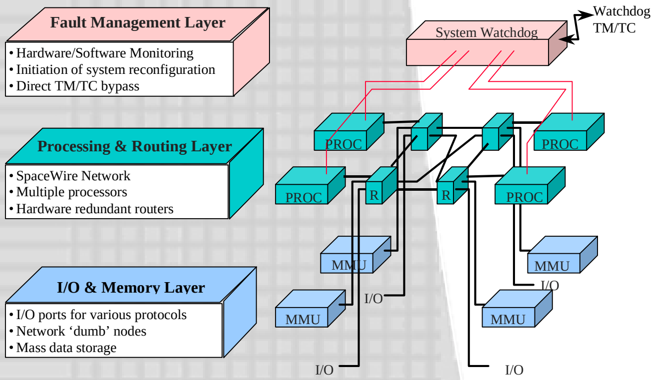
Application of SpaceWire to the Unionics Advanced On-Board Distributed Processing Architecture

Contents



Fault Management Layer
• Hardware/Software Monitoring
• Initiation of system reconfiguration
• Direct TM/TC bypass
Processing & Routing Layer
• SpaceWire Network
• Multiple processors
• Hardware redundant routers
I/O & Memory Layer
• I/O ports for various protocols
• Network 'dumb' nodes
• Mass data storage
System Watchdog
Watchdog
TM/TC
PROC
R
MMU
I/O

The Unionics Project - Objectives

BNSC initiative to develop a high-performance, high-reliability, cost-constrained spacecraft avionics architecture

- Exploitation of existing UK capability and heritage,3 led by EADS Astrium Ltd (system) with SEA Ltd (hardware) and SciSys (software)
- Development of a robust, scaleable processing architecture which is adaptable to a wide range of missions – and has now been baselined!
- Exploitation of space-qualified high-powered processors and SpaceWire standard
- Reusable hardware and software modules allowing scalability.
- Independence from a specific choice of processor.
- Optimised approach to redundancy and fault tolerance
- Open architecture with SpaceWire links that can host third party software and hardware
- Demonstration of an end to end system, one step from a flight system

Unionics – Main Features

- Increased on-board processing capability (e.g. for instruments with high data rates; autonomous behaviour; formation flying)
- Combination of high throughput payload processing and real-time avionics functions within same system
- Robust fault management approach, with high tolerance to failures of individual elements, despite increasing functionality and complexity
  - Simple, robust watchdog to monitor processor health
  - Transparent relocation of s/w applications following processor failure
- Full resource sharing for resiliance.
  - Elegant degradation under failure conditions of total processing resource
  - All peripherals accessible to all processor nodes
- High speed network using Spacewire Router and links
  - SpaceWire compatibility tested with 4 Links router
  - Redundant links provide fault tolerance, relocation of processes and sharing of processing resource

Layered FDIR and Watchdog

- **Unionics FDIR** capability designed for integration into modern, layered spacecraft FDIR:
  - Lowest level - hot swapping individual units (fail operational).
  - Second level - internal Unionics reconfigurations, (fail operational).
  - Third level - spacecraft level reconfigurations with transition to safe mode (fail safe).
- **System monitor** runs on each processor to check health
  - Runs local reconfiguration manager if a process fails
  - Provides periodic messages from processor node to hardware watchdog
- **System watchdog** monitors health of each processor
  - Generates reconfiguration messages (asynchronous, over LVDS) if message indicates error or if message not received.
- **Reconfiguration manager** is started on one healthy node, which re-starts failed applications on remaining healthy nodes

Unionics Architecture

Fault Management Layer

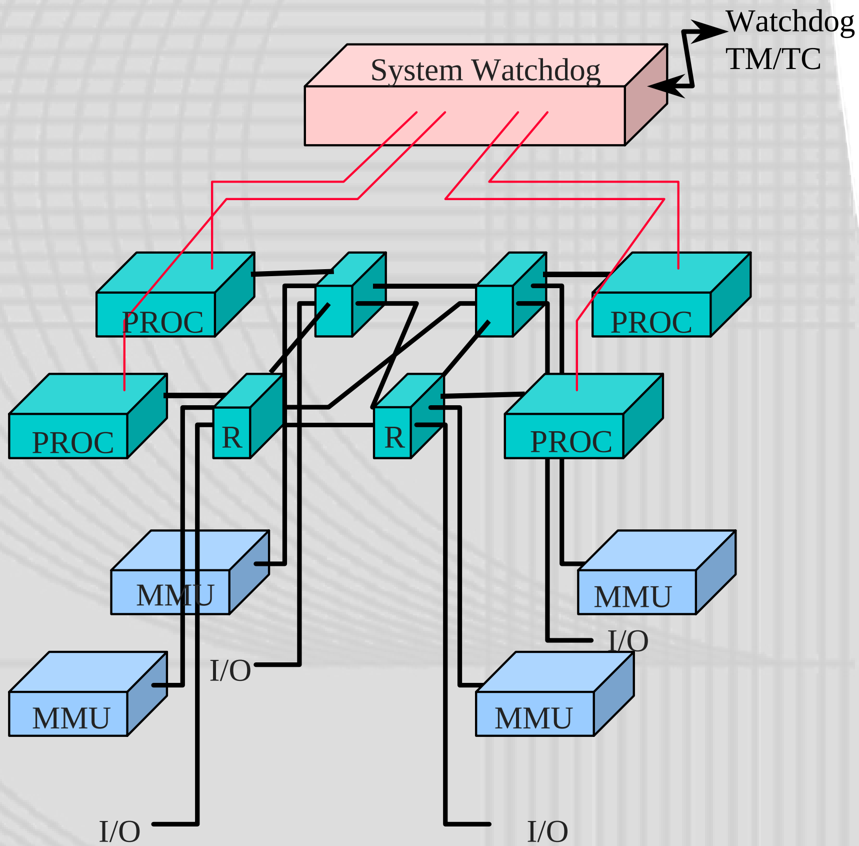
- Hardware/Software Monitoring
- Initiation of system reconfiguration
- Direct TM/TC bypass

