Embedded Multi-Core Systems for Mixed Criticality Applications in dynamic and changeable Real-time Environments

Artemis Technology Conference

October 6, 2016

MADRID

INFINEON (UK)

Zuhal Clarke

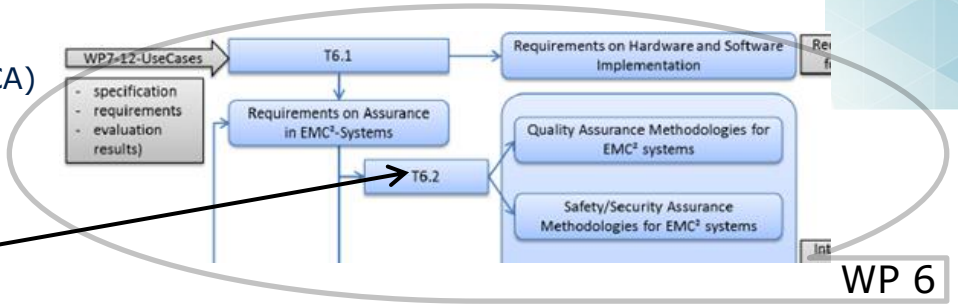
CONTENT

- EMC2 WP Partitioning
- EMC2 Goals as concept
- EMC2 Goals in Living Labs
- Demonstration Platform Introduction
- Platform Development Lifecycle

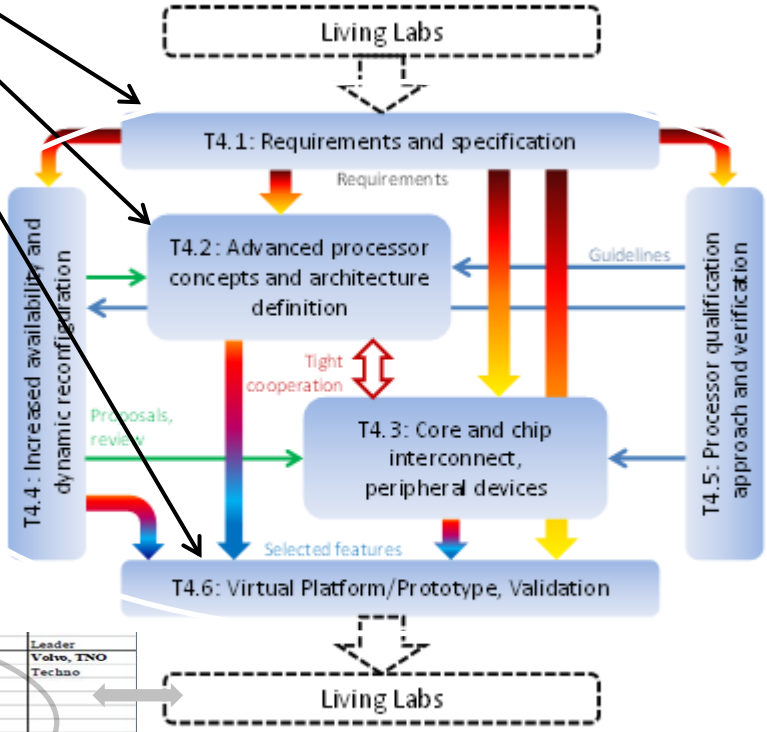
HOW IT WORKS (As defined in EMC2 TA and CA)

- Total : 203 MM (21.6 MM Subcontract)

IFGB → D6.10



WP 6 › 36 MM



WP 4 › 157 MM

ST7.1.6 UC_Methodology

WP7		Description	Leader
		LL_Automotive Applications	Volvo, TNO
	T7.1	UC_ADAS and C2X	Techno
		ST7.1.1 UC_Requirements and specification	
		ST7.1.2 UC_Define system architecture	
		ST7.1.3 UC_Implementation and integration	
		ST7.1.4 UC_Elaborate use cases	
		ST7.1.5 UC_Set up and execute trials	
		ST7.1.6 UC_Methodology	
		ST7.1.7 UC_Evaluation	

WP7

Figure 39: Task layout of WP4

WP 13 / Dissemination-exploitation (10MM)

OBJECTIVES of INFINEON with EMC2

Problem: Early Mixed Critical Embedded system development

Systems are evolved into the "multi-core world", software partitioning, resource availability, bottlenecks, integration and software debug become increasingly difficult problems to solve.

