AEROSPACE
Assuring Space Mission Success

Development and Debugging of FPGA-Based Hardware-in-the-Loop Simulation Systems

Dr. Martin Panevsky
Embedded System Applications Manager

Zeve Akerling
Advanced Degree Fellow

Xingui Zhao
Engineering Specialist

Embedded Control Systems Department
The Aerospace Corporation

Flight Software Workshop
December 10-12, 2013
California Institute of Technology

Outline

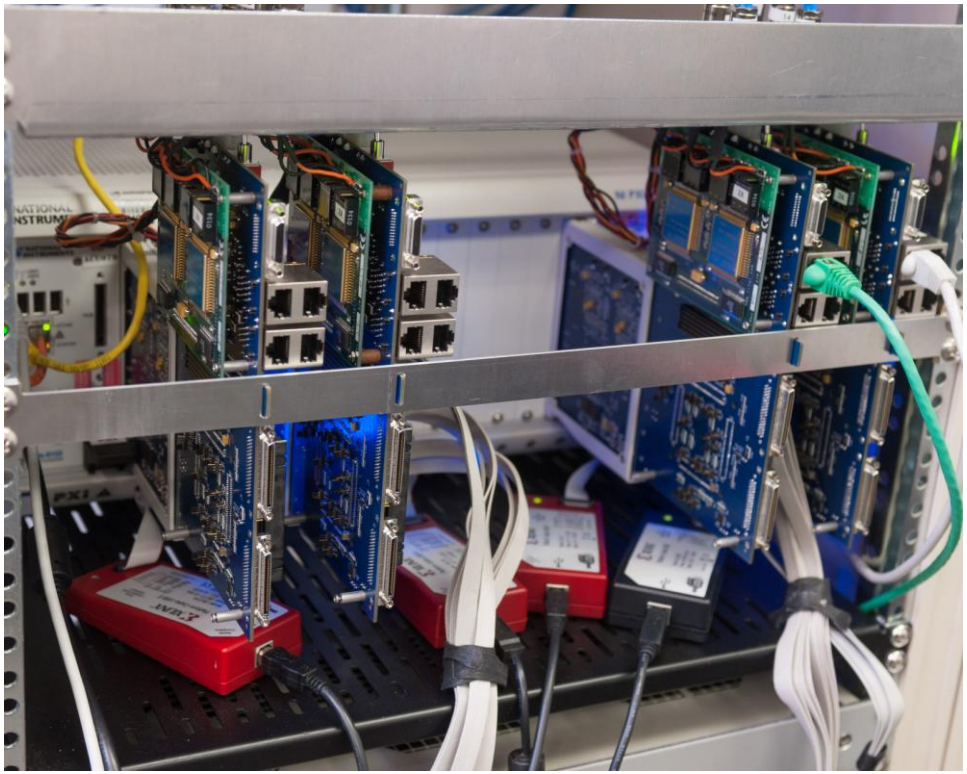
- Background
- FPGA System Architecture paradigm
- Modular architecture of FPGA firmware
- Implications of High-Level Synthesis Tools for VHDL and Bitfile Generation
- FPGA ASIC Emulation
- Leveraging the FPGA Technology for a Breakthrough in Avionics System Debugging, Integration, and Insight