Processing & Routing Layer

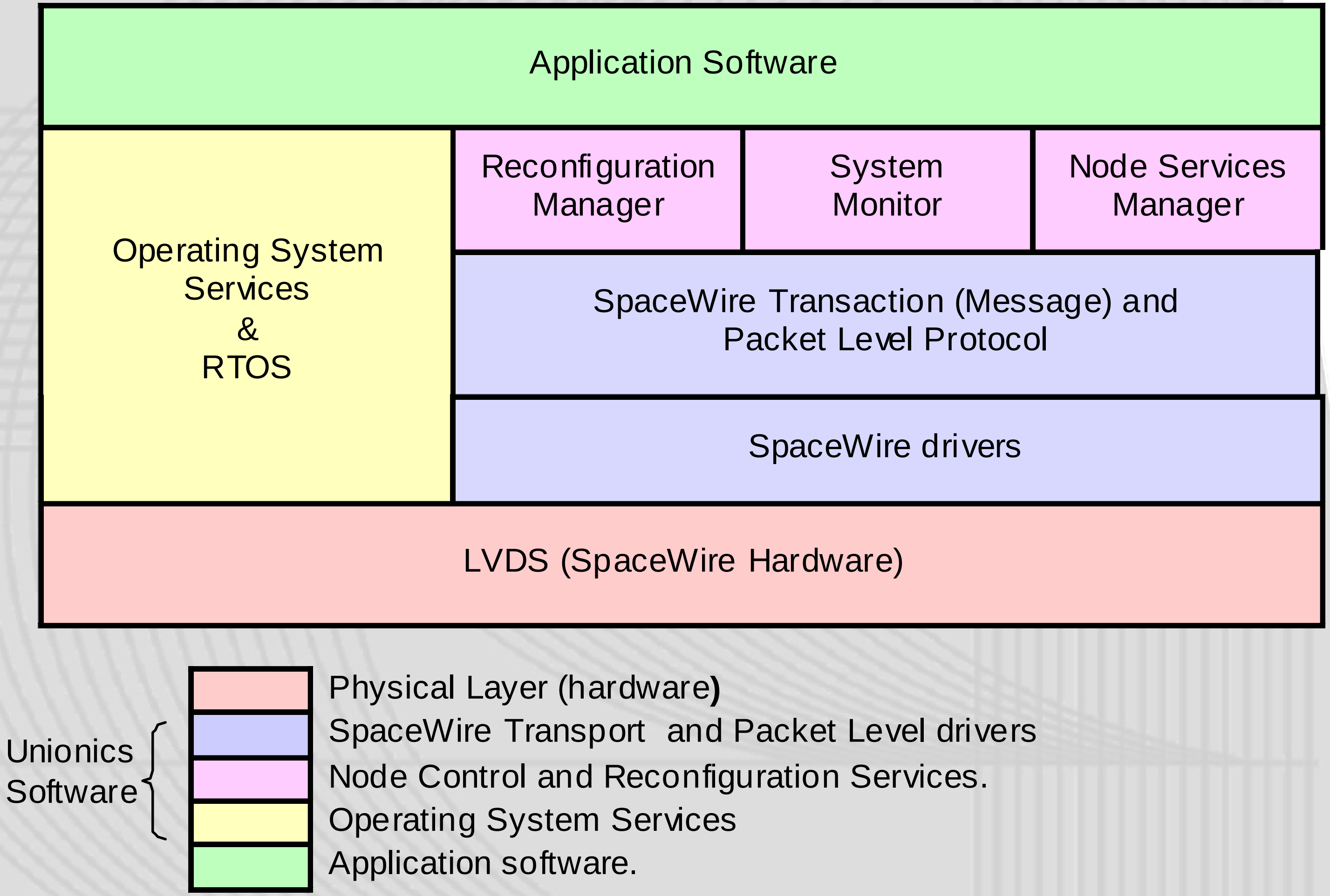
- SpaceWire Network
- Multiple processors
- Hardware redundant routers

I/O & Memory Layer

- I/O ports for various protocols
- Network ‘dumb’ nodes
- Mass data storage



Unionics Software Architecture



Unionics Software

The Unionics software forms part of the Unionics architecture:

- Real Time Operating System and Services
  - Runs standard ANSII C applications supporting third party developments, legacy code re-use and auto-coding
  - prioritised pre-emptive multitasking
- Spacewire Drivers
- Failure Detection Software Components
  - Report processor and applications status to Watchdog
- Failure Correction
  - Locally restart failed applications
  - Reconfigure Unionics domain according to pre-defined configuration tables in response to Watchdog
- Time Synchronisation
- Processor Mode Services
  - Operating System level communications between processors

Spacewire Drivers

- Bi-directional communications using Virtual Channels
- Spacewire message consists of a block of data with an added message header and a message header containing the length of the data and its sequence number
- Router design ensures that packets arrive in correct order. Missing packet causes the whole message to be dropped and the calling application to be informed
- Priority has been added to driver to allow use with mixed control and data driven systems (eg AOCS and payload processing)

