

ABSTRACT

Title of Dissertation: EMBEDDED SYSTEM FOR CONSTRUCTION
MATERIAL TRACKING USING COMBINATION OF
RADIO FREQUENCY AND ULTRASOUND SIGNAL

Won Suk Jang, Doctor of Philosophy, 2007

Dissertation Directed by: Professor, Mirosław J. Skibniewski
Department of Civil and Environmental Engineering

This study created a framework for integrating the latest innovations in wireless sensor network that automate tracking and monitoring construction assets, e.g. equipment, materials, and labor in construction sites. This research constitutes one of the few studies to incorporate emerging information and sensor network technologies with the construction industry, which has been slow to migrate away from legacy processes.

The presented research works introduce a new prototype framework of an automated tracking system that will address the needed shift from the time-and labor-intensive legacy systems to sensor- and network-based collaboration and communication systems for construction processes. Software and hardware architecture for the new tracking system was developed using the combination of ultrasound and radio signals. By embedding the external ultrasound device with a MICAZ platform, enhancements to networking flexibility and wireless communication was observed over the previous technologies used in the construction material tracking systems. Feasibility study and testbed experiment on the position estimation were implemented to verify the localization algorithm presented in this dissertation.

Cost benefit analysis based on quantitative approach implied that the presented framework can save the implementation cost of material tracking by up to 64 percent in a typical construction project. In addition to cost savings, the use of sensor-based tracking system can provide the intangible, comprehensive benefits in communication, labor utilization, document management, and resource management.

It is hoped that the present work will describe a system that can effectively be used in a range of applications for tracking and monitoring purposes and will present a clear path that engineers can take to use existing wireless sensor technology in their particular applications. The cost of such hardware will decrease rapidly, thereby permitting large numbers of application scenarios to be possible in many construction sites with improved energy consumption, hardware performance, durability, and safety.

**EMBEDDED SYSTEM FOR CONSTRUCTION MATERIAL TRACKING
USING COMBINATION OF RADIO FREQUENCY AND
ULTRASOUND SIGNAL**

by

Won Suk Jang

**Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2007**

Advisory Committee:

**Professor Miroslaw J. Skibniewski, Chair
Professor Sung W. Lee, Deans' Representative
Professor Gregory B. Baecher
Professor Mehdi Kalantari Khandani
Professor Michael Casey**

©Copyright by

Won Suk Jang

2007

Dedication

To my parents, Ji-Hun, and Hannah

Acknowledgements

I would like to thank all of those people who helped make this dissertation possible.

First, I wish to gratefully thank my advisor, Dr. Mirosław Skibniewski for all his guidance, encouragement, support, and patience. Over the past few years, he has inspired me to create the innovative research ideas to overcome all technical problems and to achieve the research work done in this dissertation. Without his support and encouragement, none of this work would have been possible. I would also like to thank Dr. William Healy, who is a group leader at National Institute of Standards and Technology. His support for experiments on the wireless sensor network was very helpful to complete this work successfully.

Grateful acknowledgement is given to my dissertation committee members, Prof. Gregory Baecher, Prof. Sung Lee, Dr. Mehdi Khandani, and Dr. Michael Casey, for their very helpful insights, comments and suggestions. I also thank the members of e-Construction Group for stimulating discussions and bringing so much fun to my life, both research and non-research alike.

Finally, I would not have had survived days of my life completing this dissertation without my parents, my wife Ji-Hun, and my daughter Hannah. My best acknowledgement is given to my lovely family.

November 21, 2007

Table of Contents

| | Page |
|--|------|
| Dedication | ii |
| Acknowledgements..... | iii |
| Table of Contents..... | iv |
| List of Tables | vii |
| List of Figures..... | viii |
| | |
| CHAPTER 1: INTRODUCTION..... | 1 |
| 1.1 Overview..... | 1 |
| 1.2 Motivation..... | 3 |
| 1.3 Research Objectives..... | 9 |
| 1.4 Contributions of the Proposed Research..... | 11 |
| 1.5 Organization of Dissertation..... | 13 |
| | |
| CHAPTER 2: LITERATURE REVIEW..... | 15 |
| 2.1 Components Tracking Systems in Construction Industry | 15 |
| 2.2 Localization Techniques..... | 27 |
| 2.3 Chapter Summary | 34 |
| | |
| CHAPTER 3: WIRELESS SENSOR NETWORKS..... | 36 |
| 3.1 Introduction..... | 36 |
| 3.2 Hardware Architecture..... | 38 |
| 3.2.1 Microcontroller | 39 |
| 3.2.2 RF Transceiver..... | 40 |
| 3.2.3 Gateway | 41 |
| 3.2.4 Sensor Board..... | 42 |
| 3.3 ZigBee with IEEE 802.15.4 Network Standard..... | 43 |
| 3.3.1 IEEE 802.15.4 and ZigBee Structure..... | 44 |
| 3.3.2 Zigbee Network Layer | 47 |
| 3.4 Software Architecture | 48 |
| 3.4.1 Execution Model..... | 49 |
| 3.4.2 Component Model | 51 |

| | | |
|--|--|-----|
| 3.4.3 | Communication Model | 53 |
| 3.5 | Deployment Metrics and Challenges | 55 |
| 3.5.1 | Life Cycle..... | 55 |
| 3.5.2 | Cost and Size..... | 57 |
| 3.5.3 | Coverage Range | 58 |
| 3.5.4 | Signal Interference | 59 |
| 3.6 | Chapter Summary | 60 |
| CHAPTER 4: FEASIBILITY STUDY..... | | 61 |
| 4.1 | System Architecture for Tracking Algorithm..... | 61 |
| 4.1.1 | Coordinating | 62 |
| 4.1.2 | Query and Response Pulse based on TOF | 65 |
| 4.2 | Feasibility Analysis..... | 67 |
| 4.2.1 | Measurement of the Distance by RF Signals..... | 68 |
| 4.2.2 | Measurement of the Distance by combination of RF and Ultrasound Signals | 70 |
| 4.3 | Simulation on the Position Estimation..... | 72 |
| 4.3.1 | Trilateration Technique..... | 73 |
| 4.3.2 | Kalman Filter | 75 |
| 4.3.3 | Feasibility Simulation for Accuracy Performance..... | 80 |
| 4.4 | Scheduling Scheme..... | 90 |
| 4.4.1 | System Overview | 91 |
| 4.4.2 | Single Communication..... | 95 |
| 4.4.3 | Multi-Communication..... | 98 |
| 4.4.4 | Mobility Model..... | 102 |
| 4.5 | Deployment Challenges in Construction Sites | 104 |
| 4.6 | Chapter Summary | 105 |
| CHAPTER 5: PROTOTYPE EXPERIMENT FOR MATERIAL TRACKING SYSTEM.. | | 106 |
| 5.1 | Overview..... | 106 |
| 5.2 | Hardware Configuration | 108 |
| 5.2.1 | Ultrasound Transducer..... | 108 |

| | | |
|--|--|-----|
| 5.2.2 | Transmitter..... | 110 |
| 5.2.3 | Receiver..... | 114 |
| 5.2.4 | Connection to MICAZ..... | 117 |
| 5.3 | Software Configuration..... | 119 |
| 5.3.1 | Software Architecture for Remote Node..... | 120 |
| 5.3.2 | Software Architecture for Beacon..... | 125 |
| 5.3.3 | Software Architecture for Base Station and Java..... | 129 |
| 5.4 | Implementation of Distance Estimation and Trilateration..... | 132 |
| 5.4.1 | Distance Estimation..... | 133 |
| 5.4.2 | Trilateration..... | 148 |
| 5.5 | Chapter Summary..... | 153 |
| CHAPTER 6: COST-BENEFIT ANALYSIS..... | | 155 |
| 6.1 | Introduction..... | 155 |
| 6.2 | Cost and Benefit Analysis..... | 157 |
| 6.2.1 | Quantitative Approaches..... | 159 |
| 6.2.2 | Qualitative Approach..... | 171 |
| 6.3 | Possible Application Scenarios..... | 173 |
| 6.4 | Chapter Summary..... | 177 |
| CHAPTER 7: CONCLUSIONS AND FUTURE STUDIES..... | | 179 |
| 7.1 | Summary..... | 179 |
| 7.2 | Recommendations for Future Research..... | 181 |
| APPENDICES..... | | 183 |
| BIBLIOGRAPHY..... | | 206 |

List of Tables

| | Page |
|--|------|
| Table 2.1: Characteristics of different RSSI-based localization techniques..... | 33 |
| Table 3.1: Frequency allocations and physical layer in IEEE 802.15.4. | 44 |
| Table 4.1: Summary of parameters used for feasibility simulation..... | 82 |
| Table 4.2: Variable notation summary..... | 94 |
| Table 5.1: ADC conversion time in ATmega128..... | 148 |
| Table 6.1: Estimation of unit working hour in three process cycles..... | 163 |
| Table 6.2: Overview of a bridge in the State of Massachusetts requiring superstructure replacement..... | 164 |
| Table 6.3: Quantity Take-Off of major bulk materials for superstructure replacement . | 165 |
| Table 6.4: Cycle estimation for typical bulk materials..... | 166 |
| Table 6.5: Qualitative benefits by adopting a sensor-based tracking system | 173 |

List of Figures

| | Page |
|--|------|
| Figure 2.1: Operational structure for on-site schedule monitoring and control..... | 18 |
| Figure 2.2: Field test configuration using RFID tags and RFID readers..... | 20 |
| Figure 2.3: Read range from field test - Chiller station and generator project..... | 21 |
| Figure 2.4: Dynamic schedule of parts and packets unification model using RFID device in construction industry..... | 22 |
| Figure 2.5: GPS positioning field test in Wanchai, Hong Kong Island..... | 26 |
| Figure 2.6: Active Badge..... | 29 |
| Figure 2.7: Active Bat..... | 30 |
| Figure 2.8: A few different Cricket units..... | 31 |
| Figure 2.9: MotionStar DC magnetic tracker..... | 32 |
| Figure 3.1: Mote family of TinyOS for wireless sensor network..... | 38 |
| Figure 3.2: Diagram of MICAZ hardware structure..... | 39 |
| Figure 3.3: CC2420 block diagram..... | 41 |
| Figure 3.4: MIB510 interface..... | 42 |
| Figure 3.5: MTS300/310 sensor board..... | 43 |
| Figure 3.6: ZigBee stack architecture..... | 46 |
| Figure 3.7: ZigBee Network..... | 48 |
| Figure 3.8: Illustration of Blink application for wiring the component structure..... | 51 |
| Figure 3.9: An example of component structure defined in TinyOS..... | 52 |
| Figure 3.10: Illustration of discharge curve in Lithium discharge curve and Alkaline batteries..... | 56 |
| Figure 4.1: Sensor network block diagram..... | 62 |
| Figure 4.2: Coordinating scheme to trigger the query pulse for measuring the distance between beacon and remote node..... | 63 |
| Figure 4.3: Superframe structure of the RF signal packet..... | 65 |
| Figure 4.4: Description of time-of-flight utilizing the combination of RF and US signals | 66 |

| | |
|---|-----|
| Figure 4.5: The schematic of the distance estimation from the beacon based on (a) the round trip time of the RF signal and (b) combination of RF and ultrasound signals. | 68 |
| Figure 4.6: Operation delay in digital clock cycle when a signal is received..... | 70 |
| Figure 4.7: Illustration of trilateration method that consists of the location a mobile remote node from its distance to three fixed points..... | 74 |
| Figure 4.8: Discrete Kalman filter cycle: two phases of Predict and Correct..... | 79 |
| Figure 4.9: System configuration for feasibility simulation | 81 |
| Figure 4.10: Tracking trajectory at 8 MHz clock cycle (RF-only scheme) | 83 |
| Figure 4.11: Measured distance variation at 8 MHz (RF-only Scheme) | 83 |
| Figure 4.12: Tracking trajectory at 25 MHz clock cycle (RF-only scheme) | 84 |
| Figure 4.13: Measured distance variation at 25 MHz (RF-only Scheme) | 84 |
| Figure 4.14: Tracking trajectory at 50 MHz clock cycle (RF-only scheme) | 85 |
| Figure 4.15: Measured distance variation at 50 MHz (RF-only Scheme) | 85 |
| Figure 4.16: Tracking trajectory (RF+US scheme) | 86 |
| Figure 4.17: Observed distance measurement from three beacons..... | 87 |
| Figure 4.18: Percentile of position errors | 89 |
| Figure 4.19: System configuration of beacon scheduling..... | 92 |
| Figure 4.20: Beacon B_m scheduling of single communication | 96 |
| Figure 4.21: Structure of a RF message..... | 98 |
| Figure 4.22: Beacon B_m scheduling of one-to-many communication | 100 |
| Figure 4.23: Mobility problem: errors in estimating the actual moving path when sensor's moving speed is very faster compared to beacon's sampling rate..... | 103 |
| Figure 5.1: Illustration of timestamping scheme for TOF approach | 108 |
| Figure 5.2: Sound pressure level at transmitter | 109 |
| Figure 5.3: Sound pressure level at receiver | 109 |
| Figure 5.4: Characteristic of directivity in 40TR16F ultrasound transducer | 110 |
| Figure 5.5: Circuit diagram for transmitter..... | 112 |
| Figure 5.6: Hardware design of ultrasound transmitter connected to 51-pin connector in MICAZ | 113 |

| | |
|--|-----|
| Figure 5.7: Output signal at hardware timer (left) and ultrasound transducer (right) measured by oscilloscope | 114 |
| Figure 5.8: Circuit diagram of receiver..... | 115 |
| Figure 5.9: Response time of signal voltage, LM358..... | 116 |
| Figure 5.10: Hardware configuration of a beacon | 117 |
| Figure 5.11: 51-pin extension connector in MICAZ | 118 |
| Figure 5.12: Illustration of timestamping and distance conversion..... | 121 |
| Figure 5.13: Flowchart diagram for operating US_TxM.nc..... | 122 |
| Figure 5.14: Example code for RFDetectMsg event | 123 |
| Figure 5.15: Example code for US and RF transmission in a remote node..... | 124 |
| Figure 5.16: RF packet structure..... | 125 |
| Figure 5.17: Flowchart diagram for operating US_TriggerM.nc | 127 |
| Figure 5.18: Example code for ADC implementation..... | 128 |
| Figure 5.19: Example code for the RF relaying structure..... | 129 |
| Figure 5.20: Flowchart diagram for operating US_BaseM.nc..... | 131 |
| Figure 5.21: Data packet conversion and variable assignments | 132 |
| Figure 5.22: Device configuration for a MIB510 interface (left), a beacon (center), and a remote node (right) | 133 |
| Figure 5.23: Configuration of setup for distance estimation test..... | 134 |
| Figure 5.24: Process of signal detection scheme at receiver | 136 |
| Figure 5.25: Screenshot of received ultrasound signal measured at 1, 5, and 10 M distance | 137 |
| Figure 5.26: Noise level measured in ultrasound receiver (50mV/div and 20us/div) | 139 |
| Figure 5.27: Plot of received signal level and noise level (left) and signal-to-noise ratio (SNR) in decibel | 140 |
| Figure 5.28: Measurements of distance that is varying from 1 to 15 meters..... | 142 |
| Figure 5.29: Plot of percentile errors and standard deviation..... | 145 |
| Figure 5.30: ADC timing diagram in single conversion mode (first conversion) | 146 |
| Figure 5.31: ADC timing diagram in single conversion mode (normal conversion) | 147 |
| Figure 5.32: System configuration of trilateration and RF relaying scheme to determine the position of a remote node..... | 148 |

| | |
|---|-----|
| Figure 5.33: Hardware configuration of three beacons and a remote node | 149 |
| Figure 5.34: Java application for visual representation of beacons and a remote node.. | 150 |
| Figure 5.35: Measuring the position of a remote node that turns around a triangular path with 1.3 meter from each vertex | 151 |
| Figure 5.36: Distance errors in radius (unit: mm)..... | 152 |
| Figure 6.1: Three cyclic procedures associated with the material tracking and handling | 160 |
| Figure 6.2: Total implementation cost for field materials tracking compared with three alternatives | 168 |
| Figure 6.3: Implementation cost savings due to the deployment of embedded sensor- based tracking methods..... | 169 |
| Figure 6.4: Unit labor cost savings in percent due to the deployment of sensor-based tracking methods..... | 170 |
| Figure 6.5: Information flow diagram for possible collaboration with project management systems..... | 177 |

CHAPTER 1: INTRODUCTION

1.1 Overview

A report published by the Business Roundtable in 1982 defines materials management as the management systems that leverage the efficient utilization of materials and equipment with all necessary efforts to ensure that the right quality and quantity of materials and equipment are appropriately controlled in a timely manner with reasonable cost and availability [Business Roundtable, 1982]. This report claimed that materials management is a distinct system that can contribute to increase the cost effectiveness of construction project. During several decades, many research efforts have been demonstrated to provide the effective materials management strategies for improvement in labor productivity and performance. Even though there was a growing awareness in the construction industry that materials management needs to involve a comprehensive and integrated coordination of management activity, previous research efforts could not afford clear establishment to justify the impact of materials management practice on productivity. This might be attributed to the fragmented functions of materials management with minimal communication and unclear responsibilities among the owner, engineer or contractor [Bell and Stukhart, 1986].

Poor materials management on construction projects is not a new problem. Perhaps the absolute criteria for relative comparison between good and poor have changed over time due to increased complexities of construction projects and continuous development of technologies, which has made the measurement of the impact of sound materials management practices difficult. However, it is generally accepted that materials management practice has gradually improved in the past several decades, and many

research activities have been focused on the improvement of materials management in increasing labor productivity and construction performance.

The main functions of materials management in construction sites include receiving, warehousing, tracking, locating, finding, and distributing the right materials to the right locations in a timely manner. Hundreds of thousands of construction materials including equipment, tools and crews are involved in the process of design, fabrication, delivery, storage, installation and inspection. These functions should be efficiently carried out in order to prevent materials shortage, surplus, cash flow problems and labor delay. For instance, if materials are delivered early, capital may be tied up and interest charges may be incurred on the excess inventory. Even worse case, materials may deteriorate or be stolen during storage unless special care is taken. In contrast, if materials required for particular activities are not available at the right time, delays or extra expenses may be incurred. The materials management system must therefore identify potential materials shortages or surpluses as early as possible to avoid lost time and resources.

One of the keynotes presented in the workshop sponsored by National Institute of Standards and Technology (NIST) in May 2003 [Saidi et al., 2003], stated that “material tracking remains a very big problem on the current construction job site and how can it be addressed today.” This conclusion implies that the current practice of tracking materials in construction is still reliant on manual legacy systems, which often causes errors and delays jeopardizing entire projects. Thus, inefficiencies, reported to be 4-6% of labor cost related to the manual operation of reporting, recording and transferring field data in current tracking systems [Bell and Stukhart, 1987], are still an important issue especially

on large construction projects. The performance of materials management could be further improved if information about materials is to be gathered more easily, and if more efficient tracking and control mechanism for construction materials is to be exerted with the help of emerging technologies.

Advancements in low power microelectronic devices and sensor network technologies provide the capability to automate and integrate individual tasks for tracking and monitoring in the construction industry. Based on this motivation, the proposed research will examine a hypothesis that the integrated and proactive materials management systems will have the potential to improve the productivity and efficiency of project performance. The research approach that will be used intends to bridge the gap between manual and automated construction materials management, and the proposed research will verify such a hypothesis.

1.2 Motivation

A 1982 Business Roundtable Construction Industry Cost Effectiveness (CICE) report entitled “Modern Management System” examined the importance of materials management and control. This report claimed that “senior management of firms in the construction industry has not always recognized the significant contribution that materials management can make to the cost effectiveness of projects.” A major reason why firms neglect the importance of effective materials management is because materials management systems are often stovepiped resulting in minimal communication and a lack of clearly established responsibilities among project participants. Therefore, many of the materials management decisions tend to be ad hoc and intuitive, not based on the

data [Formoso and Revelo, 1999]. Furthermore, manual handling and controlling of materials causes errors in the particular circumstances, e.g. the performance of individuals and the lack of systemic understanding of the organization.

In 1994, the Construction Industry Institute (CII) established the Materials Management Task Force to examine the critical issues outlined in the 1982 CICE report. The task force attempted to develop information and concepts that prove the value and cost effectiveness of integrated and proactive materials management systems. The key findings from the Materials Management Task Force include the following:

- When properly planned and executed, clearly defined materials management provides an invaluable tool to improve labor productivity
- More than 6% of all construction labor costs could be saved if materials and equipment had been available at the work site when needed
- Few published case studies have measured the impact to specific instances of good and poor materials management practices

In the meantime, their findings triggered several research efforts to examine the cost benefit of the materials management system. Bell and Stukhart examined benefits related to craft labor productivity with twenty different case study projects [Bell and Stukhart, 1987]. This research found that good materials management would reduce overall craft labor costs by 6% and increase direct labor savings by 8%. The research attributed the increased efficiencies to materials management practices providing better visibility of material availability, which allows craftsmen to schedule work accordingly. They also pointed out that, on projects lacking a materials management system, craft

foremen spent 20% of their time hunting materials and another 10% tracking purchase orders and expediting. They also found that the ineffective or fragmented materials management system could incur bulk materials surplus as high as 10%. Finally, this research concluded that integrated computer-aided systems that track bulk materials would produce an additional 4% to 6% savings in craft labor costs.

