Italian Workshop on Embedded Systems
Siena, Italy, September 13-14 2018

A safety-oriented engineering process for autonomous robotic systems

Fabio Federici, Giulio Mosé Mancuso

United Technologies Research Center

Overview

- UTC: BU needs and supporting capabilities
- Certification issues
- Proposed design flow
- Technology Evaluation
- Open points

UTC and intelligent systems

UTC Business Units

UTC Aerospace Systems

- Actuation & Propeller Systems
- Air Management Systems
- Landing Systems
- Electric Systems
- Engine Systems
- Sensors & Integrated Systems

UTC Climate, Controls and Security

- Intelligent building Technologies
- Heating & Cooling
- Fire Safety & Security
- Refrigeration

Pratt & Whitney

- Commercial and Military Aircraft Engines
- Auxiliary Power Units
- Helicopter Engines

OTIS

- Elevators
- Escalators
- Moving Walkways

Use Cases

Inspection

Assembly

Manipulation

Grinding

Deburring

Welding

Mapping

Autonomous Transportation

Capabilities

3D Reconstruction

Dense Mapping

Visual Inspection

Perception

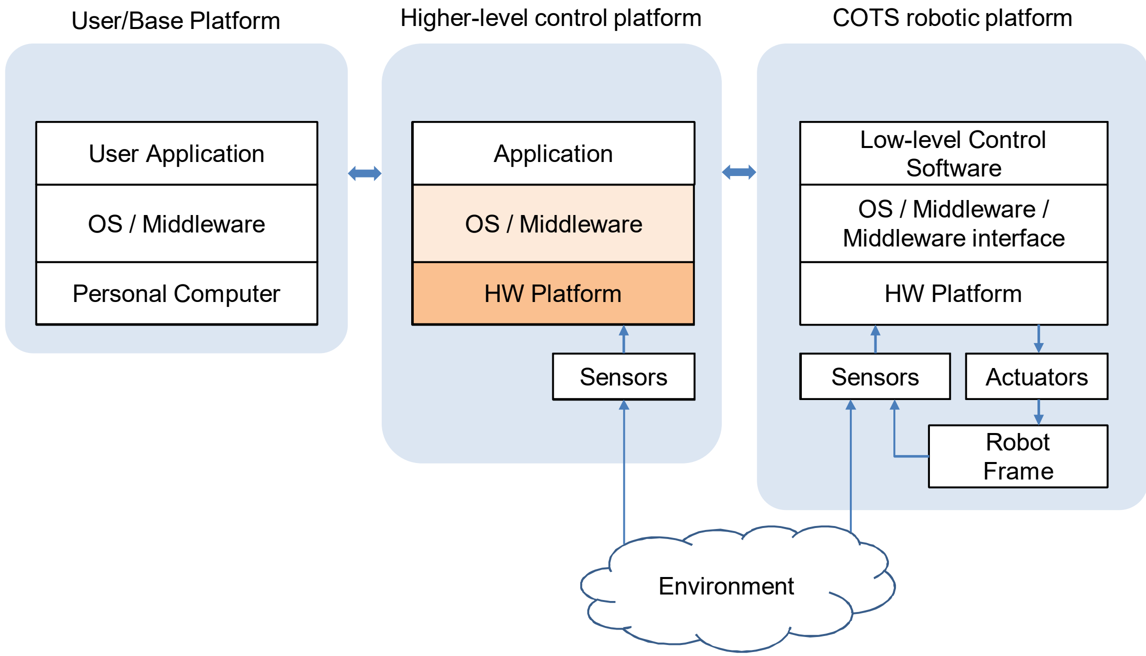
Navigation

Autonomous Exploration

Manipulation

Activity Prediction

Focus area



Example

Ground Control Station

High-level Controller

Flight Controller/ Quadcopter Frame

Relevant standards

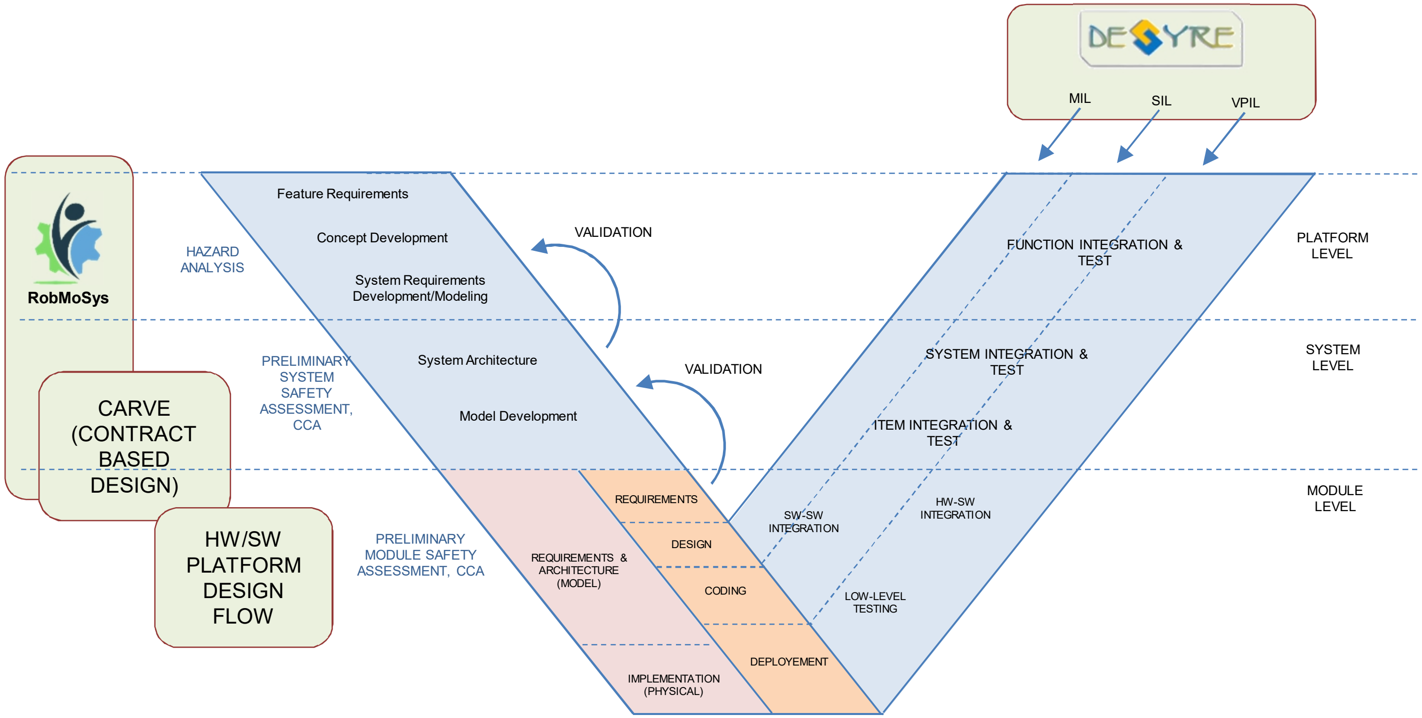
Safety related certification

- IEC 61508: Functional safety of Electrical/Electronic/Programmable Electronic Safety-related Systems

