Content Delivery in

Ad Hoc Wireless Networks

Project Plan

Dec 10-03

Client:

Iowa State University

Department of Electrical and Computer Engineering

Advisors:

Prof. Lei Ying

Ming Ouyang

Team Members:

Wyatt Brenneman

Taylor McKechnie

Prashanth Yanamandra

April 26, 2010

Table of Contents

List of Figures

List of Tables

List of Definitions

Wireless Ad-hoc Network: A decentralized wireless network which does not depend on preexisting infrastructure like routers but each node participates in data transmission.[1]

USRP/USRP2 (Universal Software Radio Peripheral): USRP/USRP2 is a general purpose RF hardware device designed by Ettus Research which provides a low-cost, readily but versatile radio functionality.

GNU Radio: GNU Radio is an open-source software development toolkit for the development of software-defined radios. It contains a variety of signal processing algorithms that we will use on the data collected by the USRP hardware.

P2P (Peer to Peer): A system where the users both supply and consume the data available on the network.

SD Card (Secure Data Card): A non-volatile memory card which would be used to store programming for the USRP2 in its standalone operation mode.

BER (Bit Error Rate): The number of erroneously received bits divided by the total number of transmitted bits. BER is unit-less and expressed as a percentage.

Executive Summary

As the understanding of the world around evolves and grows, the control and distribution of information is key to the continual growth and progress of society. Previous exchanges of information required physically receiving the information from the source, receiving it from a third party, or from large network sources such as the internet. These previous methods were not conducive to real time sharing of information between individuals. Our project will allow users to stream data onto the network for others to view immediately and straight from the sources of the information.

A team of three Iowa State University College of Engineering students will work to create an Ad-Hoc wireless network which will allow the ability of users to stream information using P2P traffic. Our group will implement this network with several USRP and USRP2 radios, wireless sensors, and the GNU Radio protocol.

