Multi-Core Processing in Flight Software

David Edell

JOHNS HOPKINS UNIVERSITY
Applied Physics Laboratory

Introduction

- **Multi-core processors are ubiquitous in the consumer world and are now entering the realm of space-qualified processors.**
  - ➢ **The Aeroflex Gaisler GR712RC and LEON4 are among the first of these processors.**
    - ❑ **GR712RC is a dual-core SBC**
    - ❑ **A quad-core Leon4 is available on the RASTA box**
- **A joint IRAD between APL and GSFC explored the capabilities of the multi-core LEON processor and how to utilize it in the cFE flight software architecture.**
  - ➢ **The Core Flight Executive (cFE) is an open source message bus and service framework for flight software created by GSFC.**
  - ➢ **We investigated Asymmetric (AMP) and Symmetric Multi-Processing (SMP) in both kernel-mode and user-mode (MMU-protected) environments.**
  - ➢ **Primary focus on SMP architecture, including VxWorks and Linux**
  - ➢ **Brief exploration of AMP capabilities under RTEMS**

Asymmetric vs Symmetric Multi-Processing

- Uniprocessor (UP) Systems
  - Single CPU per system
- Asymmetric Multi Processing (AMP)
  - Multiple CPUs per system acting independently
- Symmetric Multi Processing (SMP)
  - Multiple CPUs per system managed by a single operating system

Asymmetric Multi-Processing (AMP)

- Each CPU executes an independent instance of the operating system
- Standard IPC techniques are used for message exchange between cores
  - Shared message queues
  - Loopback network interface
- Hardware resources can be allocated to specific cores.
  - ➢ Each CPU is assigned a distinct area of memory and h/w resources
- OS Support:
  - ➢ VxWorks supports AMP, but the Gaisler BSP does not.
  - ➢ RTEMS includes full support for AMP
  - ➢ Additional steps are required to load and execute AMP images

Symmetric Multi-Processing

- A single OS operates across multiple processor cores
- OS can dispatch tasks to any core at runtime
  - ➢ By default, tasks ‘float’ between cores based on priority and CPU availability.
  - ➢ A task may be explicitly bound to a specific core(s) using an “affinity” flag
- LEON BSP for VxWorks 6.7 supports SMP
  - ➢ RTEMS support for SMP is under development, but was not in a usable state at the start of this IRAD.
  - ➢ SMP is common in many operating systems (ie:Windows, Linux, and Android)
- Linux SMP
  - – Baseline testing was done using a standard multi-core computer running the Linux port of cFE

SMP Task Affinity

- All tasks have a CPU Affinity attribute
  - Defines the set of CPUs a task or thread may execute on.
  - Set on task creation
  - May be updated at any time through an OS API
  - Task scheduling is a product of affinity and priority

- Task Affinity can be
  - Floating tasks are permitted to execute on any available CPU
  - If no affinity is set, a task will typically inherit the affinity of its parent

- Task Affinity is used on all SMP Operating Systems
  - VxWorks and Linux use similar cpuset macros.
  - Functionality is generally equivalent, though function names differ

Multi-Core Applications and cFE

- Extensions to the OSAL to support SMP
  - CPU Affinity can optionally be defined on task creation
  - OS API functions permit querying and re-assigning task affinity at runtime
    - By default, any task may adjust the affinity on itself or others
    - OSAL may be configured to restrict access based on target and caller task affinity.
  - Affinity functions have no effect on non-SMP systems
    - Functions will always return success, unless user explicitly sets affinity to a non-existent core.
  - All API changes are backwards-compatible
    - Implemented for VxWorks and Linux, stubbed-out for RTEMS
- SMP for cFE Architecture
  - Application startup file defines affinity for all applications
  - Mission has flexibility to determine how to allocate resources between processors

Task Affinity in cFE (user mode)

- MMU protected (user mode) extensions were made to cFE under previous IRAD efforts
  - Each POSIX process or VxWorks RTP contains a complete instance of the core flight executive (cFE)
  - Software Bus Network (SBN) application seamlessly bridges processes
    - Utilizes a loopback network interface.
    - May be extended for usage between cores for AMP
- SMP Extensions
  - Affinity may be defined for processes in startup file
    - Definition is the same as for individual tasks
  - By default, all tasks will inherit affinity from the root task in the thread or process.
  - Affinity can NOT be directly changed at runtime for a process
    - Each task/thread in that process must be explicitly changed
- System Architecture determined by mission needs
  - A single process can be defined per core
  - Processes can float between cores with explicit affinity set for applications

Multi-Processing Considerations

- Applications generally do not require any modifications to run in an SMP environment.
- Mutual exclusion semaphores automatically work across cores
- Spinlocks can provide improved performance in certain cases.
  - Allows a task to pend on a mutex without relinquishing the CPU
  - Available in VxWorks 6.6+ and reverts to standard mutex behavior in uniprocessor (UP) systems.
- Hardware interrupts may be bound to specific cores
- Global interrupt and task locks are not possible on a multi-core system
  - A task may lock interrupts and/or tasks on the current core.
  - Tasks and interrupts will continue executing normally in other cores.