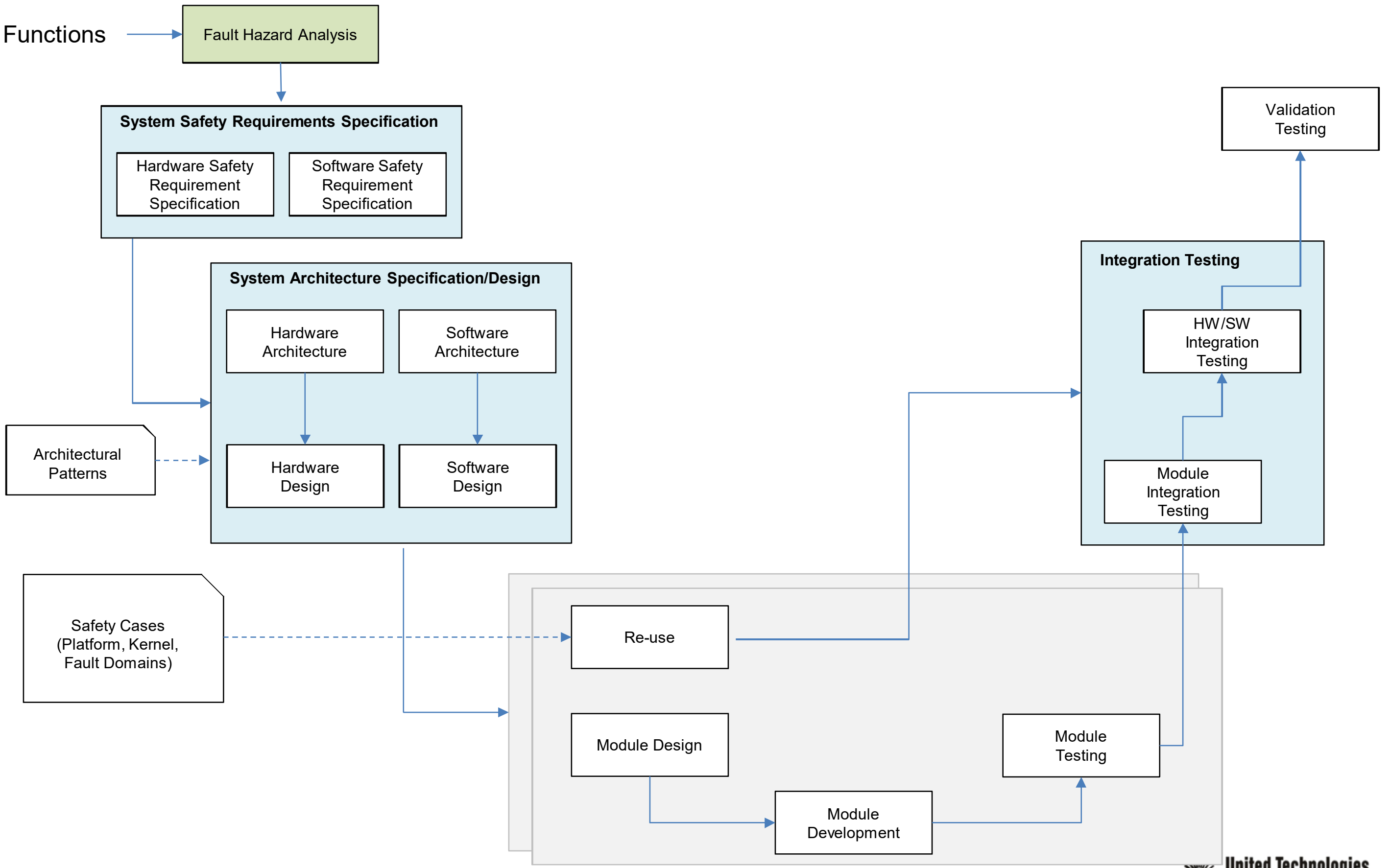
- SAE ARP 4765A: Guidelines For Development Of Civil Aircraft and Systems
    - RTCA DO 254
    - RTCA DO 178C

- ISO 10218-1: Safety requirements for industrial robots - Part 1: Robots
- ISO 10218-2: Safety requirements for industrial robots -- Part 2: Robot systems and integration
- ISO 13482: Safety requirements for personal care robots