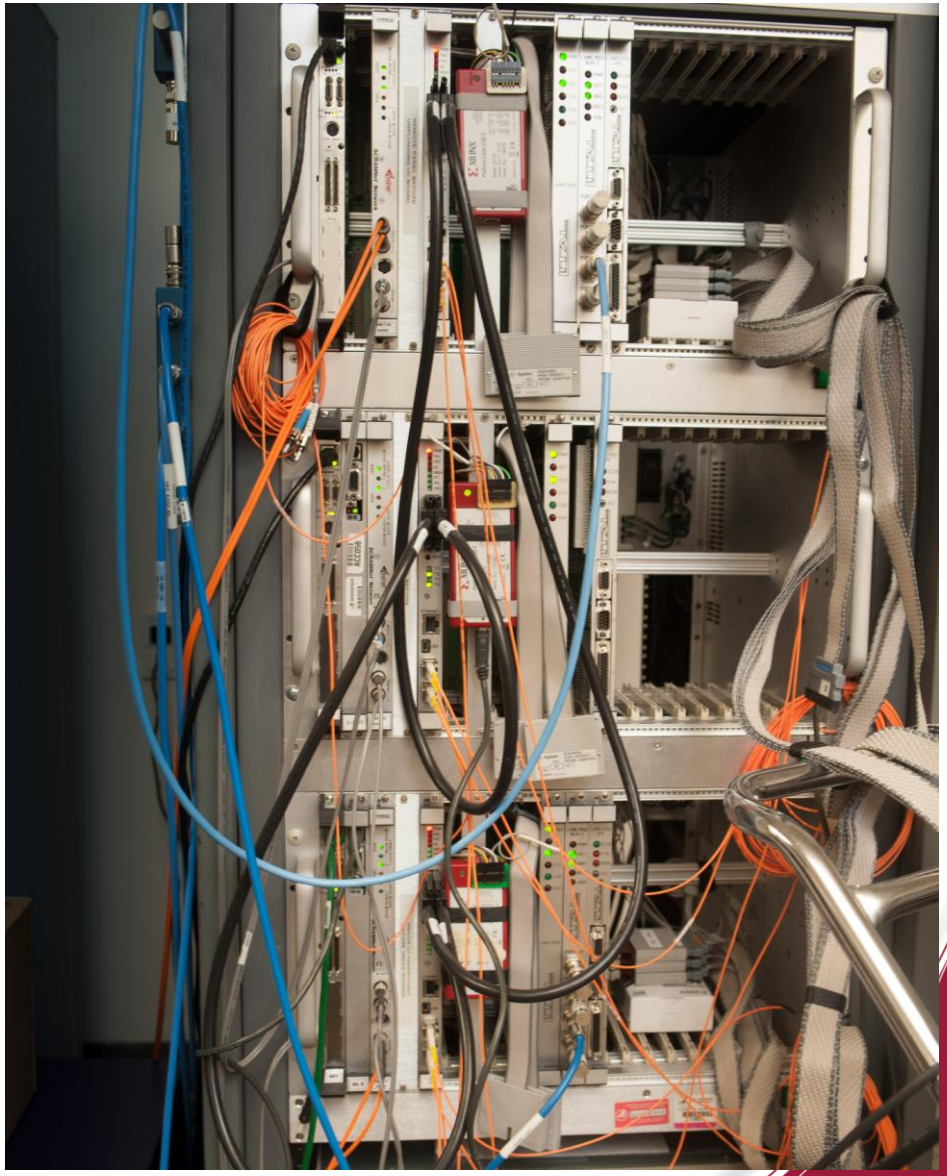
Background

- Many of the components for any Avionics HIL Simulation system are standard and reoccurring
- The need to interface with components that are unique to a specific avionics system leads to custom, expensive long lead hardware
- Traditionally this results in avionics HIL simulation systems that are costly, inflexible, and time-consuming to build
- Modern COTS hardware combined with FPGA technology create a new paradigm where most of the hardware is shifted in the COTS direction
- The FPGA technology is the interface between COTS and custom avionics hardware

