Software requirement on the next generation of space micro processors

Kjeld Hjortnaes (TEC-SWS)

Processor target domain

- The processor shall be developed for embedded Real-Time applications
- Real-time determinism is a key issue – this affects
  - Cache/no cache?
  - The cache management algorithm
  - The ability to freeze cache lines (performance optimisation)
  - Memory access model (shared/ distributed /….)
  - Fault tolerance functions.
- Availability of Commercial development tools.

The software developers view.

- The processor as seen by the software developer:
  - Available OS that makes the processor look like a “single core processor(s)” or transparent SMP (symmetric Multi-Processing)
  - Allow heterogeneous applications share the same computer resource by providing application isolation capabilities. (MMU support)
  - Linear addressing scheme
  - Pre-developed BSP, boot-up prom including self-test programs
  - Non-intrusive On-chip debugging capabilities
  - Fully compatible with mainstream software developments tools.

Software development support

- The Computer architecture and instruction set shall be compatible with commercial processors.
- Why
  - Must be able to use commercial development tools as compilers, operating systems, design tools, code, automatic code generation
  - allow for functional test and validation on commercial computers (native compilation)
- Specific processor configuration/initialisation shall be included in a pre-developed BSP/boot-up/self-test programme to be made available as Open Source from the supplier.

Cache / No cache

- Determinism of Real-Time application?
  - Cache implies indeterminism
  - For periodic applications the cache fall into repetitive pattern.
- Schedulability analysis
  - Schedulability analysis is very difficult for a cache based system.
  - Potential approach:
    - benchmark the computer performance on the application domain with and without cache.
    - perform the worst case analysis without cache accepting a smaller design margin.
- Prepare for performance optimisation functionality e.g.:
    - Cache Lines freeze function.
    - Cache partition function
    - ….