Another case study on the impact of materials management was reported. The following are conditions or events that degrade effective materials management [Thomas et al., 1989]:

- Organization of storage areas: extensive multiple-handling of materials improperly sorted or marked
- Housekeeping: trash and debris that obstruct the materials movement
- Material availability: crew slowdowns in anticipation of material shortage or rework when materials arrive
- Material handling and distribution: material losses or inefficient method to distribute materials

Due to such adverse conditions resulting from ineffective materials management practices, construction projects resulted in 18% work-hour overruns and 19% time overruns. For the affected working days, an average of 58% of the work-hours was ineffectively used. In addition, the report by Thomas et al. concluded that benefit/cost ratio amounted to 5.7 by comparing material-related investment and incurring disruptions. This implies that good materials management generates potential benefit from the materials-related investment.

In regard to the improved materials management practice, prior research examined the attributes of successful materials management systems [Bell and Stukhart, 1986]. Construction materials comprise hundreds of thousands of bulk materials, standard off-the-shelf materials, fabricated members or unit go through the process of fabrication, delivery, storage, distribution and installation. With limited storage or laydown space on a construction site, a materials management system must identify materials receipt, proper location for storage and a timely sequence of distribution to prevent material shortage/surplus and labor crew delay. Hence, productivity gain can be expected with reduction in the labor hours accumulated by storekeepers and foremen related to searching for materials and following up purchase orders. Such a system should also allow the field materials manager to promptly identify the status of materials.

By examining those attributes observed in materials management practice, Bell and Stukhart also asserted that timely information regarding actual materials status and handling of shortage/surplus should be provided to personnel related to the project [Bell and Stukhart, 1986]. For the timely communication with participants involved in the project, they pointed out that primary benefit of using computer-based materials management system is its ability to exert maximum control over material acquisition and distribution for the improvement of efficiency of crew-level work process and materials tracking practice. As a consequence, they concluded that the ability of a computer-based system and the lack of control apparent in its absence can qualitatively justify the needs and required cost of developing the system.

Another recent survey [Fletcher, 2001] indicated that the site managers wasted a lot of time manually entering data and creating reports. In fact, field supervisory

personnel spend 30-50% of work time on recording and analyzing field data [McCullough, 1997] and 2% of construction field work is categorized as manual tracking and controlling of materials handling [Cheok et al., 2000]. In those manual processes, the accuracy and process time often rely on the judgments and writing skills of the personnel collecting the data. Furthermore, the time-consuming process of manually inserting and typing the field data to the computer terminals often obstruct the anticipation of prompt decision or job activities.

Currently available computer and sensor network technologies provide potential for data acquisition and communication. It has become viable on construction sites to collect the field information about tracking of construction materials, equipment, tools, and labors with the advanced location and identification sensing technologies. In relation to the automated data collection technology (ADC), Navon and Berkovich emphasized the importance of monitoring the flow of materials in quantities and inventory levels because the problem in materials management system is the lack of up-to-date relevant information [Navon and Berkovich, 2005]. They also pointed out that a computer integrated materials management system can improve data collection, system organization, analysis and presentation to support real-time decision making process.

Based on prior research efforts in materials management presented above, key features and benefits from automated tracking and monitoring of materials, equipment, tools and labor can be summarized as follows:

- Provides real-time method of identifying, registering, collecting and communicating information related to the status of materials, equipment, tools and labors

- Reduces time searching for construction components, with up-to-date information related to the materials availability and the status of prerequisite work
- Information about materials flow can be shaped in a greater detail based on the actual flagging the instances of delivery, receipt, distribution and installation
- Reduces labor-intensive manual process relating to recording, reporting and transforming the materials tracking data
- Provides feasible methodology to control and handle the materials shortage and surplus by interacting with the database of planning, scheduling and procurement

To implement the automated tracking and monitoring of construction materials, equipment, tools and personnel, several applications of RFID and GPS have been proposed to improve the efficiency and productivity in construction site. It is generally accepted that recent technologies, such as RFID or GPS, can provide the feasibility for tracking of construction components due to their ability of automated identification and positioning of mobile objects. Thus, effective materials management system for improved labor productivity and work process time can benefit from the emerging technologies over the manual process and controlling.

While RFIDs provide an advanced materials tracking method when compared with earlier technologies, e.g. bar codes, several limitations have been observed when RFIDs are applied to construction materials tracking. The basic functionality of RFID is to present remote identification and tracking of distributed RFID tags. Because RFID was originally designed to replace the bar code technology, main functions often include the remote identification of the construction materials, where broader applications to

wireless monitoring and localization are quite limited. A recent survey conducted by RFID Journal showed that the cost of RFID tags vary from 20 cents to 6 dollars based on the tag's specification. However, most of the RFID readers cost from \$2,500 to \$3,000 depending on the various features in the device [ABI Research, 2005]. Thus, it is prohibitively expensive to cover large-scale construction sites because the communication coverage range for active RFID tags is within 15 m [Goodrum et al., 2006], which is not enough for practical use.

Even though the global positioning system (GPS) can provide somewhat improved accuracy for locating the tags' position by combining them with RFID, GPS receivers are still expensive to track and monitor large amount of construction materials in a typical construction project. In addition to high cost, localization systems based on GPS alone also suffer from the multipath and signal masking in highly dense areas. Due to these limitations, significantly large positioning errors of greater than 20m over 40% of measuring points and greater than 100m error over 9% of measuring points, have been found with the use of stand-alone global positioning techniques applied in construction-vehicle tracking systems [Lu et al., 2004]. Thus, it is unreliable to depend on the accurate localization data for material tracking and monitoring in a highly dense environment such as a construction site.

1.3 Research Objectives

Aforementioned challenges with RFID and GPS technologies lead to the following two research questions related to current practices of materials management system:

- How to overcome the limitations observed in current utilization of technologies, such as RFID and GPS, and how to improve the location accuracy and data communication efficiency with cost- and performance-effectiveness?
- How to improve the field materials management practices for effective tracking and monitoring of up-to-date information about materials identification, procurement status and delivery?

In an attempt to answer these research questions, the dissertation introduces a new prototype framework of an automated tracking and monitoring system that will address the needed shift from the time-and labor-intensive manual systems to sensor- and network-based communication systems for field materials management. The academic goal of this dissertation is to introduce and describe an improved localization technique based on a combination of radio frequency and ultrasound signal for tracking and monitoring construction components. This will be highlighted by the new investigation that utilizes the emerging technologies of wireless sensor networks supported by Micro ElectroMechanical Systems (MEMS) and IEEE communication standards. The practical goal of this research is to find cost-effective and fully automated solutions that satisfy the practical requirements of a reliable and stable system for tracking and monitoring of construction components. To implement this vision, the following constituent objectives have been identified:

- Provide improved management strategy of construction materials based on the emerging technologies of wireless sensor network that will lead to increased cost effectiveness and labor efficiency

- Provide alternative solutions with more reliable and robust, but cheaper and energy-efficient approach in real-time tracking and monitoring method compared with previously proposed techniques, such as RFID and GPS.
- Introduce an improved approach of localization methodology based on a combination of RF and ultrasound signals that will provide increased accuracy over previous proximity approach or received signal strength index (RSSI)-based techniques
- Suggest architectural framework on how to collect and acquire the field data in real time, and present the possible collaboration strategy to communicate with diverse platforms of information management systems in construction projects
- Integrate sensing components and networking framework for the development of an automated tracking and monitoring system, which will create an opportunity and possibility to implement an ubiquitous computing environment in construction sites

1.4 Contributions of the Proposed Research

The proposed research attempts to create a framework for integrating the latest innovations of ubiquitous computing to automate the tracking and monitoring of equipment, materials, and people in construction sites. Thus, we envision that such construction components can be integrated into recognizable entities that can be sensed and controlled by a distributed sensor network. This research constitutes one of the few studies to incorporate emerging information and sensor network technologies into the construction industry, which has been slow to migrate away from legacy processes.

Expected contribution to the body of knowledge in current construction research activities can be described as follow:

- The proposed research expects to present the opportunity for the improvement in automated tracking and monitoring process of construction components as well as for the suggestions of advanced information systems in terms of information acquisition and processing.
- New methodology of localization and tracking in construction processes can be integrated with functional features of information collection modules in project management systems, such as web-based PMS, 4D simulation & visualization, project scheduling software, or ERP systems.
- The understanding and knowledge of functionalities underlying the potentials of wireless sensor networks will suggest a new possibility to upgrade traditional prototypes of field data acquisition in construction industry. In addition, the proposed sensor network environments for ubiquitous computing will promise the reorganized standards and coordination toward the new paradigm of construction management.
- From a project delivery perspective, the framework for consistent management of work processes and project delivery can be implemented by an “everywhere” sensor network, enabling track and monitor project tasks and work processes based on standard information systems.
- This research may overcome the practical obstacles caused by misunderstandings and inadequate use of emerging technologies in civil infrastructure, and can

facilitate the understanding of implications that new technology has on the effective delivery of construction project.

- Localization methodology in outdoor environments can also provide an insight for the implementation of industrial practices using RF-based wireless networks, where the industrial practices can benefit from the flexible and expandable networking strategies for large scale construction job site.

1.5 Organization of Dissertation

The remainder of this dissertation presents the design, implementation and evaluation of the automated tracking and monitoring systems of construction materials. Chapter 2 presents the general background and related work on management practices and current technologies used in tracking and monitoring system on construction sites, and describes localization systems deployed in both civil engineering and electrical engineering industries. Chapter 3 describes the state-of-the art technologies of wireless sensors and networks which are deployed in construction engineering applications. Chapter 4 illustrates the feasibility study based on the numerical simulation, and shows the adverse effects of multipath and signal detection delay in RF-based localization scheme by simulating trilateration method with Kalnam filter to introduce a new, innovative approach for component tracking framework in construction. Chapter 5 introduces a new design of hardware and software for the new localization technique based on the embedded system of combining ultrasound device and ZigBee™¹ [ZigBee Alliance, 2005] product, and implements the experimental analysis for distance and

¹ High level of communication protocol based on the IEEE 802.15.4 specifications for wireless personal area networks (WPANs)

position estimation to provide the accurate and cost-efficient localization scheme for mobile construction components. Chapter 6 conducts cost-benefit analysis of three alternatives including manual, RFID and ZigBee-based tracking method, and the benefits are compared with tracking system implemented by these three alternatives. Finally, chapter 7 summarizes completed research work and presents possible directions of future research for improving the performance of material tracking in construction site applications.

CHAPTER 2: LITERATURE REVIEW

During past years, advanced computing and information technologies have drawn tremendous attention to construction industry that presents more opportunities for automation and improvement in process performance than other industries. One of the advents in those technologies deployed to construction processes can be found in the wireless sensors and networks that present the advanced methodologies in data acquisition because the real-time information systems, capable of gathering the field data wirelessly into management solution, became critical task in future project management. Many efforts in industrial practices and researches have been put to present a new framework for the applications of those technologies. This chapter examines previous works on tracking and monitoring the construction material, equipment and labor utilizing advanced technologies.

2.1 Components Tracking Systems in Construction Industry

Tracking and monitoring of information flow on a construction job site is an important, yet commonly overlooked, aspect because timely information about the status of materials, equipment, tools and labors are directly related to the successful completion of project. Realizing the potential for cost savings and up-to-date information acquisition, the Construction Industry Institute (CII) funded a formal research project in 1987 to investigate the possible application of bar code in construction industry [Bell and McCullough, 1988]. Bar code is defined as self-contained information encoded in a printed pattern, and construction application has been found in jobsite resource and activity identification. In the CII research effort in 1987, bar code technology can be

applied to materials, equipment, containers to track and control those items for improving the speed and accuracy of data entry into a computer system.

For practical application, bar code technology can provide 1) automatically tracking field data of construction materials; 2) automatically recording historical data about construction materials in the project; 3) automatically monitoring materials flow; and 4) automatically transferring field data construction materials via wireless connection [Chen et al., 2002]. To obtain timely information from the bar code, it is necessary to identify the several objects that should be interactive and stored in the bar code database system: 1) activities; 2) materials and equipment; 3) labor; 4) cost items; and 5) material suppliers [Echeverry and Beltran, 1997]. Even though main application area varies from the object categories made for specific purpose, general applications of bar code technology in construction can include quantity takeoff, field material control, warehouse inventory and maintenance, tools and consumable material issue, and document control and office operations [Bell and McCullough, 1988].

Rasdorf and Herbert propose the generic system architecture for construction information management system (CIMS) using bar code technology. The specific objectives examined in this research include:

- To explore the feasibility of bar code technology to implement tracking construction materials and equipment
- To develop an integrated construction labeling, information storage, and retrieval database system
- To determine the potentials and limitations of bar code technology in computer-integrated construction

They also pointed out that the bar code can increase the efficiency in jobsite data acquisition, resource management and tracking of labor and equipment [Rasdorf and Herbert, 1990]. Thus, CIMS with bar code technology provides the project manager with a construction information storage and retrieval database to facilitate the task of jobsite information management.

There exist many mature technologies that provide management with various alternatives for accessing data quickly and reliably. Bernold [Bernold, 1990] asserted that bar code technology is one alternative receiving more attention from the construction industry. By providing some experiences in other industries, he pointed out that bar code technology can increase the labor productivity by decreased time spent for clerical work and by less waiting time for jobs and resources, such as materials, tools and equipment. In addition to the productivity, bar code provides high survivability in construction environment due to wide variety of bar code labels, adhesives and input devices. However, careful investigation on the selection of adhesives is needed when the bar code is applied to harsh environmental condition or labeled to construction equipment or tools.

Cheng and Chen [Cheng and Chen, 2002] proposed an automated schedule monitoring system using bar code based RF technology and Geographic Information System (GIS) in major critical activity. By developing ArcSched system, they demonstrated that the efficiency of lifting operations for precast building construction has been improved in monitoring and controlling the erection progress. Because RF based technology provides a good performance in areas where environmental constraints prohibit the use of other automatic identification technologies, it is very suitable to apply

bar code with RF transmitter for gathering and transmitting the job site data in construction environment. In addition, the control center and bar code collector were used for bar code read/write, data acquisition and scheduling control. Furthermore V8 video monitoring was installed on top of the tower crane to monitor the erection operation in progress, shown in figure 2.1.

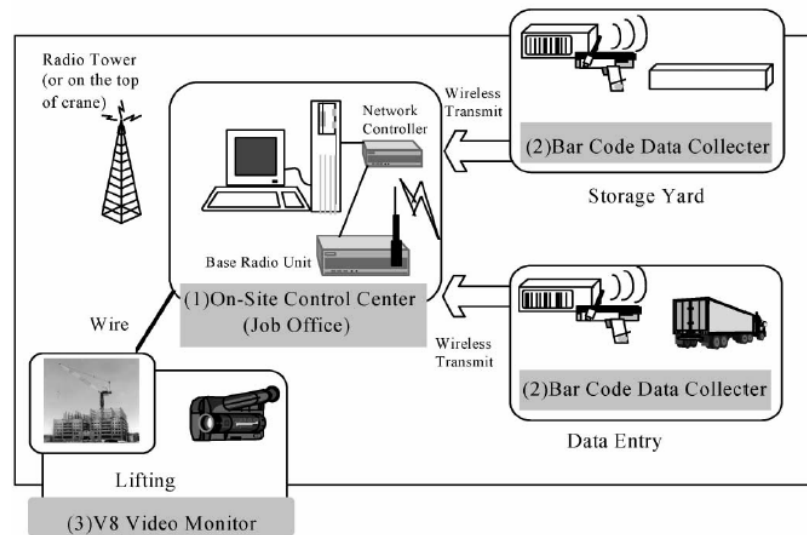


Figure 2.1: Operational structure for on-site schedule monitoring and control [Cheng and Chen, 2002]

Although bar code technology has the advantages in identifying and tracking the construction materials, tools, equipment and labor, it presents several disadvantages. First of all, bar code requires physical contact to read the information, which also requires the line-of-sight or clean environment from noise, contaminants, glare and dirt. Second drawback of bar code includes that it requires manual entry of all data related to materials and equipment coming in and out of the storage, so its utilization for automated data acquisition (ADC) can be time-consuming and fallacious [Li et al., 2005]. Finally, bar code has limited capacity of data storage and data communication, which is considered as

inappropriate selection for fully automated framework of data acquisition in construction materials tracking.

With advancement of radio frequency, RFID has received attention for possible tool for improved materials management such as tracking the construction materials, equipment, tools and labors. During several years, significant beneficial applications of RFID technology had been already found in retailing, manufacturing, transport and logistics industries, and security and access control [Song et al., 2006] due to its robust functionalities with inexpensive and non-labor intensive means of identifying and tracking products. In addition to its relatively large storage capacity compared to bar codes, it is very reliable in harsh environments such as construction industries. More recently, many of research activities related to the application of RFID could be found in construction area to develop potential applications that enhance construction and facility operations.

Jaselskis and El-Misalami (2003) demonstrated a pilot test using RFID in construction process. In this research, they illustrated a list of potential applications that include engineering design, material management, maintenance, and field operations. In order for examining the applicability to the material procurement process on a construction site, two pilot tests were conducted in power plant project in Mississippi and the construction of a catalytic cracking unit in a Texas refinery. The pilot tests showed that RFID tags reduced the time required to download data into a company's material tracking system, and RFID can be a beneficial technology to the receiving process of construction material.

In response to the need to track identified materials through the supply chain, Song et al. conducted field tests of current RFID technology to examine its technical feasibility for automatically identifying and tracking individual pipe spools in laydown yards and under shipping portals (figure 2.2) [Song et al., 2006]. The field test indicated that RFID technology could function effectively even in the construction field environment of large metal objects, and the authors expected the potential benefits in automated pipe spool tracking, which include 1) reduced time in identifying and locating pipe spools and more accurate information gathered in timely manner; and 2) reduced searching effort of misplaced pipes and increased reliability of pipe fitting schedule.

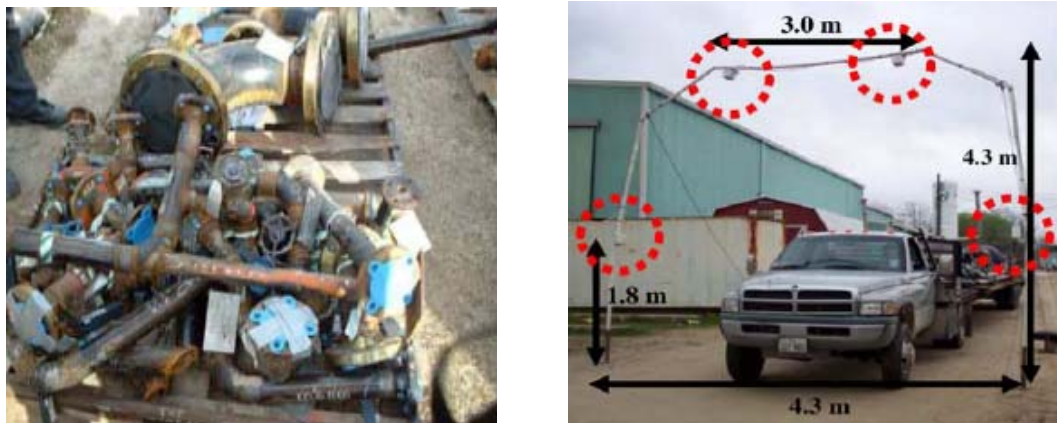


Figure 2.2: Field test configuration using RFID tags and RFID readers [Song et al., 2006]

Goodrum et al. conducted the experimental test for tool tracking system using active RFID tags with 32 Kb memory, 3.6 V battery, and antennae operating at 915 MHz [Goodrum et al., 2006]. Utilizing PDA, a prototype tool tracking system was developed to track tools in mobile environment and to inventory hand tools located in either mobile gang boxes or truck boxes. The field tests concluded that the capacity of the active RFID

tags showed quite adequate read range and durability required for a tool tracking and inventory systems (figure 2.3). Average reading ranged from 3 m to 9 m, and all tags survived daily use in the construction environment over a three month testing period. It is noted that the measurement of received signal strength index (RSSI) varied from the temperature where the active RFID tags were exposed. Measurements at low temperature experienced smaller read ranges compared to high temperature. Although the lower temperatures affected the read range of RFID tags, the paper concluded that their ability to store data was not affected.