Unionics Bench Top Demonstrator

Objectives of Unionics Demonstrator

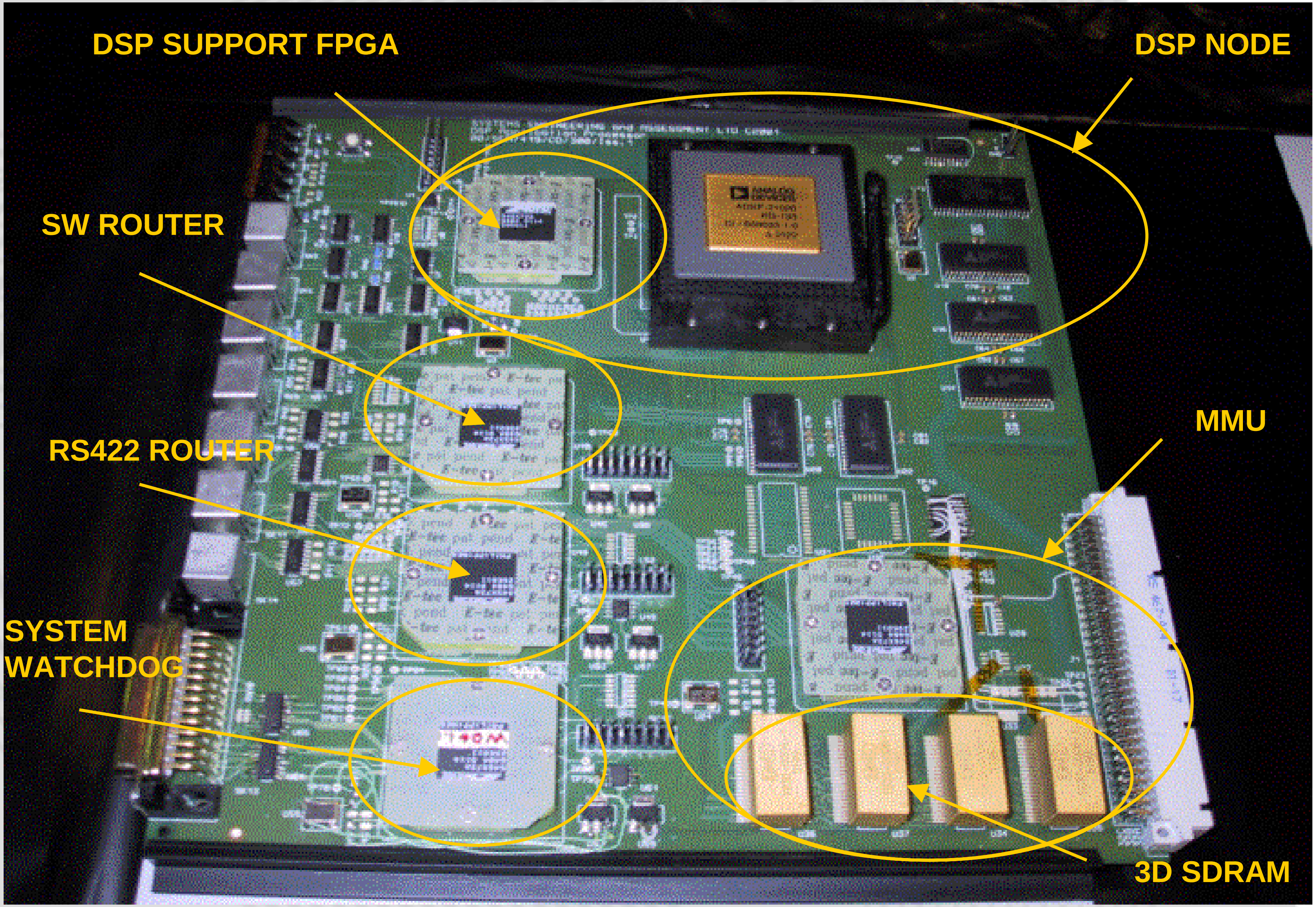
The Unionics demonstrator fulfils the following objectives:

- One step from flight prototype of Unionics hardware and software.
- Test principal Unionics capabilities.
- Test bed for different Unionics strategies, e.g. alternate failure detection & correction implementations.
- Demonstrate Unionics applicability to a range of different missions in a closed loop environment.
- Test bed for prototyping new tools and methods for low cost software application development, e.g. auto-coding
- To publicise Unionics capabilities.
- Early testing of software applications during the development of a specific mission based on Unionics.