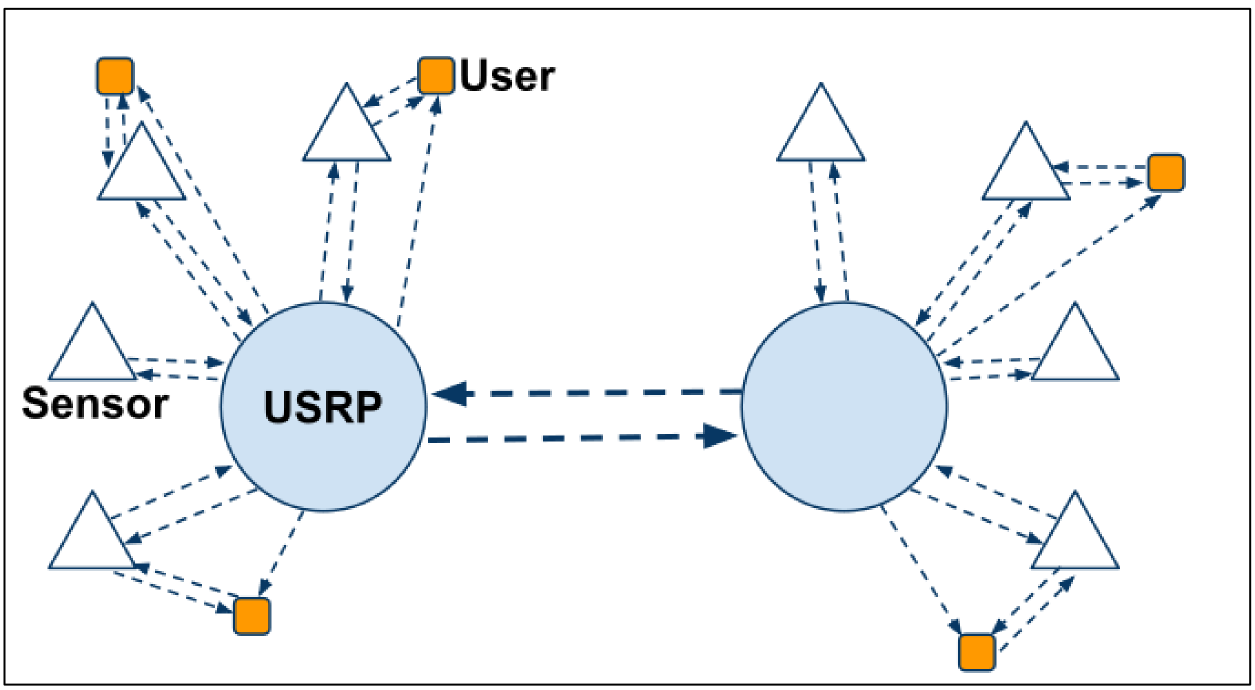
Acknowledgement

Our group would like to thank Iowa State University for providing the equipment needed for the project including the USRPs, USRP accessories, and laptops used to complete this project. We would also like to thank our faculty advisor Professor Lei Ying and his PhD student Ming Ouyang for answering our questions about the project and helping troubleshoot the errors that we have and will encounter.

Problem Statement

Wireless connectivity has become popular in our everyday life and the ability to exchange information is increasing just as fast. Examples of technologies implementing ad-hoc networks include Wi-Fi, Bluetooth and Zigbee.

In this project we will be developing our own content delivery protocols for an ad hoc wireless network. Our team will be responsible to design and implement protocol-stacks (from physical layer to application layer) on the software defined radio platform using GNU radio software and USRP/USRP2 hardware. Utilizing software defined radio significantly lowers the barrier of wireless system design by moving the design from hardware to software.



Operating Environment

The network will be providing our P2P streaming service for an individual building with the users accessing the network also inside the building. Because our system will be operating indoors to provide network coverage for the entire building we won’t have to worry about external weather conditions affecting the equipment or the transmission of the data. Since our network will be operating indoors, the network will need to overcome the many stationary physical path obstacles as well as smaller moving path obstacles. Future projects may be implemented to provide an outdoor network and will need to provide physical protection from temperature, rain, dust, weather conditions, wildlife, and vandalism. Moving the network outside will also require compensating the data signal for adverse weather conditions to improve reliability.

Intended Users and Intended Uses

Intended User	Intended Use
Non-technical unknowledgeable peripheral device user	Taking video and wirelessly broadcasting it to other users around the area who can see it in real time. Should not require any knowledge of how background network works, user only has to start filming and hit a “broadcast” button and it should broadcast the video to all users from there until the user ends filming.
Administrator	Manage feeds on the network, limiting and/or cutting the connectivity of ones that become too much of a hindrance to the network.
Commercial user	Broadcast of meeting or demonstration from on-site area to an office building or safe-zone.
Educational user	Broadcast of period of instruction by a professor to students in his/her class

Expected End Product and Other Deliverables

The end product delivered will be

- *Fully Functional network*

The network, which is capable of communicating with all the USRPs, sensors, and capable of delivering content to users will be delivered at the completion of the project.

Other Deliverables include

- *Project Plan Document*

The project plan document highlighting the proposed approach plan will be provided at the completion of the project

- *Design Document*

The design document highlighting the design approach considered for the project will be provided at the completion of the project.

- *Design Review Presentation*

The presentation made for the design review will be provided at the completion of the project

- *Project Poster*

The poster designed highlighting our achievements for the project will be provided at the completion of the project

- *User Guide*

A user manual will be provided to the user upon completion of the project in order to act as a reference on how each element of the system works. This document will include a software guide as well as instruction sets that assist with training and reference material.

Functional Requirements

- *Video streaming to peripheral users at 10fps*
    - A fully functional system must be capable of providing a bitrate that allows 10fps of streaming video to peripheral users.
- *Simultaneous Streaming*
    - The fully functional system must be capable of supporting 10 simultaneous streams at any instant of time.