Design and verification flow



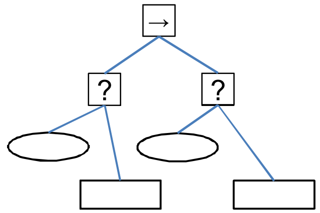
HW/SW Platform Design Flow



Robotics Architecture Design Patterns

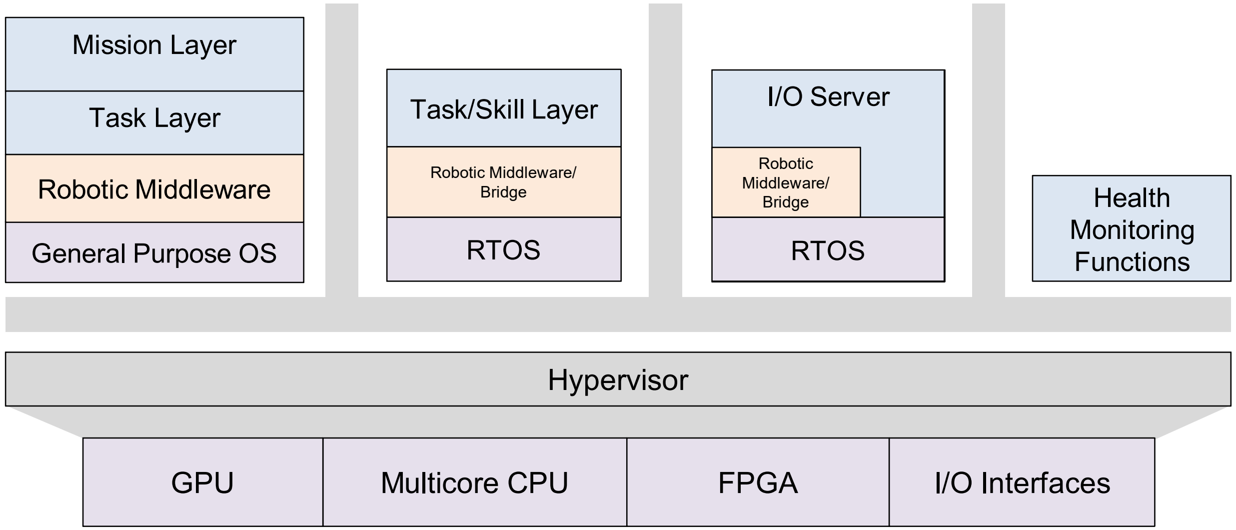
DELIBERATIVE LAYER	MISSION
SEQUENCING LAYER	TASK
	SKILL
SKILL LAYER	SERVICE
	FUNCTION
	EXECUTION CONTAINER
	OS/MIDDLEWARE
	HARDWARE

CARVE: use of behavior trees



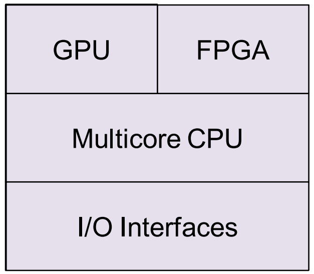
Internal research investigation

Development of HW/SW Platform



Current collaborations: UNIMORE UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA; HiPeRT Lab High-Performance Real-Time Lab

Heterogeneous platforms



Goal: use of COTS heterogeneous devices

- Low-cost
- Short time to market

Problems:

- Sophisticated (obfuscated) components
- Greater complexity
- Resource sharing potentially jeopardizing safety

TARGET PLATFORMS

NVIDIA Jetson TX2 System-on-Module

- Quad-core ARM Cortex A-57
- Dual-core NVidia Denver 2
- NVidia Pascal GPU w. 256 CUDA cores

Zynq UltraScale+ MPSoC

- Quad-core ARM Cortex A-53
- Dual-core ARM Cortex-R5
- ARM Mali 400 MP2 GPU
- 16 nm FinFET+ Programmable Logic

Need for efficient middlewares

ROS

Open-source, meta-operating system for robots Hardware abstraction,

- Low-level device control,
- Commonly-used functionality,
- Message-passing between processes,
- Package management.

Pros:

- Widely adopted
- Large community
- Out of the box support for devices
- Algorithms & Libraries

Cons:

- Lack of determinism
- Not well fit for safety critical systems

2

Fork of ROS based on the Data Distribution Service (DDS).

- DDS is suitable for real-time distributed embedded systems due to its various transport configurations (e.g., deadline and fault-tolerance) and scalability.

