EMBEDDED MULTIPROCESSOR TECHNOLOGY FOR VHSIC INSERTION

Presented at Technology for Space Station Evolution - A Workshop

January 16-19, 1990

Paul J. Hayes
NASA Langley Research Center
(804) 864-1491
VHSIC SPACEBORNE MULTIPROCESSING

ADVANTAGES
- High Speed
- Fault-Tolerant
- Ada Language
- Maturing Component Base
NASA MULTIPROCESSOR TECHNOLOGY

OBJECTIVE
Develop multiprocessor system technology providing user-selectable fault tolerance, increased throughput, and ease of application representation for concurrent operation.

APPROACH
Develop graph management mapping theory for proper performance, model multiprocessor performance, and demonstrate performance in selected hardware systems.
MULTIPROCESSING TECHNOLOGY

Concurrent Processing Theory

ATAMM

TMR
Multiple processor types
Multiple graphs

GMOS
- 1st Cut
- TMR/Simp
- Ada

VHSIC
Breadbrd (EDM)

VHSIC (ADM)

GVSC

RH-32

ADAS Multiprocessor Modeling
Graph Management Operating System (GMOS)

Features

- Distributed O/S and Nodes
- Real-Time Node Assignment
- Application Graph
- Node-Selectable Fault Tolerance
- Ada
- VHSIC 1750A

Processor Assignment for Node Execution
GMOS FUNCTIONAL FEATURES

- EXECUTES DIRECTED GRAPH - SINGLE GRAPH
  - MULTIPLE GRAPH

- GRAPH NODE CRITICALITY - TMR OR SIMPLEX

- GRAPH NODE SCHEDULING
  
  A) DATA DRIVEN - EVENT FLAG
  - SEMAPHORE (MULTIPLE EVENTS)
  - AND/OR LOGIC

  B) DEMAND DRIVEN - PERIODIC TIMER
  - ONE-SHOT TIMER

- BACKUP NODE ALLOCATION

- FAULTY PROCESSOR EXCLUSION, SELF TEST, REBOOT
ALGORITHM TO ARCHITECTURE MAPPING MODEL (ATAMM)

• A strategy for the real-time assignment of the nodes of a data-driven algorithm graph to parallel processors

• Based on Petri-Net marked graph theory

• Aimed at large-grain graph applications

• Provides:
  - deadlock-free performance
  - optimum time performance
  - operating system rules
  - performance prediction
ALGORITHM MARKED GRAPH (AMG)
FOR 7-NODE EXAMPLE GRAPH

LEGEND:
○ = TOKEN

Name

Time
NODE MARKED GRAPH (NMG) FOR DATA HANDLING

READ

PROCESS

WRITE

Tr → T3 → Tw

N3
GRAPH FOR ANALYSIS

\[ TBO = \text{TIME BETWEEN SUCCESSIVE OUTPUTS} \]

\[ TBO_M = \text{MINIMUM VALUE FOR TBO} \]
\[ = \max \left( \frac{\text{TIME}}{\text{NO. TOKENS}} \right)_i = \frac{6}{2} = 3 \]
PROCESSOR REQUIREMENT PLOT
(SEVEN PROCESSORS AVAILABLE)

TIME UNITS

NO PROCESSORS

- COMPLETION OF A NODE
■ COMPLETION OF NODE N5
PERFORMANCE MARGIN

Throughput (Arbitrary Units)

Max

No. of Processors

1 2 3 4 5 6 7 8
Normalized Throughput Versus Number of Processors

Maximum Parallelism (Performance of Purely Parallel Graphs)

Idle Time

7-Node Graph Margin (ATAMM)

Actual Performance

Number of Processors

Throughput

$\text{T}_0$ = Idle Time

10 9 8 7 6 5 4 3 2 1

1 2 3 4 5 6 7 8 9
ATAMM PROVIDES A NEW CAPABILITY SET

- Mathematically proven lock-free performance

- Operating system rules to manage the assignment of graph nodes to processors

- Prediction of graph's performance bounds
  - Maximum data rate
  - Maximum number of processors
  - Dependency of data rate on number of processors
# ATAMM DEVELOPMENT/DEMO PLANS

<table>
<thead>
<tr>
<th>STATUS</th>
<th>FEATURE</th>
<th>DEMO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial ATAMM</td>
<td>o Single graphs</td>
<td>ADM</td>
</tr>
<tr>
<td></td>
<td>o Simplex</td>
<td></td>
</tr>
<tr>
<td></td>
<td>o Identical processors (HW &amp; SW)</td>
<td></td>
</tr>
<tr>
<td>Current Update</td>
<td>o Triple Modular Redundant (TMR)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>o Graph optimization for specific no. processors</td>
<td></td>
</tr>
<tr>
<td>Future Features</td>
<td>o Multiple graphs</td>
<td>GVSC</td>
</tr>
<tr>
<td></td>
<td>o Multiple iterations of the same graph</td>
<td>and/or</td>
</tr>
<tr>
<td></td>
<td>o Multiple processor types</td>
<td>RH-32</td>
</tr>
<tr>
<td></td>
<td>o Variable node-latency times</td>
<td></td>
</tr>
</tbody>
</table>
MAJOR RESEARCH THRUST FOR FY90

- Implement ATAMM Rules into KOS
  - Simpler operating system than previous GMOS
  - Use Westinghouse Directed Graph Tool
  - Use 1553B to provide data I/O and monitor graph status
  - Improved version of Ada compiler
  - 2.5 MIPS VHSIC ADM 1750A processor

- Demo/Evaluate with Ada Algorithms
  - Test Algorithm (Scan-to-scan track correlation for SDIO)
  - Time-simulated graphs
  - Simplex/TMR
  - Fault injection and continued processing
MULTIPROCESSOR INTERFACE DIAGRAM

MICROVAX II
- COMPILe
- DOWNLOAD
- DEBUG

IEEE 488

CPU 1

1553B

CPU 2

CPU 3

CPU 4

MULTIPROCESSOR

PC/AT
- CONTROL
- DATA IN/OUT
- GRAPH STATUS
- INSERT FAULTS
- RESULTS
- DISPLAY

PI BUS
V1750A ADM
Chassis/Module
- 1.25µ CMOS VHSIC
- µ-Coded CPU Plus 256K RAM
- Surface-Mounted Devices; Dual Board Module
- 5.88 x 6.44 Inches (SEM-E)
MAJOR PROGRAM MILESTONES

1) Integrate/Demo ATAMM and SDIO Algorithm on Avionics Advanced Development Model (ADM)

2) Adapt/Demo ATAMM-Based OS on GVSC

3) Expand ATAMM Capability and Adapt/Demo on RH-32