- *Minimum data rate at all time*
  - Our team plans to provide a minimum data rate of 10kBps at all times for streaming the broadcasted data. This minimizes the lag in the real-time video when streamed.
- *Maximum bandwidth utilization*
  - The system is limited to a bandwidth of 8MHz for the USRP and 20MHz for the USRP2, which will limit our bandwidth to 8MHz when utilizing both USRP models.
- *Broadcasting range*
  - The system must be able to provide good reception for users within 100 feet of the USRP/sensors indoors.

Non- Functional Requirements

- *Physical Dimensions*
  - The dimensions of the sensors need will need to be small enough to be implemented in classrooms and labs without needing to change any layout of the room. The dimension of the USRP is fixed.
- *Equipment Protection*
  - The equipment is placed indoors at all times and does not require weather protection.
- *Power Requirements*
  - The USRP and the USRP2 need an AC to DC converter. The sensors will need 2 AA batteries to supply power or solar sensors that will convert light from interior light sources into energy used by the sensor.

Technology Considerations

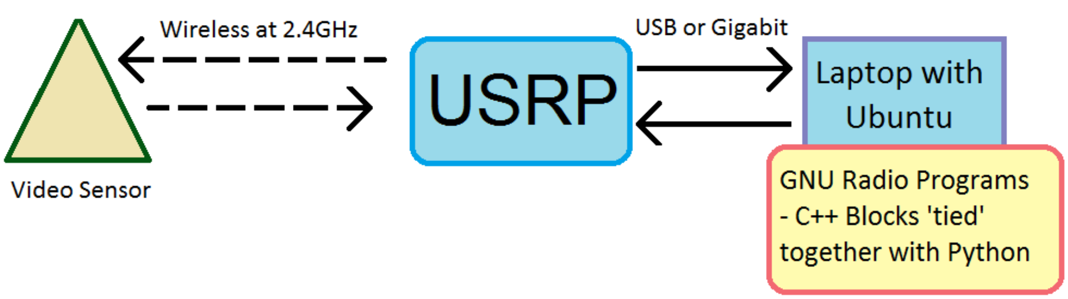
- *GNU Radio:* GNU Radio is an open-source software development toolkit which enables a variety of signal processing on the data collected by the USRP hardware. GNU radio is a software interface to the USRP. This technology is used for wireless link between electronic systems

- *USRP/USRP2*: The USRP/USRP2 is a general purpose Radio Frequency hardware device which provides a low-cost, readily but versatile radio functionality. It runs on the GNU radio platform.

- *Sensors:* Our team is planning on implementing sensors made by Texas Instruments and Crossbow. Some of the sensors considered have USB connectivity, whereas, other sensors have wireless connectivity.

