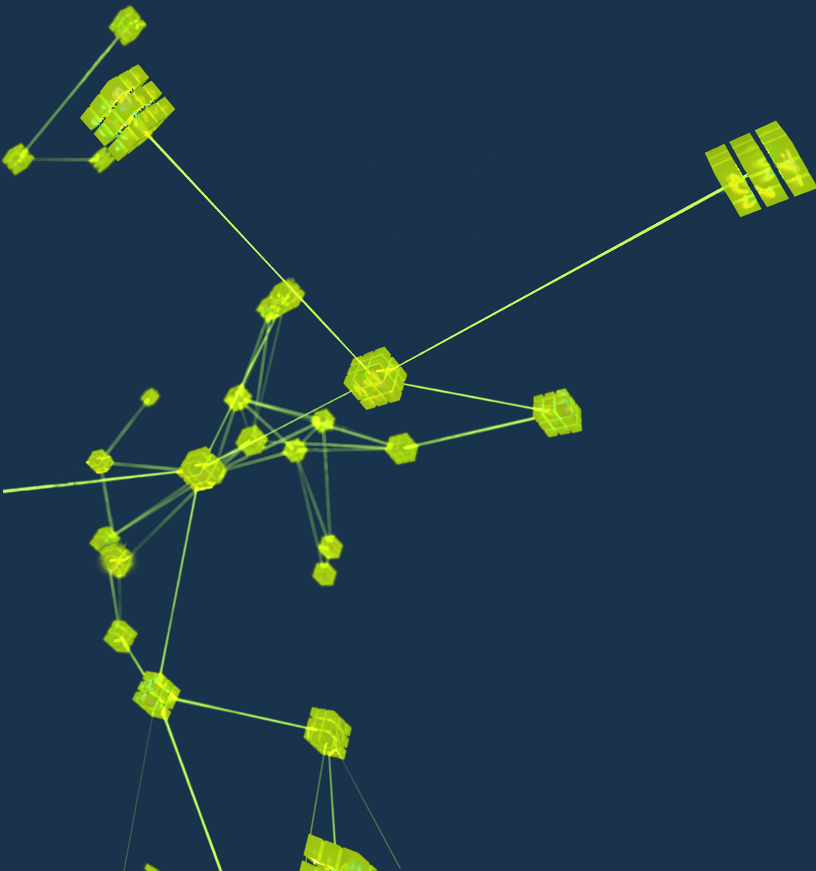


EPICS

Rtems 5
+ EPICS 7 (libcom)



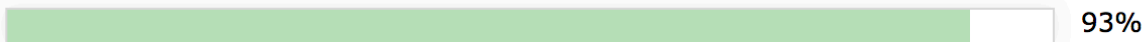
- RTEMS 5 (formerly known as RTEMS 4.12 aka RTEMS 4.11.99)
- Still not released. Release will become 5.1
(see: <https://devel.rtems.org/wiki/Release>)

Supported Releases

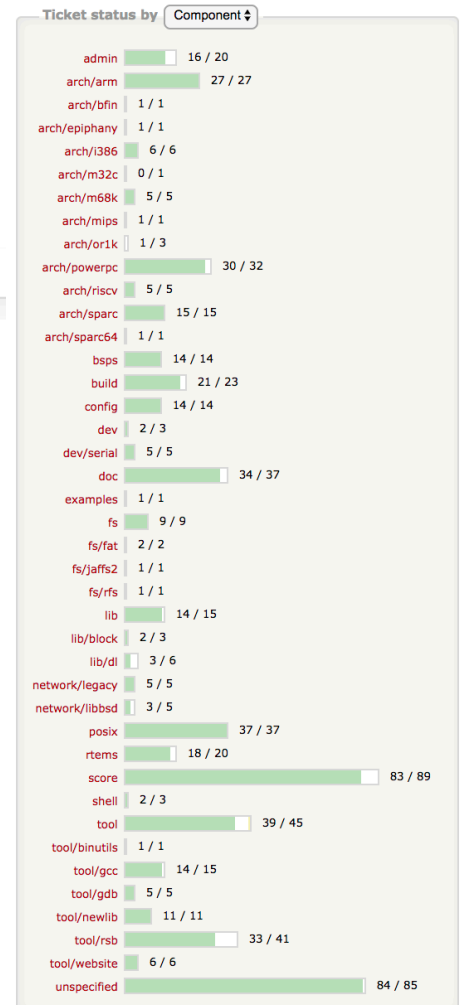
| Release (Notes) | Status | Next (milestone) |
|-----------------|------------------|------------------|
| 5 | Development | 5.1 |
| 4.11 | Current Release | 4.11.3 |
| 4.10 | Previous Release | 4.11.4 |