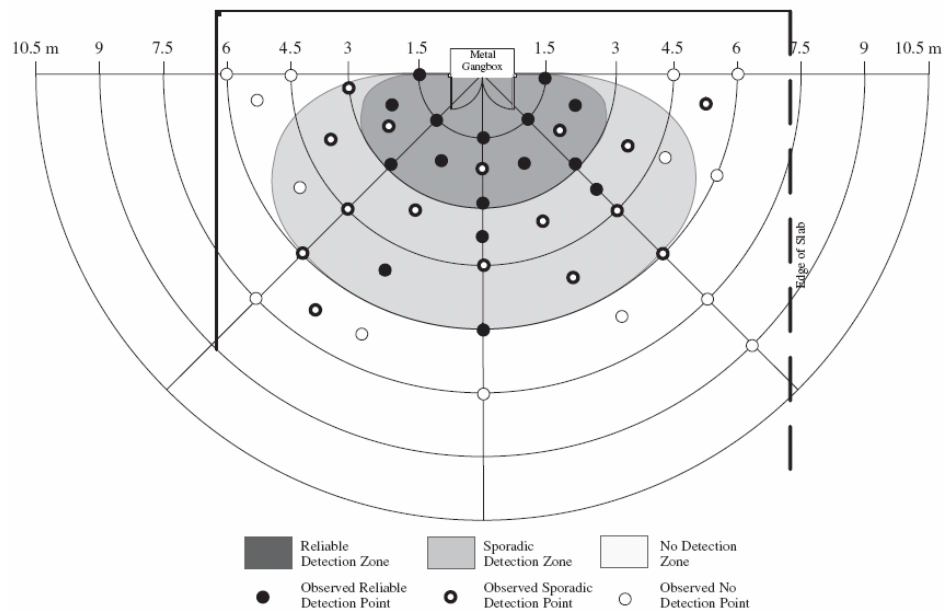


Figure 2.3: Read range from field test - Chiller station and generator project [Goodrum et al., 2006]

Yagi et al. and Umetani et al. presented RFID applications in parts and packets unification for construction automation, where RFID devices were attached to construction components to investigate the relationship between the components and their information [Yagi et al., 2005 and Umetani et al., 2006]. Construction process may be

characterized to be a project-type production, defined as course of concerted action intended or considered possible to achieve some chosen production purpose for a given period of time under some given constraints on available resource (manpower, material, money) and construction environment. It is therefore of critical importance to allocate production activities properly according to the well-planned process model and scheduling. The proposed research suggested a unification model for construction process control which deals with bulk of scheduling elements produced by multiple heterogeneous players, contractors, subcontractors, and parts makers (figure 2.4). Furthermore, by proposing Glue logic or active database as a possible control mechanism, this research describes the unification methodology of parts and packets which can be regarded as virtual robotics for the integrity of components and their information.

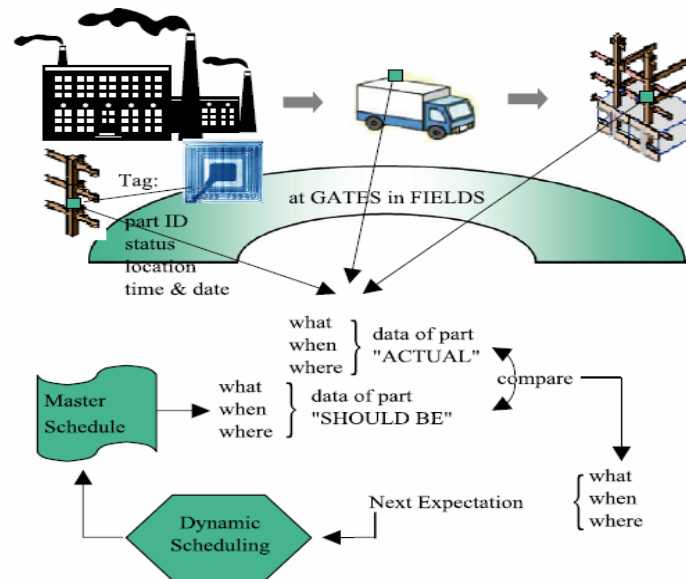


Figure 2.4: Dynamic schedule of parts and packets unification model using RFID device in construction industry [Yagi et al., 2005]

Even though the RFID technologies provide the potential benefit to construction industry for tracking the construction components, several limitations related to practical applications in construction industries can be identified:

Cost: the cost of RFID tags and readers are a major factor currently preventing their wide use in construction. According to the findings of RFID Journal [www.rfidjournal.com], a 96-bit EPC tag costs from 20 to 40 cents. Low-frequency transponders (tags) in glass capsules are about \$3.5 each, and high-frequency transponders range from about \$2.50 to \$6.00. Aside from the cost of the RFID tags, most UHF readers cost from \$2,500 to \$3,000, depending on the features in the device [ABI Research, 2005]. Thus, practical application of RFID technology is still challenging in tracking the heterogeneous type and size of construction components.

Standardization: currently standardization among RFID manufacturers' readers, tag frequencies, and recording software could not attract much confidence from the RFID consumers. Due to competitive advantage for more gaining in RFID market, technologists are not willing to provide their information and avoiding standardization (Roberts 2002). In order for the consensus in RFID standardization, the following issues must be identified (Schneider 2003): global RFID frequencies, interoperability, use of active or passive tags, data formats (syntax, data structures, and encoding), methods of identification and presentation, and communications between tags and objects and tags and readers.

Interference: signal interference in reading RFID tags is also an identifiable problem for practical use. Metal interference is a big concern in RFID communication since the metal is the most familiar material on construction worksite. Although the

RFID does not require a line-of-sight between tags and reader, interference caused by metal would result in increased reading errors and shortened coverage range due to unreliable signal propagation throughout the metal. Frequency interference may also cause problems if bulky size of construction components needs to be identified with a particular tag surrounded by a number of other tags operating at the same frequency band (Schneider 2003).

Global Positioning System (GPS) is an alternative technology that could impact the automated construction process. GPS is a radio signal based navigation system with 24 GPS satellites orbiting the earth. By measuring the amount of time that the radio signals travel from a satellite to a receiver, GPS receiver calculate the distance and determine the locations accurately in terms of longitude, latitude and altitude [Oloufa et al., 2003]. As a vital global utility, GPS has become indispensable for modern navigation on land, sea, and air around the world, as well as many diverse fields of civilian activity. However, so far potential application of mainstream civil engineering has been limited to only surveying and leveling in construction site, structural deformation monitoring, equipment tracking, and transportation systems.

Peyret et al. investigated experimental analysis using GPS for the elevation control of the profiling equipment's tool [Peyret et al., 2000]. The study indicated that GPS could provide the significant advantages; easy installation and use; highly automated positioning which can guarantee the reliability; and simultaneous provision of three spatial coordinates with perfect coherence between them. However, the raw satellite measurements are not accurate to be used in the control systems due to bias or

drift error caused by multipath effects. As a possible solutions, this research proposed that repeatability or double-base survey in real-time can be used to improve the raw accuracy of the measurements for elevation control of road construction equipment.

In order for the use in construction application, Jiang and Li [Jiang and Li, 2002] presented the allocations of GPS in traffic data collection, and conducted the analysis at freeway work zones to obtain the traffic characteristics using vehicle speed profiles and queue lengths. The GPS device was attached on a test vehicle to record the test vehicles' positions and speeds at specified time intervals, and the exact trajectory of the test vehicles was identified through satellite signals and recorded by GPS device. The research demonstrated that utilization of GPS is an efficient tool to measure the accurate and detailed information on traffic characteristics in work zone. Error of the GPS device is within 3.28 ft for distance measurements and is less than 0.10 mph for speed measurements. In addition, the GPS data showed the precise measurement of time, position and speed of the test vehicle's movement, and this method can identify the exact queue length in front of the work zone under traffic-congestion condition.

Garcia et al. evaluated the use of the GPS for collecting traffic data and for analyzing the traveling conditions to improve the safety of construction work zones on interstates in the state of Indiana [Garcia et al., 2006]. The presence of construction work zones on interstate affects both traveling public and the workers that perform the construction, which often causes the traffic congestions and accidents that involve road-users and construction crews. This research concluded that GPS provides engineer with a preliminary tool for improvement of safety and construction efficiency by collecting data

from different construction projects that represents the different conditions of construction work zones.

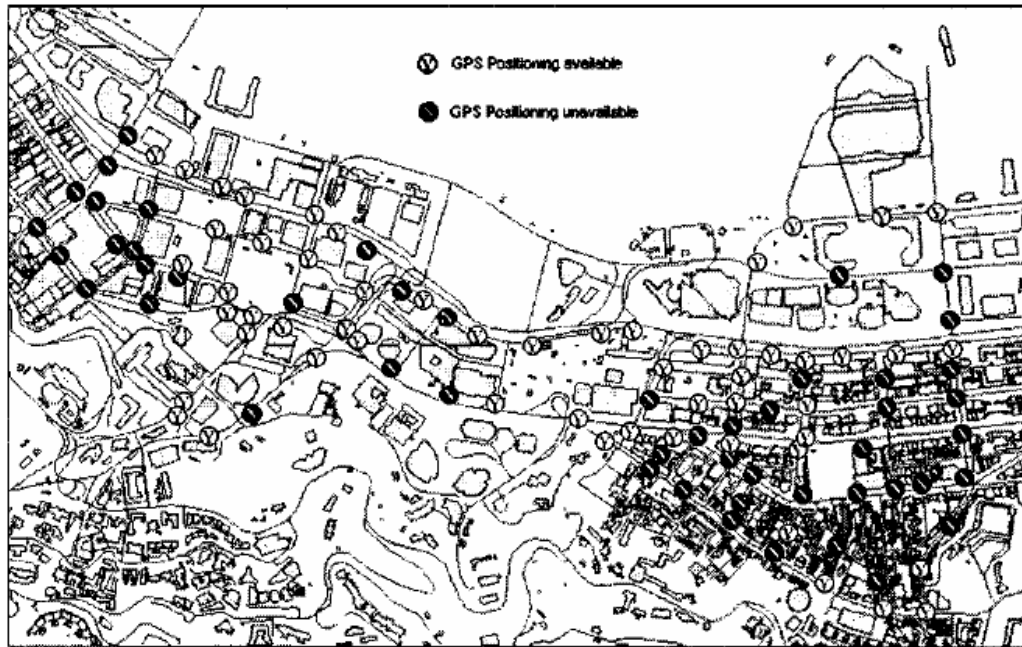


Figure 2.5: GPS positioning field test in Wanchai, Hong Kong Island (open circle - available GPS signal; and closed circle - unavailable GPS signal) [Lu et al., 2004]

Lu et al. showed that existing GPS technologies presents the key to overcoming the data obstacle and providing the sophisticated models work in a building environment [Lu et al., 2004]. Thus, they represent significant step toward the revolutionary applications to current construction management practices. However, localization system based on GPS alone also suffers from the blockage of satellite signal, multipath and signal masking in highly dense areas. Due to these limitations, the performance of GPS in these areas can be severely degraded and unreliable. In order to evaluate GPS performance in dense urban area, test zone of 1×2 is selected, and 106 control points

were established in that area as is shown in figure 2.5. According to the field experiment, around 50% of test area is not receptive of GPS signal due to signal blockage by neighboring buildings and temporary structures on the site. Due to the limitation, significantly large positioning error, over 40 % of points greater than 20 m and 9 % of points greater than 100 m, has been found in the stand-alone global positioning system for construction-vehicle tracking system, so it is insufficient and unreliable to obtain accurate localization data for material tracking and monitoring system in highly dense construction site.

2.2 Localization Techniques

Location tracking technique has fostered a growing interest in mobile systems implementing ubiquitous computing environments. Localization is a primary task for higher level of sensor network functions such as tracking, monitoring, and geometric-based routing [Elnahrawy et al., 2004]. In construction environment, the system design and development for estimating location and spatial information is challenging task because of management complexity, large scale system and harsh nature of wireless communication. Furthermore, there exist a variety of localization techniques that determine the location of objects or people at different environmental circumstances, e.g. building office, industrial job sites, metropolitan area, or vast outdoor area. Thus, it is important to examine the different characteristics of each technique for appropriate deployment to the environment of construction sites.

Global Positioning Systems are perhaps the most widely used location-sensing systems in outdoor environment. With an excellent lateration framework for estimating

geographical positions, GPS provides a reliable capability to allow receiver to calculate their position to within 15 meters on average [Garmin Ltd., 2006]. However, GPS will not function well in certain environments such as indoor or building-populated areas because receivers in such environment usually cannot detect the satellite signal. A possible solution that maintains GPS interaction in the area where signal might be obstructed is to use a GPS repeater mounted at some open site to rebroadcast the signal inside [Hightower and Borriello, 2001].

Olivetti Research Laboratory, now AT&T Cambridge, has developed the Active Badge indoor location system that consists of a cellular proximity system using diffuse infrared technology [Want et al., 1992]. Small infrared badge attached to mobile objects emits a globally unique identifier, and this data is sent to the application programming interfaces, shown in figure 2.6. A fixed infrared sensor provides absolute location information using symbolic or other infrared constraining volume, and the software architecture designed by Cambridge group controls this type of symbolic location data to represent the physical position. Active Badges, however, have a limitation on location with the environment of fluorescent lighting or direct sunlight due to the emissions of spurious infrared generated from these light sources [Hightower and Borriello, 2001]. In addition, effective coverage range of Active Badges is several meters, which limits communication area to small- or medium- sized room.



Figure 2.6: Active Badge [AT&T Laboratories Cambridge]

For more accurate physical positioning over Active Badges, Active Bat location system has been developed by AT&T researchers [Harter et al., 1999]. Active Bat uses an ultrasound time-of-flight lateration technique, in which a Bat emits an ultrasound pulse to a grid of ceiling-mounted receivers to measure the time interval from reset to ultrasound pulse arrival, shown in figure 2.7. Then the local controller forwards the distance measurements to a central controller, which performs the lateration computation. The Active Bat system shows within 9 cm resolution of Bats' position for 95 percent of measurements, and it can also compute orientation information based on predefined placement of Bats. However, there exist some disadvantages that include difficulty of deployment, cost, and limited scalability because the system requires large scale of fixed-sensor infrastructure and precise placement of sensors.



Figure 2.7: Active Bat [AT&T Laboratories Cambridge]

The Cricket Location-Support System, complementing the Active Bat system, has been developed by MIT Computer Science and Artificial Intelligence Laboratory, shown in figure 2.8 [Priyantha et al., 2000]. This research presented the design, implementation, and evaluation of Cricket system for in-building and location-dependent applications. Cricket uses beacon architecture to disseminate information about geographic space to listeners, and a beacon is placed at an unobtrusive location like a ceiling or wall. Localization technique used in Cricket system utilizes the radio frequency signal not only for synchronization of the time measurement, but also to delineate the time region during which the receiver should consider the ultrasound it receives. Like the Active Bat system, cricket uses ultrasonic time-of-flight data and a radio frequency control signal, but a grid of fixed ceiling sensor is not required because the receivers perform the timing and computation function. The advantages of Cricket system include that it does not rely on any centralized control system or explicit coordination mechanism between beacons; it provides information to devices regardless of their type of network connectivity; and each

Cricket device is made from off-the-shelf components and costs less than \$10. However, timing and processing both the ultrasound pulses and RF data may increase computation and power burden on the mobile receivers, and it is not easy to monitor the performance of receivers' performance due to decentralized management [Hightower and Borriello, 2001].

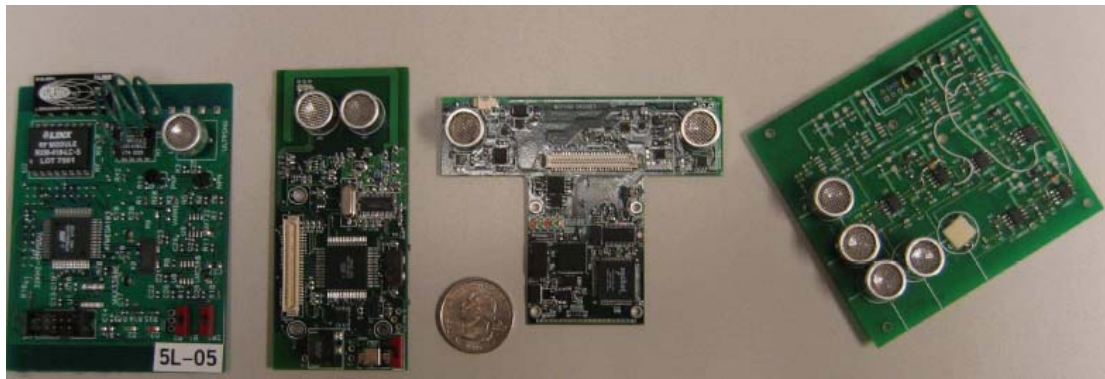


Figure 2.8: A few different Cricket units. From left to right: v1, v2, v2 done jointly with Crossbow, and a compass daughter board. The v1 and v2 units can function as either beacons or listeners [Balakrishnan et al., 2003]

Many research efforts on supporting virtual reality and motion capture for computer animation has offered the development of electromagnetic sensing technique for tracking system. MotionStar DC Magnetic Tracker, developed by Ascension Technology Corporation, tracks the position and orientation of up to 108 sensor points on an object or scene. By generating axial DC magnetic-field pulses from transmitting antenna in a fixed location, the system computes the position of the receiving antennas measuring the response in three orthogonal axes to the transmitted field pulse, shown in figure 2.9. The system provides quite precise physical positions relative to magnetic transmitting antenna, and many companies sell optical, infrared, and mechanical motion-

capture system. However, MotionStar Magnetic Tracker system is mainly designed to capture position for virtual environment, so it is not appropriate to the scalable and large location-awareness application.



Figure 2.9: MotionStar DC magnetic tracker [Ascension Technology Corporation, 2006]

The remaining majority of localization techniques are based on the received radio strength indicator (RSSI) due to its widely availability in wireless radio signal communication [Lymberopoulos et al., 2006]. Especially, RSSI-based localization has an advantage that utilization of the same radio hardware for both communication and localization would make it possible to provide efficiency in simple design framework over a specific localization infrastructure - such as ones using directional antennas or same transmission signal, and separate design of ultrasound or infrared [Elnahrawy et al., 2004]. Table 2.1 summarizes the characteristics of different RSSI-based localization techniques.

Table 2.1: Characteristics of different RSSI-based localization techniques [Lymberopoulos et al., 2006].

| Technique | Design Approach | Technology | Testbed Dimensions | Location Error |
|------------------------------------|--|------------|----------------------------|----------------|
| Ecolocation | Constrained based approach applied on the ordered sequence of raw RSSI data | MICA2 | 26 × 49(ft) (Indoor) | 10 ft |
| Probability Grid | Probabilistic. RSSI values are used to estimate the one-hop distance in a grid | MICA2 | 410 × 410(ft) (Outdoor) | 66 ft |
| Radar | RSSI fingerprint map | 802.11b | 42.9 × 21.8(m) | 15 ft |
| MoteTrack | RSSI fingerprint map | MICA2 | 18751 ft ² | 13 ft |
| LEASE | Online Fingerprinting and signal propagation modeling | 802.11b | 225 × 144(ft) | 15 ft |
| Bayesian Indoor Positioning System | Learning Based | 802.11b | 225 × 144(ft) | 20 ft |
| Stochastic Indoor Location System | Optimal positioning of a given number of clusterheads with respect to the location-detection performance | N/A | N/A | N/A |
| Monte Carlo Localization | Learning based with signal strength map | 802.11b | N/A | 7.2 ft |
| Nibble | Bayesian networks | 802.11b | 224 × 96(ft) (Indoor) | 20 ft |

In general, signal strength based localization can be divided into map-based [Bahal and Padmanabhan, 2000 and Lorincz and Welsh, 2005] and distance-based [Elnahrawy et al., 2004 and Yedavalli et al., 2005] algorithms. In the map-based algorithms, a large set of sample points that cover the whole geographical area collect one or more signal strength measurements from all visible beacon nodes. Each sample point is then mapped into either signal strength vector or a signal strength probability distribution.

Despite the map-based localization methods provide relatively small location errors, they have some limitations because they require an excessive amount of profiling processes. On the other hand, distance-based localization algorithms utilize the distance prediction model either by triangulation or trilateration method because the raw signal strength data can be mapped into distance through a signal propagation model. The

measured distance can be either used directly in order to determine the location of mobile devices or used in a probabilistic localization algorithm.

However, RSSI-based localization has a critical limitation that has been observed in the characteristics of radio signal propagation. Since raw RSSI does not provide enough accuracy of localizing the mobile objects to be used in practical application, the distance prediction requires additional investigation on the probability method or learning-based localization algorithm [Lymberopoulos et al., 2006]. The main factors associated with the inaccuracy are identified as the multipath propagation and signal fading. Multipath is the propagation phenomenon that results in two or more propagation paths between sensor and receiving antenna. Fading induces the rapid fluctuations of amplitudes, phases, or multipath delays of a radio signal over a short period of time or travel distance. The level of inaccuracy inherited from the physical properties of radio signal increases as the chance of reflections or scatterings of signal from unwanted obstacles becomes higher, such as indoor environments where signals travel with much obstruction. The localization error observed in the RSSI-based techniques shown in Table 2.1 illustrates that most of the errors range from 15 ft to 25 ft for indoor system. While construction site is considered as an outdoor system where the chaotic properties of signal propagation can be much lessened than indoor due to little obstruction, there still exists physical limitation of multipath and fading in signal propagation because of the complicated layouts of construction sites.

2.3 Chapter Summary

This chapter has presented prior research on and development of component tracking system in construction industry and localization techniques, which have

followed a variety of approaches. Using bar code, RFID and GPS technologies, automated materials tracking methods have been examined, and advantages and disadvantages have been identified for possible applications in construction sites. In addition, localization techniques, core methodologies in tracking of mobile objects, have been presented to bring forth a new investigation with different system architecture of tracking the materials, equipment, tools and personnel. There is a recognizable need for ZigBee-based location sensing in construction industry due to technical and economical drawbacks of aforementioned approaches. Chapter 3 will examine the detailed description of wireless sensor and ZigBee networks for the recognizable needs and shifts that introduce a new method for materials tracking in construction sites.