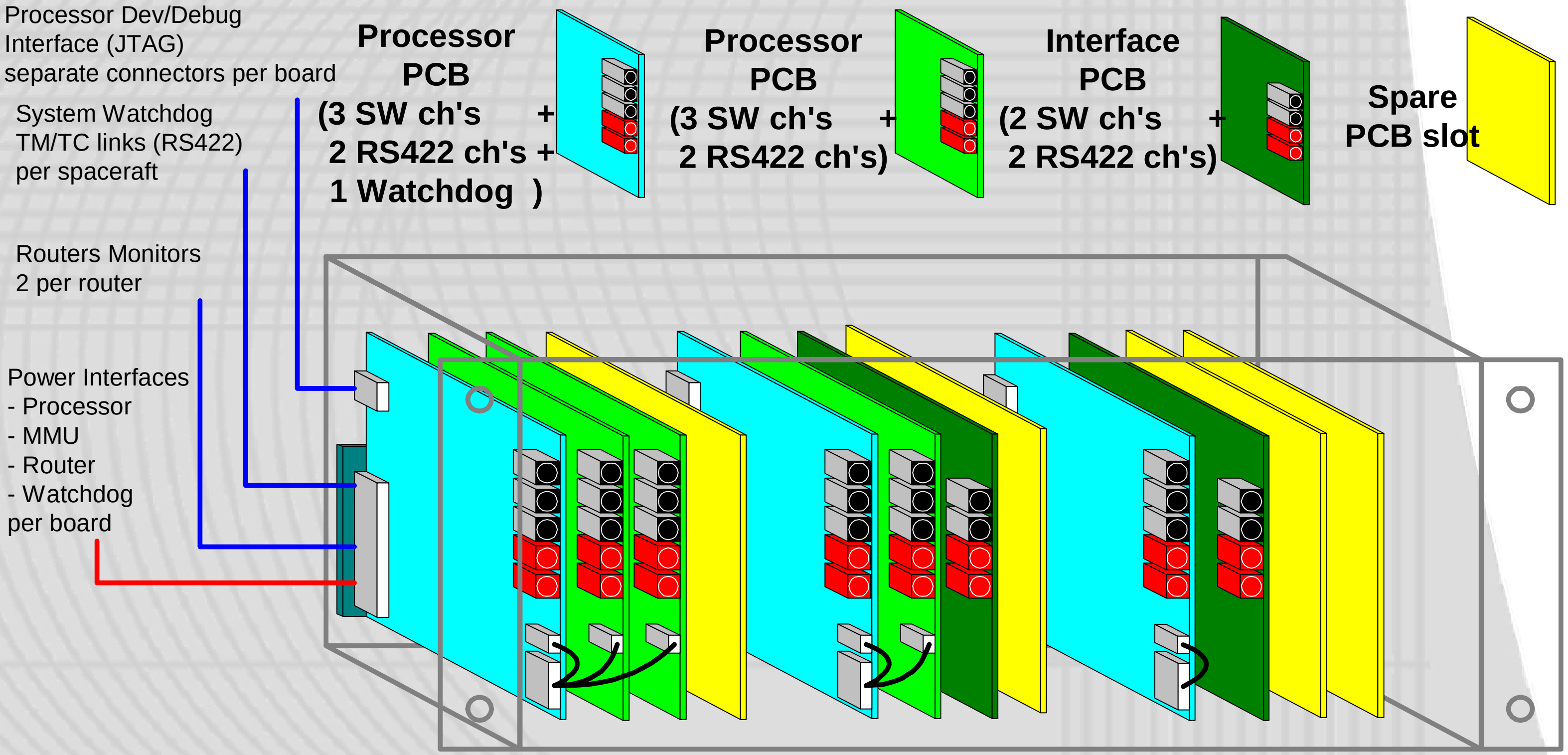
Unionics Demonstrator Scenario

- Three LEO spacecraft, which can operate independently or can utilise inter-satellite links for master/slave communications
  - Validates Unionics applicability to low cost LEO spacecraft, as well as some aspects of trans-spacecraft processing network
- Incorporates AOCS, data handling and payload processing applications.
  - Auto-coding used in AOCS development
  - Existing code reused for payload processing
- Unionics failure detection and correction capability integrated into spacecraft failure detection and correction scheme
- Test environment provides command and monitoring, closed loop dynamics simulator, simulated external units and graphical displays.