System Decomposition

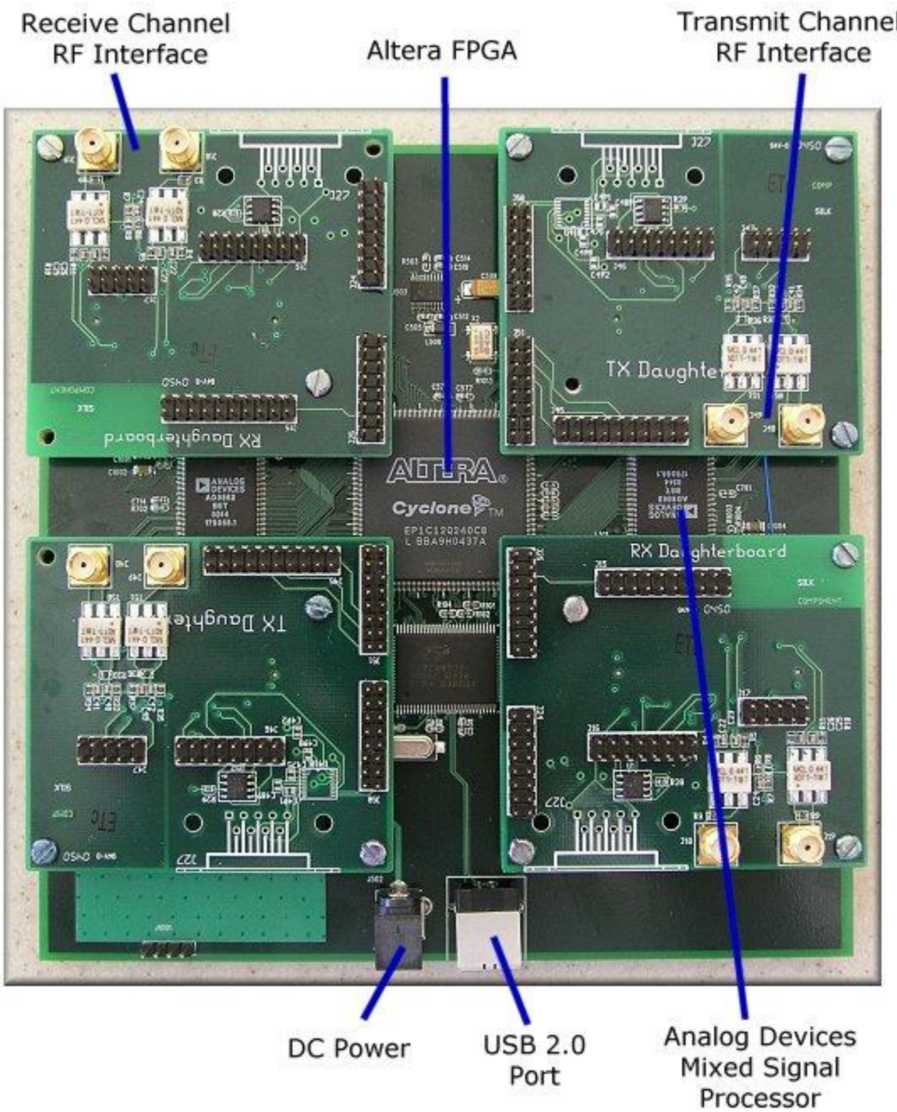
The USRP sends and receives information from the video sensor wirelessly at 2.4GHz. The data packets received from the video sensor are processed by the FPGA in the USRP. The processed data is then sent to the laptop computer through a USB cable or a gigabit cord.



Detailed Design Overview

USRP

For our project we will be using the USRP as well as the newer USRP2 to provide the hardware interface of our radio system. The main reason for using the USRP/USRP2 is that they were designed with the GNU Radio software in mind and created to maximize the capabilities of GNU Radio. The USRP/USRP2 design is also open source, which means that all of the tools we will be using to create our final product are open source. This fact allows us forgo typical software and licensing fee that could be attributed to a typical project of our nature.



Details of Hardware

For both our USRP and USRP2 modules we will be using the same antennas and daughterboards.

Antenna: We will be using the VERT2450 antenna on all of our USRPs.

- Dual band 2.400 – 2.800 GHz and 4.900 – 5.900 GHz.

Daughterboard: We will be using the RFX2400 Transceiver daughterboard in all of our USRPs.

- Fully controllable through radio software or FPGA
- Transmitting power of 50mW (17 dBm)
- Able to transmit and receive through one antenna and board connection
- 30 MHz transmit and receive bandwidth
- Frequency range of 2.300 – 2.900 GHz

