Fault-Tolerant Architecture for High Performance Embedded System Applications

Author:
Gul N. Khan
Division of Computing Systems
School of Applied Science
Nanyang Technological University, Singapore

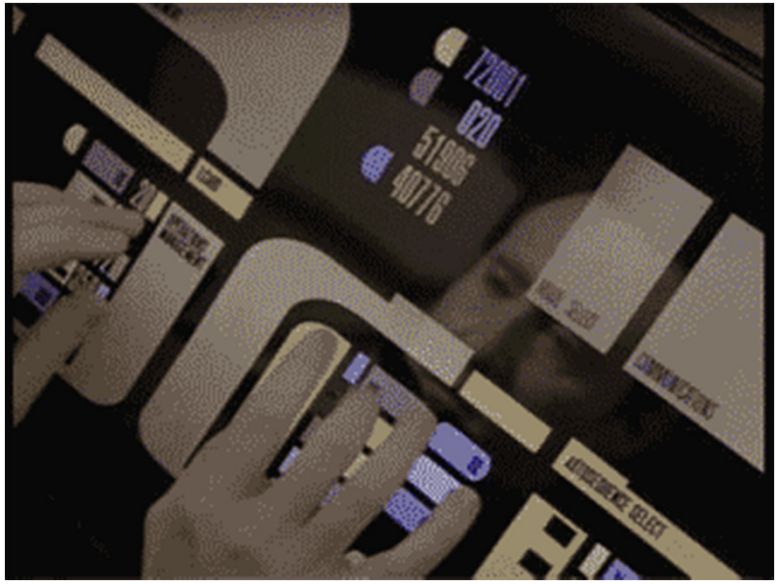
P*resented by*: Matthew D. McClellan
ECGR 6185 Adv. Embedded Systems
April 3rd, 2013

Agenda

- Purpose
- Abstract
- Architecture
- Fault Detection
- Fault Containment and Recovery
- System Reliability Analysis

Purpose

- Overkill for most systems
  - Cellular Phones
  - Printers
- Needed for safety-critical systems
  - Medical Systems
  - Avionics
  - Astronautics



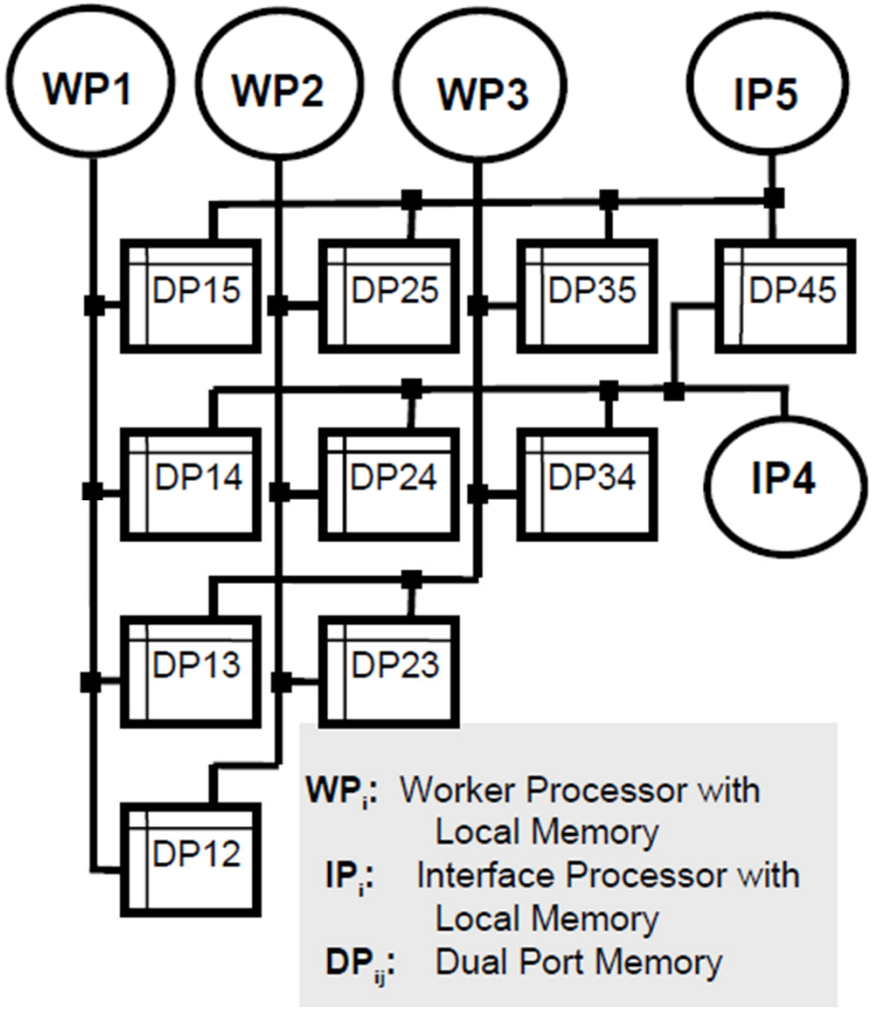
[1]