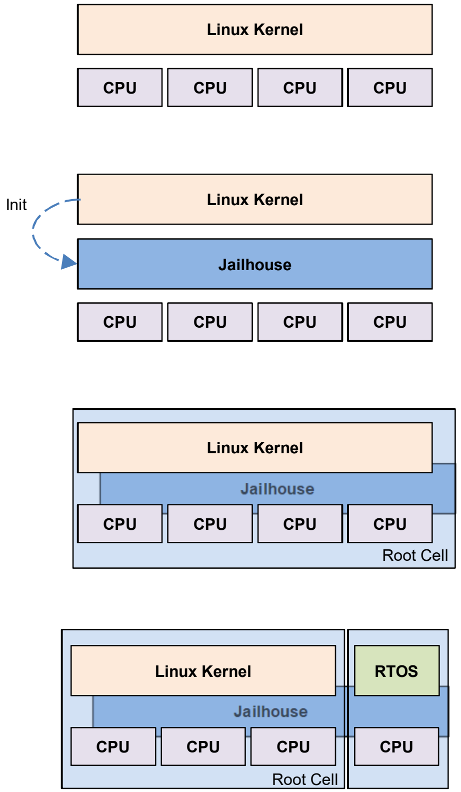
Pros:

- Real-time, deterministic
- Support for multiple communication middlewares
- Compatibility with ROS

Cons:

- Maturity level
- Adoption

Jailhouse partitioning hypervisor



Jailhouse:

- Partitioning Hypervisor based on Linux.
  - Able to run bare-metal applications or (adapted) operating systems.
- Originally developed by Siemens
- Released as Free Software (GPLv2) since November 2013

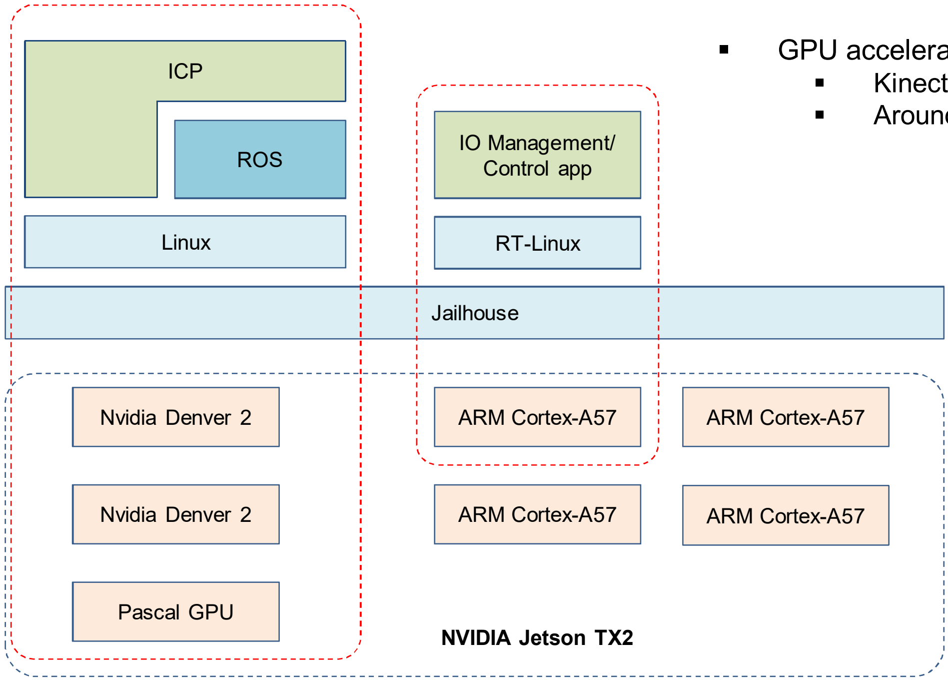
Pros:

- Native support for the Linux kernel
- Low latencies, good performance
- Open Source (GPL v2)
- Ported on several embedded platforms (Xilinx Zynq, Nvidia Jetson TX1/TX2)

Limitations:

- System boot depends on the Linux Kernel
- No partition scheduling, only static resource assignment
- Limited maturity

Ongoing activity on demo Platform



- Root cell running ROS executed on the Denver Cluster
- GPU accelerated ICP:
  - KinectFusion algorithm
  - Around 108 Hz execution speed

Summary and Open Points

Activities

- Definition of a safety oriented flow for robotics systems
- Analysis and design of a robotic hardware/software architecture
- Assessment of open-source technologies

TODOs & Open points

- Consolidation of MBD flow
  - Bringing in RobMoSys approach
- Additional isolation mechanism to be introduced in Jailhouse
  - Long-term need: mature, certifiable hypervisor
- Verification

Questions?