An Open Architecture for an Embedded Signal Processing Subsystem

7th Annual Workshop on High Performance Embedded Computing

MIT Lincoln Laboratory
23-25 Sept 2003

Stephen F. Shank
Principal Member
Engineering Staff
Project Summary

- **The Objectives:**
  - Utilize High Performance Embedded Computing To Replace Legacy Signal Processor Equipment In Future Radar Programs
  - Assemble A Project Team To Define, Develop And Code The Key Functions Of The Open Architecture Digital Processor
  - Demonstrate A Prototype In 15 Months

- **The Players:**
  - Lockheed Martin – Radar Design Agent And System Integrator
  - INDRA – Spanish Radar Company And Software Developer
  - CSPI - COTS Hardware Supplier And Investment Partner
  - VMETRO - COTS Data Recorder
  - Primagraphics - COTS Display

- **Lockheed Martin Tasks:**
  - Develop The Hardware / Software Architecture
  - Define Target Radar Characteristics And Provide Specifications, Matlab Models,
  - Conduct Integration And Test Activities

- **INDRA Tasks:**
  - Design, Develop, Code, And Test Key Functions Of The COTS DSP
  - Support Integration & Test

- **CSPI Tasks:**
  - Provide Training To INDRA
  - Provide Hardware And Software Development Environment
  - Develop Radar Interface Boards
  - Provide Development Support

- **VMETRO:**
  - Provide Recorder Equipment

International Development Team Assembled
Project Plan: 
Reconfigurable Generic Search Radar Digital Signal Processor (RGSD)

• Define radar characteristics, specifications, Matlab Models and system interfaces
• Develop a flexible hardware / software architecture
  — Software is reusable and scalable
  — Hardware is scalable and refreshable
• Conduct Integration and Test activities in radar test bed

Demonstrate RGSD in a Legacy Radar in 15 months
Open Architecture Digital Processor

**Digital Processor (DP) Subsystem**

**ASP/ DSP / RCP Application Software**

API, Common Services, OA Middleware (MPI & VSIPL)

- Analog Signal Processor
- High Speed Data Recorder
- FPGA – Based Processing
- Scalable Waveform Processing
- OA Embedded Network
- Scalable Data Processing
- OA Interface (PMC)
- OA Interface (PMC)
- OA Interface (PMC)
- OA Interface (PMC)
- OA Interface (PMC)

- **Software - Object-Oriented, C/C++**
- **Requirements Management – Telelogic DOORS**
- **OO Modeling – Rational Suite (Rose)**
- **Configuration Management – Rational ClearCase**
- **Integration & Test – VxWorks Tornado 2**

- **Standard API, OA Middleware**
  - Open Message Passing Software
    - MPI & TCP/IP
  - Standard Signal Processing Libraries
    - VSIPL
  - Support for Open Architecture Standards
    - VME 64, Fibre Extreme, PCI/PMC capable, Myrinet

 Independent, Scalable, Reusable Software
RGSD Development Methodology

- Determine Processing Requirements for Waveform Suite
- Partition Processing Requirements into 5 Functional Groups
  - Radar Interface Component
  - Display Interface Component
  - Coherent Waveform Processing
  - Non-Coho Waveform Processing
  - Detection
- Map Algorithm Functionality to Processor Configuration
- Identify Potential Risk Areas
  - Processing Intensive (e.g. Match Filtering)
  - I/O Intensive
- Design Software using
  - High Level Language (C/C++)
  - Common Application Programmer’s Interfaces (API) such as MPI/VSIPL for scalability and portability
- Validate Software against MatLab Hardware Model
Non Coherent Processing Architecture
-Two Options:

Pipeline

Round Robin

4 Interfaces
31 G4 PPCs

2 Interfaces
7 G4 PPCs
Coherent Processing Architecture

-Two Options:

1. **Pipeline**
   - Radar Interface
   - FP Conv
   - PC
   - Pipeline
   - Clutter Vel Corr
   - Doppler Filtering & Mag
   - CFAR
   - MIC Blanker Display Interface
   - From Radar
   - 5 Interfaces
   - 42 G4 PPCs

2. **Round Robin**
   - Radar Interface
   - FP Conv Limit / PC / Mag / GOF / CFAR
   - Display Interface
   - From Radar
   - 2 Interfaces
   - 10 G4 PPCs
Top Level RGSD Use Case Diagram