CHAPTER 3: WIRELESS SENSOR NETWORKS

Emerging technology of wireless sensor networks has been gradually replacing the traditional paradigm of wired sensor networks during the past several years. Wireless sensor networks are characterized as the combination of processing, sensing, computing, and communication throughout distributed mesh networks. Tiny embedded devices with CPU, memories, radio components, and supply power form a flexible network by peer-to-peer communication protocols, such that the individual devices are interconnected throughout the network where data is seamlessly routed among all the nodes. While the capability of any single device for networking is not surprising at all, the composition of hundreds of single nodes offers new technological possibilities. This chapter introduces the current technologies of wireless sensor network that consists of hardware and software architecture under TinyOS platform.

3.1 Introduction

In sensor networks, the special purpose of individual sensors is designed to provide flexibility and scalability while required to be as small and inexpensive as possible. Generic wired sensor networks require extensive efforts of installation and maintenance for long-term deployment. For example, a periodic bridge health inspection for the wired sensor network takes 2-4 hours to install and removal of the sensor infrastructure, such as sensor devices, wires, power supply, and PC station [Kim et al., 2000]. If the bridge health inspection for long-term deployment is considered, extra efforts are required for data transmission to remote destination, safety equipment for sensors, and durable wires for survival of connectivity.

Unlike traditional wired sensor networks, deployment costs and efforts can be minimized. Wireless systems not only reduce the cost of wires, that otherwise require thousands of feet routed through protective conduits, but also provides a simple method of installation and removal due to the compact size and radio communications. In addition, the power of wireless sensor networks lies in the scalability and flexibility to deploy large numbers of tiny nodes for extended applications. For example, up to 65,000 sensor nodes can be linked and formed under a network topology [Skibniewski and Jang, 2006], and this capability enables a flexible scalability for a wider range of applications. Thus, thousands of sensor nodes could communicate with adjacent nodes hop-by-hop to efficiently deliver any sensing information, to detect equipment faults and adverse events, to diagnose problems, and to respond to hazardous conditions in real-time.

Traditional network abstractions for wireless systems are generally not suitable for wireless sensor networks. For example, items such as cell phones, personal digital assistants, or laptops with IEEE 802.11 cost hundreds of dollars, targeting specialized applications, and rely on the pre-deployment of extensive infrastructure support. In addition, unlike traditional operating systems, integration of wireless connectivity should be provided to the operating systems for wireless sensor networks. For instance, in TinyOS, robust component and execution model exploit advanced compiler technology to simultaneously provide efficiency and reliability. As a result, more traditional operating systems in gateway-class and high-bandwidth systems are now incorporating these concepts [Crossbow Technology, Inc.].

3.2 Hardware Architecture

Most current deployments of sensor networks include small size generic sensor devices, e.g. square-inch-size. Their network capability consists of an interconnected mesh tied to the Internet through multiple gateway-class devices. The recent developments in first-generation wireless sensor networks are now evolving into a new generation of hardware capable of dealing with complex data streams and robust communication. Platforms for sensor network hardware available today are listed in figure 3.1.

| Mote Type | WeC | Rene | Rene 2 | Dot | Mica | Mica2Dot | Mica2 | Micaz | Telos | |
|-----------------------------------|----------------------|------|-----------|------|----------------------|----------|--------|------------------|--------|----|
| Year | 1998 | 1999 | 2000 | 2000 | 2001 | 2002 | 2002 | 2004 | 2004 | |
| | | | | | | | | | | |
| Microcontroller | | | | | | | | | | |
| Type | AT90LS8535 | | ATMEGA163 | | ATMEGA128 | | | MSP430 | | |
| Program Memory (KB) | 8 | | 16 | | 128 | | | 60 | | |
| RAM (KB) | 0.5 | | 1 | | 4 | | | 2 | | |
| Active Power (mW) | 15 | | 15 | | 8 | | 33 | | 3 | |
| Sleep Power (μ W) | 45 | | 45 | | 75 | | | 6 | | |
| Wakeup Time (μ s) | 1000 | | 36 | | 180 | | | 6 | | |
| Storage | | | | | | | | | | |
| Chip | 24LC256 | | | | AT45DB041B | | | ST M24M01S | | |
| Connection Type | I ² C | | | | SPI | | | I ² C | | |
| Size (KB) | 32 | | | | 512 | | | 128 | | |
| Communication | | | | | | | | | | |
| Radio | TR1000 | | | | CC1000 | | CC2420 | | | |
| Data rate (kbps) | 10 | | | | 40 | | 38.4 | | 250 | |
| Modulation Type | OOK | | | | ASK | | FSK | | O-QPSK | |
| Received Power (mW) | 9 | | | | 12 | | 29 | | 38 | |
| Transmit Power at 0dBm (mW) | 36 | | | | 36 | | 42 | | 35 | |
| Power | | | | | | | | | | |
| Minimum Operation (V) | 2.7 | | | | 2.7 | | | 1.8 | | |
| Total Active Power (mW) | 24 | | | | 27 | | 44 | | 89 | 41 |
| Programming and Sensor I/O | | | | | | | | | | |
| Expansion | none | | 51-pin | | none | | 51-pin | | 10-pin | |
| Communication | IEEE 1284 with RS232 | | | | IEEE 1284 with RS232 | | | USB | | |
| Integrated Sensors | no | | yes | | no | | | yes | | |

Figure 3.1: Mote family of TinyOS for wireless sensor network [Levis and Culler, 2002]

We adopted the MICAZ mote module produced by Crossbow Technology Inc., a brief diagram of the hardware structure is shown in figure 3.2. The Berkeley motes are a notable development in wireless sensor networks, with the MICAZ the most recently

developed commercially available product, constructed from off-the-shelf components to provide efficient flexibility. MICAZ supports a ZigBee compliant RF transceiver, i.e. IEEE 802.15.4, and it works between 2.4 and 2.4835 GHz, a globally compatible ISM band. Its DDSS radio offers both high speed capabilities (250 kbps) and hardware security (AES-128). The MICAZ 51-pin expansion connector supports Analog Inputs, Digital I/O, I2C, SPI and UART interfaces. These interfaces make it easy to connect to a wide variety of external peripherals, including a variety of sensor, data acquisition boards and gateway.

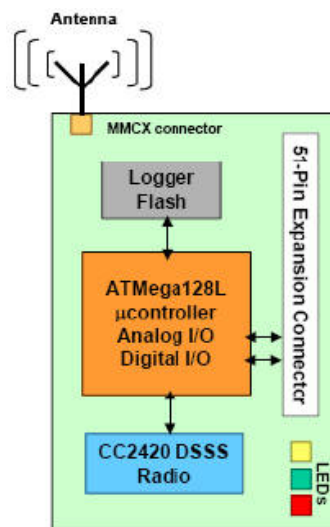


Figure 3.2: Diagram of MICAZ hardware structure [Crossbow Technology, Inc.]

3.2.1 Microcontroller

The main microcontroller is Atmel ATMEGA128L running at 4MHz and performing four million instructions per second (MIPS). High-performance and Low-power AVR 8-bit Microcontroller provides 128-Kbyte of In-System Programmable Flash with Read-While-Write capabilities. 32 general purpose working registers are connected

to ALU directly, and 4-Kbyte static RAM, 4-Kbyte EEPROM are provided. In addition, ATMEGA128L provides the following features: two external universal asynchronous receiver transmitters (UART), a serial peripheral interface (SPI) port linked to CC2420, a byte oriented Two-wire Serial Interface, internal 8-channel 10-bit analog-to-digital converter (ADC) with operational differential input state with programmable gain, three hardware timers, 51-pin expansion connector to interface with a variety of sensing and programming board, and six sleep modes for power restriction.

For more expanded connectivity to external devices, the I/O subsystem interface is installed with a 51-pin expansion connector that consists of eight analog lines, eight power control lines, three pulse width modulated lines, two analog compare lines, four external interrupt lines and an I²C-bus. The expansion connector also contains a standard UART interface to control or provide to any RS-232 protocol-based device.

3.2.2 RF Transceiver

The CC2420 is the industry's first single-chip 2.4 GHz RF transceiver designed for low-power and low-voltage wireless applications [Atmel Corporation]. It is compliant with the IEEE 802.15.4 standard, and ready and compatible to be used in ZigBee products. The CC2420 includes a digital direct sequence spread spectrum baseband modem providing a spreading gain of 9dB and an effective data rate of 250 kbps. In addition, the CC2420 provides extensive hardware support for packet handling, data buffering, burst transmissions, data encryption, data authentication, clear channel assessment, link quality indication, and packet timing information. The SPI interface

configures the interface and the transmit/receive FIFOs of the CC2420 to achieve read-while-write operation, shown in figure 3.3.

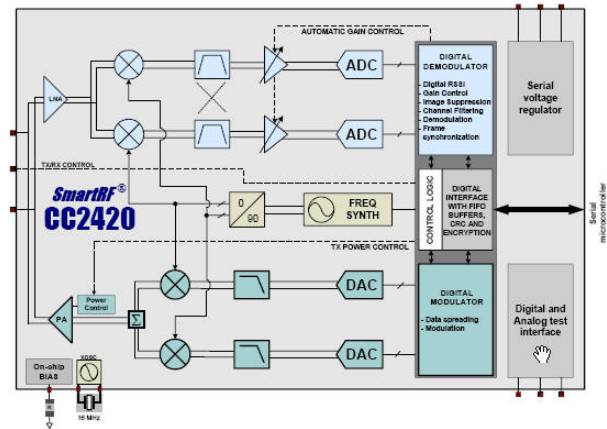


Figure 3.3: CC2420 block diagram [Texas Instrument]

The CC2420 radio allows the controller direct access to the signal strength of the incoming RF transmission as well as sampling the level of background noise during periods when there is no active transmission. These features can improve efficiency for multihop networking applications by providing a link with good signal to noise ratio. Because the interactions between transmitter and receiver are predictable, it is possible to achieve low duty cycle without global coordination or complex time slotting.

3.2.3 Gateway

The MIB510 interface with serial port (shown in figure 3.4) bridges the wired and wireless network, and allows for the aggregation of sensor network data on a PC as well as other standard computer platforms. A MICAZ node can function as a base station when it is connected to the MIB510CA serial interface that also provides an RS-232

serial programming interface. The MIB510 has an onboard processor to achieve programming for the Mote processor/radio boards. Two 51-pin expansion connectors are embedded to allow sensor boards to be attached for monitoring or programming development.

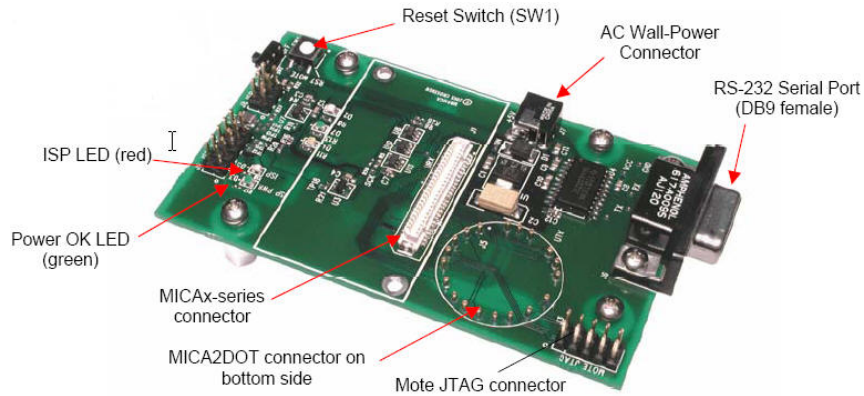


Figure 3.4: MIB510 interface [Crossbow Technology Inc.]

3.2.4 Sensor Board

The MTS300/310 is a set of flexible sensor boards with a variety of sensing modalities, and provides capabilities to develop sensor networks for a variety of applications, including vehicle detection, low-performance seismic sensing, movement, acoustic ranging, robotics, and other applications. It consists of several embedded sensors that include light, temperature, acoustic and sounder (both MTS300/310) as well as dual-axis magnetometer and dual-axis accelerometer (MTS310 only), shown in figure 3.5.

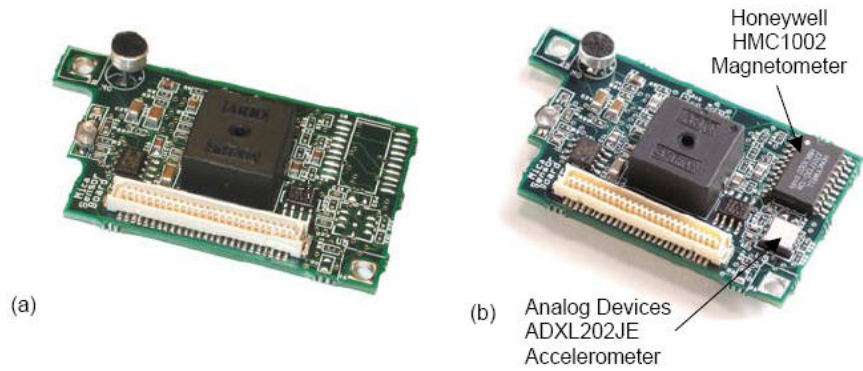


Figure 3.5: MTS300/310 sensor board: (a) MIB300 and (b) MIB310 [Crossbow Technology Inc.]

3.3 ZigBee with IEEE 802.15.4 Network Standard

There are a multitude of standards that address mid to high data rates for voice, PC LANs, video, etc. However, until recently there has not been a wireless network standard that meets the unique needs of sensors and control devices. Sensors and controls do not need high bandwidth but they do need low latency and very low energy consumption for long battery lives and for large device arrays. There are a multitude of proprietary wireless systems manufactured today to solve a multitude of problems that also do not require high data rates but do require low cost and very low current drain. These proprietary systems were designed because there were no standards that met their requirements. These legacy systems are creating significant interoperability problems with each other and with newer technologies.

ZigBee is an emerging network technology, as a wireless communication standard, capable of realizing the ubiquitous environment to satisfy such requirements. ZigBee is the product of the ZigBee Alliance [ZigBee Alliance], an organization of manufacturers dedicated to developing a new networking technology aimed at converting industrial and home application into wireless connectivity. The ZigBee specification has been released

to the public in June 2005, and products supporting the ZigBee standard are now being introduced into the market. The ZigBee specification takes advantage of the IEEE 802.15.4 wireless protocols as the basic communications method, and expands on it with a robust mesh network, application profiles, interoperability, and device description. The promise of the ZigBee application can be found in robust and reliable, self-configuring and self-healing networks that provide a simple, cost-effective, and battery-efficient approach to add wireless to mobile and fixed communication devices.

3.3.1 IEEE 802.15.4 and ZigBee Structure

The IEEE 802.15.4, released in May 2003, was selected by the ZigBee Alliance as a lower-rate version of the IEEE 802.15 series standard for wireless communications of the Wireless Personal Area Networks (WPAN). The IEEE 802.15.4 standard defined the RF capacity with the system operating in three license-free bands for global communication. The RF link parameters are shown in table 3.1.

Table 3.1: Frequency allocations and physical layer in IEEE 802.15.4 [Adams, 2005].

| Frequency band | 2.4 GHz | 915 MHz | 868 MHz |
|--------------------|-----------|---------|---------|
| Number of channels | 16 | 10 | 1 |
| Bandwidth (kHz) | 5,000 | 2,000 | 600 |
| Data rate (kbps) | 250 | 40 | 20 |
| Symbol rate (ksps) | 62.5 | 40 | 20 |
| Modulation method | O-QPSK* | BPSK** | BPSK |
| Diffusion method | DSSS*** | DSSS | DSSS |
| Available regions | Worldwide | USA | Europe |

O-QPSK* (Offset Quadrature Phase Shift Keying)

BPSK** (Binary Phase Shift Keying)

DSSS*** (Direct Sequence Spread Spectrum)

The physical layer of the IEEE 802.15.4 has functional capabilities, such as measurement of reception power, notification for link quality, and Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA). A radio with data to transmit initially seeks the channel and then transmits its packet when the channel is clear. On the other hand, if the channel is occupied or interfered by either 802.15.4 or non-802.15.4 station, the radio holds off from transmitting to the channel for a certain period of time before again checking the channel for occupancy. In this manner, it is possible to conduct high-quality communications in a platform that is shared with other systems using the same frequency band. At the Medium Access Control (MAC) layer, IEEE 802.15.4 controls access to the radio channel using a CSMA-CA mechanism. Network beacons bind a superframe, allowed by IEEE 802.15.4, and divide it into 16 equally segmented slots to synchronize the attached devices, to identify the personal area network (PAN), and to describe the superframe structure. Hence, the slotted CSMA-CA provides a communication method to any device with which other devices compete within the networks.

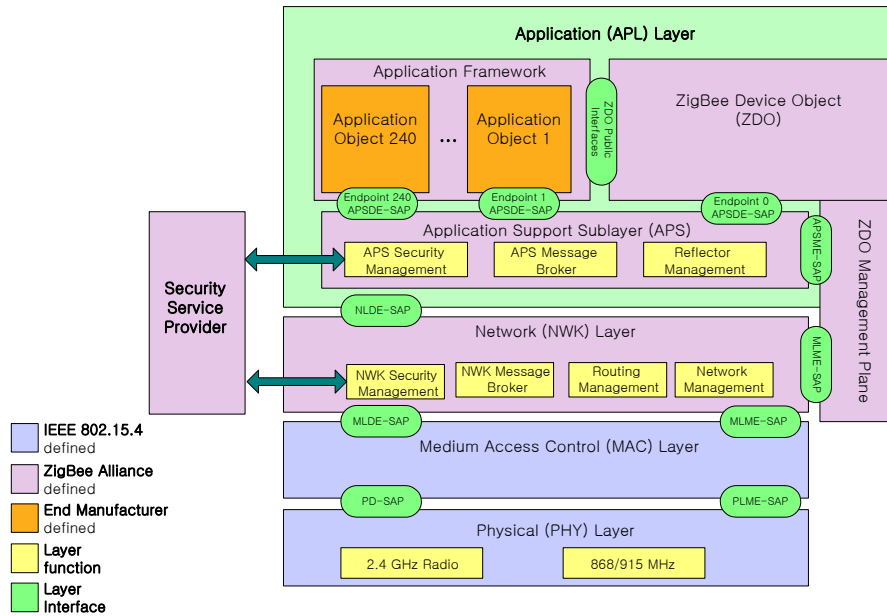


Figure 3.6: ZigBee stack architecture [ZigBee Alliance].

Figure 3.6 shows the relative organization of the IEEE radio with respect to the ZigBee functionality. In addition to the basic IEEE 802.15.4 standard, ZigBee defines the upper layers of the system. The physical and MAC layers take full advantage of the physical radio specified by IEEE 802.15.4, which describes a peer-to-peer radio using Direct Sequence Spread Spectrum (DSSS). The data rates, channelization, and modulation techniques are employed in those layers. In the logical network, security, and application software layer, the ZigBee networking stack creates the mesh networking capability. Each microcontroller /RF chip combination requires its own ZigBee stack due to the differences in microcontrollers and RF chips, and the ZigBee stack provided by the chip vendor is included with either the microcontroller or RF chip. The application layer is defined by two types of profiles: public profiles are certified by ZigBee Alliance for interoperability purpose, and private profiles are for use in closed systems.

3.3.2 Zigbee Network Layer

ZigBee is intended to support three network topologies: star, mesh, and hybrid networks. The star network is very common, and provides the simple configuration of topology. However, when the size of network is large and the network becomes complex, the star network may not have an advantage to be used. Mesh network (or peer-to-peer) networks provides a high degree of reliability when the signal packets are routed through a variety of nodes. This feature of a mesh network can be an advantage, especially when clear lines of sight between nodes are required due to obstacles that block the radio signals. Mesh networks also make possible for communication to pass through any set of nodes between the sender and receiver. This capability makes the mesh network self-healing and self-configuring even when a node breaks down and is out of service. Finally, a hybrid network is essentially a combination of star and mesh networks which is usually deployed to address the most complex network requirements.

A ZigBee network, shown in figure 3.7, consists of ZigBee coordinators, ZigBee routers, and ZigBee end devices. The coordinator and routers form a star network configuration using PAN coordinator functions, and it is possible to form a multi-hop network by simultaneously configuring a mesh network between the coordinator and routers. On the other hand, the end devices take part in the network communications by linking to the coordinator and routers through star-link networks. The end devices conduct multi-hop communications via connected routers to communicate with other devices connected to the networks.

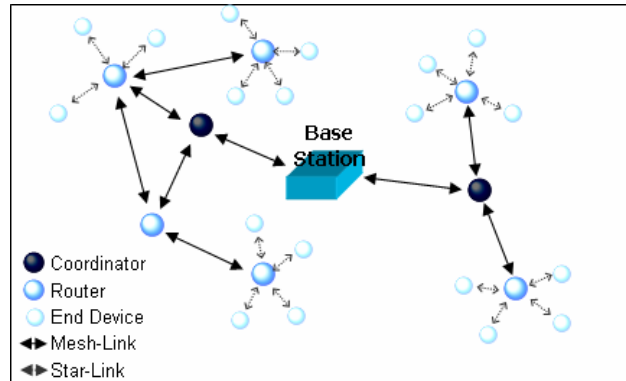


Figure 3.7: ZigBee Network [ZigBee Alliance].

3.4 Software Architecture

A critical consideration for the implementation of a wireless sensor network is the design of software architecture that bridges low-level hardware with the operating system. A good design of software architecture allows the wireless sensor network to be efficient with respect to the computation process, memory allocation, and power management, so that it meets expected requirements for various applications. Unlike the hardware design that deals with size, power, and RF transmission, the software architecture resides in an operating system that manages hardware capabilities by supporting concurrent and event-driven operations to achieve efficient modularity and robustness. Design of the software should allow multiple applications or events to simultaneously use system resources, e.g. communication, computation and memory. This is because sensor networks consist of hundreds of tiny, low-power nodes that execute reactive and concurrent operations with extensive power and memory constraints. The aforementioned requirements for software design of wireless sensor network drive the design of TinyOS to scale with the current technology trends of integrated hardware design and software components.