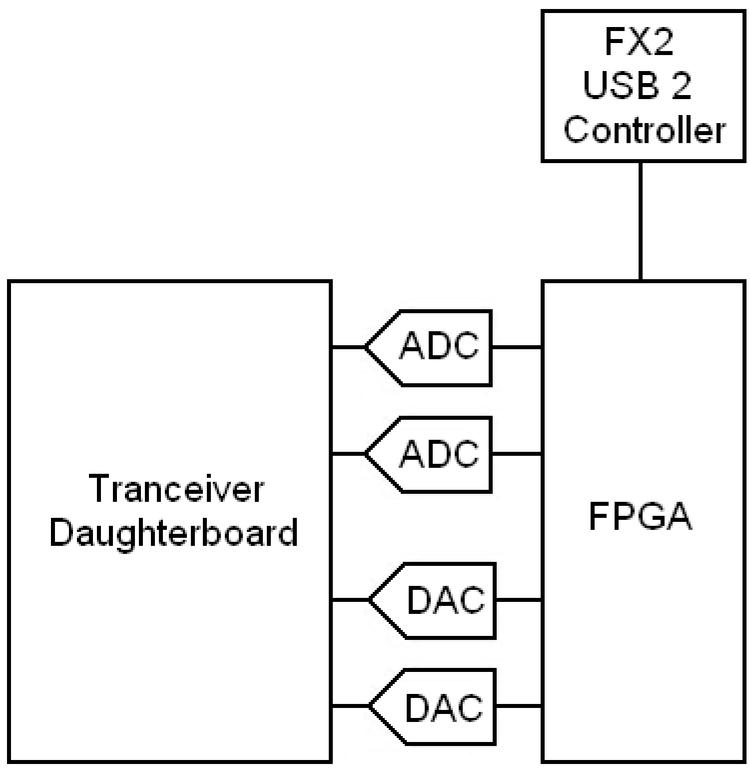
The first generation USRP utilizes 4 high-speed ADCs and DACs which are connected to an Altera Cyclone EP1C12 FPGA. The USRP also uses USB2 to interface with laptop.

Analog to Digital Converters

- 64 Ms/s at 12-bit

Digital to Analog Converters

- 128 Ms/s at 14-bits



The second generation USRP2 also utilizes 4 high-speed ADCs and DACs, which are then connected to a Xilinx Spartan 3-2000 FPGA which then connects to a Gigabit Ethernet interface.

Analog to Digital Converters

- 100 Ms/s at 14-bit

Digital to Analog Converters

- 400 Ms/s at 16-bits

Software Design

Our team plans on using Orthogonal Frequency Division Multiplexing for our transmission scheme. This scheme has advantages over other communication techniques. This method is considered as it reduces the cross-talk between different communicating devices. This is due to the fact that OFDM makes each of the signals orthogonal to each other, thereby minimizing the interference.

Advantages of using OFDM signaling scheme:

- This scheme has high spectral efficiency as compared to other communication techniques.
- This scheme has less interference, so this when messages are decoded at the receiver; the Bit Error Rate is less, which results in a high percentage of success.
- This scheme can be implemented efficiently using Fast Fourier Transform.
- GNU Radio has number OFDM blocks included that can be modified

GNU Radio is a widely used software platform for educational and research processes at numerous universities and offers a reliable and flexible communication platform. There are numerous resources available that will help us learn from and apply efficient transmission, receiving, and processing techniques using this platform.

GNU Radio software is an open-source project with an extensive library of signal processing and radio related building blocks. Since GNU Radio is open-source we will have access to the code of these blocks, which will allow us to simply modify them to fit our needs instead of having to program them all from scratch.

The signal processing blocks are programmed in C++, and the links between the blocks forming the communications system are written in Python. Python is new to us but it is used because of it easy to read and understand syntax. C++ is widely used as design language and we are all somewhat familiarly with it because we have programmed in C or C++ before.