Abstract Requirements

- High Performance
  - Exploit Parallelism with Multithreading
  - High Throughput for Real-Time Critical Applications
- Reliability
  - System Failure at Most 10^{-10} per Hour (MTF: 1.1 Million Years)

HPEC Architecture

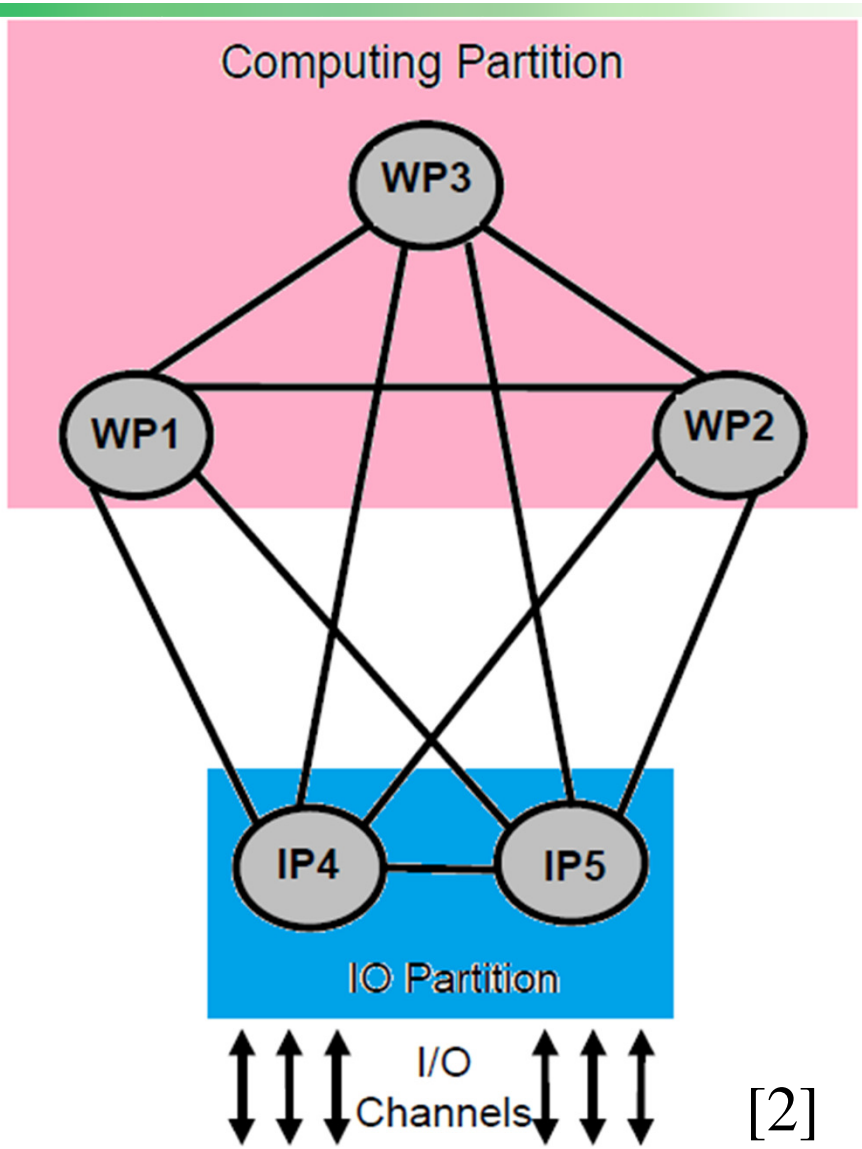
- High Performance Embedded Computer (HPEC)
  - Five Processors
    - Three Worker Processors (WP_1, WP_2 & WP_3)
    - Two Interface Processors (IP_4 & IP_5)
  - Ten Dual Port Memory
    - Connecting Each Processor Pair (DP_{12}, DP_{13}, …, DP_{45})



[2]: HPEC System Architecture

HPEC Architecture (cont.)

