in the divider</td>
<td>0.002</td>
<td>0.01</td>
<td>0.002229</td>
<td>0.00002</td>
<td>2.51%</td>
<td>0.008508</td>
<td>15%</td>
<td>0.007232</td>
<td>0.001276</td>
<td>2.42%</td>
<td>90%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>CPU</td>
<td></td>
<td>Incorrect Instruction Result caused by a fault in the adder</td>
<td>0.002</td>
<td>0.01</td>
<td>0.001256</td>
<td>0.00035</td>
<td>1.42%</td>
<td>0.006779</td>
<td>15%</td>
<td>0.005763</td>
<td>0.001017</td>
<td>1.93%</td>
<td>90%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>CPU</td>
<td></td>
<td>Corrupt data or value caused by a fault in the register bank shadow</td>
<td>0.030</td>
<td>0.01</td>
<td>0.029292</td>
<td>0.00030</td>
<td>33.09%</td>
<td>0.115579</td>
<td>15%</td>
<td>0.098242</td>
<td>0.017337</td>
<td>32.85%</td>
<td>90%</td>
<td>STL</td>
<td>90%</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>CPU</td>
<td></td>
<td>Incorrect Instruction Flow caused by a fault in the pipeline controller</td>
<td>0.029</td>
<td>0.01</td>
<td>0.028984</td>
<td>0.00029</td>
<td>32.70%</td>
<td>0.115579</td>
<td>15%</td>
<td>0.098242</td>
<td>0.017337</td>
<td>32.85%</td>
<td>90%</td>
<td>CTRL FLOW, WD</td>
<td>40%</td>
<td>CTRL FLOW, WD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>FETCH</td>
<td></td>
<td>Incorrect Instruction Flow caused by a fault the branch logic (Wrong Branch Prediction)</td>
<td>1606</td>
<td>44</td>
<td>0.001</td>
<td>0.01</td>
<td>0.001025</td>
<td>0.00001</td>
<td>5.35%</td>
<td>0.003422</td>
<td>15%</td>
<td>0.002908</td>
<td>0.015639</td>
<td>4.86%</td>
<td>25%</td>
<td>STL, WD</td>
<td>15%</td>
<td>WD</td>
</tr>
<tr>
<td>11</td>
<td>BUS</td>
<td></td>
<td>Incorrect Instruction Flow caused by a fault the fetch logic</td>
<td>0.018</td>
<td>0.01</td>
<td>0.018115</td>
<td>0.00018</td>
<td>94.65%</td>
<td>0.071387</td>
<td>15%</td>
<td>0.060679</td>
<td>0.015639</td>
<td>95.42%</td>
<td>19%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Diag. Cov.**: HW REDUNDANT RANGE CHK

**HW Safety Mechanism**:

1. An SM can cover more than one FM
2. One FM can be covered by multiple SMs
Automotive SoC verification challenges

- Systematic Failure Verification
- Concurrent SW Development
- Requirements Traceability
- Use Case Verification
- Performance Verification
- Security Verification
- Automotive Protocol Verification
- Mixed Signal Verification
- Functional Safety Verification
- Random Failure Verification
Automotive Functional Safety challenges

- Safety-Certified IP
- Failure Mode Definition
- Safety Mechanism Design
- Fault Campaign Planning
- Fault Reduction
- Safety Requirement Traceability
- Fault Execution
- Re-use of FV Environment
- Metric Calculation

**ADAS SoC Example**

Multiple verification engines and FMEDA Integration
Safety verification solution

- Unified functional + safety verification flow and engines
- Integrated fault campaign management across formal, simulation, and emulation
- Common fault results database unifies diagnostic coverage
- Proven requirements traceability, enabling FMEDA integration
TUV SUD ISO 26262 certified documentation kits with TCL1 level confidence

- TCL1 reflects the highest confidence that tool malfunctions will not cause violations of safety requirements

- A tool-chain that evaluates to TCL1 will reduce the complexity, cost, and time required of our customers to certify their work products

Use Case 1
Use Case 2
Use Case 3
... Use Case n

Can a safety violation be caused by the tool?

Prove tools do not cause a safety issue
Summary

Complex and demanding compute requirements

Increasing need for security

Rising functional safety requirement
Thank You!
Danke!
Merci!
谢谢!
ありがとう!
Gracias!
Kiitos!
감사합니다
धन्यवाद