FPGA-Based HIL Evolution

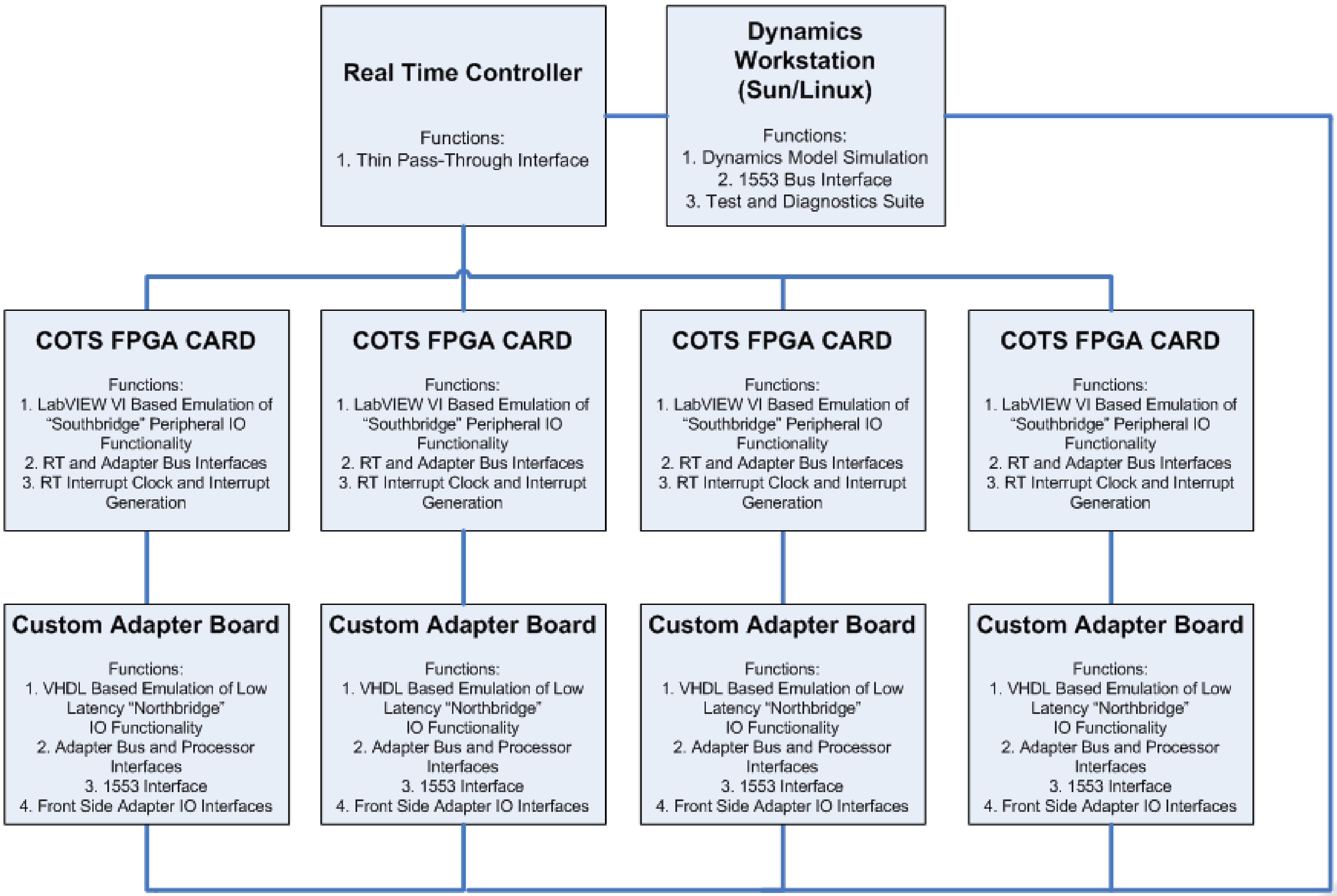
- Legacy HIL has fixed custom CPU boards and fixed custom I/O boards
- FPGA HIL has flexible custom CPU boards, flexible COTS I/O boards, and COTS software environment