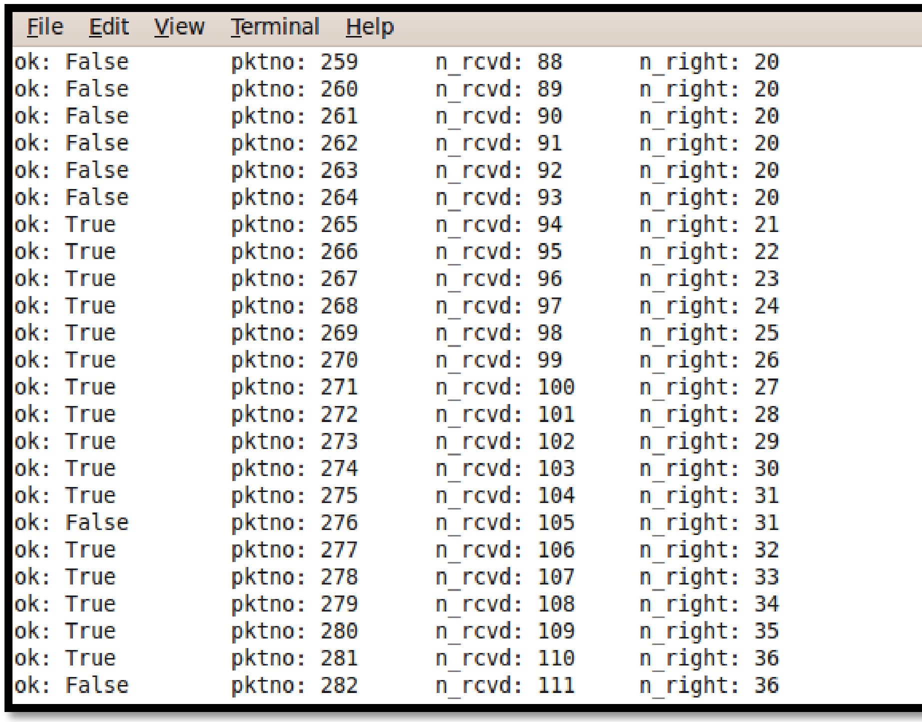
We also chose GNU radio software considering the fact that it is a free software and with no license constraints of fees to use this software. This can save our team some money from purchasing expensive software.

The new installations of GNU Radio also include the GNU Radio Companion (GRC) which is a graphical user interface. We foresee using this tool to quickly test small parts of our system in order to save time in programming ideas that will not work. GRC also allows new users to learn some basics of how GNU Radio works before diving into programming Python and C++ code.

Testing and Evaluation

Testing will be performed in various stages during the course of the project. The initial stage of testing includes testing the system by sending data packets from one USRP to another USRP. Once this is successfully tested, communication will be established between more than one USRP, wherein data packets will be sent from one USRP and will be received by the others. This method will result in eliminating interference between devices.

The data sent from one computer will be received by the other computer and will be analyzed. The analysis of the data includes checking for errors in transmission and the number of packets successfully received. The data is being transmitted through air and this will result in the data being corrupted by noise. The data when analyzed by the receiver will test for the Bit Error Rate. Transmission and receiver delays will also be analyzed.



```
File  Edit  View  Terminal  Help
ok: False       pktno: 259      n_rcvd: 88      n_right: 20
ok: False       pktno: 260      n_rcvd: 89      n_right: 20
ok: False       pktno: 261      n_rcvd: 90      n_right: 20
ok: False       pktno: 262      n_rcvd: 91      n_right: 20
ok: False       pktno: 263      n_rcvd: 92      n_right: 20
ok: False       pktno: 264      n_rcvd: 93      n_right: 20
ok: True        pktno: 265      n_rcvd: 94      n_right: 21
ok: True        pktno: 266      n_rcvd: 95      n_right: 22
ok: True        pktno: 267      n_rcvd: 96      n_right: 23
ok: True        pktno: 268      n_rcvd: 97      n_right: 24
ok: True        pktno: 269      n_rcvd: 98      n_right: 25
ok: True        pktno: 270      n_rcvd: 99      n_right: 26
ok: True        pktno: 271      n_rcvd: 100     n_right: 27
ok: True        pktno: 272      n_rcvd: 101     n_right: 28
ok: True        pktno: 273      n_rcvd: 102     n_right: 29
ok: True        pktno: 274      n_rcvd: 103     n_right: 30
ok: True        pktno: 275      n_rcvd: 104     n_right: 31
ok: False       pktno: 276      n_rcvd: 105     n_right: 31
ok: True        pktno: 277      n_rcvd: 106     n_right: 32
ok: True        pktno: 278      n_rcvd: 107     n_right: 33
ok: True        pktno: 279      n_rcvd: 108     n_right: 34
ok: True        pktno: 280      n_rcvd: 109     n_right: 35
ok: True        pktno: 281      n_rcvd: 110     n_right: 36
ok: False       pktno: 282      n_rcvd: 111     n_right: 36
```