Visual Modeling maximizes the team’s development productivity
## Architecture Comparison

### Latency (µs)

<table>
<thead>
<tr>
<th>Waveform</th>
<th>Estimate Pipeline</th>
<th>Estimate Round Robin</th>
<th>Actual Round Robin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Coho 1</td>
<td>7,140</td>
<td>5,540</td>
<td>2,270</td>
</tr>
<tr>
<td>Non-Coho 2</td>
<td>3,570</td>
<td>3,710</td>
<td>1,970</td>
</tr>
<tr>
<td>Non-Coho 3</td>
<td>3,570</td>
<td>1,920</td>
<td>900</td>
</tr>
<tr>
<td>Coho 1</td>
<td>14,480</td>
<td>19,760</td>
<td>15,620</td>
</tr>
<tr>
<td>Coho 2</td>
<td>15,360</td>
<td>22,130</td>
<td>18,210</td>
</tr>
</tbody>
</table>

### Number of PPCs (G4)

<table>
<thead>
<tr>
<th>Waveform</th>
<th>Estimate Pipeline</th>
<th>Estimate Round Robin</th>
<th>Actual Round Robin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Coho 1</td>
<td>31</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Non-Coho 2</td>
<td>25</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Non-Coho 3</td>
<td>11</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Coho 1</td>
<td>42</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Coho 2</td>
<td>35</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

### Processing (%)

<table>
<thead>
<tr>
<th>Waveform</th>
<th>Estimate Pipeline</th>
<th>Estimate Round Robin</th>
<th>Actual Round Robin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Coho 1</td>
<td>49</td>
<td>96</td>
<td>89</td>
</tr>
<tr>
<td>Non-Coho 2</td>
<td>62</td>
<td>94</td>
<td>87</td>
</tr>
<tr>
<td>Non-Coho 3</td>
<td>50</td>
<td>88</td>
<td>71</td>
</tr>
<tr>
<td>Coho 1</td>
<td>58</td>
<td>91</td>
<td>89</td>
</tr>
<tr>
<td>Coho 2</td>
<td>55</td>
<td>90</td>
<td>87</td>
</tr>
</tbody>
</table>

### I/O (%)

<table>
<thead>
<tr>
<th>Waveform</th>
<th>Estimate Pipeline</th>
<th>Estimate Round Robin</th>
<th>Actual Round Robin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Coho 1</td>
<td>51</td>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td>Non-Coho 2</td>
<td>38</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>Non-Coho 3</td>
<td>50</td>
<td>12</td>
<td>29</td>
</tr>
<tr>
<td>Coho 1</td>
<td>42</td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td>Coho 2</td>
<td>45</td>
<td>10</td>
<td>13</td>
</tr>
</tbody>
</table>

---

**Cost Drivers**

Round Robin Meets Requirements with Fewer Processors
RGSD Development System Configuration

Open Architecture with Scalable Performance
Dual Radar and Display Interface

- Provides in a PMC Form Factor
  - RS-422 Interface to Radar Processor and Display console
  - User programmable CPLD
  - High performance (64/66) PCI controller providing a high bandwidth/low latency connection between the CPLD and the PMC connectors

Radar Interface Personality
- Buffers and packetizes I / Q data
- DMA’s packets to host memory for access by MPI
- Supports Test Data Injection
- Round-Robin queuing of radar data to destination software component based on waveform

Display Interface Personality
- DMAs data from host memory
- Sorts packets
- Buffers packet in preparation for display
- Restores time characteristics for proper display
- Generates output signals (data and synchronization) to display console

Hi-Performance Programmable Interface
Project Summary

- **RGSD Prototype was successfully integrated at Lockheed Martin**
  - System Integration and Test completed in less than three weeks
  - Successful use of Matlab model of legacy hardware substantially reduced I&T effort

- **RGSD will be leveraged for future radar programs**
  - Addresses production cost and Diminishing Material Supply (DMS) issues of current systems by replacing legacy equipment with COTS
  - Software based OA design provides the ability to enhance or modify system operation without the need for major redesigns

- **Project validated benefits of High Performance Embedded Computing**
  - Reduces Cost for:
    - Development effort
    - Acquisition / Life Cycle Cost
  - Provides:
    - Scalable and Reusable Signal Processing Software applicable to a wide variety of radar applications

Cost Effective use of OA Standards for Real Time Radar Applications