VxWorks SMP Experiences

- Board Support Package
  - Gaisler SMP support requires VxWorks 6.7
  - Previous Leon3 development has used VxWorks 6.5
  - User-mode (MMU) support in LEON3 VxWorks BSP lacks maturity

- Several BSP issues were encountered
  - Significant time was spent communicating with Gaisler to address various BSP issues.
    - Majority of issues related to user-mode
  - BSP issues were primarily split between 6.7 SMP issues, general MMU issues and related system configuration.

VxWorks SMP Debugging

- Multi-core system can make it difficult to isolate the cause of an issue.
  - Once isolated and reproducible in a test case, Gaisler has been very responsive at correcting issues with the BSP in a timely fashion
- In most cases, isolating the issue was the difficulty.
  - compounded by the uncertainty of whether issues were caused by an application, or the OS/BSP
  - Certain issues only occurred sporadically
- Debugger limitations
  - VxWorks Workbench debugger has limited usage in tracking CPU exceptions
  - Grmon output can be cryptic
    - ❑ If a trap occurs outside of the currently selected core, it will be reported as an “undefined watchpoint”
    - ❑ Gaisler support was required in identifying and correcting BSP issues
- Additional Tools:
  - Quad-Core LEON4 useful in further isolating applications
  - Additional Serial Ports can be used to provide application-specific debug output ports.

VxWorks SMP for LEON

- BSP Conclusions
  - All confirmed issues have been corrected with support from Gaisler.
    - Several BSP patches generated in response to our queries
    - Some issues caused by configuration issues in our software
  - Early development encountered many stability issues
    - Majority of issues involved usermode
    - All issues have been resolved, or can no longer be reproduced.
  - VxWorks SMP support for the Leon is a new product
    - Stability and reliability in our experience has increased substantially over the past year
    - Further development and testing is required to verify the maturity of the system.

RTEMS AMP Experiences

- Started with existing single-core RTEMS configuration
  - Contains a set of applications configured for execution in 8MB SRAM
- Single OS image per core
  - For AMP, we cloned the original project and adjusted the memory map
    - ➢ Switched to 128MB SDRAM configuration of GRMON
    - ➢ Added RTEMS shared memory (SHM) region at start of RAM
    - ➢ Memory map of second image modified to load at a higher address
  - Added RTEMS AMP configuration options
    - ❑ Initialize shared memory for RTEMS IPC via message queues
    - ❑ Assign Ethernet to core-0 image
    - ❑ Assign distinct serial ports for each image
- Each core executes independent cFE instances
  - Shared message queues can be used to communicate between cores
  - APL Asynchronous Message Service (AMS) prototype used to demonstrate communication between cores as a proof of concept.
  - cFE images successfully executed, each running distinct application sets.
    - ➢ SBN app can be used to send messages between cores.
    - ➢ Due to time and scope constraints, we did not complete debugging this interface.

RTEMS AMP Summation

- Pros
  - Applications are fully isolated between cores in terms of CPU utilization
  - In the event of an OS or application failure, the other cores are not necessarily affected.
  - Reduced complexity in multi-core synchronization compared to SMP
  - Initial Setup can be more complex than for VxWorks SMP, but no issues were encountered due to OS or BSP bugs in our brief exploration
- Cons
  - AMP Loading process for development is more complex
    - A Grmon startup script is used to automate the process.
    - Each CPU image is individually loaded and its execution and stack points set prior to execution.
  - An application instance cannot utilize more than a single core.
  - Debug considerations discussed for SMP apply, but are mitigated by the fact that applications cannot float between cores.

Power Savings

- Multi-core systems offer power savings capabilities.
  - Power and performance can be dynamically scaled to mission phase
    - Ie: Reduce number of active cores in cruise versus encounter
- To effectively disable a core in flight on an SMP system:
  - Task Affinity must be explicitly set on all tasks to exclude the disabled core(s) as applicable
  - A custom idle task may place its associated core into low-power mode
    - Low-power ‘sleep’ is a function of the LEON processor core
    - The core will resume normal operation upon receiving any interrupt.
- For an AMP system
  - Low-power mode can be enabled per-core using an idler, when applicable
  - Cores may be disabled at boot if no image is loaded

Questions?

- **References:**
  - http://gaisler.com
  - GSFC cFE and OSAL http://code.nasa.gov/project/core-flight-executive-cfe/