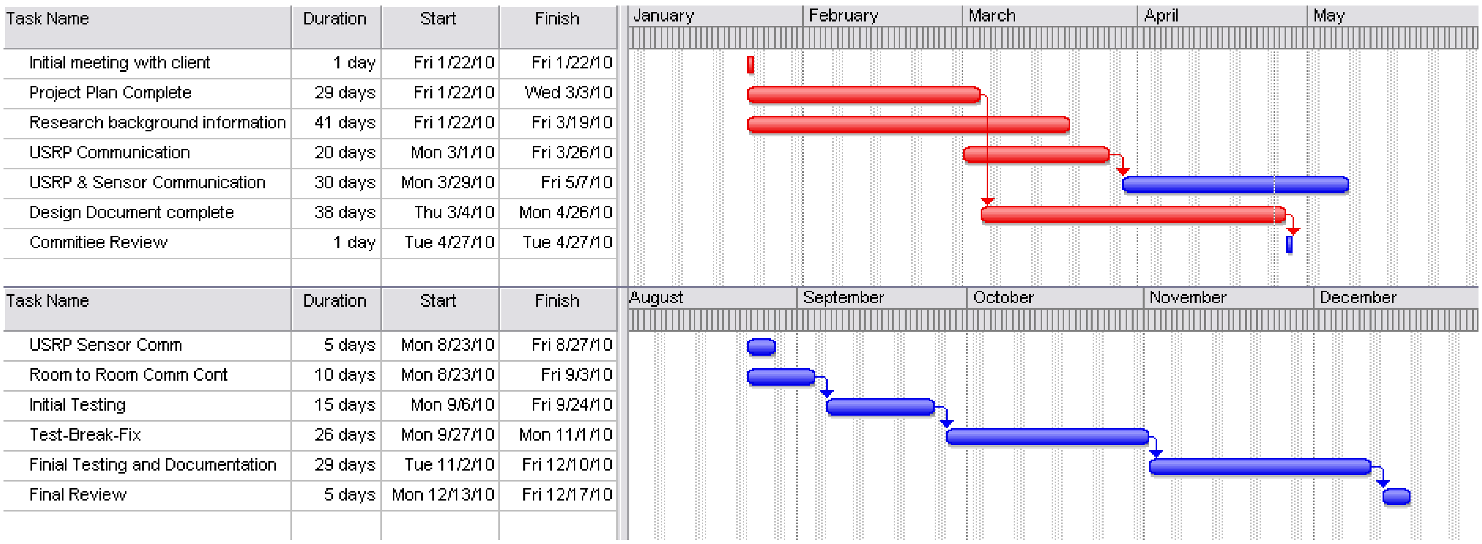
Testing Phases

Communication will be established between USRPs within the same room. Once this is successfully completed, then communication will be established between USRPs in neighboring rooms. This will test the system to deliver a constant data rate within a certain range as specified in the functional requirements. The signals tend to weaken when they pass through concrete structures and must be received efficiently with least amount of errors.

Once this stage is complete, the sensors will be added to the network which boost the signal strength and transmit over a farther distance. This network will then be tested in the same floor with USRPs separated by a greater distance.

Once we are able to send and receive data packets between USRPs in the same floor, we will then extend the systems to different floors and testing for all the necessary parameters. Eventually, we will extend the system to the entire building.