TinyOS is a flexible and application-specific operating system for the platform of wireless sensor networks. It has become a primary solution that integrates flexibility and modularity with an execution model, component model, and communication model that provides complex and safe concurrent operations. TinyOS forms a core component of ambient intelligence systems that overcome the challenges of limited resources and event-driven concurrent applications. It supports an event-driven concurrency model based on split-phase interfaces, asynchronous *events*, and deferred computation called *tasks*. TinyOS defines a number of important concepts that are described in nesC. First, nesC applications are built from components with bi-directional interfaces. Second, nesC defines a concurrency model based on tasks and hardware event handlers, and detects data races at compile time. A concurrency model is important for low-power wireless sensor applications because many event-driven executions in TinyOS require efficient CPU allocation for power management and improvement of execution time. A component provides and uses interfaces that declare a set of functions called commands and events. For a component to call the commands in an interface, it must implement the events of that interface. For a concurrency model, TinyOS executes only one program that consists of selected system components and custom components needed for a single application. Tasks and hardware interrupt handlers are the two threads of execution for this model.

3.4.1 Execution Model

TinyOS supports event-driven execution to attain the higher level of operating efficiency required in wireless sensor networks. This event-driven execution handles

concurrent operations with a minimal amount of memory space that requires pre-allocated memory stack for each execution context. This event-driven domain of wireless sensor networks provides a concurrency in which events arrive at any time, interacting with the continuous computations. The fundamentals of the execution model consist of run-to-completion tasks and interrupt hardware handlers. The former, tasks, represent the ongoing computation in which a program sends a task to the scheduler for execution with the post operator, and the tasks are executed in a particular order according to a run-to-completion rule. The latter, interrupt hardware handlers, are signaled asynchronously by hardware. Low-level components handle the complexities associated with an interrupt-based programming code to meet their real-time requirement. Normally, tasks are not preempted and run to completion, and this is characterized as atomic with respect to other tasks. However, tasks are not atomic with respect to interrupt handlers, commands, and events they invoke. A different application can use tasks to ensure that all data and code modification occurs atomically when viewed from the context of other tasks.

The design of the structuring concepts and execution model of TinyOS is implemented by nesC, an extension to the C programming language to support the event-driven operating system in wireless sensor networks. As the basic concepts behind nesC, programs are achieved out of components that are wired to perform whole programs. Components are passed from the threads of operation that are rooted in a task or hardware interrupt through interfaces, so that it is possible to construct and composite event-driven components separately. The interfaces link different components by ‘provide’ or ‘use’, and the provided interfaces represent the functionality for providing

the component to its users, and the used interfaces represent the functionality to make its user use the component. Thus, the interfaces specify a set of functions to be implemented by the interfaces' provider and user by providing bidirectional interaction so that a single interface can represent a complex interaction between components. As a consequence, a component cannot call a send command separately unless it implements a sendDone event beforehand. Figure 3.8 illustrates an example of wiring the component structure to blink LED once a second.

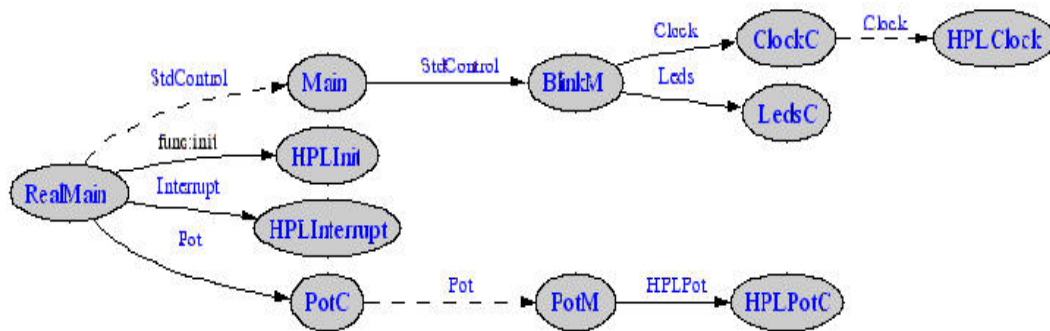


Figure 3.8: Illustration of Blink application for wiring the component structure

3.4.2 Component Model

In addition to providing high efficient event-driven execution, TinyOS also provides a component model that aims at highly efficient modularity and easy composition. This model is important to increase reliability and performance, and an application developer would be able to easily combine components into a whole program structure. A wiring specification independent of component implementations is used to connect a specific application, and it defines the required set of components that the application may use. A provider or user of the aforementioned interfaces defines a model,

a specific function, through an interface type. Figure 3.9 shows a simplified form of TimerM component that provides StdControl and Timer interface, and uses a Clock interface.

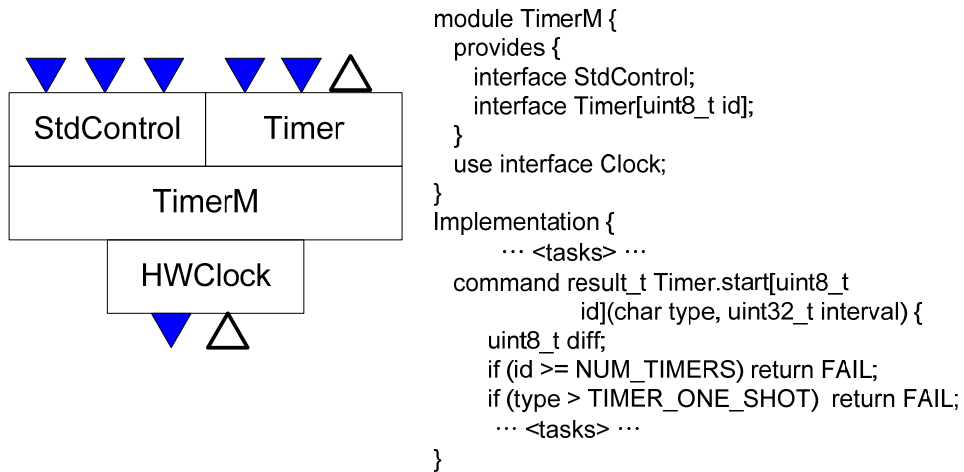


Figure 3.9: An example of component structure defined in TinyOS

In TinyOS, commands are non-blocking requests implemented to lower level components. Typically, a command stores request parameters into its local frame and conditionally posts a task for later execution. Sometimes, it can also invoke lower commands, but it must not wait for indeterminate latency actions to take place. Handlers in the lowest level components are connected directly to hardware interrupts, which may be external interrupts, timer events, or counter events. An event handler can deposit information into its frame, post tasks, signal higher level events, or call lower level commands. In addition, a hardware event triggers a process that goes upward through events and can reverse downward through commands. Normally, commands are not allowed to signal the events to avoid cycles in the command/event chain. Both

commands and events are intended to perform a small, fixed amount of work, which occurs within the context of their component's state.

Interface namespace is assigned to each component, and refers to the commands and events that it uses. When a configuration wires interfaces together, the connection between the local name of an interface used or provided by another local name is made in compilation. As a consequence, a component can invoke an interface without explicit reference to its implementation, which makes it easy for a new component to use and provide the same interfaces. Another characteristic of interfaces is a multiple-time of wiring to different interfaces. For example, a StdControl interface in Main can be wired to TimerC, Multihop, and LedsC, which gives the capability of providing transparency to the caller. Thus, the multiple wiring provides a flexible combination of the results in each function call by returning success or fail flags, using result_t logic.

The TinyOS component model also allows easy migration of the hardware/software boundary because the event-driven software model is complementary to the underlying hardware structure. In this way, hardware-based implementation can request the use of a fixed sized, pre-allocated memory storage. The feature of this migration from software to hardware is particularly important for networked sensors because it is feasible to apply the tradeoffs between the scale of integration, power requirements, and the cost of the system.

3.4.3 Communication Model

A critical aspect of TinyOS design is the communication model it supports. Active Messages (AM) structure is the primary networking feature in TinyOS, capable of

providing a simple and extensive message-based communication for distributed computing systems. An Active Message is small data packets of 36-byte associated with 2-byte destination address, 1-byte AM handler ID, 1-byte group ID, 1-byte message length, and 29-byte message payload. When an Active Message is received, a node carries out the message to one or more handlers registered to receive those types of messages. The handler function extracts the message from the network by either integrating the data into the computation or sending a response message. Minimal buffering for the message provides the network a “pipeline,” and buffering difficulties in the communication mechanism can be removed. Otherwise, they may block the communication protocols or the special send/receive may be congested. Thus, message handlers should execute quickly and asynchronously in order to ensure performance from any network congestion.

Active Messages (AM) originated from high performance parallel computing, but the low power design regime for wireless sensor network was adopted by using basic concepts of integrating communication and computation. Thus, the AM communication model fit to a distributed event model that nodes in the networks, can communicate with each other’s events. In this way, Active Messages, inherited from the lightweight architecture of parallel computing, is able to provide an extensible communication framework with high efficiency and agility. In order to match communications to hardware capabilities, it is necessary to overlap the computation work with high-level communications, such that the event-driven function of Active Messages allows the execution to interact with sensor communication or the execution of other applications.

Another noticeable feature of Active Messages is compatibility from different types of physical communications. Networked sensor platforms require an extensive effort to provide message transmission with certain functionalities, e.g. message acknowledgement, addressing, and dispatch. However, a higher level of application layers could reside on top of the message transmission efforts, and the developers can decide a variety of devices that have different physical communication capabilities and needs. This is because the TinyOS provides the component model that allows the communication tasks as separate components chosen by developers, so the application developers are able to create their applications into different hardware platforms by performing encryption, flow control, and packet fragmentation.

3.5 Deployment Metrics and Challenges

It is critical to understand the capability of a system platform, which the wireless sensor network requires, to improve and optimize performance and applicability. Multidimensional needs and requirements in various application scenarios should be examined in order to achieve a successful application deployment. This chapter describes a set of metrics and challenges that exist in the current stage of wireless sensor network, and this effort could leverage the application strategy for developers to understand the possible tradeoffs between performance and requirement.

3.5.1 Life Cycle

The Power supply for wireless sensor platforms is the primary limitation for long-term deployment. With current energy supply forms, the central station cannot provide adequate power supply to the distributed, local nodes. Each sensor node should be

designed to manage and provide its local power supply in order to maximize the total life cycle of the networks. In many situations, a battery is used as a self-powered architecture, in which the total life cycle could last for a couple of years according to power management in the TinyOS. A comparison between Lithium and Alkaline batteries is described in figure 3.10. Lithium batteries have a compact power source with a constant voltage supply that is drained slowly. On the other hand, alkaline batteries suffer from non-constant voltage during their service life, so its discharge characteristics are considered an essential factor in designing a system that tolerates a wide range of input voltages.

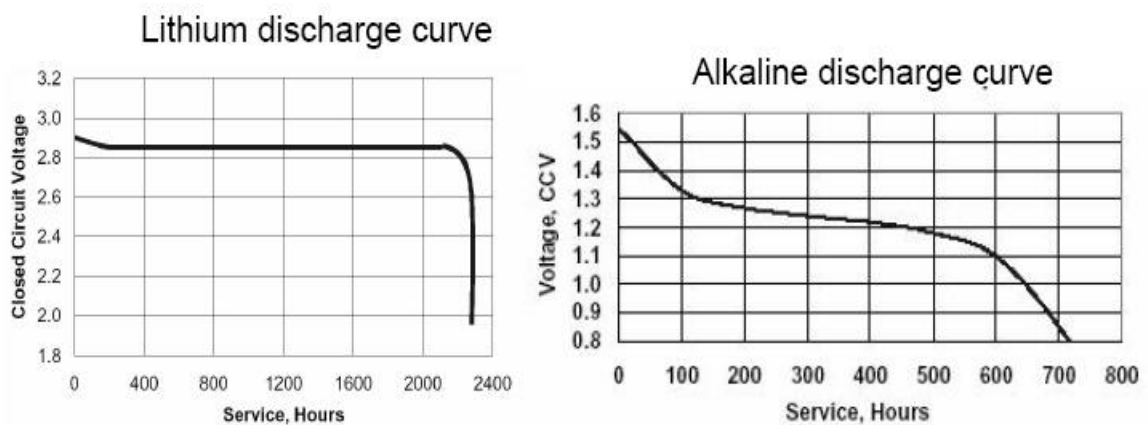


Figure 3.10: Illustration of discharge curve in Lithium discharge curve and Alkaline batteries [Hill, 2003]

Usually, power consumed by a radio subsystem is the most critical factor in determining the life cycle of the energy supply. Modern low-power transceivers consume approximately the same amount of energy, between 15 and 300 milliwatts of power, when sending and receiving. This power consumption rate could be reduced by decreasing the transmission output power or by decreasing the radio duty cycle.

However, it is interesting to note that if the receiver is on at full power during the periods of intermittent communication, the overall cost of radio communications may be the single largest one in the system. This is because a significant amount of energy in the transmitter is consumed by internal operations, whereas the actual power emitted from the antenna in the transmitter takes a small fraction of the transceiver's total energy consumption. Thus, efficient algorithms and protocols should be addressed to reduce the power consumption rate in a radio transceiver.

3.5.2 Cost and Size

With the development of the micro electromagnetic system (MEMS), it is possible to design a very tiny sensor device to be used in industrial applications. However, practical design of a sensor device varies according to the deployment strategy and protocol specifications. As a consequence, detailed categories and attributes related to the application scenarios, that define the sensor types and network specifications, must be addressed in order to satisfy the needs and requirements. In addition to the size of the sensor device, the economics associated with the device cost plays a critical role to the decision maker in a management point of view. In most cases in the real world, adoption of wireless sensor network technologies often rely on the total cost of ownership and initial deployment costs. Thus, a cost reduction in individual sensor node will result in the increased applicability of a highly dense network that could increase link survival rates and connectivity.

3.5.3 Coverage Range

Coverage is one of the essential components when deploying the wireless network. When the physical domain of an application becomes larger, the coverage range increases its value to the end user because the coverage range is often considered the network's capability for communications. However, it is important to note that the network coverage can be extended into a large scale of physical area because the feature of multihop communication overcomes the limited capabilities of an individual wireless link. Thus, most wireless networks refer to their communications capability as the scalability, a network's ability to scale a large number of nodes beyond the range of the radio technology alone.

Sensitivity and transmission power are used to calculate the transmission range in dBm. The dBm is a logarithmic scale where 5 dB increases 5 times the power and 1 watt power is represented as 30dBm. When a transmitter sends a signal at 0dBm, the transmission range can be 25-50 meters in free space with receiver sensitivity of -85dBm. If a receiver's sensitivity is -110 dBm, then the transmission range is 100-200 meters.

In general, the characteristics of civil infrastructure systems include a large scale of physical domain, and this scalability becomes a more critical component for the deployment of a wireless sensor network than indoor network systems. For example, a bridge monitoring system using wireless sensors requires up to a several-mile network link to collect reliable measurements obtained from the behaviors of the whole bridge structure. As a consequence, a user should keep in mind that the network coverage is capable of scaling to meet the eventual needs of an application. However, multihop networking protocols may increase the power consumption rate of the nodes at a given

transmission range because an increased number of nodes with higher density will impact the life cycle of the networking system. Furthermore, increased density of the network will also decrease the efficiency of the sampling rate such that more sensing points can cause a delay in a network scheduling scheme of a large scale network. This is because a data collection scheme in a network, a parent node should determine a child node for relaying a mechanism in multihop. If one parent node has 10 successive child nodes, then the parent node should handle 10 times as much data as a single network because of the multiplicative increase in data communication. As a result, special care should be taken in the tradeoffs between scalability and sampling rate according to the purpose and needs of the various types of applications.

3.5.4 Signal Interference

Signal interference is a possible challenge if multiple protocols of networks are used in the same application domain. The specification of a radio signal recommended by IEEE 802.15.4 utilizes a 2.4 GHz frequency with a low transmission power of up to 1 W, which is allowed by FCC regulations. A low power transmission scheme with low duty cycles (under 1%) provides reduced interference by other devices and systems, practically, in-network systems [IEEE Standards, 2003]. Also, other types of radio signals that use a similar frequency range, such as a 2.4 GHz cordless phone or 2.4 GHz Wi-Fi communication, etc, would not interfere with the signal of IEEE 802.15.4 because a different scheme of band structure, prevents the possible interference between them.

In a highly dense network, however, signal collision problem could arise, especially when several nodes try to transmit data simultaneously. The proposed

research will identify the factors that might possibly cause a signal interference or a collision during practical applications. This issue is widely studied in the area of communication theory, and the recommended Media Access Control (MAC) schemes for IEEE 802.15.4 is equipped with state-of-the-art collision avoidance techniques, which enable it to operate efficiently in a high interference environment with dense deployment of sensor nodes.

3.6 Chapter Summary

Emerging technology of wireless sensors and networks has drawn interests on their deployment in many construction applications. Among a variety of benefits, wireless data communication between the sensors or to a storage location opens up a range of possibilities because of the ease and the low cost by which the sensors can be deployed. Despite the expected benefits of a wireless sensor network, most construction and civil engineers are not aware of the details behind the broad picture of the wireless sensor network. This is because of the great complexities required, in addition to the advanced knowledge of embedded programming and hardware design, to achieve the level of performance desired. Such knowledge is not common among construction and civil engineers, which are often tasked to simply use the devices on construction sites. Thus, this chapter introduced the current technologies of wireless sensor network that consists of a hardware operating system, software architecture, hardware components in wireless sensor modules, and wireless network protocols. It is hoped that the knowledge described in this chapter encourages development and deployment of wireless sensor networks into a variety of civil engineering applications.

CHAPTER 4: FEASIBILITY STUDY

The multipath effect changes the direct signal target, causing errors or low quality of localization. In addition to multipath, fast speed of radio signal propagation generates the delays of signal detection in microprocessor. These problems motivated to introduce new system architecture of IEEE802.15.4-based localization scheme with the combination of RF and ultrasound for the construction materials tracking. Ultrasound device is well known as a good tool for distance measuring with accurate, simple hardware design. Hence new proposed system makes use of high performance of ultrasound as a method of distance estimation, and at the same time 2.4 GHz radio frequency signal is used to provide general wireless communication with the help of recent technology provided by IEEE 802.15.4 network standard.

In addition, application scheme of wireless sensor monitoring is described based on the embedded system. According to different application scenario, different sensors, such as temperature, humidity, displacement, or strain sensor, can be attached to the main device system to provide extended functionality of wireless monitoring system in construction. This chapter introduces new tracking and monitoring system architecture to justify the necessity of adoption by comparing the previous tracking method, and implements the feasibility study on accuracy performance by simulating the distance measuring algorithm and trilateration.

4.1 System Architecture for Tracking Algorithm

In the RF-based wireless sensor network such as ZigBee, elimination of undesirable multipath components and fading is an important issue to achieve the

localization of the distributed objects. While a construction site is considered as an outdoor environment where severe multipath of radio signal propagation is somewhat reduced compared to an indoor environments, there are still major concerns about complicated properties of signal propagation due to the reflection from ground, buildings, equipment, and materials. This research focuses on the new methodology to mitigate the unwanted components of signal propagation for accurate and reliable measurement of the location of distributed sensor devices. Additionally, it also proposes a new approach to a potential deployment of ZigBee network to the automated tracking and monitoring system in construction site (figure 4.1).

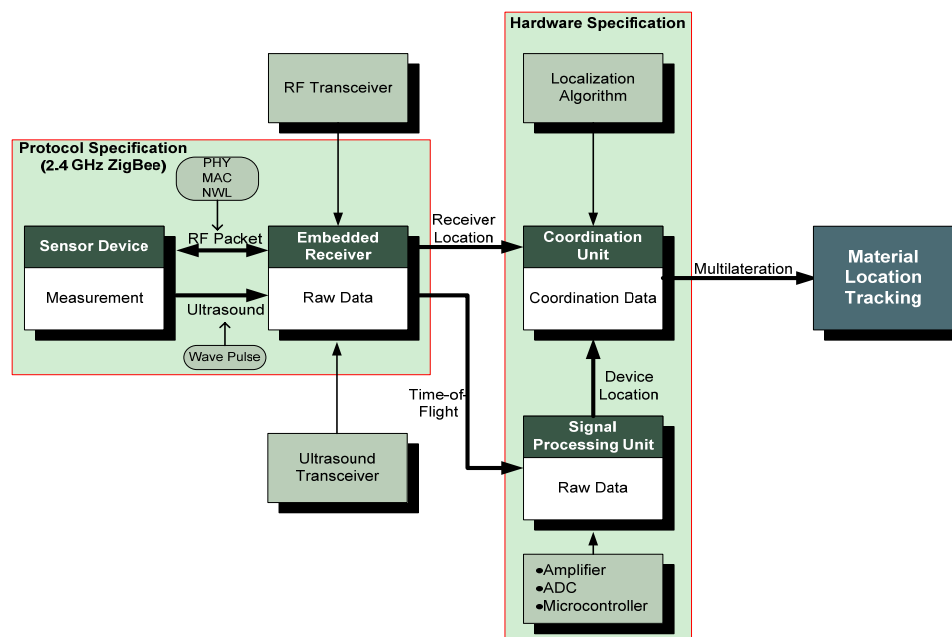


Figure 4.1: Sensor network block diagram – Example of embedded system for construction material tracking and monitoring

4.1.1 Coordinating

Coordinating is one of the most important tasks in wireless communication networks. To select the paths and to advertise the identity of a remote node (or sensor),

different coordinating algorithms impose over communication overheads. In the ZigBee networks, the remote nodes transmit the radio signal with 2.4GHz frequency and 250 kbps data rate within the coverage range of 10-100 m, and this specification of transmission determines the level of power and network topology to be communicated between remote nodes and beacon. Particularly, our investigation approach includes the localization of construction components in a large scale outdoor environment, so the allowable coverage range and power consumption rate would be the main issue for the reliable localization technique.

