Contiki and RS-mote
Challenges and Experiences

A Brief Technical Report

Mary Nsabagwa
Emmanuel Kondela
Maximus Byamukama

Version change

Version Number	Comments	Date
1.0	Initial set up of the experiment	2nd Oct 2015

1. Introduction

In this report, we give our experiences of getting started with contiki operating system programming for RS2-Mote sensor node. RS2-Mote is a Universal IEEE 802.15.4 Sensor for WSN and IoT applications. We have deployed the first prototype, which we call the Bergen prototype running on a set of RS2-Motes. The RS2-Motes run contiki operating system. We would like to identify areas that will improve performance of the RS-Mote. This calls for a better understanding of the structure of contiki in order to suggest areas of improvement or even exploit its features.
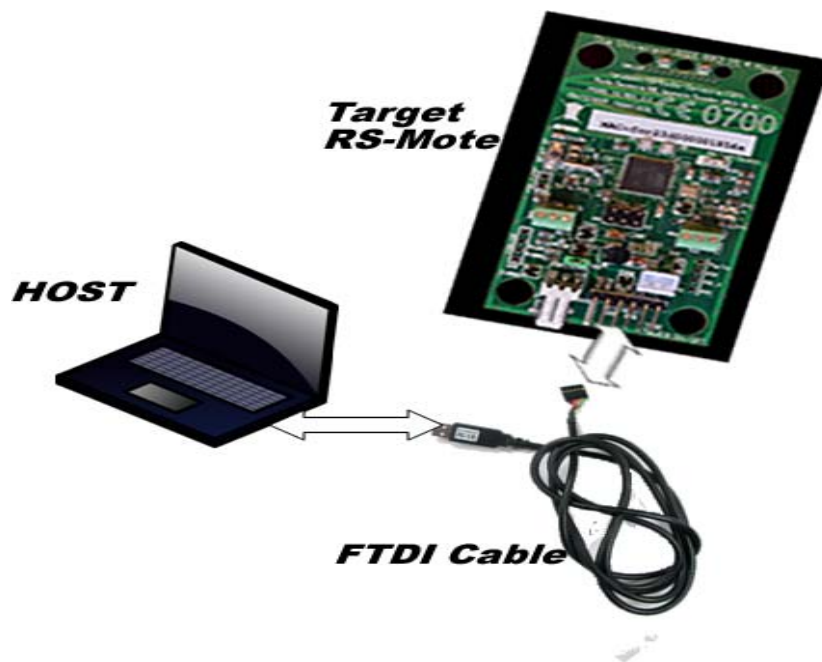
2. Experiment Set up

Figure 1 Set up of the experiment

The set up consists of three main components
   i.   Computer running software given in detail
        We used Ubuntu, a Linux distribution to run the experiments. We had to attain and install software that performs the following tasks
        • Clone contiki version that supports RS-Mote from[2]. We downloaded and installed git to perform cloning of the repository.
        • Compile and flash contiki to RS-Mote. Avrdude tool chain[4] facilitated flashing of contiki to the RS2-Mote.

```
pi@raspberrypi ~ $ sudo apt-get install gcc-avr binutils-avr gdb-avr avr-libc avrdude
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libelf1 libftdi1
Suggested packages:
  avrdude-doc task-c-devel gcc-doc gcc-4.2 gdb-doc
The following NEW packages will be installed:
  avr-libc avrdude binutils-avr gcc-avr gdb-avr libelf1 libftdi1
0 upgraded, 7 newly installed, 0 to remove and 39 not upgraded.
Need to get 25.6 MB of archives.
```

Figure 2 Installing the AVR toolchain in ubuntu

```
pi@raspberrypi ~ $ dpkg -l | grep avr
ii  avr-libc                          1:1.8.0-2                        all
for Atmel AVR development
ii  avrdude                           6.1-2+rpi1~bpo8+1                armhf
mming Atmel AVR microcontrollers
ii  binutils-avr                      2.20.1-3                         armhf
pporting Atmel's AVR targets
ii  gcc-avr                           1:4.7.2-2                        armhf
(cross compiler for avr)
ii  gdb-avr                           7.4-1                            armhf
r avr
pi@raspberrypi ~ $
```

Figure 3 Confirm if the avr tool chain has been installed

ii. RS2-Mote: The mote on which contiki is tested.
iii. FTDI cable connecting the RS2-Mote to the computer

3. Getting started with Contiki on RS2-Mote

3.1 Setting up Contiki
Below are the steps required to test contiki on RS2-Mote after installation of the required software

- Clone contiki-RS from github using
  git clone https://github.com/herjulf/contiki-RS
- Study and understand contiki directory structure
- Compile and flash contiki application onto the RS2-Mote.

```
pi@raspberrypi ~/contiki-RS/platform/avr-RSS2/apps/rime-bc $ ls
bc.avr-RSS2  bc.hex             contiki-avr-RSS2.map  hih6130.c  obj_avr-RSS2  TWI_Master.c
bc.c         contiki-avr-RSS2.a  flash.sh            Makefile   README.bc      TWI_Master.h
pi@raspberrypi ~/contiki-RS/platform/avr-RSS2/apps/rime-bc $
```

- Flashing contiki to the RS2-Mote
  We called a script that performs both the compiling and flashing of the generated contiki

binary file. After compiling, a .hex file is generated. The file is written to the mote using a command

Before flashing contiki to the mote, press the reset button of the mote for a few seconds. During this time, the mote will start booting and light red LED. Before the red LED goes off, flash the generated .hex file to the mote.

Figure Visual message showing progress in flashing on contiki

- Testing new OS over the serial line

```
&: E64=fcc23d000000231f  T_MCU=28.9  V_MCU=3.08
&: E64=fcc23d000000231f  T_MCU=27.8  V_MCU=3.08
&: E64=fcc23d000000231f  T_MCU=28.9  V_MCU=3.08
&: E64=fcc23d000000231f  T_MCU=27.8  V_MCU=3.08
&: E64=fcc23d000000231f  T_MCU=28.9  V_MCU=3.08
```

3.2 Blinking LED lights

4. Challenges

- Flashing the mote while at the same time reading from the terminal emulator. This made the flashing of firmware to fail and instead continuously light red. When reading from the serial port was stopped, pressing the reset button for 3 seconds, the firmware was successfully written.

References

1. http://radio-sensors.com/
2. https://github.com/herjulf/contiki-RS
3. http://www.nongnu.org/avrdude/
4. http://avr-eclipse.sourceforge.net/wiki/index.php/The_AVR_GCC_Toolchain