Multi/many core in Avionics Systems

4th TORRENTS Workshop
December, 13th 2013

Presented by
Jean-Claude LAPERCHE - AIRBUS
Agenda

• Introduction

• Processors Evolution/Market

• Aircraft needs

• Multi/Many-core Drawbacks & Possible ways

• Other challenges for multi-core in avionics system

• Conclusion
Introduction

- Improvement of Aircraft (A/C) Safety over last 50 years,

- Many factors: Training, Regulation rules, ...
  New A/C, New functions (Fly-by-wire, ...),

- **Safety Improvement is the result** of the COTS use and above all **COTS µPs** allowing more and more A/C functions.
Processors Evolution

- From 1970 to 200x: Mono-core architecture
  - Processing power by frequency, Cache but thermal dissipation limit.
  - In 2005, Intel stops the “Ghz” race for the new "performance per watt" race. Objective: consumption ÷ 10 by using
    - Electronic integration
    - Multi-Core Processor (MCP) architecture
Processors Market

- All µP/FPGA Manufacturers offer Multi-core µprocessor products
  - Consumer Market and Telecom, Automotive, Medical markets
  - Low power consumption multicore boosted by Tablet, Smartphone, movable device (medical, military..)
Processors Market

Avionics does not lead the processors market

Other targeted markets do not have the same constraints

Only few multi-core COTS are eligible for avionics

COTS’ internal architectures are in constant evolution
  o Towards “Systems-On-Chip”
  o Increase of power computing ⇒ multi-core ⇒ many-core
Aircraft trends

• New functions for navigation optimization, Synthetic visualization, Data transmission,
• Increased software size,
• High Speed Communication Buses (1Ghz Network ➔ 10000 A429 (100Kb/s))
• Security Data management,
• Integration: incorporate more and more functions in one computer (less weight, volume, watts, ...),
• Obsolescence management,

➤ Multi/Many-cores could:
  ▪ Be an answer for these needs,
  ▪ Help Aircraft Manufacturer to continue improving Safety
Multi/Many-core Drawbacks & Possible ways

Timing variability
- Shared Cache (L3),
- Memory controllers: accesses slowed down if simultaneous requests from ≠ cores,
- I/O controllers.

Impact on WCET

Possible ways
- Limit functional usage domain (same as mono-core µP) of the Multicore SoC to decrease demonstration efforts,
- Find an adequate execution model to meet our predictability requirements,
- Determine how to support robust partitioning (temporal, space, I/Os).

WCET: Worst case Execution Time
Multi/Many-core Drawbacks & Possible ways

**Self-reconfiguration:**
- Frequency self modification in case of overheat,
- ARM Big/Little Architecture:
  Self selection of the appropriate Core
  *with no indication to the software*

**Possible ways:**
- to deactivate these functions if possible,
- To use these functions under software management (with a complete behavior change)
Multi/Many-core Drawbacks & Possible ways

**Software Integration**: Lack of observability & Verification Means,

*Possible ways:*

- Simulation, Instrumentation/monitoring, internal debug features,

**SEU/MBU (cosmic radiation) management**

- If SEU rate should be stable, the ratio SEU/MBU ratio could increase.
- Analysis more complex, SEU/MBU effect on common features?

*Possible Ways:*

- Duplication/Triplication of application to detect/correct impact of SEU/MBU
  - Impact on processing power, time variability, etc
  - Easier with many-cores than with multi-cores?

---

SEU: Single Event Upset  MBU: Multiple Bit Upset
Multi/Many-core Drawbacks & Possible ways

**Reliability & Life Time:**

- Multi/Many-core first components “using” electronic integration.

**Impact on:**

- Infant mortality phase: failures during Final Assembly Line
- Reliability: MTBF
- Wear Out Phase: DMC(Direct Maintenance Cost), Prediction models?

**Possible ways**

- System Architecture (Redundancy/ Backup)
- Possibility to increase Reliability/Life time with a specific use:
  - cores switch off/on, cyclic use, lower voltage, lower frequency?
- New screening method to improve “Infant Mortality Phase”
Multi/Many-core Drawbacks & Possible ways

Safety demonstration:

- Quantitative analysis: Today based on constant failure rates. Tomorrow need to take into account wear-out phase (mean failure rate, Weibull law, )?  
- Multi/many-core not developed according to aeronautical standards (as usual for COTS component),
  ➔ Safety analysis depends on “erroneous behaviour” detection coverage

Possible ways

➔ Set up mitigation means to limit the Multi/Many-Core COTS undesired effects impacts (Architecture mitigation):
  ➔ At equipment level,
  ➔ At function/system level.
Mitigation means at equipment level:

Example of monitoring on mono-core SOC: Not linked with A/C function

- Validation of ALL exchanged Data (CRC)
- Global behaviors monitoring (watchdog)
Multi/Many-core Drawbacks & Possible ways

Mitigation means at equipment level:

- For Multi-core: same principle with cross-check between cores and 1 final check by an external proven component.

Diagram showing interconnects and components involved.
Multi/Many-core Drawbacks & Possible ways

Mitigation means at Function - AC Level:

→ Example of monitoring on mono-core SOC COTS

CPU zone

Complex Peripheral Zone

F: Function

MF: Monitoring Function

Computer 1

Computer 2

Servo-Control
Mitigation means at Function - AC Level:

» Example of monitoring on mono-core SOC COTS

F1: Monitoring of COTS behavior => data known, same spatial and timing that functional data linked with safety objectives.

» Monitoring could be adapted to Multi-core/manycore
Other challenges for multi/many-core in avionics

Fault Tolerance concept – [For safety critical system (as FlightByWire)]

- Multi/many cores could integrate more and more A/C functions
  - Total Component failure: simultaneous system reconfigurations!
  - Partial component failure (1 or more cores) or fault due to SEU:
    - reconfiguration to another core,
    - should be done without impact at computer/aircraft level

- Because, in case of failure, System Designer has to master
  - Computers/A/C functions reconfiguration,
  - impact on hydraulic or electric networks,
  - A/C zonal analysis,
  - Crew alerting.

⇒ Internal reconfiguration of Multi/Many-core could be set up only to reach reliability/safety objectives at component/equipment level.

With no or limited impact on aircraft flight
Other challenges for multi/many-core in avionics

System Architecture

- Multi-core COTS could have significant impact on system architecture:
  - To be adapted to multi-core:
    - WCET demonstration, parallel programming
  - Due to Architecture Mitigation
    - Dissimilarity? New Monitoring? Fault tolerant architecture

Certification aspects:

- Mainly derived from technical aspects and ways to use the COTS
  (confidence in COTS is based on experience)
- Airworthiness authorities could request more justifications
Other challenges for multi/many-core in avionics

**Industrial aspect:**

- Faultless support of component manufacturer (access to design data)
- To design «long lasting» solutions by limiting adherence to internal COTS architectures
- To Maintain competence during 30-50 years
- Avoid only one multi/many-core manufacturer for avionics,
Conclusion

- Multi/many-Core Processor:

  ➤ Normal evolution for avionics systems,

  ➤ Complexity will continue to increase,

  ➤ Complex/long studies to develop skills/solution/certification justifications,

  ➤ Potential impact on system architecture,

  ➤ **Multi/Many-cores could:**

    - propose opportunity for new A/C functions,
    - Help Aircraft Manufacturer to continue improving Safety.
Thank You