Project Schedule



Task Name	Duration	Start	Finish
Initial meeting with client	1 day	Fri 1/22/10	Fri 1/22/10
Project Plan Complete	29 days	Fri 1/22/10	Wed 3/3/10
Research background information	41 days	Fri 1/22/10	Fri 3/19/10
USRP Communication	20 days	Mon 3/1/10	Fri 3/26/10
USRP & Sensor Communication	30 days	Mon 3/29/10	Fri 5/7/10
Design Document complete	38 days	Thu 3/4/10	Mon 4/26/10
Commitiee Review	1 day	Tue 4/27/10	Tue 4/27/10

Task Name	Duration	Start	Finish
USRP Sensor Comm	5 days	Mon 8/23/10	Fri 8/27/10
Room to Room Comm Cont	10 days	Mon 8/23/10	Fri 9/3/10
Initial Testing	15 days	Mon 9/6/10	Fri 9/24/10
Test-Break-Fix	26 days	Mon 9/27/10	Mon 11/1/10
Finial Testing and Documentation	29 days	Tue 11/2/10	Fri 12/10/10
Final Review	5 days	Mon 12/13/10	Fri 12/17/10

The red bars indicate completed tasks and blue bars indicate tasks yet to be completed.

Task Breakdown

Semester 1

	Research	Hardware Implementation	Software Design	Documentation	Total
Wyatt Brenneman	35	5	10	30	80
Taylor McKechnie	40	0	10	25	75
Prashanth Yanamandra	35	5	10	30	80
Total	105	10	30	90	215

Semester 2 Projections

	Transmission	Receiving	Signal Processing	Testing	Documentation	Total
Wyatt Brenneman	50	20	10	50	30	160
Taylor McKechnie	10	10	60	50	30	160
Prashanth Yanamandra	20	50	10	50	30	160
Total	80	80	80	150	90	480

Client/Advisors Information

Professor Lei Ying
3219 Coover Hall
Ames, IA 50011
leiying@iastate.edu
515 294-5353

Ming Ouyang
2215 Coover Hall
Ames, IA 50011
mouyang@iastate.edu
515 294-2664

Project Team Information

Wyatt Brenneman
232 Walnut Ave Unit 16
Ames, IA 50010
wgbn01@iastate.edu
319 981-7003

Taylor McKechnie
225 Hyland Ave Apt 12
Ames, IA 50014
thrp1st@iastate.edu
402 926-1033

Prashanth Yanamandra
143 University Village Apt F
Ames, IA 50010
psy@iastate.edu
321 704-6321

Closing Summary

In a world where the need of fast and reliable exchange of information is growing our network will showcase one possibility of how this can be achieved. Our network will rely on the flexibility of P2P sharing and combine it with the immediate gratification of streaming data. By streaming the data the users will not be required to wait to view the information they need as well and continue to propagate the data through the network.

References

[1] Crossbow Wireless Modules Portfolio – Crossbow Technology. Retrieved March 1, 2010. From the World Wide Web : < http://www.xbow.com/Products/productdetails.aspx?sid=156>.

[2] Ettus Research LLC. Retrieved March 1, 2010. From the World Wide Web : <http://www.ettus.com/>.

[3] "GNU Radio". Retrieved March 1, 2010 from the World Wide Web. <http://gnuradio.org/trac>.

[4] "GNU Radio." *Wikipedia, the free encyclopedia*. Retrieved March 1, 2010 from the World Wide Web: <http://en.wikipedia.org/wiki/GNU_Radio>.

[5] ISU Electrical and Computer Engineering: ECpE Building Addition and Coover Hall Renovations Project. Retrieved March 1, 2010 from the World Wide Web: <http://www.ece.iastate.edu/typo3temp/pics/6552a907c0.gif>.

[6] Jackey, John. FPGA for MRFM Cantilever Control. Retrieved April 26, 2010 from the World Wide Web :< http://www.research.cornell.edu/KIC/events/MRFM2006/pdfs/Jacky%20talk/jackytalk.html>.

[7] "Peer-to-peer." *Wikipedia, the free encyclopedia*. Retrieved March 1, 2010 from the World Wide Web: <http://en.wikipedia.org/wiki/Peer-to-peer>.

[8] "Universal Software Radio Peripheral." *Wikipedia, the free encyclopedia*. Retrieved March 1, 2010 from the World Wide Web: <http://en.wikipedia.org/wiki/Universal_Software_Radio_Peripheral>.