- Divisions of Tasks
  - Worker Processors for Intense Computations
  - Interface Processors for Real-Time I/O
- Additional Processors Possible

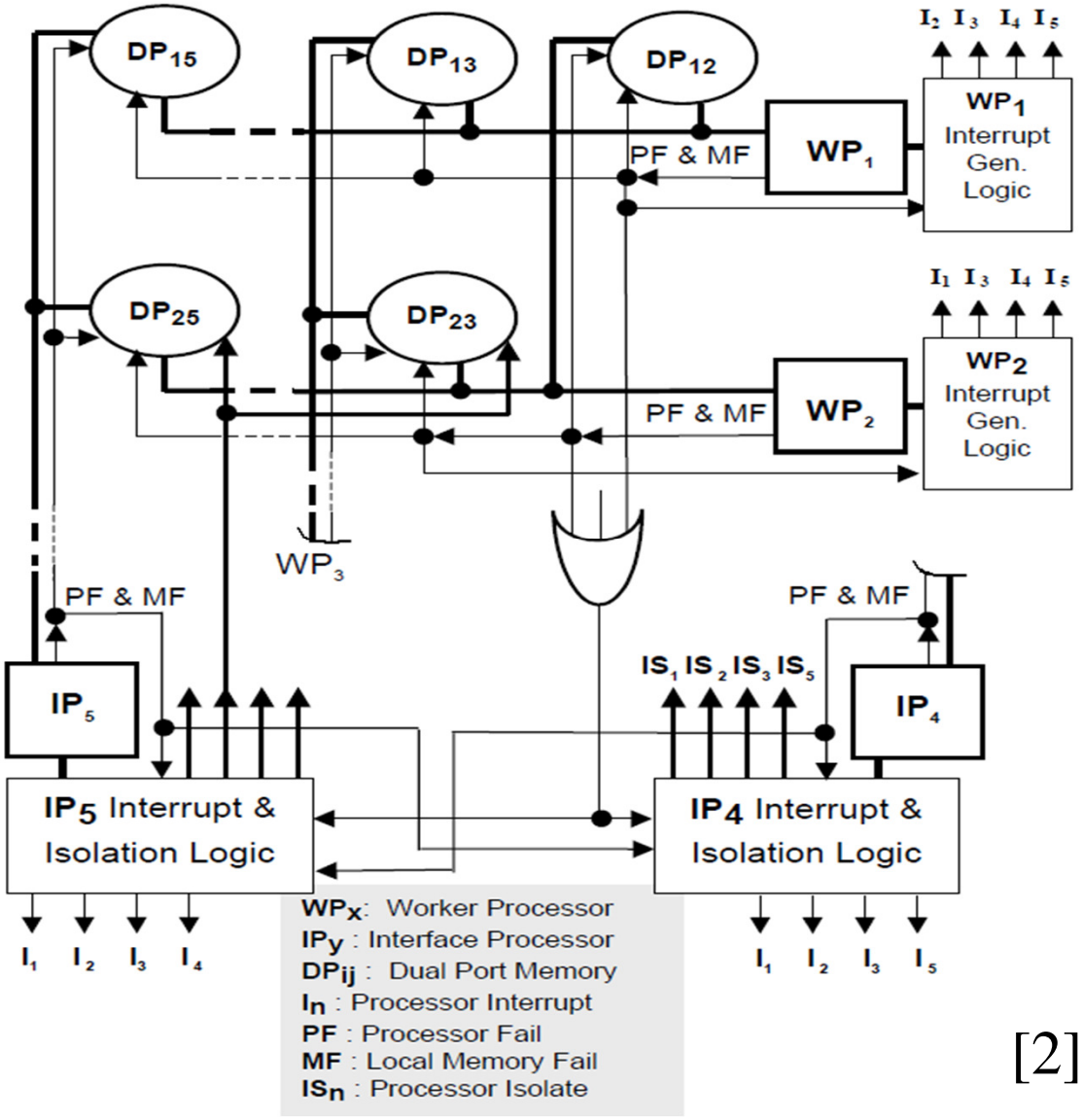


[2]