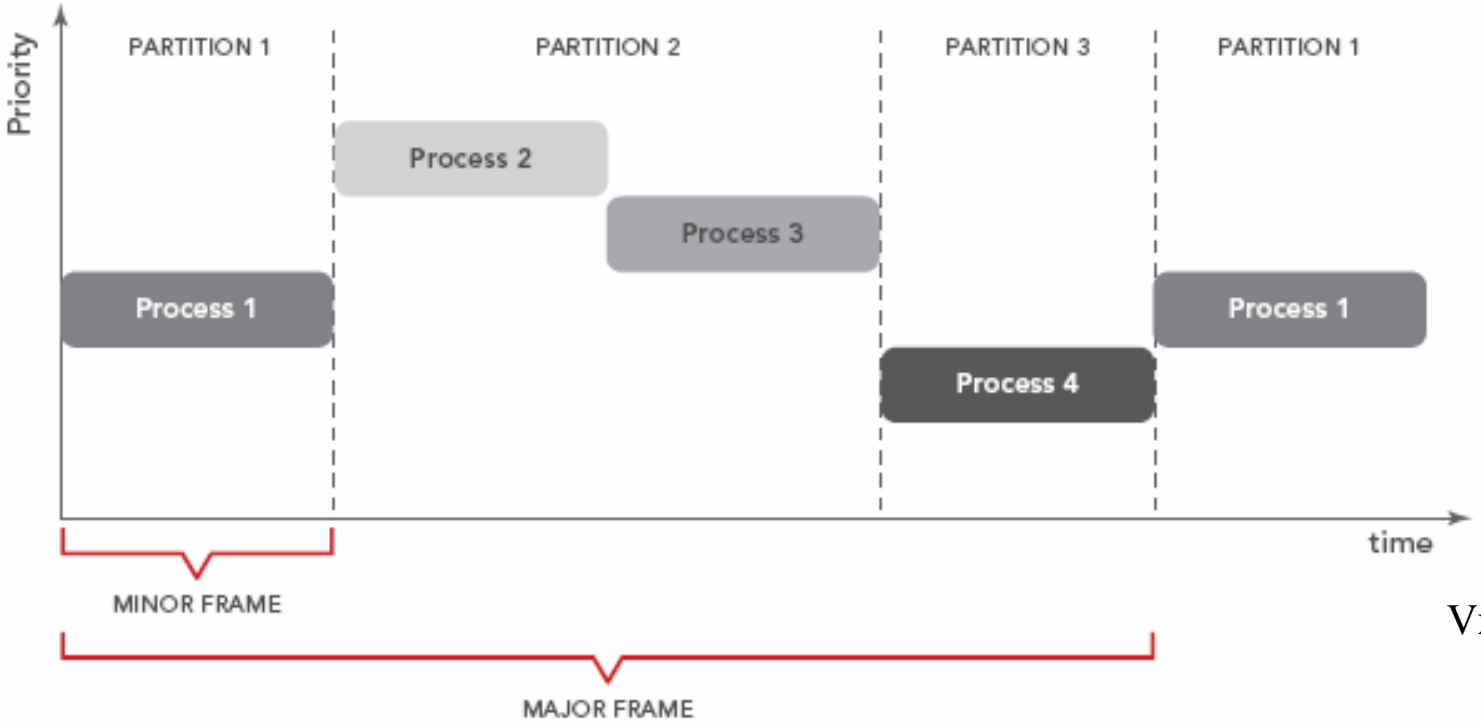
Single core / multi core

- Trend in the computer world is towards multi-core processors with cache
- <u>But in which configuration</u>
  - N independent processors integrated into one chip
    - Each programme is tested and validated for a single processor
    - Shared memory partitioned by MMU
  - A Master Slave multiprocessor system.
  - A fully Symmetric MultiProcessing (SMP)
    - Multi threading and automatic load sharing transparent to the user.
    - cache coherence implemented in hardware.
    - Require SMP compliant operating system.
  - Memory models
    - DSM (Distributed Shared Memory) / NUMO (Non-Uniform Memory Access) / Shared memory / …..

Integrated Modular Avionics (IMA)

- Defined in aviation specification ARINC-653
- Used in security and safety critical applications
  - Deployed in aviation industry e.g. A380, Boeing777.
- IMA principle.
  - multiple applications shares a common computing resource by providing spatial and temporal partitioning.
  - Provides isolation of applications from interfering with each other
  - Provide isolation of the application to the underlying hardware.

Integrated Modular Avionics.



VxWorks

- Spatial partitioning (and isolation) is provided by MMU
- Temporal partitioning (and isolation) of multiple applications running concurrently is provided by OS by allocating a time slot.
- Within each time-slot can pre-emptive or cyclic scheduling can take place.

Software validation support

CPU (sorted by increasing CPU speed)	Software Simulation Speed (100% is real-time) (Higher is better)	Needed Host performance Speed in order to run target simulator in real-time (Lower is better)
ERC-32@25 MHz (ESA)	100-200 %	1 times (Present host speed Ok)
TX49-64@25MHz (JAXA)	4-10%	10-25 times faster
Leon@100 MHz (ESA)	20-40%	2.5-5 times faster
HR 5000 64 bit@200 MHz (JAXA)	0.4-0.6 %	166-250 times faster
Quad Leon@ 250 MHz (ESA)	2-4 % ??? (extrapolated)	25-50 times faster ?

ESA's figures are based on the fastest available AMD (3.8 GHz) host machine.
JAXA's figures are based on a 3 GHz host machine.

- Using software simulators might not be possible
- We still need to produce low cost software validation facilities (with minimal HIL)
  - Provide an SVF architecture that can be "cloned" at low cost.

Software validation support

- Support for debugging is required – preference for non-intrusive debugging.
- White-box testing of the on-board software " in context" is required.
  - Potential to have monitoring watch-points assigned to code or data (might be difficult/impossible in a cache based system)
  - Provide a link between processor I/O and the environment simulation
  - Provide the means to secure synchronisation between on-board software and environment simulation (Is a CPU go/stop functionality the answer?)
  - Provide mechanism for suspend resume (e.g. set registers and Instruction pointer)

Open points

- Scalability ?
- Support for software validation facilities to provide “in context” validation.
  - Integration of environment simulation with the target microprocessor (as co-processor)
- What is the amount of physical memory shall be available per processor (at “ zero” wait-states)
  - Is there a need to have large data storage mapped into CPU memory?
- Shall the “BIOS” contain standard drivers e.g. Spacewire, support for hot redundancy configurations, external configurable start-up (multiple images)?
- Massive parallel computing – do we have those applications?
- What I/O bandwidth should we anticipate?