Milestone 5.1

No date set



Total number of tickets: 620 - closed: 573 - in progress: 45 - new: 2



RTEMS Libbsd

- Tracking latest FreeBSD
 - Started with FreeBSD 6, now at 12
- SMP performance improvement with self contained object
- Configuration follows FreeBSD system administration
- Simplified initialization via `/etc/rc.conf`
- USB with these Device Classes:
 - Input, Serial, Networking, Wlan, Storage
- Flattened Device Tree (FDT)
- MMC (eMMC)
- Networking:
 - IPv6, IPv4, IPsec, Packet Filter
 - Wireless
 - OpenCrypto, OpenSSL
 - Drivers: RE, FXP, E1000, DC, SMC, Broadcomm, plus some SoC specific including QorIQ DPAA and 10GigE
 - Commands: tcpdump, ifconfig, route, ping
 - mDNSResponder (Bonjour)
 - DHCPD
 - VLAN
- Over 50 examples and tests


“Not at the moment. I am blocked by libbsd and the FreeBSD 12 release which is 4th Dec 2018 ...

<https://www.freebsd.org/releases/12.0R/schedule.html>

This is something that I did not really appreciate until recently.” (Chris Jones)

RTEMS Improvements

- SMP performance and stability improvements
 - SPARC, ARM, PowerPC, and RISC-V
- POSIX improvements
 - headers now in Newlib which eases building third party packages including GNU Ada and FORTRAN
 - POSIX now enabled by default
 - 64-bit *time_t* (2038 here we come!)
- Self-contained objects added
- C11 thread support
- JFFS2 updated to upstream
- Flattened Device Tree (FDT) support
- xz decompression support
- 64-bit PowerPC support added
- Build system improvements
 - Removal of pre-installing header files
 - Flattening of library build under cpukit
 - Massive improvements in parallelism
- Test suite improvements
 - Improved annotation of test executable for test auditing
 - Addition of 90+ test executables
- Documentation
 - New Users Guide to help getting started
 - New RTEMS POSIX Compliance
 - Newlib libc/libm docs on path to be included in RTEMS POSIX Guide
- Many examples added



RTEMS - Real Time Operating System

[Home](#)
[Documentation ▾](#)

Welcome to the RTEMS Documentation Project

RTEMS documentation is written in [reStructuredText](#) and built using the [Sphinx](#) tools which are available for most host systems. The documentation source is available from the [rtems-docs.git](#) Git repository and the [RTEMS Wiki](#) details how to build the documentation.

The documentation project is open and welcomes new documents, chapters as well as any fixes no matter how big or small. If you see something wrong please create a [ticket](#) and if possible attach a patch.

If you need any help please ask on the [RTEMS Users](#) mailing list. You can join the list [here](#).

[Active Branches](#)
[All Releases](#)
[All Documentation](#)

Development

The **Master** branch of the documentation is the development source. You can access the latest documents below. Click below.

12nd November 2018
5.0.0 (master)

| Online | PDF | Single Page |
|---|---|----------------------|
| RTEMS User Manual |  | HTML |
| RTEMS Source Builder |  | HTML |
| RTEMS Classic API Guide |  | HTML |
| RTEMS BSP and Driver Guide |  | HTML |
| RTEMS POSIX API Guide |  | HTML |
| RTEMS POSIX 1003.1 Compliance Guide |  | HTML |
| RTEMS Filesystem Design Guide |  | HTML |
| RTEMS Networking User Manual |  | HTML |
| RTEMS Shell Guide |  | HTML |
| RTEMS CPU Architecture Supplement |  | HTML |
| RTEMS Development Environment Guide |  | HTML |
| RTEMS Eclipse Manual |  | HTML |
| RTEMS CPU Kit Doxygen | | |