Solution:

1. System representation of pre-silicon device
    1. Reflecting system complexity
    2. Containing safety critical mechanisms
    3. Allowing software development and execution

Use Emulation Platforms

Emulation Platform

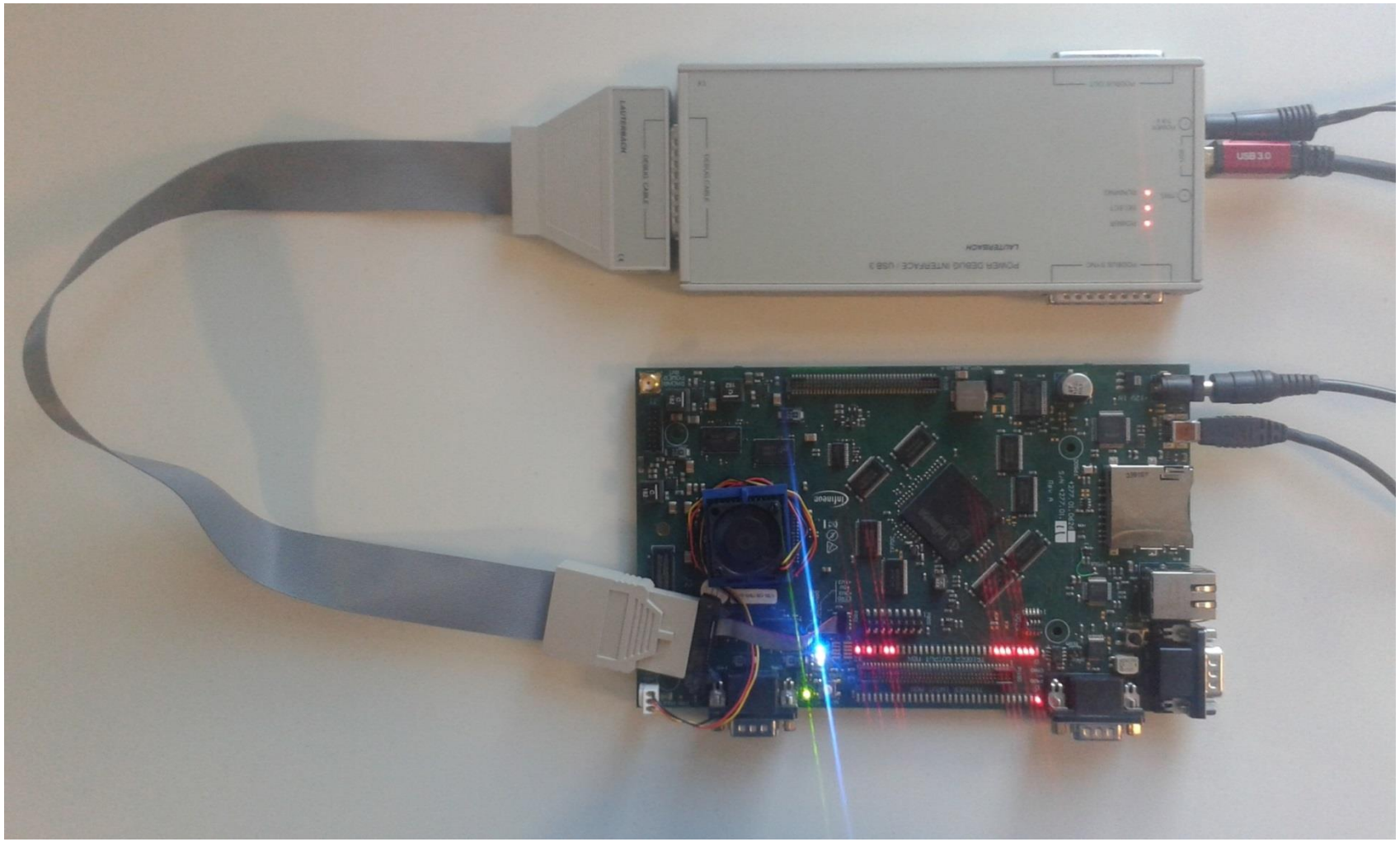
Concept of Emulation Platforms

1. A hardware platform where the multicore system usage of the SOC device and complexity can be demonstrated in advance of a Silicon product. Avoiding high cost of error
2. Provide an accurate software target for application developers

