

# **RTEMS Roadmap 2012**

**Joel Sherrill, Ph.D.**

OAR Corporation

September 2012

# RTEMS in a Nutshell

- RTEMS is an embedded real-time O/S
  - deterministic performance and resource usage
- RTEMS is free software
  - no restrictions or obligations placed on fielded applications
- Supports open standards like POSIX
- Available for 15+ CPU families and 150+ BSPs
- Training and support services available

# RTEMS Applications



# Outline of Remainder

---

- Community Driven Focus
- Development Activities
- Wish List for Future Improvements
- Active Release Branch Updates
- OAR Support Subscriptions & Legacy Releases
- Conclusion

# Community Driven Focus

- Without users, project has no reason to exist!
- Users drive requirements
  - Please let us know what you need
- Users provide or fund many improvements
  - Again those reflect your requirements
- Most bug reports are from users

**RTEMS Evolves to Meet Your Needs**

# RTEMS Project Participation In Student Programs

- Google Summer of Code (2008-2012)
  - Almost 40 students over the five years
- Google Code-In (2010-2011)
  - High school students did ~200 tasks for RTEMS
  - Included only twenty FOSS projects
- ESA Summer of Code In Space (2011-2012)
  - Small program with only twenty FOSS projects involved



# RTEMS Features

- RTEMS is a Commercial Grade Real-Time Operating System
- Truly free in price, liberty, and end user requirements
  - All source code for OS, support components, tests, documentation, and development environment
  - Test coverage is openly reported
- High performance with deterministic behavior
- Low overhead with predictable resource consumption
  - Full executables currently as small as 16K
- Highly configurable with unused features left out by linker

# How to Get Your Feature Merged

- RTEMS is free software
  - You are free to modify or add to it
- No guarantee anything will get merged
  - Must meet coding and design standards
- Improve your situation!!!
  - Get a core developer involved
  - Support them implementing your feature
- Recent user funded projects
  - RFS, Run-Time Loader, SMP, and TCP/IP Upgrade



# RTEMS Project Process Improvement

- RTEMS Project is over twenty years old
  - Very long time in computing terms
- We rely on other tools to meet project goals
- New approaches and tools introduced
  - Doxygen did not exist when project began
  - Buildbot did not exist
- New tools available to address old needs
  - For example, RCS -> CVS -> SVN -> git, etc.
- Tools may fall out of favor or unmaintained
  
- Unplanned transitions are PAINFUL
  - GNATS -> Bugzilla was too late

# Project Process Improvement Efforts

- New website in Drupal
  - Old site used MetaHTML which is now unmaintained
  - Now looks fresher and is easier to update
- Build bot for regression testing
  - Each check in will be tested for regressions
  - Automated test of all configurations impacted tested
- Transition from CVS to git
  - Driven by build bot which needed atomic change sets

# New Features – Scheduler Framework

- Pluggable Scheduler Infrastructure
  - Allows application to select thread scheduling algorithm
  - Framework to develop new algorithms
- Multiple single-core Schedulers now in tree
  - Deterministic Priority Scheduler
  - Simple Priority Scheduler
  - Earliest Deadline First (EDF)
  - Constant Bandwidth Server (CBS)
- RTEMS Scheduler Simulator to aid in writing one

# New Features – SMP

- Symmetric Multiprocessing Support
  - Initially tested up to 4 cores on LEON3 and pc386
  - GSOC student added ARM M3 support
- Opportunities for community sponsorship
  - Other target architectures
  - Processor affinity
  - Alternative SMP scheduling algorithms

# New Features – Time Representation

- Sixty-Four Bit Internal Time
  - Represents nanoseconds since POSIX epoch
  - 584.55 years before overflow
- POSIX API still thirty-two bits in places
  - RTEMS uses same C Library as Cygwin and we will follow Cygwin in changing POSIX time types
- Classic API time types OK until after 2100

