The Space-Cube Architecture

A Road Map For The Future

Gary Bolton
Avionic Equipment Section 344
Flight Computer Development Group
SPACE-CUBE ARCHITECTURE

Available Flavors and others ...

SINGLE PROCESSOR

MULTIPROCESSOR

MIN. INSTRUMENT PROCESSOR

Example

FEATURES
- Processor Independent
- Upgradeable
- Maintainable
- Interchangeable Modules
- Can Be Easily Configured To Different Architectures
  - Single String
  - Dual String
  - TMR
  - Multiprocessors
  - Distributed
  - NonDistributed
  - Others
  - CPU Module Inheritance from AFC
  - Modules Within Manufacture Capability Of Several Vendors
    - LORAL
    - HUGHES
    - nChip
    - Others

- Design Consists of Several Standard Features
  - Module Size
  - Interconnect Bus
  - Interconnect Method

Spacecraft Example (Distributed Architecture)

High Speed Serial Fault Tolerant Network

1553, FDDI, etc
The Space-Cube Architecture

- Introduction of the Space-Cube Architecture
- Implementation of the Space-Cube Architecture using today's technology

Agenda
The Space-Cube Architecture

Why Modularize?

• **Can** Take Advantage Of Commercially Available Products
• Resulting System Is Easier To Maintain and Repair
• Encourages:
  – Standard Designs
  – Hardware/Software inheritance
• Modules That Are Used Frequently **Enough To Justify The Cost Can Be Miniaturized (i.e. MCM ASIC).**
• Smaller Modules Are Easier To Manufacture
The Space-Cube Architecture

Why Stack Modules?

- Resulting Volume Is More Compact
- Interconnect Wire Length Is Minimized
- More Structurally Sound

- The Space-Cube Architecture Is Possible
Road Map For The Future

- Single PWBS
- Stacked PWBS
- Hybrid of Stacked MCMs & PWBS
- Stacked Dies
- Stacked MCMs
FEATURES

- Regular Polygon (i.e. Triangle, Square, Hexagon, Octagon, etc.)
- Stackable
FEATURES

- Provides The Interconnection Between Modules
- Single Bus per Side
- Flexible Interconnection
- Busses May Be Selectively Conducted Or Broken
- Each Flavor allowed in all 4 possible orientations
INTER-MODULE PACKAGING

FEATURES

- Single Bus per Side
- Removable Modules
- Flexible Interconnection
REQUIREMENTS

- Must Fit On a Single Side
- Power Pins Must Be Included

OPEN ISSUES

- Fault Tolerant
  - EDAC on Address
  - EDAC on Data
  - Redundant
- Bus Width
  - Required Bandwidth
  - Standard/Non-Standard
  - Multiplexed/Non-Multiplexed
  - Support Multiple Masters
  - Number Of Pins
FEATURES

- Allows for up to 2 Concurrent Connections
- Arbitration done on a cycle by cycle basis
- Latency equal to two buffer delays if not blocked
**Super Gateway**

**Features**

- Allows for up to 4 concurrent connections
- Arbitration done on a cycle by cycle basis
- Latency equal to two buffer delays if not blocked
FEATURES

- Single String Architecture
- *Nothing New* Here
Single String Example

FEATURES

● Single String Architecture

● *Nothing New Here*
NOTE: 1) FDU controls gateway.
2) Gateway and FDU can be merged into one module.
3) Redundant Fdus and gateways could be used.

**FEATURES**

- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
FEATURES

- Dual String Architecture: Each With Own Bus
- All With No Crossing Wires!!!
Dual String Example

FEATURES

- Dual **String** Architecture; Each With Own Bus
- All With No Crossing **Wires!!!**
Dual String Example

FEATURES
- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
Dual String Example

FEATURES

- Dual String Architecture; Each With Own Bus
- All With No Crossing **Wires!!!**
Dual String Example

FEATURES

- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
FEATURES

- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
Dual String Example

- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
**FEATURES**

- Dual String Architecture; Each With Own Bus
- All With No Crossing *Wires!!!*
Dual String Example

FEATURES

- Dual String Architecture; Each With Own Bus
- All With No Crossing Wires!!!
Dual String Example

FEATURES

- Dual String Architecture; Each With Own Bus
- All With No Crossing **Wires**!!!
Features

- Triple String Architecture
- Each String Has Its Own Bus
  Simple & Easy
**Triple Module Redundant**

**FEATURES**

- Triple String Architecture
- Each String Has Its Own Bus
  - *Simple & Easy*
Triple Module Redundant

FEATURES

● Triple String Architecture
● Each String Has Its Own Bus
● Simple & Easy
Triple Module Redundant

FEATURES

Triple String Architecture
Each String Has Its Own Bus
Simple & Easy
FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
  - *Simple & Easy*
Triple Module Redundant

Features

- Triple String Architecture
- Each String Has Its Own Bus
  *Simple & Easy*
**Triple Module Redundant**

- **URES**
  - Triple String Architecture
  - Each String Has Its Own Bus
  - *Simple & Easy*
FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
  - Simple & Easy
Triple Module Redundant

FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
  - *Simple & Easy*
FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
- Simple & Easy
Triple Module Redundant

FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
  - Simple & Easy
**Triple Module Redundant**

**FEATURES**

- Triple String Architecture
- Each String Has Its Own Bus
- *Simple & Easy*
**Triple Module Redundant**

Features:

- Triple String Architecture
- Each String Has Its Own CPUs
  
*Simple & Easy*
Triple Module Redundant

FEATURES

- Triple String Architecture
- Each String Has Its Own Bus
  *Simple & Easy*
Possible Architectures

- Pipeline
- MIMD Cluster
- Multiple Clusters
- Binary Tree
- Fault Tolerant Architectures
  - 0
  - and many others

All using this approach and all with no intersecting wires!!!!!!
The Space-Cube Architecture

- Processor Independent, Upgradeable
- Simple Interchangeable Modules
- Can Be Easily Configured into Complex Architectures
- Can Take Advantage Of Commercially Available Products
- Reduces The Cabling Between Modules To Simple Board To Board Connections