Fault Detection

- Watchdog Timer Detects Processor Failure
  - Generates PF Flag
- Local Memory uses Error Correction Circuits
  - If Error Cannot be Corrected, Generates MF Flag
- DP Memory Checked by CRC
- One Interface Processor Acts as Controller
  - Load Balancing in Addition to I/O
- Second IP Monitors the First

Fault Detection (cont.)



[2]

Fault Containment and Recovery

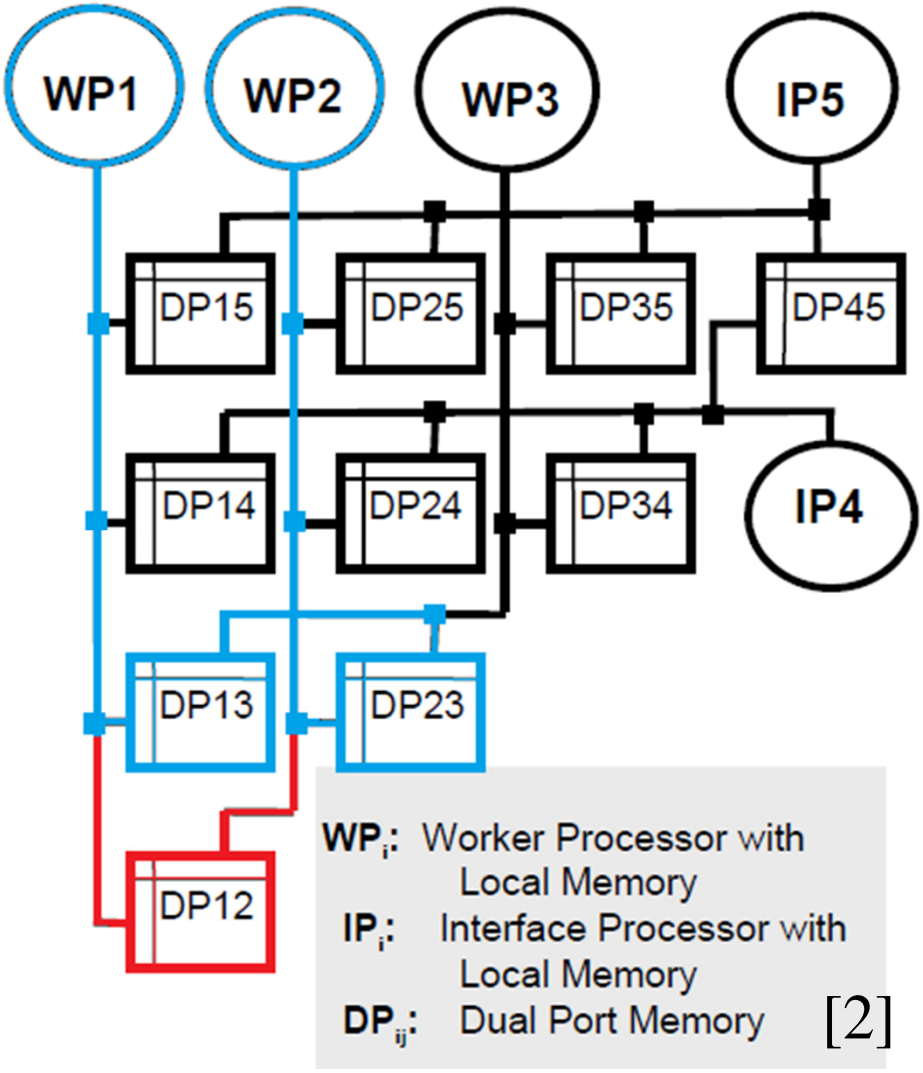
- Failing Node Interrupts the Controller
- If Fault is a Processor:
  - Disable Interrupt capability of the node
  - Generate Isolation Signal (IS_n)
    - Prevents the Node from Effecting DP Modules
  - Broadcast Failure to All Nodes
  - Attempt Diagnostic Subroutine to Recover the Processor
    - If Successful, Return the Node to Service
    - If Unsuccessful, Redistribute Tasks to Other Nodes
  - Redo All Suspect Tasks
    - Both IPs Keep Records of Work Done by All Nodes

Fault Containment and Recovery (cont.)