# New Features –Ports & BSPs

- New ports
  - SPARC64 (e.g. V9)
  - NEC V850
  - Microblaze (slow volunteer activity)
- Ports with significant additions
  - ARM Port had Cortex-M3 Support added
  - MIPS Interrupt Processing upgraded to PIC
  - Improved NIOS II Support
    - External interrupt controller support added
- OVER 40 new BSPs since 4.10

# Ongoing Effort – Test Coverage

- Instruction and branch level coverage reports for
  - Ten BSPs covering six architectures
- For RTEMS “*proper*”
  - Instruction coverage hovers near 100%
  - Branch instruction taken/not taken is ~96%
- Current effort is to broaden code analyzed
  - Goal is all CPU Kit except third party code and shell
  - File systems and more support libraries
- Reports now included with releases

# Coverage Wish List

- More fully utilize *gcov* output from *covoar*
  - Multiple reporting tools support *gcov* format
  - Support added by ESA SOCIS student
- Output from *covoar* in *gperf* format
  - Same ESA SOCIS student added initial support
- Assistance from domain experts to
  - Get DC and MC/DC coverage reports from *covoar*
  - Produce reports in formats useful to RTEMS users
- Help from any user to improve coverage
  - Submit test cases... please

Coverage is a long journey with many small steps



# Active Effort – USB Stack

- Basic features
  - OHCI (USB 1.1) and EHCI (USB 2.0)
  - Supports removable mass storage
  - Tested on ARM, PowerPC, and x86
- Based on a FreeBSD 8.x kernel
  - Attempt to NOT modify the FreeBSD sources to improve the ability to import updates
  - Provide as RTEMS adaption layer for FreeBSD kernel services (e.g. threads, mutex and condition variables)
- Available from RTEMS Source Code Repository
  - Will be merged concurrently with TCP/IP upgrade

# Active Effort – TCP/IP Stack Upgrade

- Stack includes IPV6, IPV4, and IPSEC
  - only testing IPV4 and IPV6
- Based on same FreeBSD 8.x source as USB Stack
  - Uses much more of low level kernel sources
- Includes update of user space networking support
  - C library networking methods, client services
  - Networking commands (e.g. ping, ifconfig, route, netstat)
- Large project with numerous complex issues and need community support to complete project

# Active Effort – Run-Time Loader

- Based on the POSIX *dlopen()* family of calls
- Implemented as a run-time link editor (RTL)
  - Locates and fixes up ELF object files in the target
- Designed to have a small target footprint
  - Minimizes heap fragmentation when loading a multiple object files
  - Minimizes target resident overhead

# RTEMS Project Release Policy

- Free software project performs activity on:
  - development head
  - current release branch with bug fix releases
  - one older release branch with bug fix releases
- Limited changes allowed on a release branch
  - new BSPs may appear on current branch and head
- Release branch is made when
  - new features are ready, and
  - test quality is high enough

**All support from the community is on a volunteer basis**

# OAR Support Subscriptions

- Bring RTEMS into your project faster and simpler
- Direct Contact with the RTEMS Engineers
- Timely Responses from the RTEMS Engineers
  - normally less than 24 hours
  - always within 2 business days
- Rapid Problem Response
  - solutions as quickly as technically possible
- Assistance with the RTEMS Environment

All communication is considered confidential

# OAR Legacy Support Releases

- OAR offers legacy support for older branches
- Currently available for 4.6, 4.7, and 4.8
  - 4.6.9 includes 54 fixes not in 4.6.6
  - 4.7.5 includes 34 fixes not in 4.7.3
  - 4.8.4 includes 22 fixes not in 4.8.2
- Branch supported as long as needed by users

# Conclusion

- RTEMS Project is driven by user needs
- Continual effort to improve our processes
- Improvements will continue with user support
- Current and legacy release support available

**Please Support RTEMS Enhancement Activities**

# Contact Information

---

**Joel Sherrill, Ph.D.**

Director of Research and Development

**On-Line Applications Research (OAR) Corporation**

7047 Old Madison Pike Suite 320

Email: [joel.sherrill@oarcorp.com](mailto:joel.sherrill@oarcorp.com)

Voice: (256) 361-9375

Fax: (256) 722-0985