ENHANCING REAL TIME CAPABILITIES OF NANO-RK FOR TELOSB PLATFORM

CSC714-FALL 2011 PROGRESS REPORT

NORTH CAROLINA STATE UNIVERSITY
INSTRUCTOR: DR. FRANK MUELLER

TEAM MEMBERS
Devendra K Modium
Krishna Priya Kolla

OVERVIEW

Nano-RK currently runs with full functionality on the FireFly Sensor Networking Platform as well as the MicaZ motes. Our aim is to enhance the capabilities of Nano-RK for the TelosB platform (TI MSP430F1611). This project aims at implementing the radio stack and light and temperature sensor modules for Nano-RK on the TI MSP430F1611 platform.

Telos Rev. B platform has the following components, chipcon CC2420 (radio transceiver), a photo synthetically active radiation sensor (light sensor) and a temperature sensor along with other sensors. Currently Nano-RK doesn't have support for the above mentioned components on Telos Rev. B platform. Our project goal is to provide support for the radio transceiver, light sensor and temperature sensor for the TI MSP430F1611 platform under Nano-RK. Along with the above support our aim is to implement the power save mode under Nano-RK for this platform.

TIMELINE

Status	**Deadline**	**Team Members**	**Task**
Completed	October 24	Devendra, Krishna	Project proposal submission
Completed	November 1	Devendra, Krishna	Study required data sheets and refer to implementation of Radio stack in MantisOS
Completed	November 8	Devendra, Krishna	Implementation of radio stack modules in Nano-RK
Currently working on it	November 16	Devendra,	Implementation of Light & Temperature sensor modules
Currently working on it	November 16	Krishna	Implementation of power save mode module in Nano-RK
Incomplete	November 23	Devendra	Testing radio communications, light sensor and temperature sensor functionality with same text cases as in HW1
Incomplete	November 23	Krishna	Testing radio communications and power save mode
Incomplete	November 29	Krishna, Devendra	Writing final report and project presentation

RADIO STACK IMPLEMENTATION

The goal of this module to is to extend the nano-RK to have a working radio stack so that motes can communicate using RF in nano-RK. The base nano-RK code, upon which we started working on, has the higher level APIs to send and receive packets. After analyzing the code, we have found the base nano-RK doesn’t have the essential functionality of selecting and initializing USART module for SPI interface to CC2420, configuring CC2420 and hardware abstraction library(hal) functions of communicating CC2420 over SPI bus. Necessary functionality is added to the nano-RK code to make radio stack working.

Configuration of the CC2420 is performed using general I/O pins and the MSP430's SPI interface. Basic initialisation of the CC2420 is done by asserting the RESET pin active low for at least 1 µs. The VREG_EN signal must also be set for the transceiver to operate. After these two conditions are met, the MCU can start operating the RF transceiver over the SPI bus. Before any further action can take place, the CC2420 oscillator must be allowed time to stabilize. Bit 6 of the status byte holds the state of the oscillator.

MSP430F1611 has two identical USART modules USART0 and USART1. In our implementation, USART1 module is used in UART mode for asynchronous serial communication with terminal and USART0 module is used in SPI mode for synchronous communication with hardware module CC2420.

MantisOS along with MSP430F1611 hardware data sheets are taken as reference for extending nano-RK to include radio stack functionality wherever needed. To integrate the RF communication with nano-RK periodic task model, polling is used in the RF receiver. In each cycle, RF receiver checks for the packet arrival and reads the packet from CC2420 hardware RX buffers if one exists

TESTING

Basic testing of Radio module is done to make sure correct functionality is added to nano-RK. In this test, both transmitter and receiver contain a periodic task with configurable timing parameters (period, phase, execution time). In each cycle the transmitter sends a packet and toggles an LED. In each cycle, the receiver task checks if packet arrived over radio and toggles an LED if a packet is received. Since polling method is used, the frequency of receiver task is made high than the transmitter task Results shown that the two motes are able to successfully communicate over radio and good synchronization is achieved over radio synchronization by configuring the timing parameters of transmitter and receiver tasks. We have posted the source code long with the test cases on project website. The basic_rf project folder contains the required test cases.

Apart from this test, separate tests were done where large messages (size 40 bytes) are sent from one mote to other over radio. The CRC check passed for these messages. We also tested them using UART. Byte by byte of payload from transmitter is checked with receiver to make sure that no bytes got corrupted.

PENDING WORK

Devendra is currently working on the implementing the light and temperature sensor modules and Krishna is working on implementing the power save mode for nanoRK. We will first study the data sheets of the component along with those of Telos Rev.B platform specific hardware to implement the sensor modules and the power save mode. Along with them we will also be referring to other similar embedded operating systems (MantisOS and TinyOS), which have the above modules implemented for the Telos Rev. B platform.

WEBSITE

http://www4.ncsu.edu/~kkolla/CSC714/proj.html

REFERENCES

- *Nano-RK: an Energy-aware Resource-centric RTOS for Sensor Networks,* 26th IEEE International RealTime Systems Symposium RTSS05 (2005) by Anand Eswaran, Anthony Rowe, Raj Rajkumar.
- MSP430x1xx Family Data Sheet
- MSP4301611 Part Specific Data Sheet
- Nano-RK website www.nanork.org
- Mantis OS website www.mantisos.org
- Wireless Sensor Networks http://en.wikipedia.org/wiki/Wireless_sensor_network
- CSC 714 Lecture Slides