RTJ on Raspberry Pi Documentation

Release 1.0

Oct 28, 2017

Brian Neely
Contents

1 Setting up the Raspberry Pi 3
2 Building the RT examples 5
3 Running the RT examples 7
4 Installing and Running with the Fiji VM 9
5 Installing RTEMS 11
This page will get you setup programming in Java on the Raspberry Pi. It also contains a few example tests for Java’s real time capabilities on the RPi, and instructions on how to get setup with a real-time Java VM, the Fiji VM.
Once you have your Raspberry Pi HW, follow the instructions on the RPi’s guide for setting up Raspbian, a Debian distribution for the RPi.

Now, install Java on the Pi. Run the following on the Pi (you need an internet connection):

```
sudo apt-get install openjdk-7-jre
```

Note that this will only install the Java runtime on the Pi, since we will be compiling our code elsewhere. Now your Raspberry Pi is ready to run Java.
To build the examples in this tutorial, you will need a Raspberry Pi cross compiler. Here is a link for how to install.

Once this is done, follow the instructions in the .java file for how to compile each example. The java examples are located in the Java directory. There are three examples:

1. Main directory: Generates a 500 Hz pulse on GPIO pin 0
2. ./GCStress: Does (1), while running three threads that stress the garbage collector
3. ./DDStress: Does (1), but does it through a random sequence of method dispatches, stressing the VM’s dynamic dispatch capabilities.

Once built, simply copy the files to the Raspberry Pi (I recommend using ssh). Note that the GPIO library (wiringPi.so) has a hardcoded path at /home/pi/PiGpio/wiringPi.so. This isn’t the best implementation, but you will need to place the library there for it to work.
Running is easy. Simply run:

```
sudo java <your_desired_example>
```

Note that root privileges are required to access the GPIO pins. The output pulse will be on GPIO pin 0. See the Pi4J site for the GPIO pinouts.

To measure and record the data, I used a TLA-704 Logic Analyzer. Something of this nature will be required.
The Fiji VM is a real-time Java VM. Code examples for running the above real-time Java tests are included with this example (in the Fiji directory), but the Fiji compiler/runtime cannot be included here. Contact Fiji Systems here for more info.
RTEMS is a real-time operating system with basic support on the Raspberry Pi (as of this writing). If you wish to install RTEMS on the Pi, see this article. A GPIO example in RTEMS can be found in this GitHub repository.