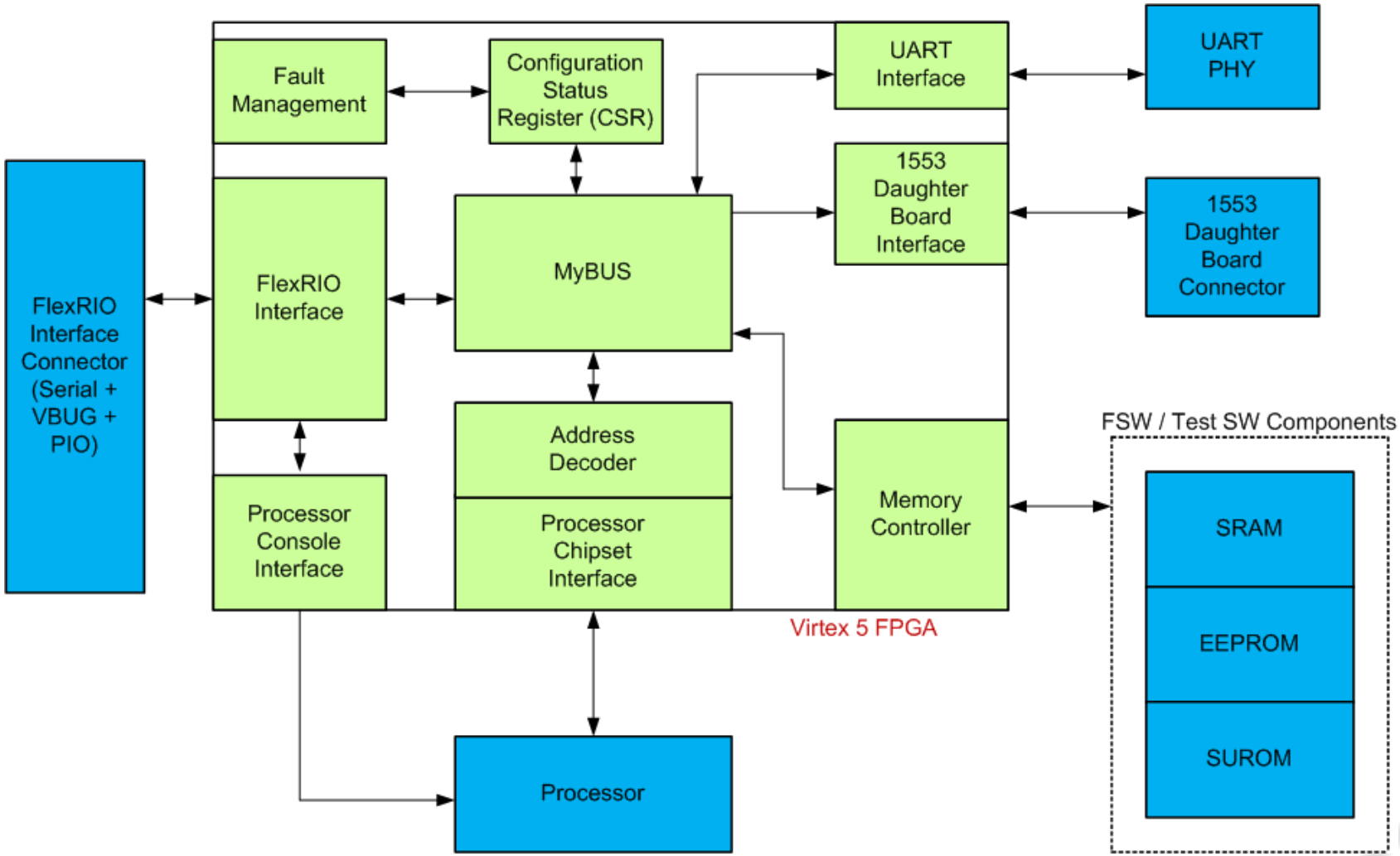
COTS FPGA Hardware Combined With the Minimum of Custom Hardware Creates a Cost-Effective and Flexible HIL Simulation