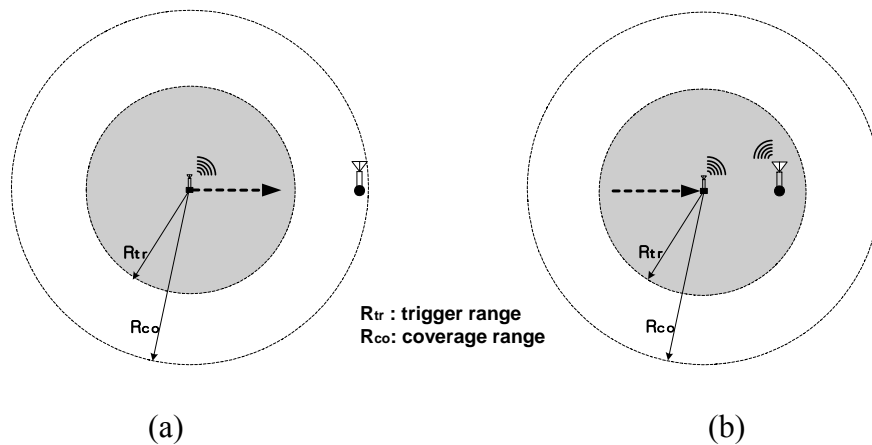


Figure 4.2: Coordinating scheme to trigger the query pulse for measuring the distance between beacon and remote node: (a) remote node is out of trigger range and (b) remote node is within trigger range

One approach to address this issue is to use the concept of *radio frequency indicator* (RFI) that advertises the identity of remote node by transmitting the device ID and null data packet to the fixed beacon. The schematic of the system is shown in figure 4.2. When a beacon receives a radio signal from a remote node, it switches to a ready mode (power save mode includes both ready mode and inactive mode) to trigger a query pulse for measuring the distance of the remote node. In order to discard the unwanted

low-strength signals that exit around the beacon, a level of threshold, trigger index (I_{tr}), needs to be determined with the ratio of SS_{tr} and SS_{co} , so it is assured that a level of signal strength for reliable communication can be obtained with increased duration of power save mode.

$$I_{tr} = \frac{SS_{tr}}{SS_{co}} \quad (4.1)$$

where, SS_{tr} and SS_{co} are the received signal strength within trigger range and coverage range, respectively.

If we consider additional functionality of remote nodes such as measurements of temperature, vibration, or chemical detection, as well as location, then the remote node might be equipped with sensing unit (MEMS sensor) inside the nodes, enabling transmission of sensing data between remote nodes and beacon. After a transceiver, designed and equipped inside the remote node, detects a particular query pulse from beacon, then it returns the response pulse to the beacon. At the same time, the transceiver starts transmitting the sensing data to the beacon. In addition, multiple numbers of remote nodes can advertise their IDs to a particular beacon in a way that the level of signal strengths received in a beacon differentiates each of the remote node's IDs by processing the pinging mechanism and the arrival time sequence.

A performance study and evaluation of RFI and trigger index under the different characteristics of geometries and complexities in construction site are subjects for further investigation of the proposed research.

4.1.2 Query and Response Pulse based on TOF

In the proposed research, data communication technique with localization of a remote node is a *combination of RF and US signal* on a different channel. While localization method based on a received signal strength index (RSSI) simplifies the device design in most case, the fluctuation of RSSI due to multipath and fading of the radio signal propagation often results in a poor accuracy. We propose an innovative method for eliminating the multipath property in signal propagation for measuring the distance. When a beacon confirms the RFI by coordinating scheme expressed previously, it emits a RF triggering signal to invoke a US signal for distance estimation of the remote node by time-of-flight method. After the beacon detects the first arrival of the US signal pulse, the traveling time of the round trip pulse enables to measure the distance between a beacon and a remote node by eliminating the undesirable multipath property of the signal. At the same time, the remote node starts to transmit the data messages to the beacon through radio signal packet recommended by IEEE 802.15.4 with 250kbs at 2.4 GHz frequency band illustrated in figure 4.3, immediately after it transmits the US signal to the original beacon.

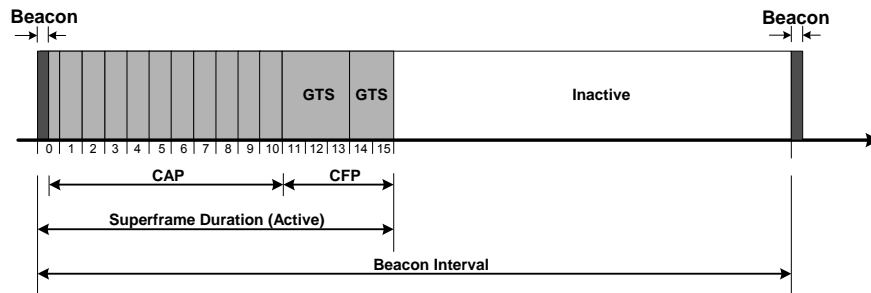


Figure 4.3: Superframe structure of the RF signal packet [Zheng and Lee, 2004].

We propose two alternative approaches for estimating the distance of a remote node from a beacon based on time-of-flight: one includes the use of *radio frequency signal* for travel time calculation; and the other scheme uses the combination of *ultrasound and radio signal* (figure 4.4). The radio signal as a trigger is designed to carry the unique characteristics of the original beacon's ID, and another RF signal is sent back to original beacon after a remote checks-in the RF trigger signal. On the other hand, combination of RF and US uses a 40 KHz ultrasound pulse as a means of time estimation. First arrival of the ultrasound response pulse to a beacon is used to eliminate the unwanted waves that may be reflected or refracted from the obstacles. Once the beacon recognizes the remote node's ID (by RF message packets) and distance (by US pulse), the geographic coordination of the remote node will be obtained by trilateration technique that uses three beacons as known reference points.

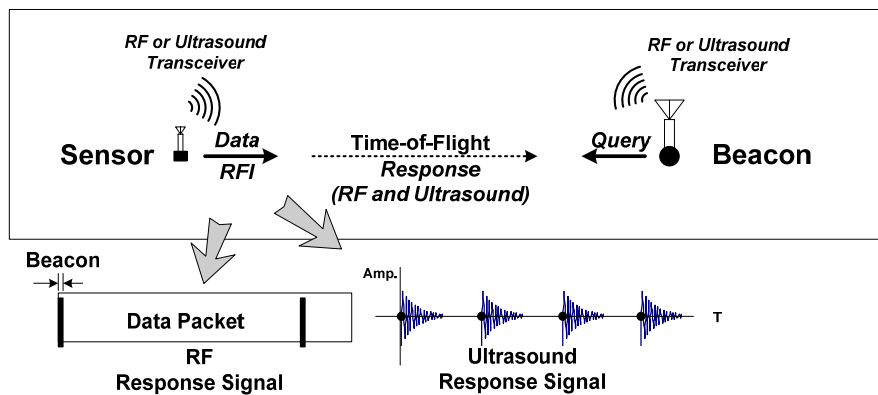


Figure 4.4: Description of time-of-flight utilizing the combination of RF and US signals

While both localization approaches are able to provide an innovative technique to eliminate the effect of multipath propagation of the radio frequency signal, it is noted that

two concerns about the localization error and the size of the additional device needs to be identified. The size of the RF-only device, e.g. MICA series modules, might be smaller than the device with combination of US and RF since additional ultrasound device needs to be installed, but the localization error caused by the instruction cycle of microprocessor should be considered because the traveling time of the radio signal at speed of light (3×10^8 m/s) is considerably smaller than the ultrasound pulse at speed of sound (340 m/s at 15°C). In the feasibility study, the investigation on the measurement error is performed to provide a feasible deployment of both methodologies.

4.2 Feasibility Analysis

In this section we consider two scenarios in our feasibility analysis. In the first scenario, all the signaling between the beacon and the remote node are performed by using RF signals.

Due to the fact that the RF signals travel at the speed of light, small delays in signaling and response of the remote nodes or the beacon may introduce error in the measurement of distance between a remote node and a beacon. In order to overcome this problem, we introduce a second scenario in which the signaling between the beacon and the remote node is performed by using RF signals, but measurement of distance is done through the ultrasound waves that travel at a lower speed and give a much higher accuracy in measurement of distance. We compare the advantages and disadvantages for each scenario.

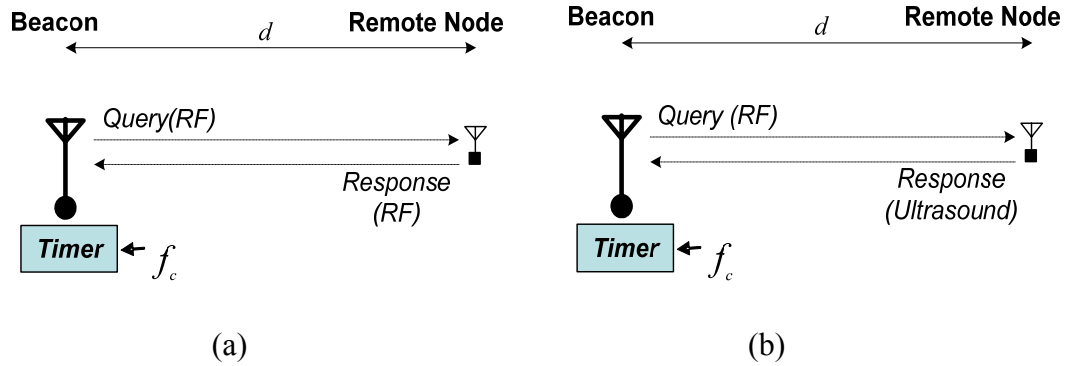


Figure 4.5: The schematic of the distance estimation from the beacon based on (a) the round trip time of the RF signal and (b) combination of RF and ultrasound signals.

4.2.1 Measurement of the Distance by RF Signals

In this method, we assume that the beacon is able to measure the distance of an object by measuring the round trip time of the RF signal from it to the remote node. A schematic of the distance measurement based on this method is shown in figure 4.5(a). As can be seen in this figure, for the purpose of finding the distance of a remote node from the beacon, the beacon sends a query packet containing the tag of the object of interest, and simultaneously, it starts a timer. Among all the remote nodes that receive this query, only the one matching the tag number on the query responds to the beacon. The beacon stops the timer as soon as receiving the response of the remote node of interest. The measured time in the timer gives the round trip time of the packet, and hence, the distance of the remote node from the beacon can be calculated by using this quantity.

In order to make a more detailed performance study of the above system, assume that it is desired to estimate the distance from the beacon with accuracy of 1 meter, and also assume that the maximum distance of remote nodes from the beacon is about 200 meters. As shown in figure 4.5(a), we assume that the timer is implemented by a counter

that counts up with a fixed clock frequency f_c . The required resolution of 1 meter means that the timer needs to have the resolution of 200 divisions, which means that it needs to have at least 8 bits. On the other hand, each increment happens in time $1 / f_c$ and accuracy of 1 meter is possible if $1 / f_c$ is smaller than the round trip time of the RF signal for a remote node in distance of 1 meter. This will lead us to the following equation:

$$f_c \geq \frac{3 \times 10^8 \text{ m/s}}{2 \times 1 \text{ m}} = 150 \text{ MHz} \quad (4.2)$$

The calculation in (4.2) shows that an increment of $1 / f_c = 6.6 \text{ ns}$ causes a 1 meter error in estimation of the distance from the beacon. One possible shortcoming is that if the undesirable errors and processing delays happen in scheduling of the sensors, it will cause increased error in measurement of the distance. All operations of a sensor are performed according to an on-board oscillator that generates timing of the internal processor of the sensor (figure 4.6). The frequency of such an oscillator is typically 10-100 MHz for current sensor technologies [MICA2DOT, 2006]. For example, if the clock frequency of the microprocessor module of a sensor is $f_p = 50 \text{ MHz}$, then this means that the remote node can have the worst case delay of $1 / f_p = 20 \text{ ns}$ in responding the query of the beacon after receiving it. This is because the fact that if the micorprocessor's operations can only be triggered on the rising and falling edges of the digital clock generated based on the oscillator frequency. A delay of 20 ns can generate up to 3 meters error in measurement of distance.

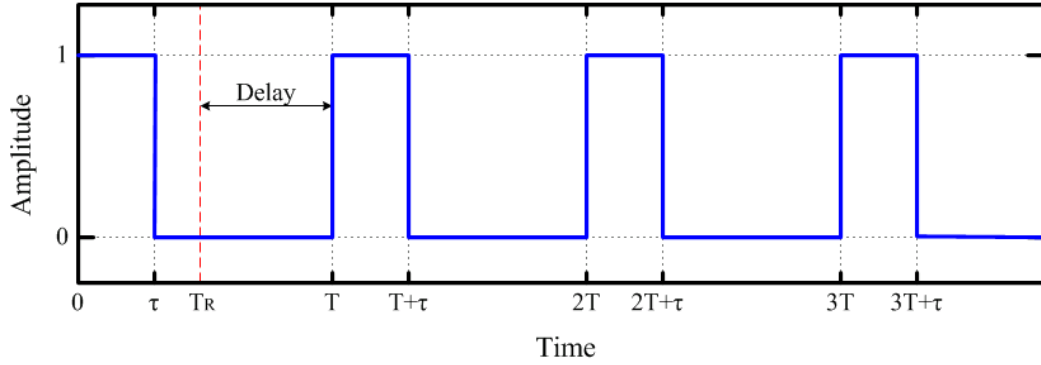


Figure 4.6: Operation delay in digital clock cycle when a signal is received at time T_R

The error in estimation of distance of a remote node from the beacon based on the measurements of RF round trip time delay is a result of very fast traveling speed of RF signals in the space. In order to resolve this issue, we study a second scenario, in which an alternative ultrasound signaling is used to measure distance of remote nodes to the beacon.

4.2.2 Measurement of the Distance by combination of RF and Ultrasound Signals

In this scheme, we use a method similar to the method introduced in previous section; however, we introduce a second signaling mode based on ultrasound for a more accurate measurement of distance, shown in figure 4.5(b). In this case, estimation of the distance is similar to the previous case. The beacon sends a query with the tag of a remote node of interest and at the same time starts a timer. Among all the remote nodes that receive the query of the beacon, only the one that matches the tag on the query responds by sending an ultrasound response back. There are two major differences in this case with the previous one: the first difference is that the speed of ultrasound signals

is in the about 340 *m/s* which is significantly smaller than the speed of light, therefore, small delays introduced by scheduling of a remote node do not cause an error in estimation of distance. The second difference from the scheme in previous section is that here the ultrasound signal does not carry in digital information, and it does not have any form of modulation. Note that in this case, a remote node needs to send an ultrasound signal within a short period of time after observing a query of the beacon containing its tag, and no digital information is needed to be exchanged in the backward direction from the remote node to the beacon.

In order to perform similar calculation to those in previous section, and under similar assumption of measurement of distance up to 200 meters with the accuracy of 1 meter, we need the timer to be an 8 bit timer with the following clock frequency:

$$f_c \geq \frac{340m/s}{1m} = 340Hz \quad (4.3)$$

Note that in this case the signal travels with the speed of light in the forward direction and with the speed of ultrasound in the backward direction. Since the speed of light is about one million times faster than the speed of ultrasound, we ignore the component of the delay introduced by the propagation delay of RF signal in the forward direction, or the small processing or scheduling delay at the remote node. Therefore the dominant component of the round trip delay of the signal is the time of traveling the ultrasound signal from the remote nodes to the source.

It is useful to note that in this case, a 1 meter increase in the distance of the remote node from the beacon results about 3 *ms* increment in the value of round trip time.

This increment is significantly larger than the typical travel time of RF between the beacon and the remote node (about $1.3 \mu s$ for maximum distance of 200 meters), or the typical processing and scheduling delays at the microprocessor of remote node (typically in order of 10-100 ns). Therefore, we predict that the estimation of error based on combination of RF and ultrasound waves to give a high performance and a very small error in estimation of the distance of remote nodes from the beacon.

A final issue in regard to the system based on combination of RF and ultrasound signals is dependence of the speed of ultrasound signals to the ambient parameters such as temperature and pressure. Such dependences can be taken into consideration by integrating proper sensors that are integrated to the beacon and make compensation for the changes in the environmental parameters.

4.3 Simulation on the Position Estimation

This section examines the numerical simulation based on the feasibility study performed in previous. Brief background of trilateration is introduced to determine the unknown position of remote nodes from three reference points. For accuracy simulation, tracking algorithm is applied to illustrate the observed trace of a remote node that is set to move along with the square path. In order to compare the accuracy, we introduce the Kalman filter that is used for an efficient recursive means to estimate the state of a positioning process. Finally, comparisons between RF-only method and RF+US method are examined to justify the feasibility study performed in previous section.

4.3.1 Trilateration Technique

Locating a remote node, ranging the measurements, or estimating the coordination is a common operation in the object tracking systems. This is often known as trilateration method in which one can determine the position of a remote node from three beacons located at know reference points, shown in figure 4.7. Traditionally, trilateration problem has been solved by algebraic or numerical methods and been deployed to common operation in robot ranging, aeronautics, or GPS applications. If we assume that the positions of the n beacons $\mathbf{x}_i=(x_i, y_i, z_i)$, $i=1 \dots n$ and the distances d_i , $i=1 \dots n$ between them, and the unknown location $\mathbf{x}=(x, y, z)$, then trivial solution can be found by estimating the intersection of n-spheres following the quadratic equation systems:

$$\begin{aligned}
 (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 &= d_1^2 \\
 (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 &= d_2^2 \\
 &\vdots \\
 (x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2 &= d_n^2
 \end{aligned} \tag{4.4}$$

or

$$\begin{bmatrix}
 (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 \\
 (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 \\
 \vdots \\
 (x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2
 \end{bmatrix} = \begin{bmatrix}
 d_1^2 \\
 d_2^2 \\
 \vdots \\
 d_n^2
 \end{bmatrix} \tag{4.5}$$

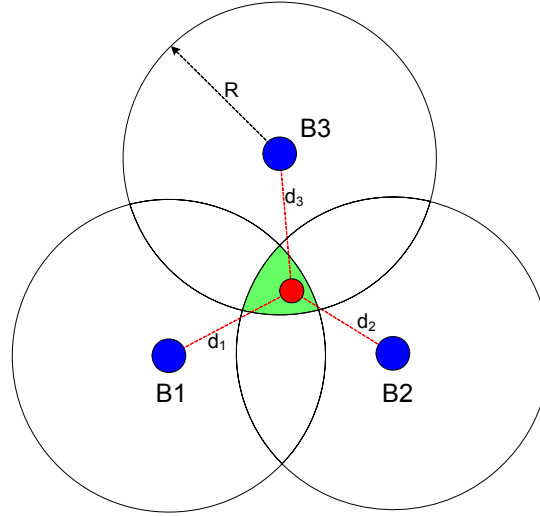


Figure 4.7: Illustration of trilateration method that consists of the location a mobile remote node from its distance to three fixed points

The quadratic form of expression intersecting n-spheres can be easily simplified to the intersection of a line and a sphere, and the system equation can be expressed by two linear equations and a quadratic equation. If we compensate obtaining the z-coordination, then 2-D coordination system finding x and y coordinates can be expressed by (n-1) linear equations:

$$A \begin{bmatrix} x \\ y \end{bmatrix} = B \quad (4.6)$$

where,

$$A = \begin{bmatrix} (x_1 - x_n) & (y_1 - y_n) \\ (x_2 - x_n) & (y_2 - y_n) \\ \vdots & \vdots \\ (x_{n-1} - x_n) & (y_{n-1} - y_n) \end{bmatrix}, \quad B = \frac{1}{2} \begin{bmatrix} -d_1^2 + d_n^2 + x_1^2 - x_n^2 + y_1^2 - y_n^2 \\ -d_2^2 + d_n^2 + x_2^2 - x_n^2 + y_2^2 - y_n^2 \\ \vdots \\ -d_{n-1}^2 + d_n^2 + x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 \end{bmatrix}$$

The result of (n-1) linear equations then becomes a least square system (equation 4.6) that can be used to estimate the position $\mathbf{x}(x, y)$ using the following expression:

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = (A^T A)^{-1} A^T B \quad (4.7)$$

The performance simulation is achieved in the form of the least square system, and MATLAB solver to obtain the solution of the position is used in next section. It is important to note that at least three reference points are needed to avoid ambiguity about the unknown location, and the position accuracy can be improved if one uses multiple reference points.