This is achieved by the use of FPGA (Field Programmable Gate Array) technology

A virtual SOC platform that allows mixed critically hardware and software to be emulated in advance of commitment to manufacture

Typical Virtual Platform



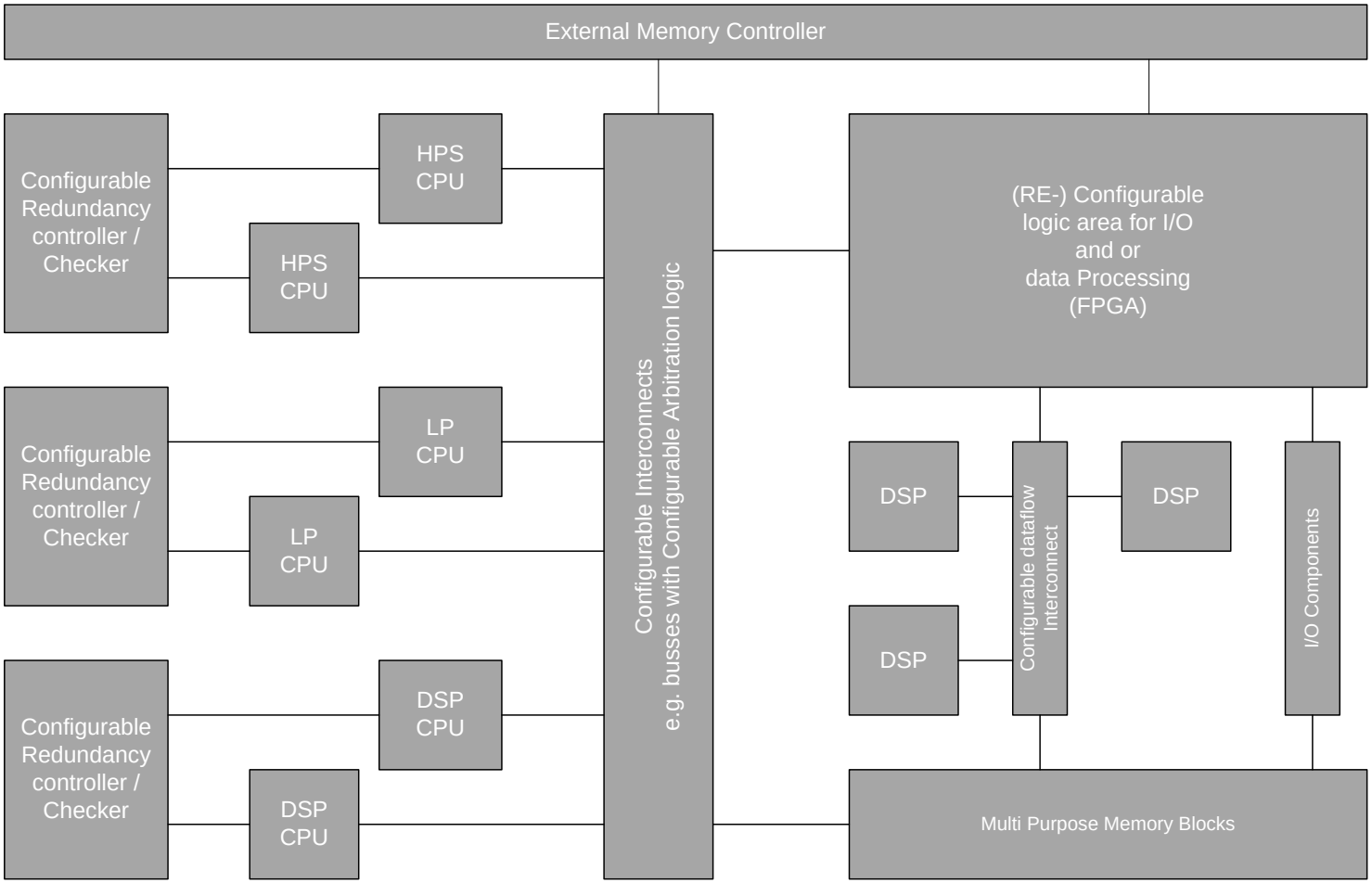
Using Emulation Platforms resolve the following issues :