Unionics Demonstrator Card Layout



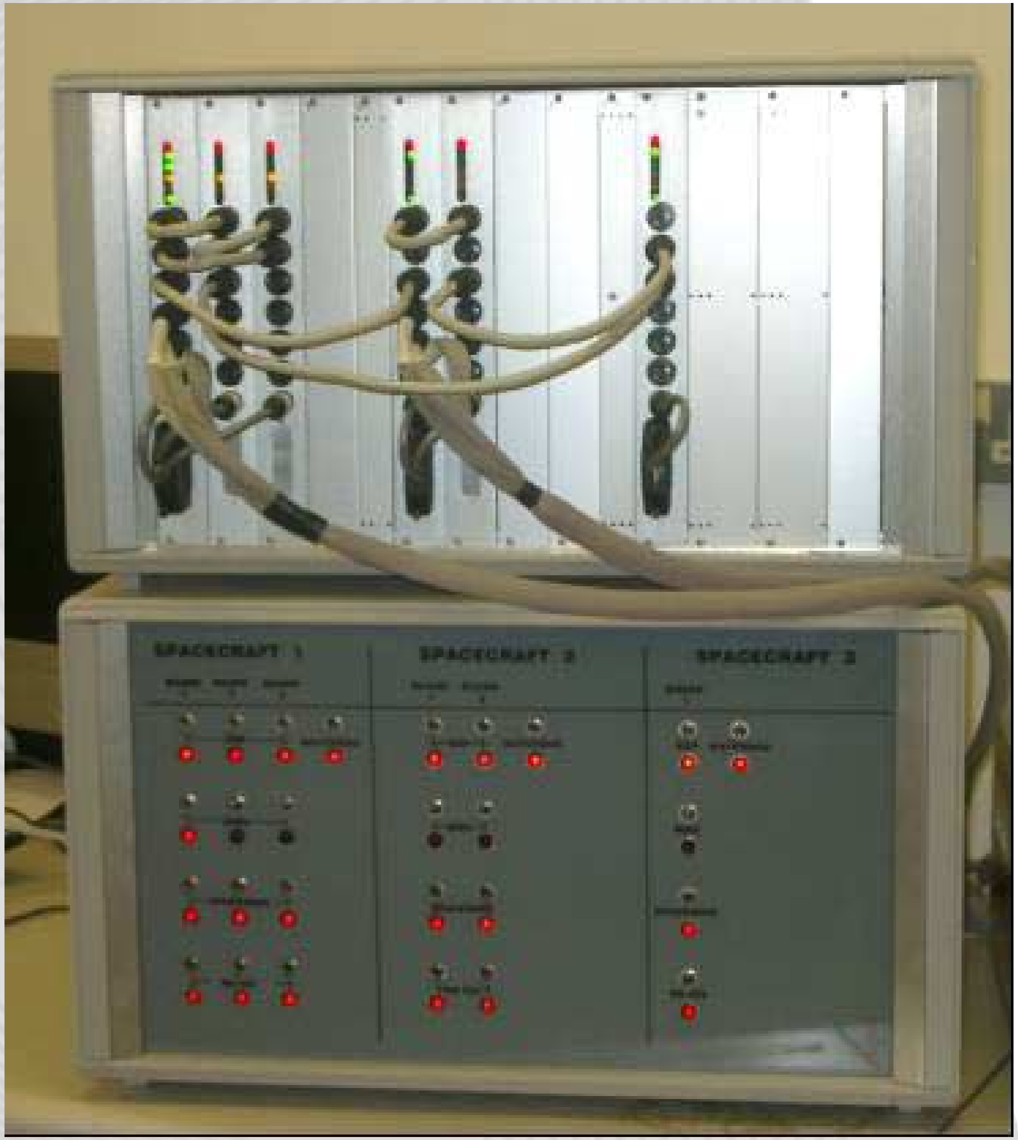
Rack Configuration



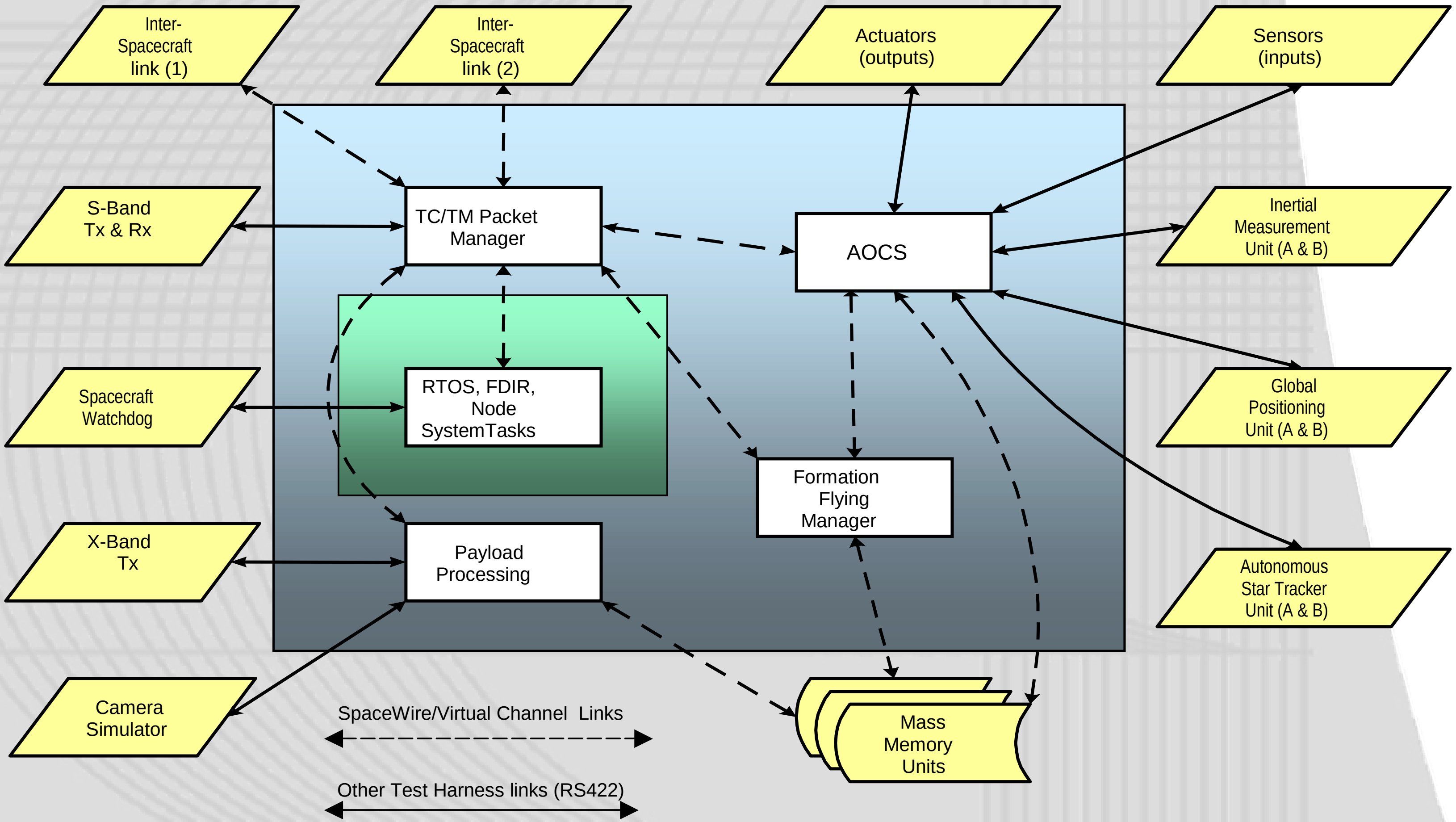
Unionics and Power Supply

- The Unionics demonstrator rack represents 3 spacecraft
  - S/C #1 – 3 nodes
  - S/C #2 – 2 nodes
  - S/C #3 – 1 nodes

- The power supply unit allows interruption of power to any unit to simulate failures



Unionics Demonstrator Software



Unionics Demonstrator Environment

Real Time Simulator:

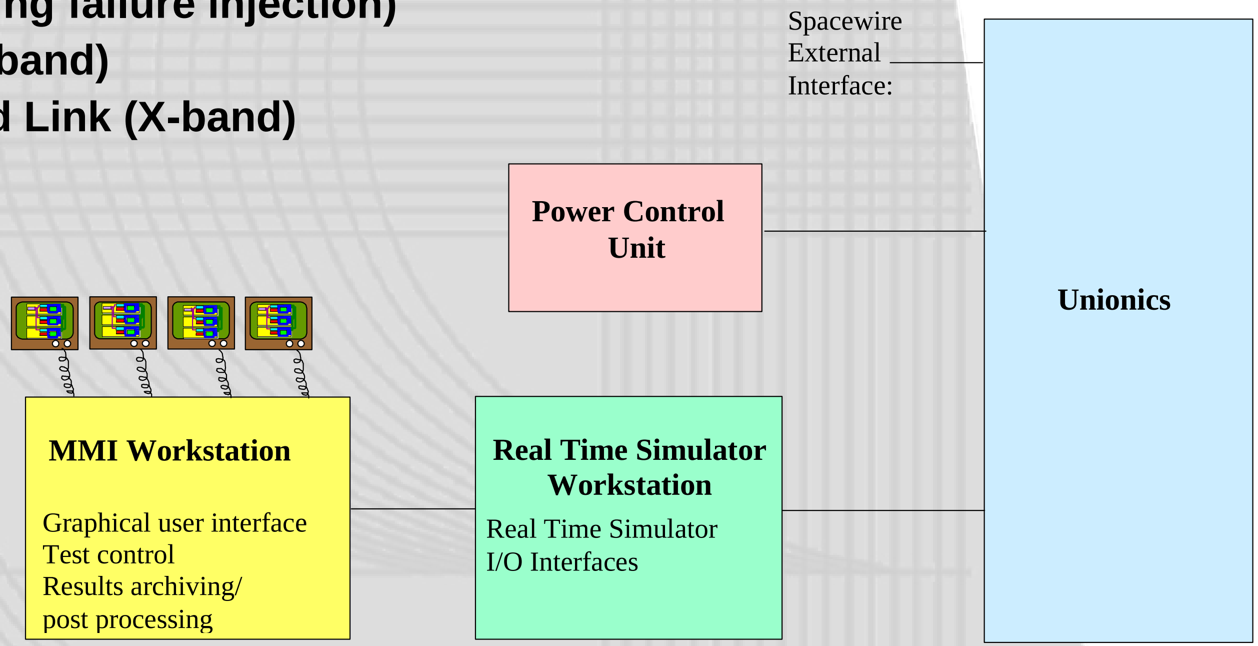
- Unionics I/O
- Spacecraft Dynamics
- On-board Units (including failure injection)
- Ground TM/TC Link (S-band)
- Payload Data Download Link (X-band)
- Payload Observations