4.3.2 Kalman Filter

R.E. Kalman published a famous paper contributing a recursive solution to discrete-data linear filtering problem [Kalman, 1960]. By specifying a linear dynamic system, he accomplished the prediction, separation, or detection of a random signal. Kalman filter is an efficient recursive filter that estimates the state of a dynamic system from a series of incomplete and noisy measurements. The algorithm provides accurate continuously-updated information about the position and velocity of an object given only a sequence of observations about its position, each of which includes some error.

The Kalman filter addresses the estimation of the state $x \in R^n$ of a discrete-time controlled process governed by the linear stochastic difference equation [Welch and Bishop, 1995]. Assumes that true state at time k is evolved from the state at (k-1) according to

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (4.8)$$

Then, observation z of the true state x can be expressed below:

$$z_k = Hx_k + v_k \quad (4.9)$$

where,

A: the state transition model applied to previous state x_{k-1}

B: the control-input model applied to control vector u_{k-1}

w : the process noise drawn from a zero mean multivariate normal distribution with covariance Q

H: the observation model which maps the true state into the observed space

v : the observation noise assumed to be zero mean Gaussian white noise with covariance R

The random variables w_k and v_k explains the process and measurement noise, respectively, and they are assumed to be independent each other with normal probability distributions:

$$\begin{aligned} P(w) &\sim N(0, Q) \\ P(v) &\sim N(0, R) \end{aligned} \quad (4.10)$$

If we define *a priori* state estimate as $\hat{x}_k^- \in R^n$ at step k with the given knowledge of the process prior to step k , and define *a posteriori* state estimate as $\hat{x}_k \in R^n$ at step k given measurement z_k , then *a priori* estimate error covariance and *a posteriori* estimate error covariance can be expressed by:

$$\begin{aligned} P_k^- &= E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T] = E[e_k^- e_k^{-T}] \\ P_k &= E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] = E[e_k e_k] \end{aligned} \quad (4.11)$$

where, *a priori* and *a posteriori* estimate errors can be defined as:

$$\begin{aligned} e_k^- &\equiv x_k - \hat{x}_k^- \\ e_k &\equiv x_k - \hat{x}_k \end{aligned} \quad (4.12)$$

In order to derive the equation for the Kalman filter, *a posteriori* state estimate \hat{x}_k is computed as a linear combination of an *a priori* estimate \hat{x}_k^- and a weighted difference between an actual measurement z_k and a measurement prediction $H\hat{x}_k^-$ as shown by Equation 4.13:

$$\hat{x}_k = \hat{x}_k^- - K(z_k - H\hat{x}_k^-) \quad (4.13)$$

In Equation 4.13, the difference $(z_k - H\hat{x}_k^-)$ is defined as the residual that reflects the variation between the predicted measurement $H\hat{x}_k^-$ and the actual measurement z_k .

The matrix K is defined as the gain factor that minimizes the *a posteriori* error covariance, and can be expressed by Equation 4.14:

$$\begin{aligned} K_k &= P_k^- H^T (H P_k^- H^T + R)^{-1} \\ &= \frac{P_k^- H^T}{H P_k^- H^T + R} \end{aligned} \quad (4.14)$$

The Kalman filter has two distinct phases: Predict and Correct, shown in figure 4.8. The predict phase uses the state estimate from the previous time step to produce an estimate of the state at the current time step. In the correct phase, measurement information at the current time step is used to refine this prediction to arrive at a new, more accurate state estimate, again for the current time step. Calculating the measurement noise covariance R is usually practical and some off-line sample measurements are used to determine the variance of the measurement noise. Sometimes, it is much more difficult to determine the process noise covariance Q because direct observation of the estimating process is typically impossible, so a relatively simple process by injecting uncertainty into the process by selecting Q may generate acceptable results.

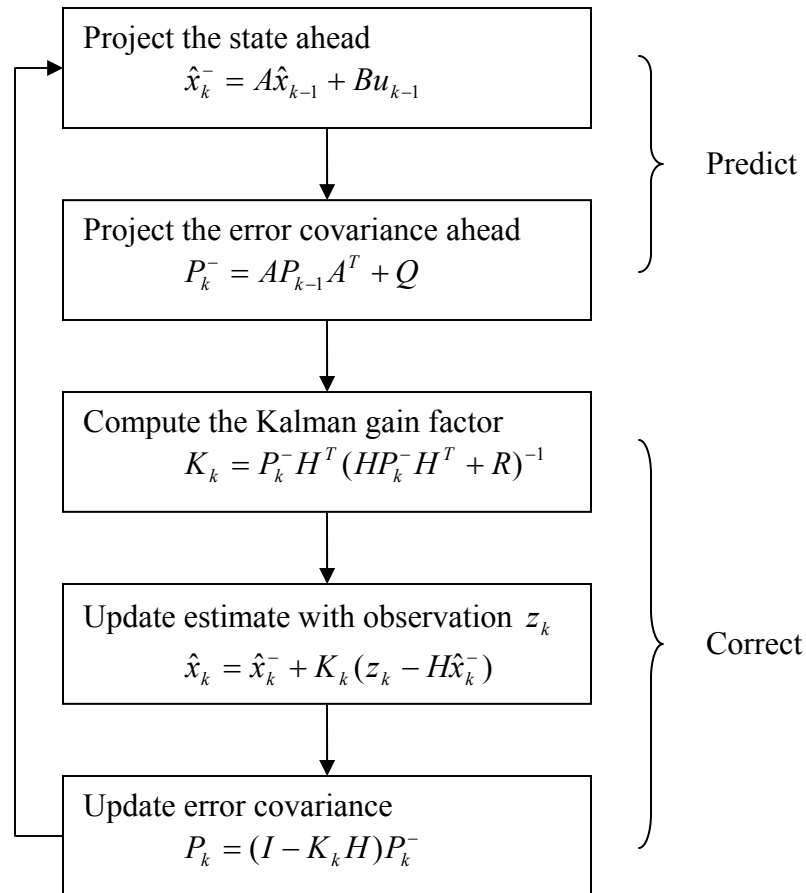


Figure 4.8: Discrete Kalman filter cycle: two phases of Predict and Correct [Welch and Bishop, 1995]

Kalman does not consider the important problem of smoothing: The filtering and prediction solution allows one to estimate current and future values of the variables of interest [Rauch et al., 1965]. By permitting to estimate the past values, Rauch et al. put forward Rauch-Tung-Striebel (R-T-S) fixed-interval optimal smoothing algorithm in 1965. R-T-S smoothing algorithm is a backward-pass recursion to a forward-pass Kalman filter, and makes use of Kalman filter output. R-T-S smoother is often applied because the simplicity of the formulas related to the smoothing problem.

4.3.3 Feasibility Simulation for Accuracy Performance

This section elaborates the performance simulation with respect to the measurement accuracy based on the feasibility study achieved in the previous section. The multipath effect is avoided by utilizing the first-arrival signal as a signal detection scheme. Also, the measuring cycle is sufficient to avoid any interference with the internal signal processing time. More importantly, it is assumed that coverage range in a both radio and ultrasound signal can be up to 100 meters for this simulation. This is often not the case with practical application without first improving the circuit and power supply, and extensive efforts should be put on the design of the ultrasound devices by increasing the transmission power and higher signal-to-noise ratio to achieve this assumption. However, it is worth pursuing the feasibility simulation with such an assumption because this simulation describes the accuracy performance based on the microprocessor's resolution. Examining the instruction cycle operation within the microprocessor can provide an understanding of how much the operation affects the measurement accuracy when comparing the utilization of a radio signal with that of an ultrasound signal as a distance estimator.

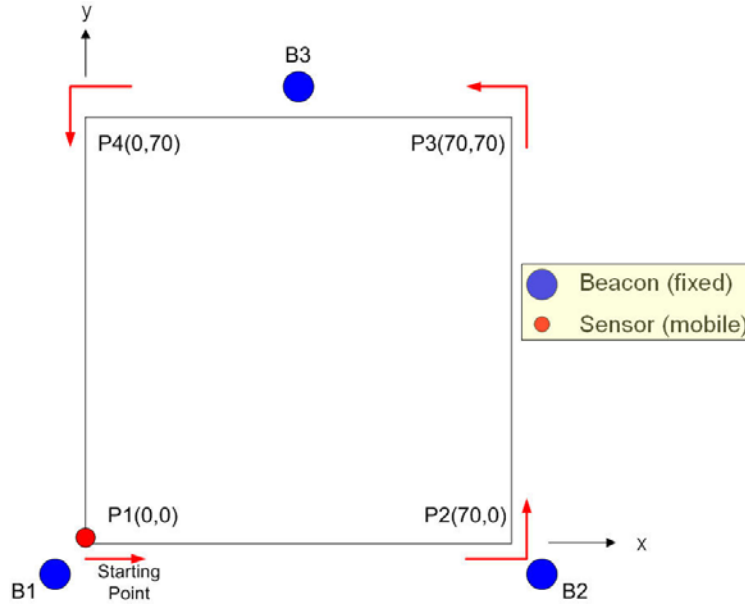


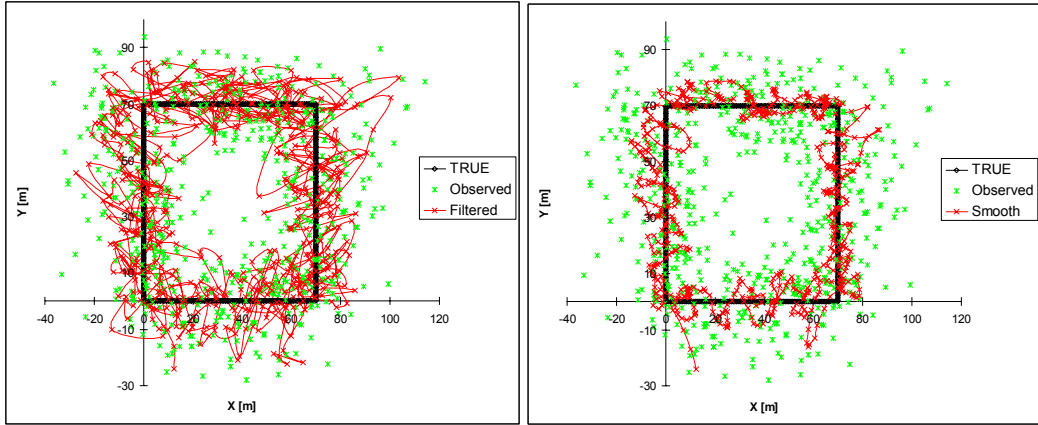
Figure 4.9: System configuration for feasibility simulation

Figure 4.9 describes the simulation configuration where three fixed beacons are located at each vertex point of a triangle, and remote node is set to move along with the square-shape path of 70-by-70 meters. Measured samples at the rate of the microprocessor's instruction cycle are then compared with the two different tracking schemes, e.g. RF-only and RF+US. In addition, a Kalman filter and a smoothed Kalman filter are applied to obtain the improved trajectory tracking using a Kalman filter toolkit developed from the Computer Science and Artificial Intelligence Lab. in MIT [Murphy, 1998]. During simulation, three clock frequencies of microprocessor, e.g. 8, 25 and 50 MHz, are selected, and travel speed is set at 3×10^8 m/s for radio signal propagation and 340 m/s for ultrasound signal propagation. A mobile sensor moves along the square path at the speed of 1 m/s, and the sampling cycle of estimating each position is set at 500 ms. The table 4.1 illustrates the summary of the parameters used for the feasibility simulation.

Table 4.1: Summary of parameters used for feasibility simulation

| Parameter | RF-only Scheme | RF+US scheme |
|-----------------------|-----------------|--------------|
| Clock Frequency [MHz] | 8, 25, 50 | 8, 25, 50 |
| Travel Speed [m/s] | 3×10^8 | 340 |
| Sampling Cycle [ms] | 500 | 500 |
| Object Speed [m/s] | 1 | 1 |

After executing the Kalman filter toolbox, it is observed that the accuracy of the position depends significantly on the tracking method used and clock frequency cycle in the microprocessor. Especially, a huge improvement in accuracy is observed when a RF+US scheme is used in comparison to a RF-only scheme. Notably, measurements on a RF-only scheme shows poor accuracy even when a Kalman filter is applied, and the accuracy declines as the lower clock frequency is selected (figure 4.10 to 4.15). The error variation obtained from the first case, at 8 MHz with a RF-only scheme, shows error variation up to 40 meters from the actual path, and extremely sparse measured points was observed. Even if the clock frequency is increased to 50 MHz, the estimated error variation indicated up to 7 meters. On the other hand, if the combination scheme of radio and ultrasound is considered, enhancement to measurement accuracy was observed with much less fluctuation (figure 4.16). For example, estimated position trace with RF+US shows very accurate positioning results regardless of a change in clock frequency, and error variation indicated around 60 cm maximum in the selection of a three clock frequency.



(a) With Kalman filter

(b) With Smoothed Kalman filter

Figure 4.10: Tracking trajectory at 8 MHz clock cycle (RF-only scheme)

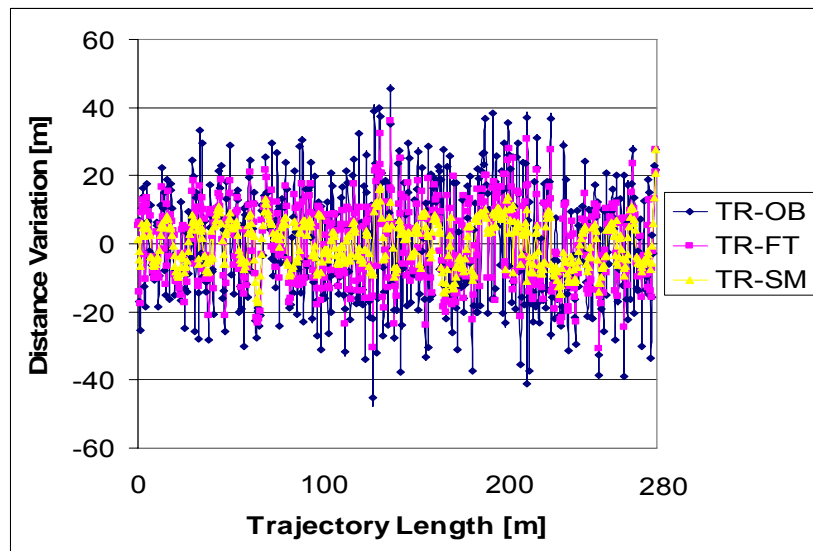
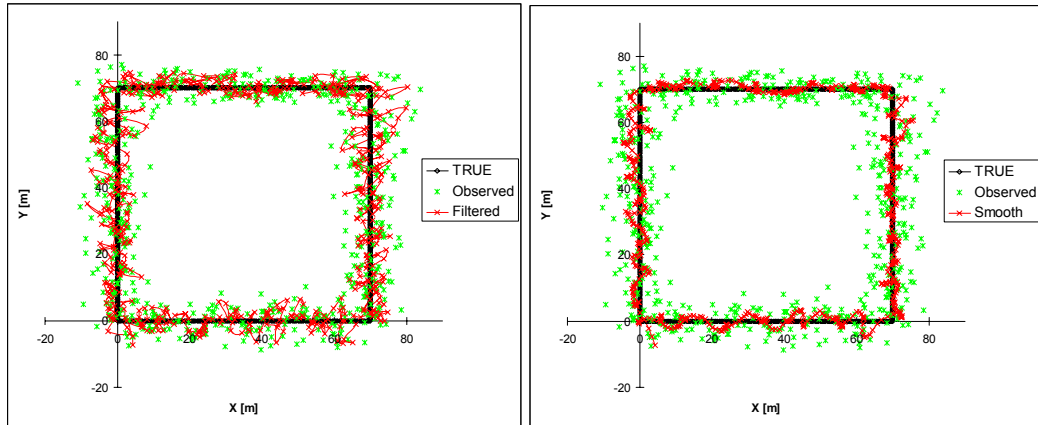


Figure 4.11: Measured distance variation at 8 MHz (RF-only Scheme)



(a) With Kalman filter

(b) With Smoothed Kalman filter

Figure 4.12: Tracking trajectory at 25 MHz clock cycle (RF-only scheme)

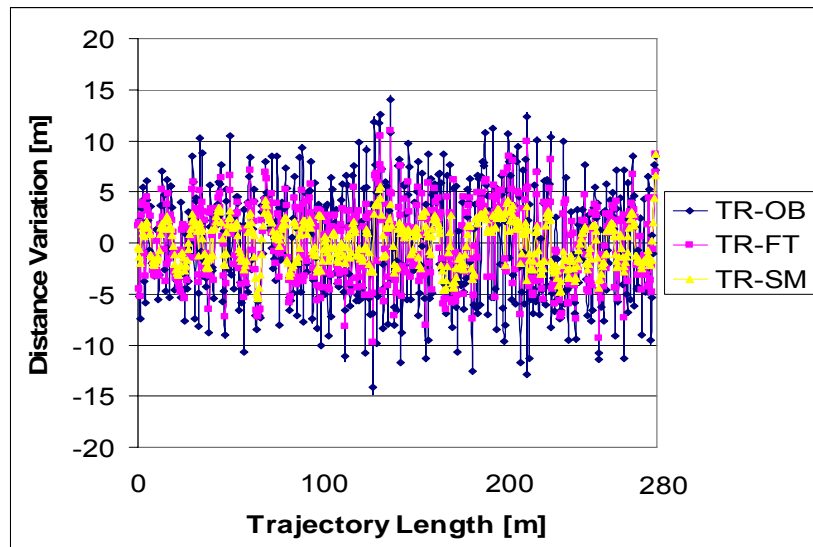
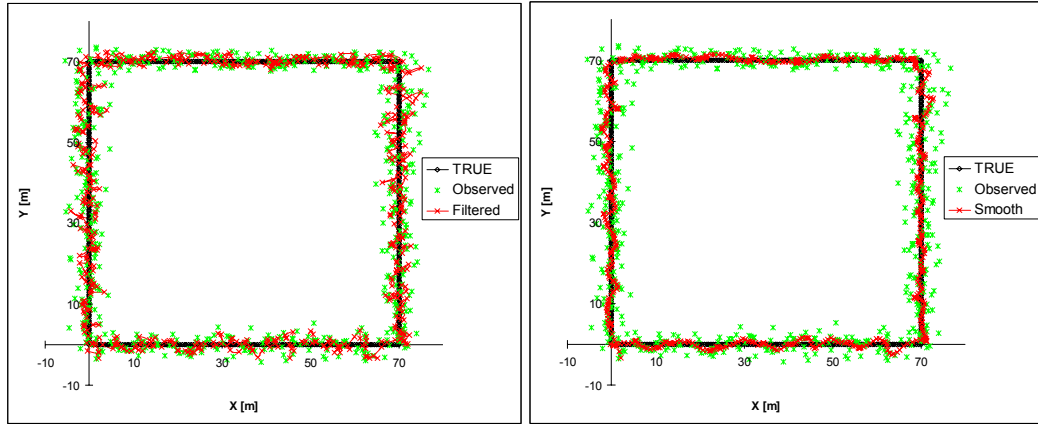


Figure 4.13: Measured distance variation at 25 MHz (RF-only Scheme)



(a) With Kalman filter

(b) With Smoothed Kalman filter

Figure 4.14: Tracking trajectory at 50 MHz clock cycle (RF-only scheme)

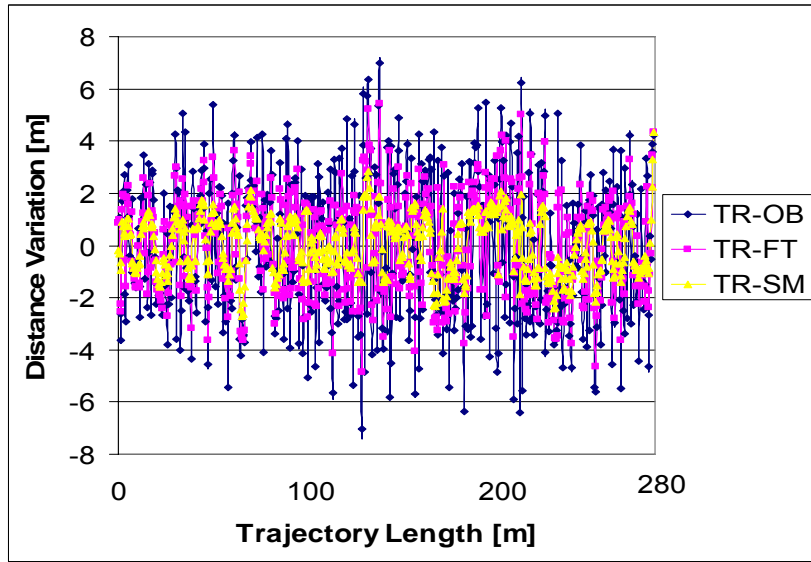
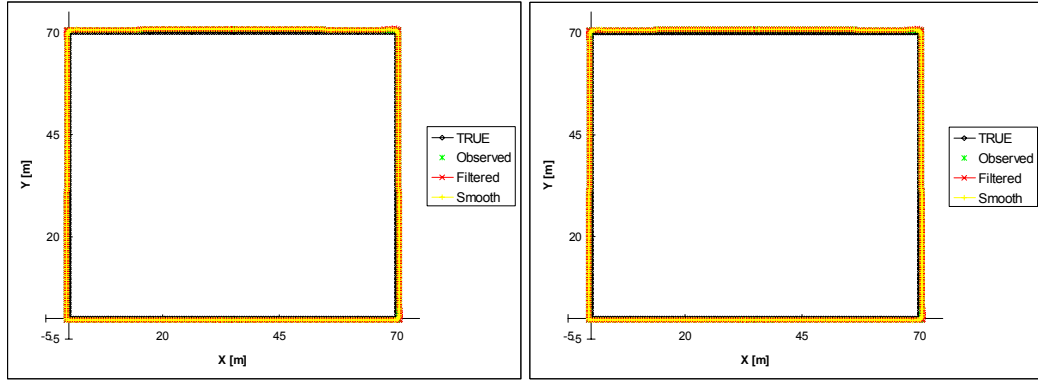
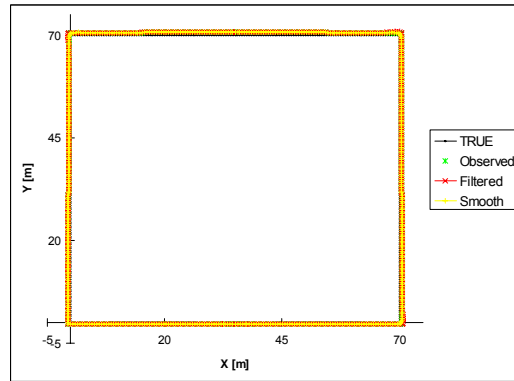