FPGA-Based HIL System Architecture



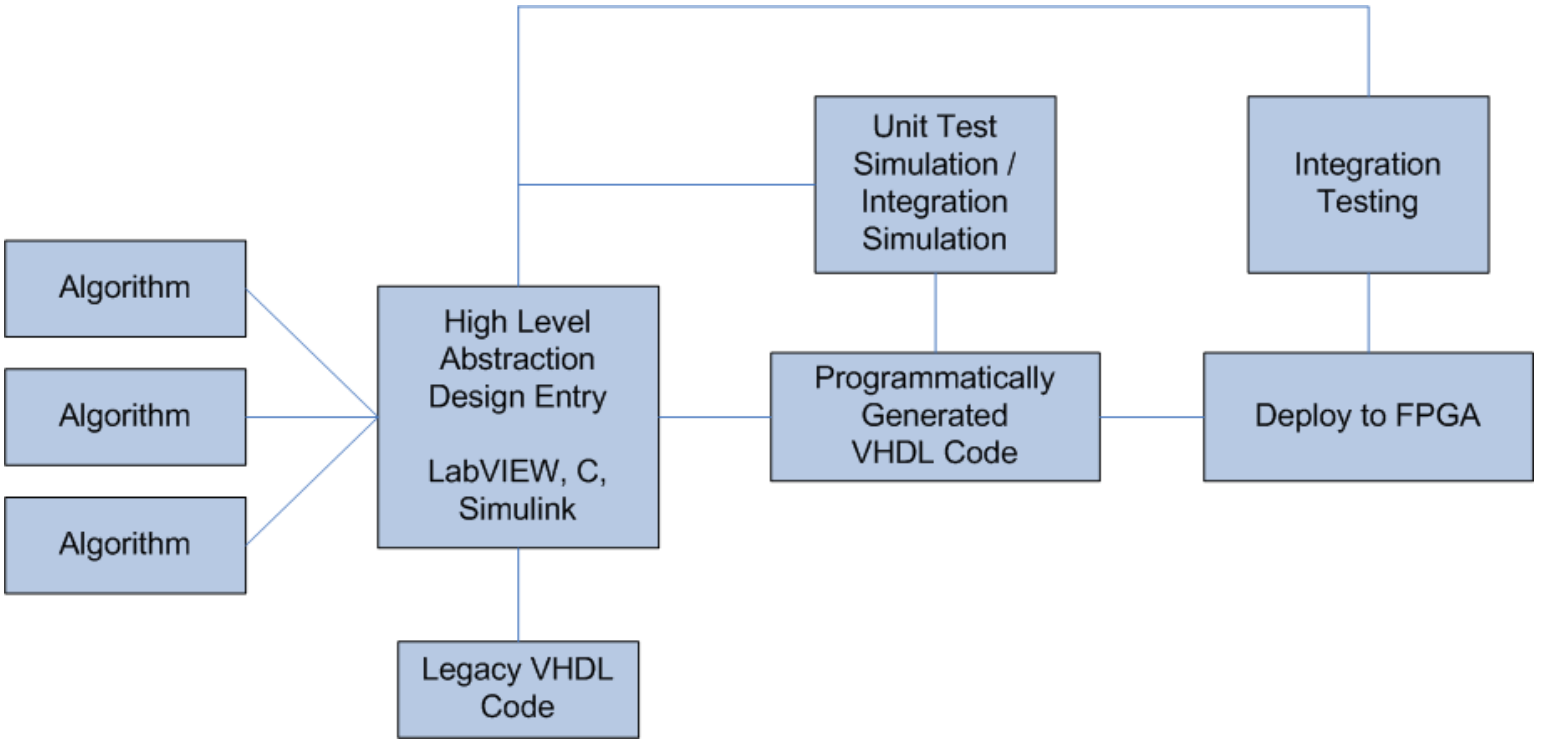
The FPGA technology is the "glue" that binds the system together

FPGA Firmware Modular Design



FPGA firmware modular architecture allows for standardization and reuse of components

High-Level Tools for VHDL Synthesis



- High-level COTS tools dramatically simplify the traditional challenge of FPGA programming
- Organic modular design facilitates code reuse