Manual control of individual Unionics power supplies

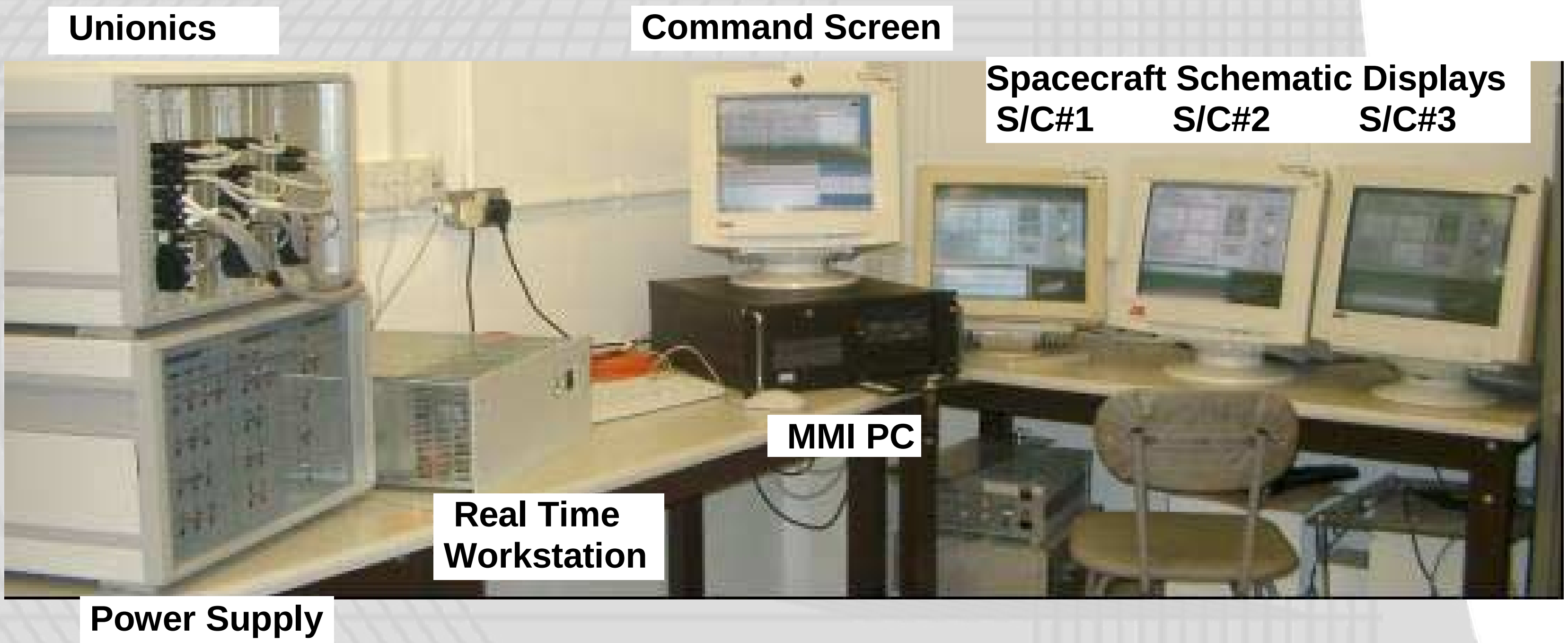
Graphical User Interface

Autonomous test scripts and manual control

Results archiving and post processing



Demonstrator Hardware

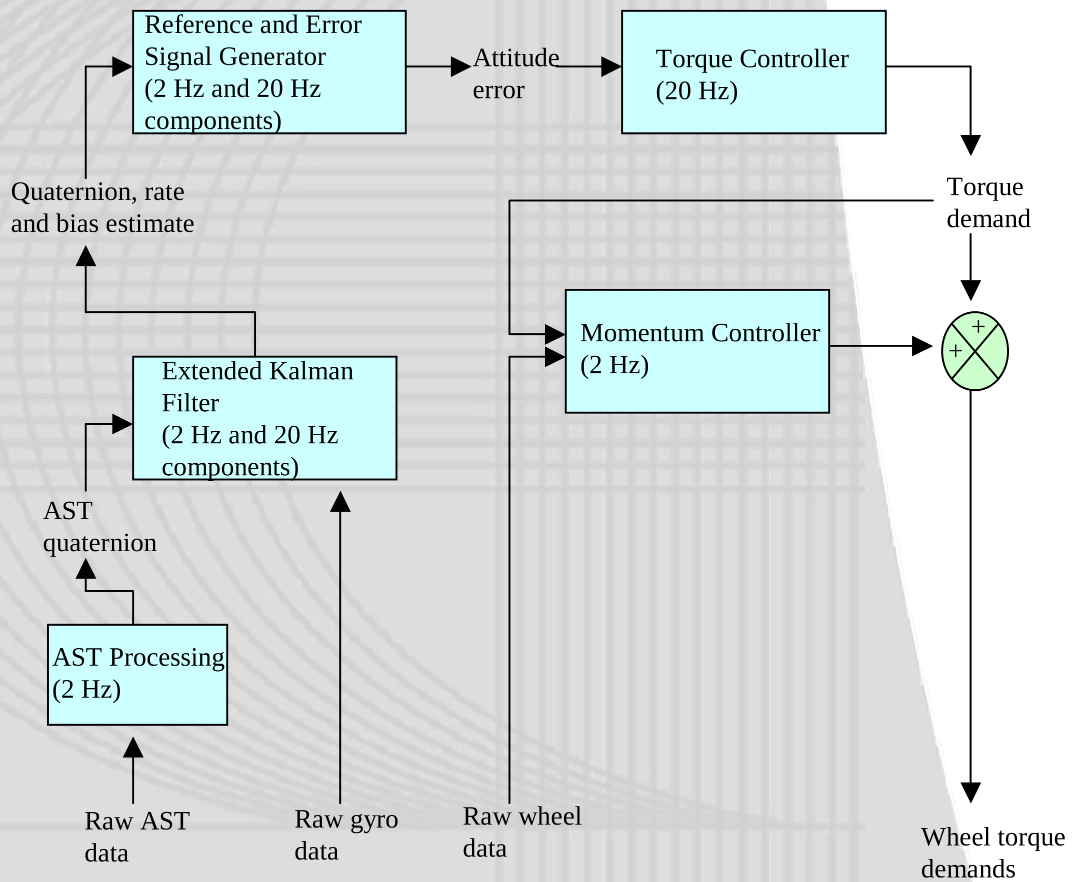


Closed-loop real-time AOCS

- AOCS and dynamics simulator developed and tested in Simulink.
- AOCS autocoded using Real Time Workshop (RTW), compiled in Microsoft Visual C++, and validated against Simulink model.
- Autocode compiled/linked with handcoded AOCS functions (I/O to sensors/actuators etc) using visual DSP. Loaded onto Unionics
- Simulink model dynamics simulator autocoded, and integrated onto test set real time workstation (Dspace).
- AOCS on Unionics validated closed loop with dynamics simulator on test set.

AOCS Architecture

- Represents normal mode utilising Autonomous Star Tracker (AST), Gyro and wheels.
- State estimation utilising extended Kalman filter
- 10 Hz control law frequency, with 2 Hz updates for AST and wheels



Introduction to the Unionics Demonstration

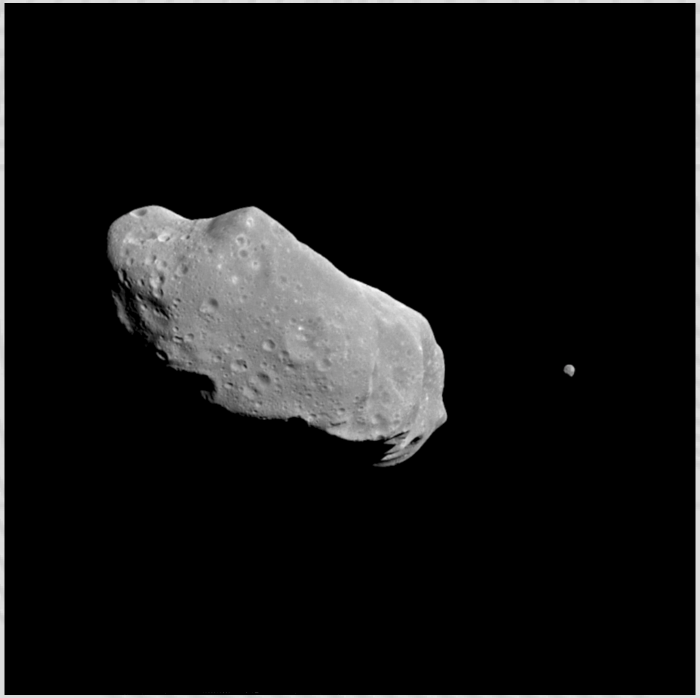
Unionics Demonstration (1)