- System configuration flexibility
- A faster "Simulation" environment x100+ w.r.t RTL simulation
- Real time behaviour demonstrated – Contention, Bottlenecks and system interfaces
- Critical interactions between hardware and software validated
- Early software development process – significant reduction is R+D costs
- Significant Risk reduction in real time multicore platform development – integration into primary application

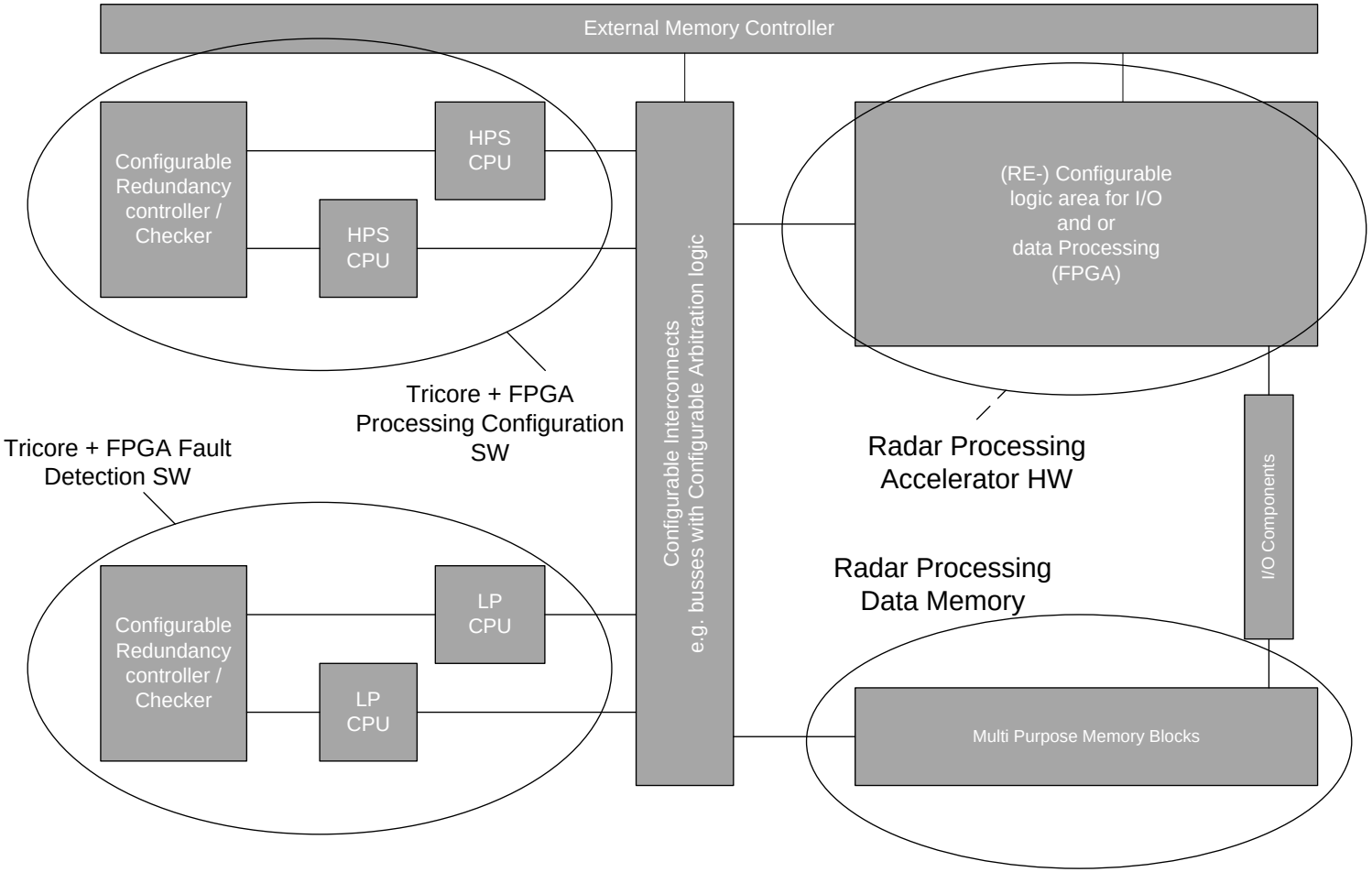
IFX LIVING LAB - EMC2 DEMONSTRATION GOALS

- Demonstration of Heterogeneous Multiprocessor SOC Architecture
- Development of safety critical ADAS Radar Processing Architecture
- Validation Fault detection Hardware and Software Methodology