<https://docs.rtems.org/>

Architecture and BSP Highlights in 5.1

- Added:
 - Architectures: RISC-V 32 and 64 bit
 - BSPs: atsamv, imx7, qoriq_e500, qoriq_e6500 (32 and 64 bit), at697f, gr712rc, gr740, ut699, ut700
- BSPs supporting SMP:
 - SPARC (1-4 cores): GR712C and GR740
 - PowerPC (1-24 cores): QorIQ (e.g. P1020, P2020, T2080, T4240, etc)
 - ARMv7-A (1-4 cores): Altera Cyclone V, Xilinx Zynq, Raspberry Pi2
 - RISC-V
- Removed:
 - Architectures: AVR, H8/300, M32C, and M32R
 - BSPs: gba, gp32, nds, Edison, gen68302, idp, mvme136, od368302, sim68000, simcpu32, genmongoosev, ep1a, mbx8xx, sis

Recent RTEMS Mission Launches



NASA Parker Solar Probe

- ▶ Launched 12 August 2018
- ▶ SPARC Flight Computer runs RTEMS
w/Saurabh Gadia's GSoC 2015 project backported to 2011 version

NASA ICESat-2

- ▶ Launched 15 September 2018
- ▶ Advanced Topographic Laser Altimeter System (ATLAS)
runs RTEMS on a mix of SPARC and PowerPC CPUs

DLR Eu:CROPIS

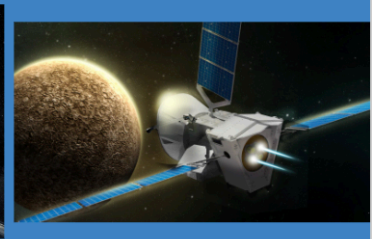
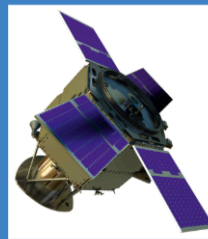
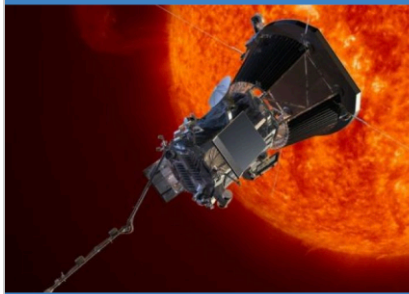
- ▶ Launched 19 November 2018
- ▶ SPARC based life-support, growing tomatoes in space

ESA BepiColombo

- ▶ Launched 8 October 2018
- ▶ RTEMS on at least MERTIS (MErcury Radiometer and Thermal infrared Imaging Spectrometer)

UAE KhalifaSat

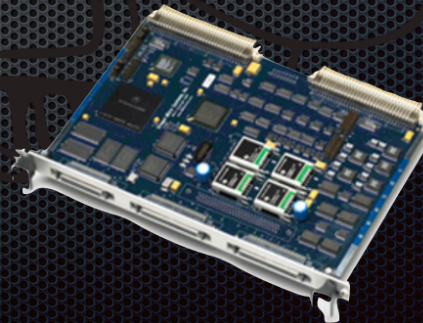
- ▶ Launched 29 October 2018
- ▶ SPARC LEON3 based imaging



- RTEMS does not provide any form of memory management or processes. In POSIX terminology, it implements a single process, multithreaded environment. This is reflected in the fact that RTEMS provides nearly all POSIX services other than those which are related to memory mapping, process forking, or shared memory
- RTEMS closely corresponds to POSIX Profile 52 which is "single process, threads, filesystem"
- Therefore very fast task switches
- Super easy to program e.g. DMA (on PCI and VMEbus) with the help of devLib2
- Can be run on embedded (small) systems

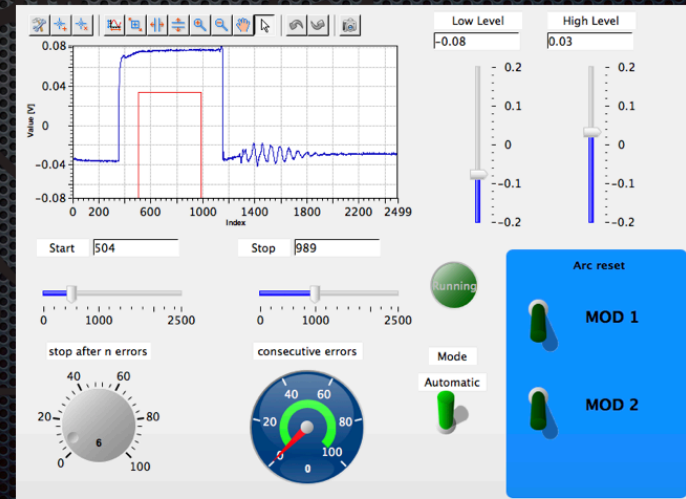
Details: Cavity stabilization

- For cavity stabilization we are using a Agilent High Resolution VMEbus Laser Axis Board (Interferometer).
- Integrated into the the VME-bus System which holds the MAXv-Card. This allows fast in-loop-control on the IOC running RTEMS.