- If Fault is a Local Memory:
  - Disable Interrupt capability of the node
  - Generate Isolation Signal (IS_n)
    - Prevents the Node from Effecting DP Modules
  - Broadcast Failure to All Nodes
  - Run Diagnostic Program from DP Memory
    - If Recoverable, Return the Node to Full Service
    - If Unrecoverable, Node is Used in Degraded Mode
      - DP Memory Used Instead of Local Memory

Fault Containment and Recovery (cont.)

- If Fault is in DP Memory:
  - System Controller Isolates the DP Module
  - All Memory Traffic is Routed Through Other Modules
    - *Ex*: If DP_{12} fails, WP_1 and WP_2 can communicate through DP_{13} to DP_{32}



[2]

System Reliability Analysis

- Assume Each Processing Node Has Failure Rate λ_p
- Assume Each DP Memory Unit Has Failure Rate λ_m
- Assume All Failures are Independent
- Performance Measure is thus Defined:
  - $C_{Max} = f(P, M) = \frac{[P*(P-1)*(M+1)]}{k}, where\ k \cong 20$
- For *p* Faulty Processor Nodes and *m* Faulty DP Memory Nodes:
  - $C = f(P - p, M - m) = \frac{[(P-p)*(P-p-1)*(M-m-1)]}{k}$
- Finally, Normalize:
  - $C_{Nor} = f(P - P, M - m + 1)/C_{Max}$

System Reliability Analysis (cont.)

- Determine System Reliability:
  - $R(t) = \sum_{p=0}^{P} \sum_{m=0}^{M} r(p) * r(m) * C_{nor}$
- Approximate r(p) & r(m) with Poisson Distributions:
  - $r(p) = \frac{(\lambda_p t)^p}{p!} * e^{-\lambda_p t}$
  - $r(m) = \frac{(\lambda_m t)^m}{m!} * e^{-\lambda_m t}$
- After lots of Fancy Math….
  - $R_{sys} = (1 - 2\lambda_p t) * (1 - \lambda_m t)$
  - System Reliability Affected More Strongly by Processor Failure

System Reliability Analysis (cont.)

- Analysis of the System by Subsystem:
  - Worker Subsystem Functions When 1 out of 3 WP Nodes Works:
    - $R_{comp} = {R_{wp}}^3 - 3{R_{wp}}^2 + 3R_{wp}$
  - I/O Subsystem Functions When 1 out of 2 IP Nodes Works:
    - $R_{io} = 2R_{ip} - {R_{ip}}^2$
  - System Functions When 1 out of Each Subsystem Works:
    - $R_{sys} = R_{comp} * R_{io} = \cdots = {R_p}^2(6 - 9R_p + 5{R_p}^2 - {R_p}^3)$
  - System Can Function With a Single IP Node (Degraded):
    - $R_{sys} = R_{io} = 2R_{ip} - {R_{ip}}^2$

Conclusion

- HPEC System is Incredibly Powerful
  - Processing Power Gracefully Degrades as Nodes Fail
- Easy to Implement
  - 5 Processors and 10 DP Memory Units to Build the System
- Weakest Point is the I/O Partition
  - To Further Improve System Reliability, Focus on the Improving the IP Processor's Fault Tolerance

References

[1] (2013) Uncyclopedia “Image - Startrek-BSoD.gif”
[Web Photo] Retrieved from http://uncyclopedia.wikia.com/wiki/File:Startrek-BSoD.gif

[2] G. N. Khan, “Fault-Tolerant Architecture for High Performance Embedded System Applications“

[3] G. N. Khan, K. Mahmud, M. S. Iqbal and H. U. Rashid, "RSM - A restricted shared memory architecture for high speed interprocessor communication", *Microprocessors and Microsystems* **18**(4) 1994 pp. 193-203.