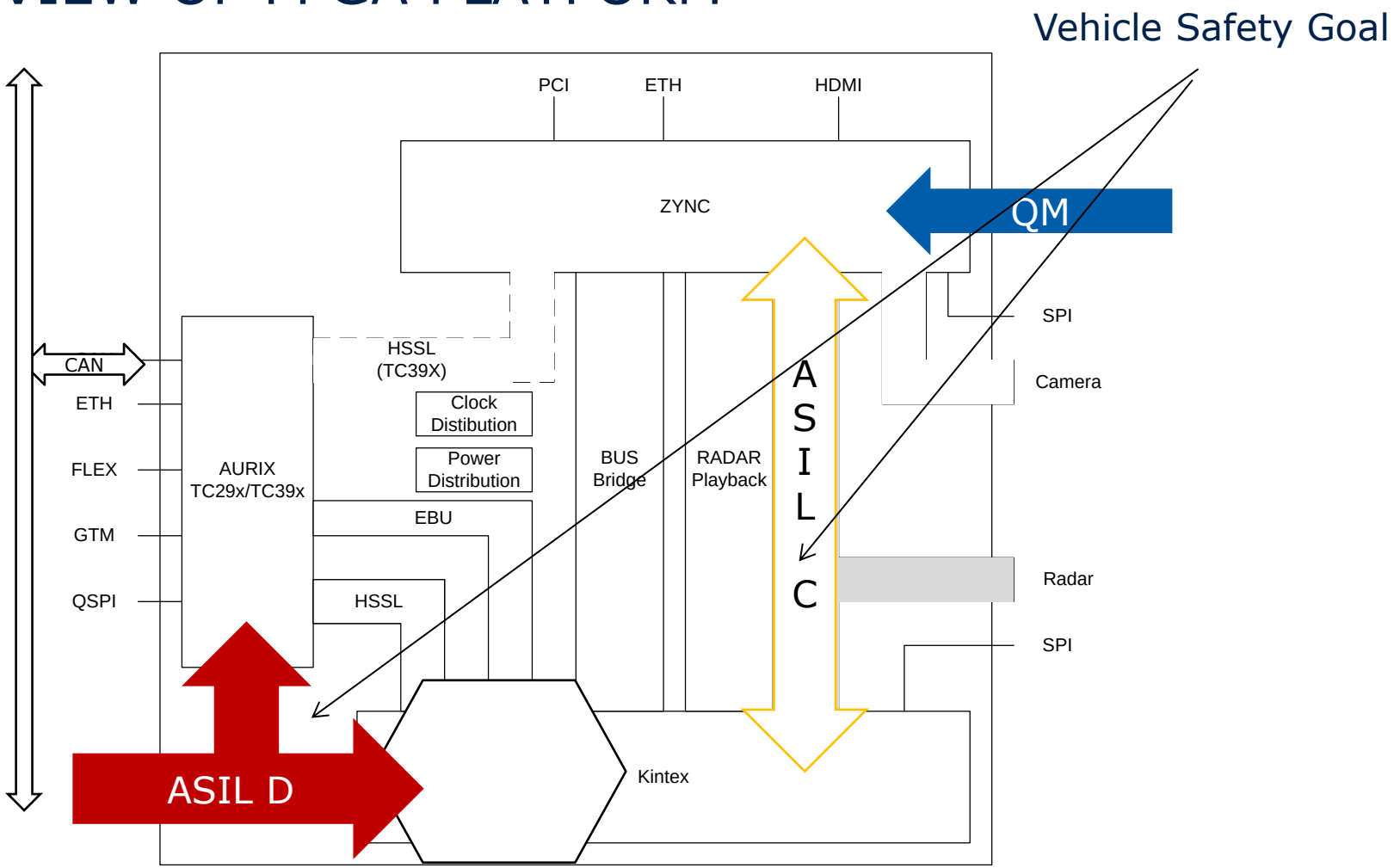
EMC2 GENERIC HARDWARE TEMPLATE



IFX Hardware / Software Application Mapping



OVERVIEW OF FPGA PLATFORM



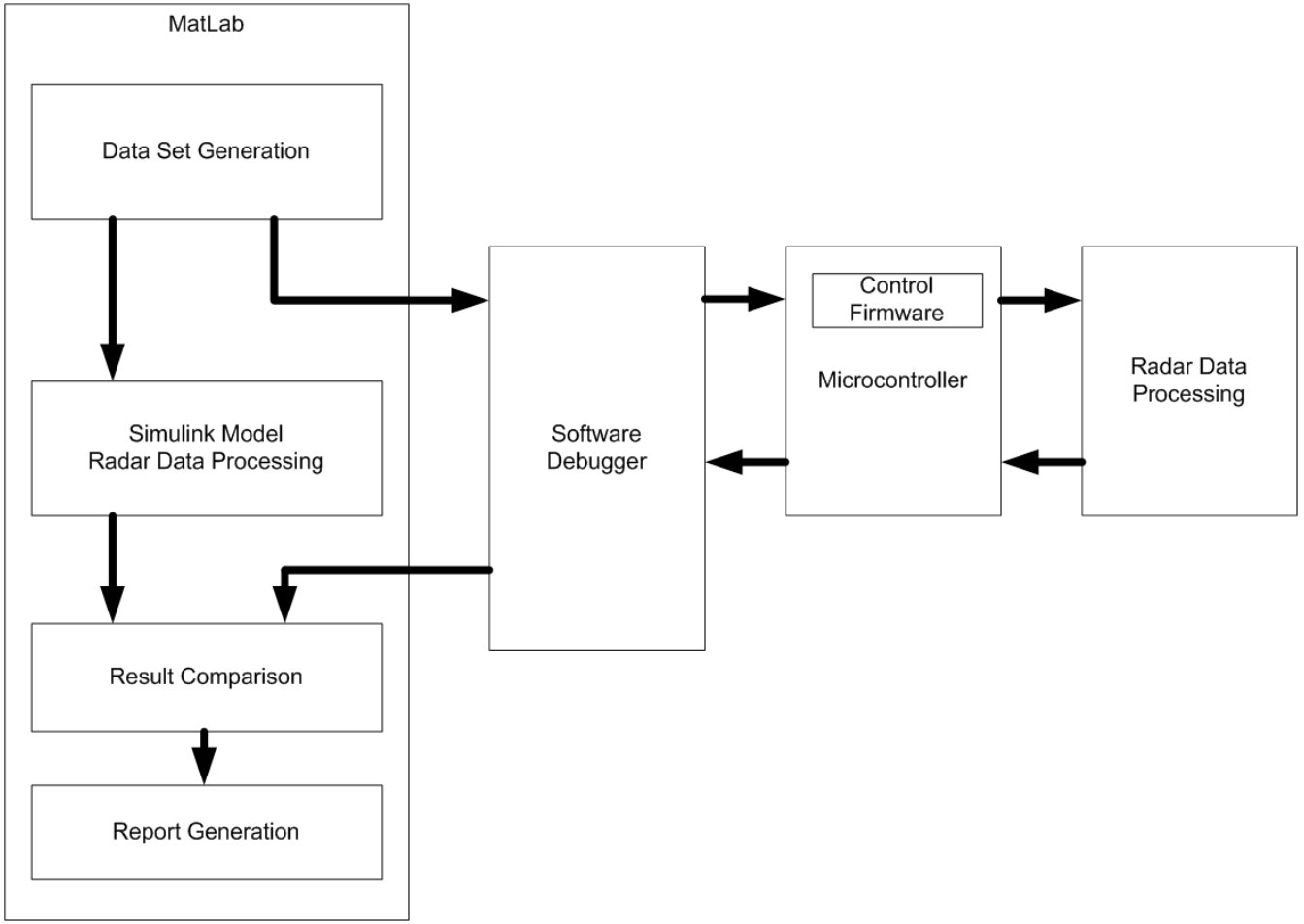
HARDWARE UPGRADES

- To use application boards to capture radar data in real time.
- To replay radar & Visual data, captured from vehicles or fixed objects, under lab conditions
- 40MS/channel radar data at 16 bit/channel with up to 8 channels.
- validating Radar processing upgrades against “Real world” data sets.