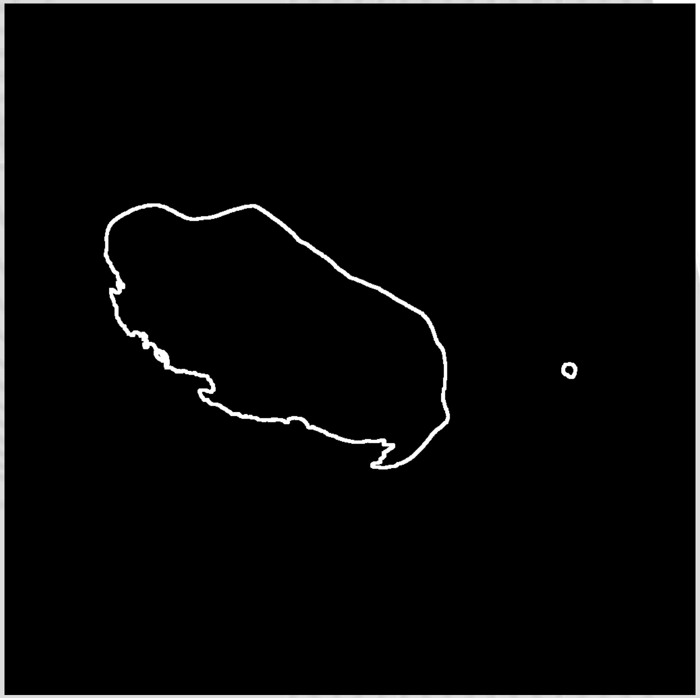
Three key features of the Unionics demonstrator will be presented:

- Simultaneous operations of data handling, simplified AOCS and payload processing, on a spacewire network based distributed processing architecture.
- Trans-spacecraft network employing spacewire inter-satellite links, and a change of the formation master spacecraft
- Unionics inherent FDIR automatically detecting processor failures and reconfiguring processes to processors mapping.

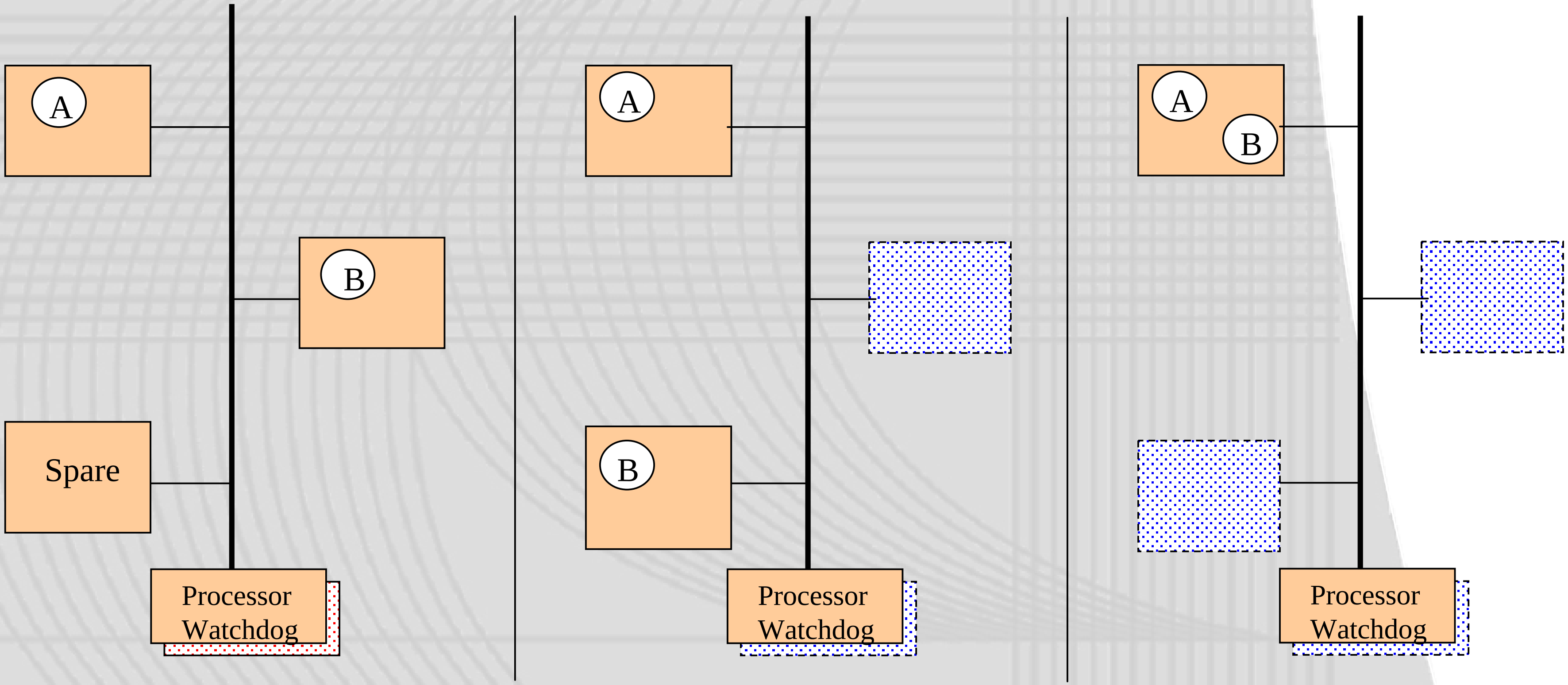
Unionics Demonstration (2)

- Simultaneous payload processing
  - Reads in an asteroid image and computes its outline, and returns the processed image.





Graceful Degradation



Nominal Configuration

First Failure

Multiple Failures