Details: Machine Protection

- Continuously monitoring electron beam from gun to dump
- Struck 3316 digitizer compares signals from transformer toroids and Faraday cup of the beam dump with expectation values, EPICS device support uses Ethernet-UDP
- If comparision fails, RF will be switched off





EPICS V4 inside

CRYVISIL



X, Y, Z (const. height)

X, Y (coarse STM)

I/V, LockIn, other

Ethernet, ca://, asyn

Ethernet, pva://

gateway

pva2pva

NTNDArray

archiver
appliance

appliance

“fast STM” - control

VMEbus

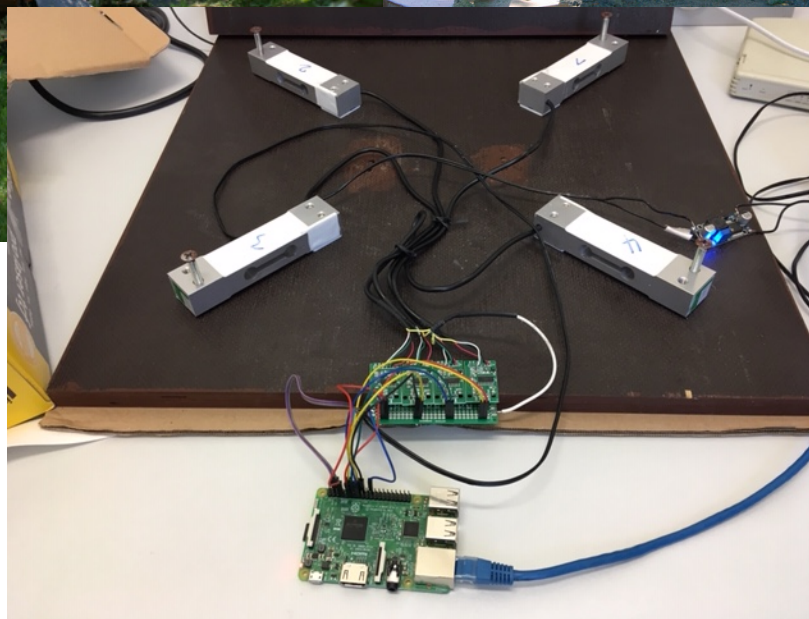
Highland V375, 4-channel arbitrary waveform generator, 15 MHz point step rate

struck SIS3316, 16 channel digitizer, 14bit, 250 MS/s per channel, 64 MSamples memory/ch

MVME2500, QorIQ, 8GB,
4 Gbit/s Ethernet

MVME6100, PowerPC, 2GB, 2 Gbit/s
Ethernet, 2 PMC-X





- Will us LoRaWAN to send the data

- The RTEMS shell commands will be provided in the Epics shell
e.g. directory commands like
 - `blksync` - sync the block driver
 - `cat` - display file contents
 - `cd` - alias for `chdir`
 - `chdir` - change the current directory
 - `chmod` - change permissions of a file
 - `chroot` - change the root directory
 - `cp` - copy files
 - `dd` - convert and copy a file
 - `debugrfs` - debug RFS file system
 - `df` - display file system disk space usage
 - `dir` - alias for `ls`
 - `fdisk` - format disks
 - `hexdump` - format disks
 - `ln` - make links
 - `ls` - list files in the directory
 - `md5` - display file system disk space usage
 - `mkdir` - create a directory
 - `mkdos` - DOSFS disk format
 - `mknod` - make device special file
 - `mkrfs` - format RFS file system
 - `mount` - mount disk
 - `mv` - move files
 - `pwd` - print work directory
 - `rmdir` - remove empty directories
 - `rm` - remove files
 - `umask` - Set file mode creation mask
 - `unmount` - unmount disk

<https://docs.rtems.org/branches/master/shell>

- Most of the existing libcom (posix) is used for RTEMS 5 with only some minor changes, does not interfere with existing RTEMS 4.x

```
# CONFIG_SITE.Common.RTEMS
#
# Site-specific information for all RTEMS targets
#-----
```

```
# RTEMS Series and Version
```

```
#RTEMS_SERIES = 4.9
#RTEMS_VERSION = 4.9.2
```

```
#RTEMS_SERIES = 4.10
#RTEMS_VERSION = 4.10.2
```