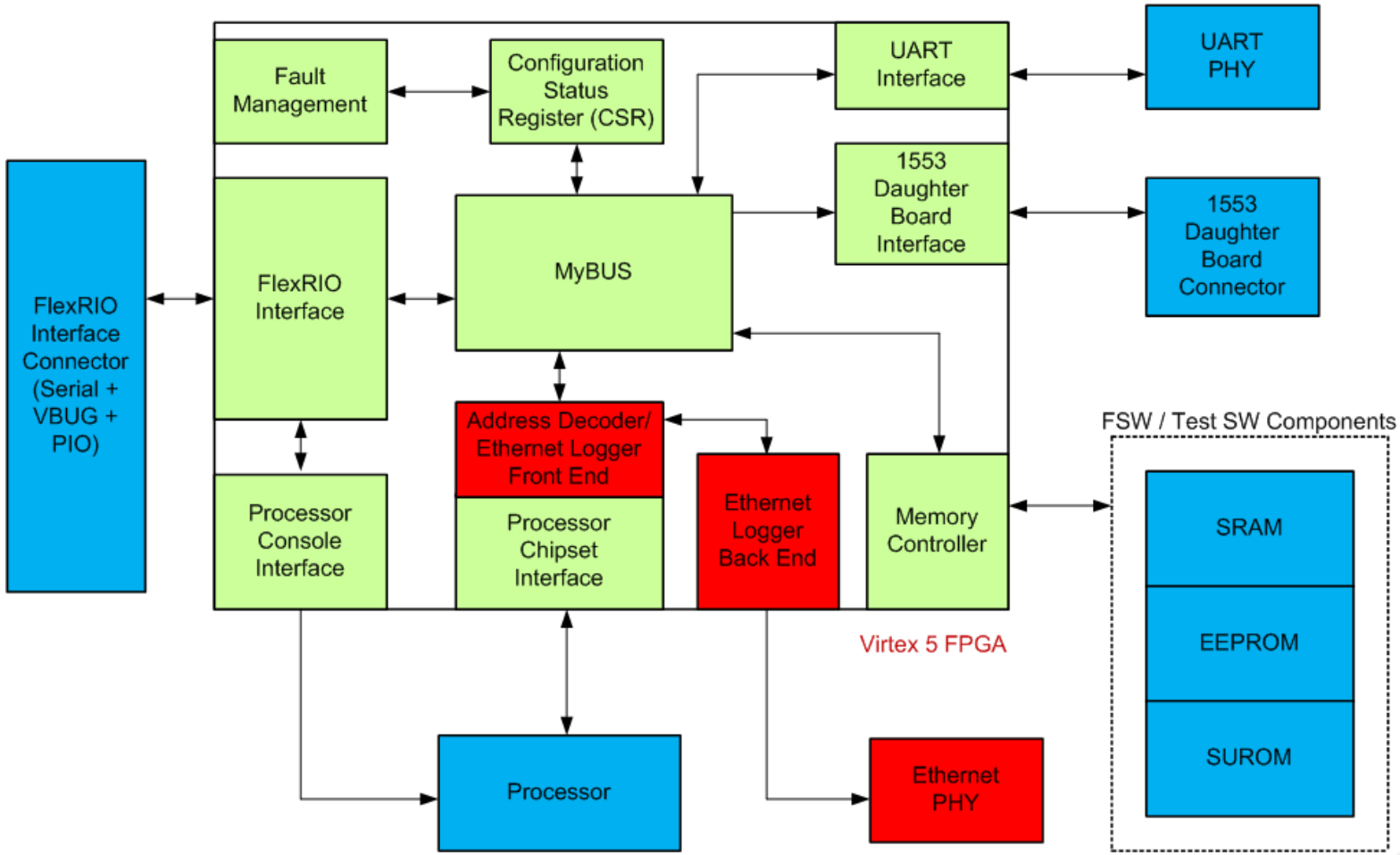
High-level FPGA synthesis tools allow various subject matter experts to directly code their own designs in VHDL

Applications: ASIC Emulation Using FPGA Technology

Purpose:

- Coding FPGA VHDL to ASIC specifications in order to validate and verify functionality before or after the ASIC has been fabricated
- The VHDL proof of concept ASIC design can be directly transitioned to an operational hardware ASIC when the VHDL code is deployed to a radiation hardened FPGA chip
- The validated ASIC design can be used to address parts obsolescence when implemented in a radiation hardened FPGA
- FPGA implementation of an ASIC design provides the ability to test and stress the system by recreating almost any conceivable hardware anomaly in the FPGA firmware

