Introduction and Installation in RTEMS

Kuan-Hsun Chen

LS 12, TU Dortmund

27,07,2015

Outline

- Introduction of RTEMS
- Installation of RTEMS
- Example of rate-monotonic multitasking.
- Exercises

What is RTEMS?

- Real-Time Executive for Missile Systems? X
- The Real-Time Executive for Multiprocessor Systems (RTEMS) is an open source Real Time Operating System (RTOS) that supports open standard application programming interfaces (API) such as POSIX.
- It is used in space flight, medical, networking and many more embedded devices using processor architectures including ARM, PowerPC, Intel, Blackfin, MIPS, Microblaze and more.
- Commercial support is available from US and European companies, and free support comes via the active global community.

Features of RTEMS

- The list of features:
  - multitasking capabilities
  - homogeneous and heterogeneous multiprocessor systems
  - event-driven, priority-based, preemptive scheduling
  - optional rate monotonic scheduling
  - intertask communication and synchronization
  - priority inheritance
  - responsive interrupt management
  - dynamic memory allocation
  - high level of user configurability
- Please check:
  https://docs.rtems.org/doxygen/cpukit/html/modules.html

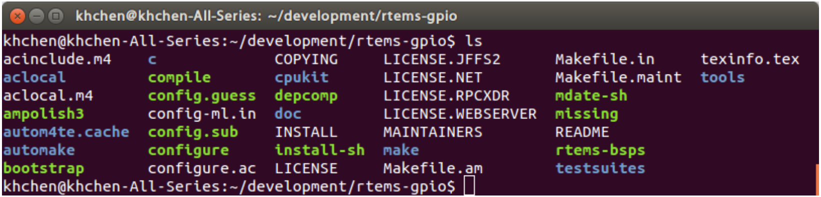
How to install RTEMS? (1/5)

1. First of all, we have to build up the cross-compiling tool chains on your host-computer.
   - We have already prepared the environment for you to ease the complexity of installation.
   - If you want to implement on somewhere, please adopt RTEMS Source Builder `ftp://ftp.rtems.org/pub/rtems/people/chrisj/source-builder/source-builder.html` to aid you building packages.
2. Then, check out the repository from Github:

```
git clone https://github.com/c0066c/rtems-gpio.git
```

3. Now you should have the source tree in your destination.

Look into the source tree (2/5)



```
khchen@khchen-All-Series:~/development/rtems-gpio$ ls
acinclude.m4    c              COPYING      LICENSE.JFFS2      Makefile.in     texinfo.tex
aclocal         compile        cpukit       LICENSE.NET        Makefile.maint  tools
aclocal.m4      config.guess   depcomp      LICENSE.RPCXDR     mdate-sh
ampolish3       config-ml.in   doc          LICENSE.WEBSERVER  missing
autom4te.cache  config.sub     INSTALL      MAINTAINERS        README
automake        configure      install-sh   make               rtems-bsps
bootstrap       configure.ac   LICENSE      Makefile.am        testsuites
khchen@khchen-All-Series:~/development/rtems-gpio$ 
```

- Some important directories to us:
  - cpukit/score/src: Provides services for all APIs (SuperCore).
  - cpukit/rtems/src: Provides RTEMS Classic APIs.
  - testsuites: Some testing programs released by RTEMS.
  - Please check the doxygen generated documentation: `https://docs.rtems.org/doxygen/cpukit/html/modules.html`

Hello world! (3/5)

- The source code of hello world can be found in ./testsuites/samples/hello/init.c
- Init() is similar as the main() in the standard C program.
- In general, the init task is used to fork the multi tasks and set up the environment. Then call rtems_task_delete(RTEMS_SELF) to terminate itself after initializing the system.
- We recommend you to check the example of "Ticker" and see how to do the multitasking.

Generating the kernel imaging (4/5)

1. We first have to build up RTEMS by the source tree.
2. Under the source directory, type the command:

   ```
   ./bootstrap
   ```

   to run a self-sustaining process for getting the configure files processed, e.g., automake.
3. Trigger the configure under the building directory:

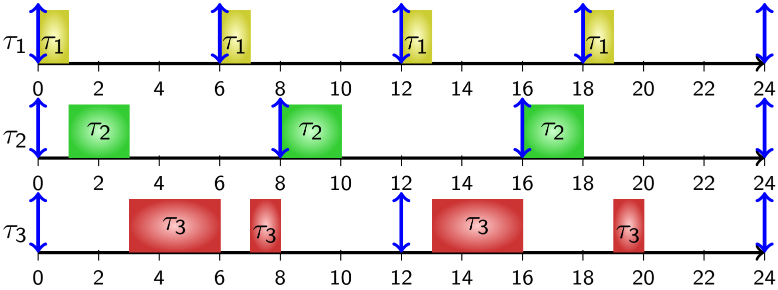
   ```
   ../rtems-gpio/configure --target=arm-rtems4.11 \
   --enable-rtemsbsp=raspberrypi \
   --enable-tests=samples \
   --enable-posix \
   --prefix=$HOME/development/rtems/4.11
   ```

Upload and execute the example on Raspberry Pi (5/5)

Rate-Monotonic Scheduling Example

Priority Definition: A task with a smaller period has higher priority, in which ties are broken arbitrarily. In RTEMS, the priorities of tasks need be defined when you create the tasks.

Example Schedule: $\tau_1 = (1, 6, 6)$, $\tau_2 = (2, 8, 8), \tau_3 = (4, 12, 12)$. $[(C_i, T_i, D_i)]$



Exercises

1. Please follow the tutorial and install RTEMS on your computer. Then upload the generated kernel on Raspberry Pi to execute. We suggest you to take a look in RTEMS.
2. Implement the Rate Monotonic example in p.10 and display the corresponding behaviours on the debug terminal.