HARDWARE VALIDATION

- Firmware ensures configure connection to FPGA
- Data sets are register sets are transferred to FPGA under control from Matlab
- Data is processed by FPGA hardware units.
- Matlab compares hardware results against Model results

HARDWARE VALIDATION

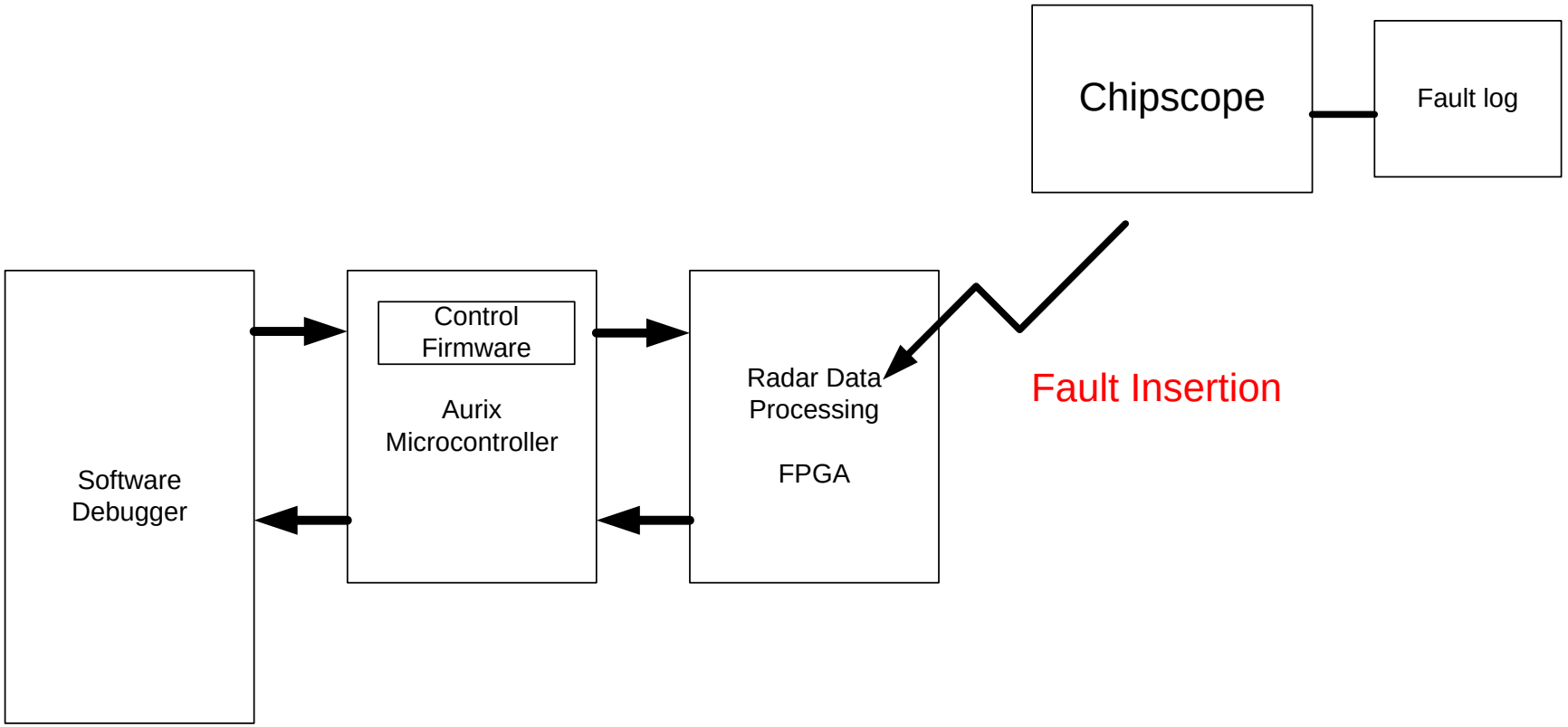


Safety Measures – Fault Injection & Detection

- Firmware provide schedule for data sets for SPU processing.
- Data sets ensure high level of coverage and fault sensitivity
- CRC provides data and register paths for error detection

- Chipscope Virtual I/O is used to model hardware ‘stuck at’ fault.

FAULT INSERTION & DETECTION



Conclusion

- To support physical vehicle systems in development – at customer or partner site
- Understanding risks in implementing and interfacing with real components
- Build up a durable and realistic Test Environment and explore risks in verification and validation of system operation
- Test real-time response of Safety Mechanisms and interference from non-critical applications
- Built a Reference for Safety Performance Measures providing basis for certification
- Exercise Safety Measures and feedback for Hardware Design Changes
- Exercise Safety Measures and feedback for embedded Software Development
- Speed up System SW development at the customer site prior to Silicon release