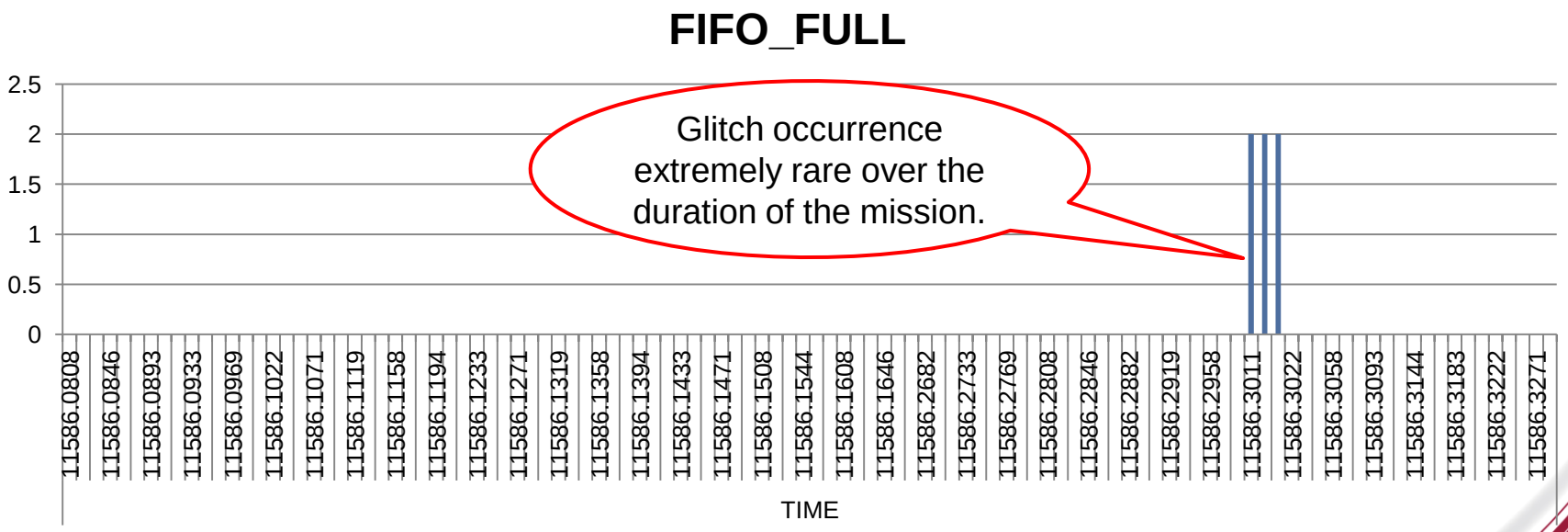
FPGA-Based System Data Logger for Debugging, Anomaly Detection and Prediction



FPGA-Based HIL enables a new paradigm of system integration and debug

FPGA-Based System Data Logger for Debugging

- System logic analyzer built into the FPGA logic
- Every CPU transaction, bus communication, and hardware interaction is recorded and archived to disk in real-time for the duration of the mission
- Post-processing and analyzing the data is invaluable for system debugging
- Ability to record and discern rare glitches that happen only once a mission



System-level avionics tracing and debugging of entire missions

FPGA-Based System Data Logger for Anomaly Detection and Prediction

- Ability to capture rare events or events that happen once and randomly in a mission – very difficult to capture with traditional tools
- Data post-processing and plotting
- Data analysis is the first step towards creating predictive models of the system behavior
- Creating statistical regression models for anomaly and failure prediction
- Creating statistical models for requirements violation prediction

FPGA-based system logic analyzer provides powerful debugging and predictive capabilities to the system developer

Future Work: Incorporating Elements of Artificial Intelligence Into Avionics Fault Management

- The increasing complexity of flight software and hardware creates a challenge for the avionics and software designers
- Trying to anticipate all the ways in which the system fails and responds becomes increasingly difficult
- Creating deterministic software to handle all scenarios is ultimately a dead end
- Some elements of AI incorporated into the FSW could offer a breakthrough
- Initial simulations will focus on the use of GPUs for creating a specialized neural network for fault management
- The ultimate goal is a FPGA-based neural network for avionics fault management
- The FPGA implementation combines the processing parallelism of GPUs and the relatively low power consumption of FPGAs

FPGA-based artificial neural networks could create a new paradigm in fault management of avionics systems

Summary

- Shifting the balance between COTS vs. custom hardware
- Modern development tools make the FPGA technology user-friendly
- FPGA ASIC Emulation of avionics systems offers a system-level platform for testing and debugging
- Incorporating system monitoring and diagnostics into the FPGA firmware
- Parallelism and low power consumption make the FPGA technology a promising candidate for complex avionics systems

FPGA technology offers system-level testing and integration with COTS tools

AEROSPACE
Assuring Space Mission Success

Thank You