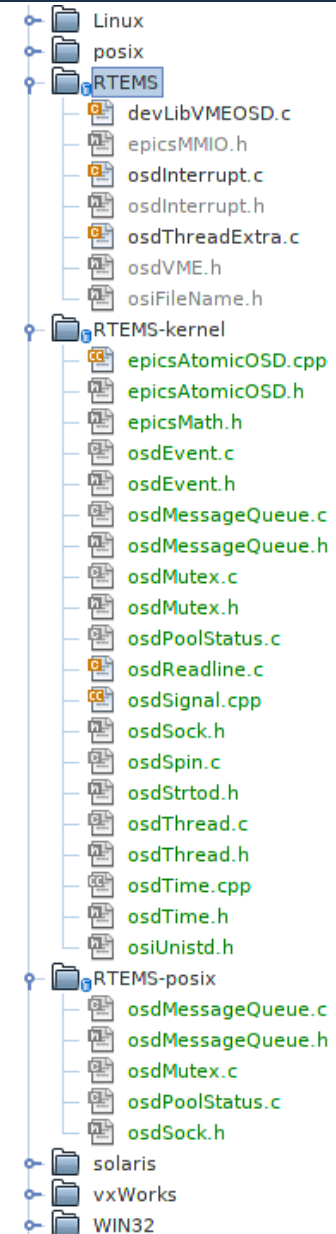
```
#RTEMS_SERIES = 5
#RTEMS_VERSION = 5
```

```
In CONFIG.Common
```

```
...
#-----
# vpath directories
POSIX_YES = os/posix
OS_IMPL_DIRS = $(if $(OS_API),os/$(OS_CLASS)-$(OS_API),) os/$(OS_CLASS)
GENERIC_SRC_DIRS = .. $(SRC_DIRS)
OS_SRC_DIRS += . $(foreach dir, $(GENERIC_SRC_DIRS), \
    $(addprefix $(dir)/, $(OS_IMPL_DIRS) $(POSIX_$(POSIX)) os/default ))
```

```
In CONFIG.Common.RTEMS
...
#-----
# operating system API (src/os/<os_class>-<os-api>)
OS_API_4.7 = kernel
OS_API_4.8 = kernel
OS_API_4.9 = kernel
OS_API_4.10 = kernel
OS_API_4.11 = $(error RTEMS-4-11 is not currently supported)
# Later RTEMS versions will use posix, no need to specify
OS_API = $(firstword $(OS_API_$(RTEMS_SERIES))) posix
...
#-----
# Operating system flags
OP_SYS_LDLIBS += -lrtemscom -lrtemscpu -lcom -linfo -lm

OP_SYS_LDFLAGS_posix += -u POSIX_Init
OP_SYS_LDFLAGS_kernel += -u Init \
    $(PROJECT_RELEASE)/lib/no-dpmmem.rel \
    $(PROJECT_RELEASE)/lib/no-mp.rel \
    $(PROJECT_RELEASE)/lib/no-part.rel \
    $(PROJECT_RELEASE)/lib/no-signal.rel \
    $(PROJECT_RELEASE)/lib/no-rtmon.rel
OP_SYS_LDFLAGS += $(CPU_CFLAGS) $(OP_SYS_LDFLAGS_$(OS_API))
```



- EPICS 7 at all works well
- Problems with mixed up output on console (RS232)
overlapping output is happening on stdout and stderr

```
gonzo> epicsSocketDestroy: failed to close a socket because "Device or 2r0e1s8o/u0r6c/e1 2b u1s1y:"1
1:18.994 [sis33160ps.cpp:624] sis0: updaftea tthraeadl stsopopiungr
c20e18:/0 6R/T12E M11S:1_1F:A19T.0A89L _[Ssi0s33U1R6C0Ep_s.EcpXp:I5T
bsp_fatal_extension(): RTEMS terminated -- no way back to MotLoad so I reset the card
Printing a stack trace for your convenience :-)
```

- Lack of a dynamic link loader, no gdb
- “A shell and loading may just be part of that. Debugging? Tracing? etc.
A wishlist for that would be great. We can turn them into tickets at least.
Getting them funded would be even better. :)” (Joel Sherill)