Figure 4.15: Measured distance variation at 50 MHz (RF-only Scheme)



(a) At 8 MHz

(b) At 25 MHz

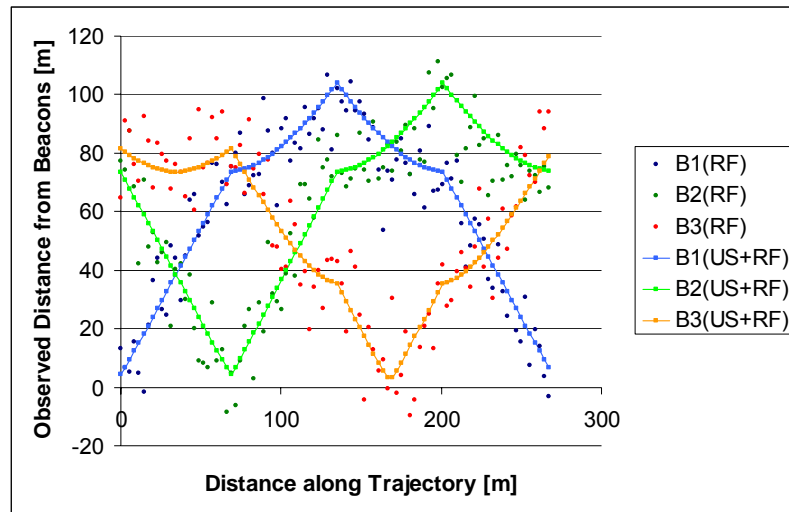


(c) At 50 MHz

Figure 4.16: Tracking trajectory (RF+US scheme)

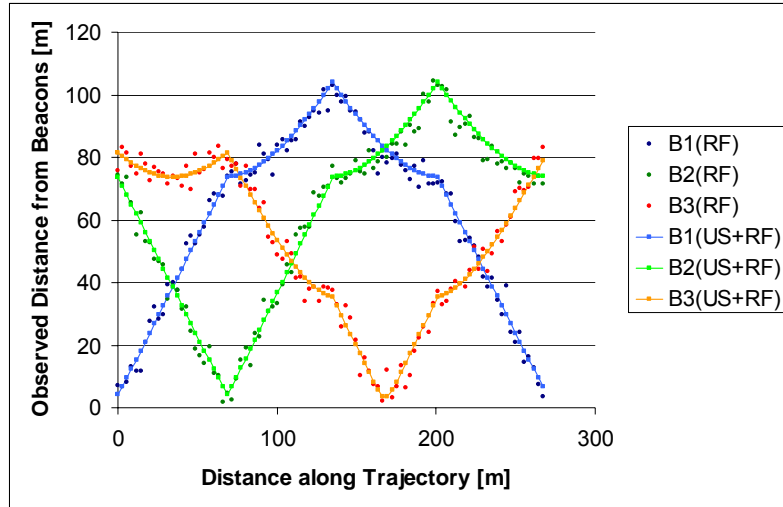
Figure 4.17 illustrates the observed distance measurement from individual beacon showing accuracy performance in single distance estimation, and the measurement variation follows the similar pattern observed in the position estimation, i.e. lower clock frequency decreases the accuracy and RF+US indicates much better results. The individual measurement is also important to examine because, in the trilateration method, the individual distance used in the least square system is assumed to have the exact values of distance from a remote node to beacons. Poor distance measurements in the

single distance estimation could consequently result in worse position estimation. Nevertheless, this requirement in calculating each distance does not always cause the worse position estimation. For example, the mechanism of least square system infers that 10% shorter distances actually measured from three beacons can result in the same coordinates as the exact ones because the least square system prorates the solution according to the measured distances unless three intersections are found. As a consequence, a random process with more than 500 sampling measurements could somewhat mitigate the error variation in position estimation, but the single distance estimations could deliver an explicit bias in determining the coordinates of an unknown object.

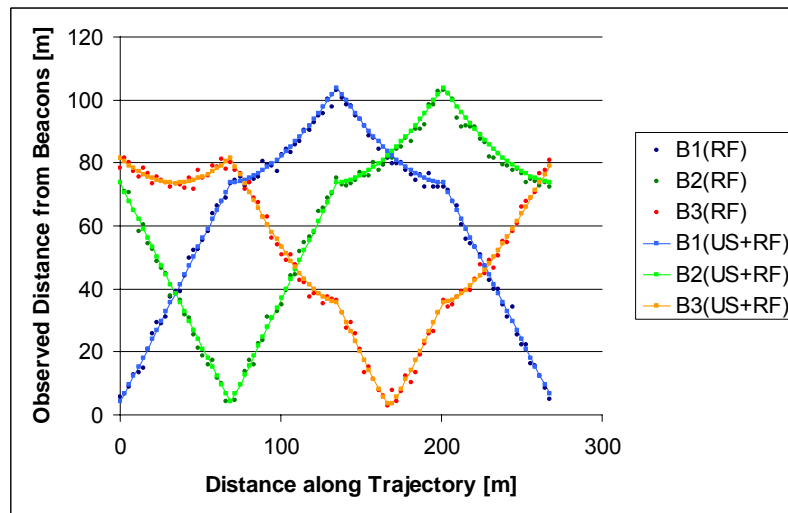


(a) At 8 MHz

Figure 4.17: Observed distance measurement from three beacons



(b) At 25 MHz



(c) At 50 MHz

Figure 4.17: Observed distance measurement from three beacons (cont.)

Percentile error variation provides a comprehensive illustration of the measurement distribution over the ranked order. The combination scheme using radio and ultrasound signal presented in this dissertation results in relatively high accuracy

ranges in the order of few tens centimeters. However, if a RF-only scheme is considered, 75 percentile errors measured at 8 MHz clock frequency increase about 20 meters. It is interesting to mention that the root-mean-square (RMS) values often provide a good reference in the situation where the variates are marked positive and negative from the exact values. Thus, randomly varying quantities in position estimation can be expressed in terms of positive magnitude, providing more inclusive representation of variance than arithmetic means. In figure 4.18, RMS values measured from the combination of RF and US indicates 58.6 cm in all range of clock frequencies. However, a RF-only scheme results in 2.8, 5.8, and 17.4 meters at 50, 25, and 8 MHz clock frequencies, respectively. If the clock frequency decreases to several hundreds of KHz, one cannot justify the rationale of deploying the RF-only scheme in distance and position estimation.

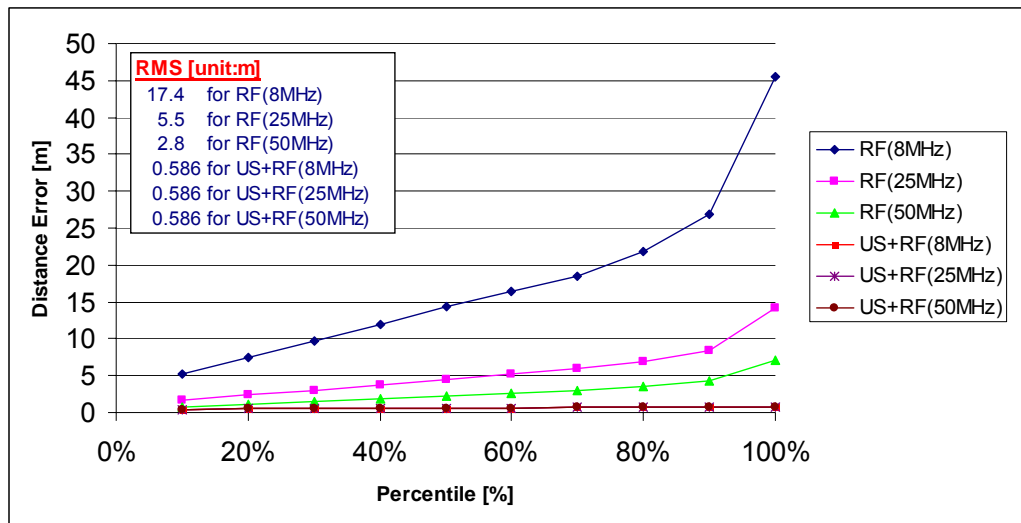


Figure 4.18: Percentile of position errors

Overall, a distance and position scheme combined with a radio and ultrasound signal provides better accuracy performance than a RF-only scheme, even though the

distance ranges up to 100 meters between beacons and a remote node. This outcome gives an implication to motivate the design of hardware and software, encouraging us to implement the prototype experiments based on the feasibility study proposed in this dissertation. As a dominant component in time calculation, the nature of ultrasound speed gives a better probability to reduce the signal delay associated with the clock cycle of a microprocessor. However, it should be noted that the actual design of hardware and software may face a challenge to be implemented because of the assumption made in the feasibility study. For example, it is not easy to find a 100-meter measuring range with current ultrasound device technology even if a one-way communication scheme is applied. Another example may include the clock cycle of current microprocessor. Most of today's cutting-edge technologies in wireless sensor modules feature clock cycles with 8 MHz, e.g. Tmote Sky [Moteiv Corporation], and the device cost will rise with higher clock frequency. This fact may cause an implementation challenge in many practical applications. In addition to the microprocessor, several technical challenges can arise, including: high quality ADC device, high signal-to-noise ratio (SNR), noise filtering techniques, improved signal detection algorithm, multi-communication scheme, and line-of-sight. Detailed explanation of these technical challenges will be examined in later implementation sections.

4.4 Scheduling Scheme

This section presents a prototype of the combination scheme using RF and US for tracking and monitoring the construction materials. Real-time tracking and monitoring of construction components is an important task for understanding the ubiquitous computing

environment, such as querying location information, communicating messages, and spontaneous inquiry of any relevant action. With the underlying tracking mechanism introduced in the previous chapter, it is a challenge to schedule a beacon signal transmitted from/to the distributed remote nodes in the form of a radio frequency and ultrasound. This is because multiple remote nodes are trying to communicate with one or multiple beacons at the same time, as long as they are in the communication range. If a RF and ultrasound signal are not properly scheduled, two types of collision may be issued: 1) if there is a overlap between ultrasound signals from adjacent remote nodes, a beacon cannot measure the distance of each remote node, hence the overlapping ultrasound signals may be considered garbage; and 2) if RF messages between adjacent beacons overlap, a relay mechanism in multi hop topology among beacons cannot be reliable. In order to prevent any collision among multiple remote nodes while minimizing the process delay and ensuring the best performance of real-time tracking, a noble mechanism of scheduling between the beacon and remote node should be examined.

4.4.1 System Overview

If we apply the automated tracking and monitoring architecture to construction sites, it is necessary to provide the entire system configuration of the particular construction environment. Let us first describe the placement of a beacon which is responsible to collect information, such as sensory data and ultrasound signals from distributed remote nodes. A number of beacons (B_i) are placed on top of a light pole on a construction site in a way that directional RF and ultrasound antenna is mounted in the beacon to cover a certain range of the construction site. A higher location allows the

beacons to communicate with remote nodes because this allows for the increase possibility of a clear line of sight, much like a satellite with a GPS. A number of sensor devices (S_j) that include the function of ultrasound and RF transmission are attached to construction materials to support the distance measuring and monitoring capability.

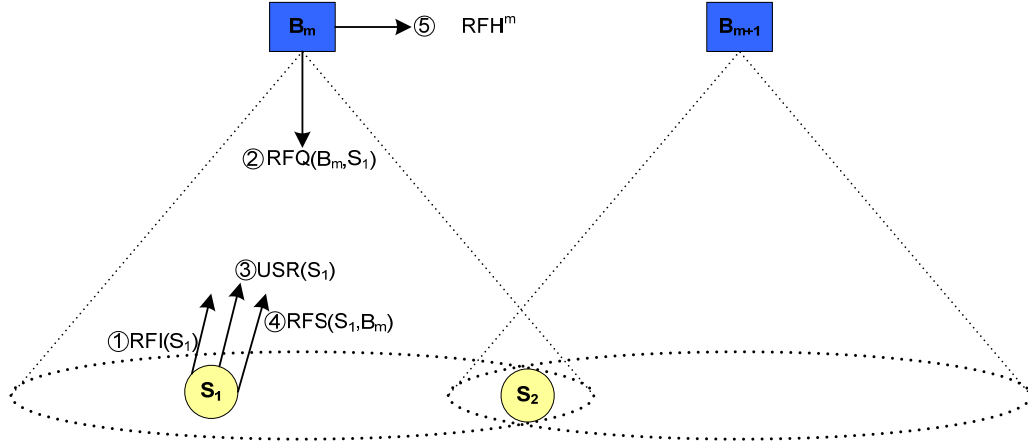


Figure 4.19: System configuration of beacon scheduling

If we define the total number of beacons and remote node as B_N and S_N , respectively, then each beacon can be denoted as B_i ($1 \leq i \leq B_N$) and remote node S_j ($1 \leq j \leq S_N$). Figure 4.19 shows that each remote node S_j , with its own ID, consists of a RF transceiver and ultrasound transmitter as well as a sensor device, if any. The initial scheduling mechanism starts with each remote node that transmits radio frequency indicator (RFI). RFI advertises the sensor's ID by radio signal, and once the beacon receives the RFI then the beacon starts transmitting command signals to the sensor, which is discussed in a previous section. In this mechanism, we assume that the actual scheduling mechanism starts when a beacon receives a RFI from certain sensors because the beacon does not have the actual starting time of RFI. In addition, the RFI signal acts

as a trigger to a beacon so that a ready mode in the beacon is to be set from idle mode for measuring the distance from the beacon and remote node, and this is the correct scheduling time of the beacon side.

Beacon B_i starts sending a query pulse $RFQ(B_i, S_j)$ to the sensor S_j from which the specific RFI came and the actual beacon scheduling is initiated immediately after the receipt of the RFI. Among sensors distributed in the geographical domain, only the sensors that receive the RFQ response to the RFQ because some of the sensors are located far away from the coverage range of beacon B_i . The $RFQ(B_i, S_j)$ contains two different messages – such as query pulse to the sensor from which RFI has been sent for measuring distance and a command message for sensory data. When $RFQ(B_i, S_j)$ is received in sensor S_j , the sensor S_j generates an ultrasound response pulse and starts sending an ultrasound signal $USR(S_j)$ to the beacon B_i . At the same time, sensor S_j transmits RF-based sensory data $RFS(S_j, B_i)$ to beacon B_i . Since the ultrasound signal $USR(S_j)$ is a physical waveform propagated through air media, any meaningful information cannot be modulated into $USR(S_j)$. Instead, RF-based sensory data $RFS(S_j, B_i)$ will include ultrasound information such as: sensor's ID, sending time, and ultrasound signal shape and frequency.

If multiple sensors are trying to respond to a query pulse RFQ that comes from beacon B_i , a frequency division multiple access (FDMA) scheme will be used to differentiate multiple ultrasound signals for estimating the distance of each sensor. Once beacon B_i receives both an $USR(S_j)$ and a $RFS(S_j, B_i)$, it starts calculating the distance by transforming the travel time of the signal then transmits a report message RFH^k that includes both sensory data and distance information to the next hop beacon (B_k). By

relaying report message RFH in the form of a multi hop network, the host computer gathers all the relay messages of the beacons. The host computer translates the distance measurements, filtering out any noise by using a filter tracking algorithm, e.g., Kalman filter, and estimates the sensor's geographical coordinations by triangulation. The above notation and definition used in the beacon scheduling is summarized in table 4.2.

Table 4.2: Variable notation summary

| Notation | Definition | From | To | Data |
|-------------------|---------------------------|--------------|------------------|--------------------------------|
| RFI(S_j) | RF indicator | Sensor S_j | Beacons | Sensor's ID |
| RFQ(B_i, S_j) | RF query pulse | Beacon B_i | Sensor S_j | Sensor's ID, command packet |
| USR(S_j) | Ultrasound response pulse | Sensor S_j | Beacons | N/A |
| RFS(S_j, B_i) | RF sensory data | Sensor S_j | Beacon B_i | Sensory data, US information |
| RFH ^m | RF relay messages | Beacon B_m | Beacon B_{m+1} | Relay messages of beacon B_m |

Each of the signal notations described in table 1 represents the total duration from the sending and receiving points, and the latency of the signal communication varies over time and device. Each signal duration has four major times [Römer, 2005]:

1) Send time

This is the delay time when the device issues a “send” command to the operating system. A raw network message in digital form is constructed in every instruction cycle of the processor, and then is delivered to the communication medium. Physical delay factors of send time include: oscillator's frequency, operation of switches, and possible hardware interrupts.

2) Medium access time

Typically, medium access time is required to allocate multiple network nodes in the MAC layer. Hence, immediate access may not be possible and each command message waits for the access sequence, causing additional variable delay.

3) Propagation time

Different signal forms have their own propagation speed from sender to receiver throughout the propagation medium. The high propagation speed of radio frequency signals may not be a problem, but the low speed of ultrasound propagation may affect the communication scheduling algorithm.

4) Receiving time

This is the delay time from a receiver antenna to the point where the device can actually recognize the arrival of the message. Similar delay factors, such as processor's instruction cycle, switches, and hardware interrupts, need to be considered.

4.4.2 Single Communication

Beacon scheduling in single communication is quite simple. A beacon is scheduled with only one remote node, so we only consider one cycle time of scanning during which the beacon can process with one cycle an ultrasound signal and a RF signal for distance estimation and data communications, respectively. The sequence of a signal process is shown in figure 4.20. Beacon B_m starts transmitting a RF query pulse $RFQ(B_m, S_1)$ to sensor S_1 immediately after it receives a $RFI(S_1)$. At this step, it recognizes that the $RFI(S_1)$ is not involved in the schedule of the beacon B_m because RFI is limited to

triggering the RF query pulse $RFQ(B_m, S_1)$. Thus, we assume that the RFI time duration is not considered in the schedule of beacon B_m .

After a sensor S_1 receives the RF query, sensor S_1 transmits both an ultrasound signal $USR(S_1)$ and RF sensor data $RFS(S_1, B_m)$ to beacon B_m . Once the beacon B_m receives an ultrasound signal, the beacon is scheduled to transmit the RF relay messages to the next adjacent beacon B_{m+1} for multi hop. It is worth noting that since there is a big difference in propagation time between ultrasound and RF signals, the arrival time of $USR(S_1)$ and $RFS(S_1, B_m)$ has a time gap. In order to make the scheduling efficient, we need to assign some other signal spot in the time gap. Thus, we assign the previous cycle of RF relay message RFH^{m-1} that has been transmitted from beacon B_{m-1} for multi hop. Since beacon B_m is responsible for forwarding the RFH^{m-1} to the next hop of beacon B_{m+1} , the best location to assign RFH^{m-1} is somewhere in the empty spot between $RFS(S_1, B_m)$ and RFH^m . The next question for this procedure then becomes: how many slots for beacon relay messages are available in the empty spot?

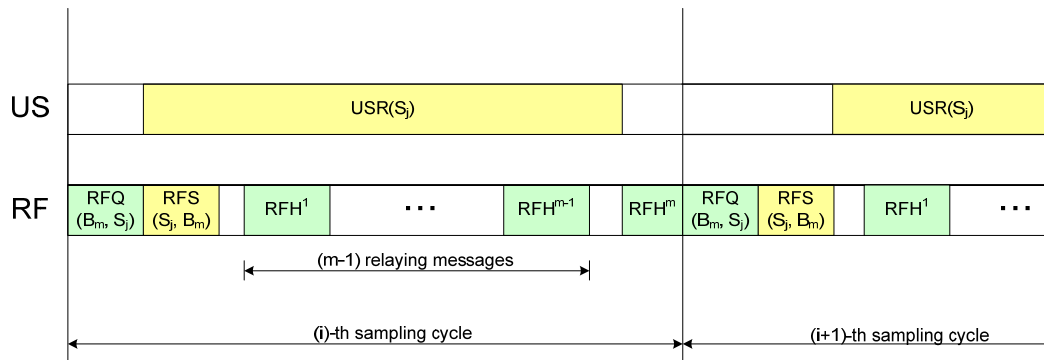


Figure 4.20: Beacon B_m scheduling of single communication

If we assume that the maximum coverage range of ultrasound communications is at most 50 m, then the maximum one-way travel time of ultrasound (at the speed of 340 m/s) can be simply computed as follow:

$$T_{US} = \frac{50 \text{ m}}{340 \text{ m/s}} = 147.06 \text{ ms} \quad (4.15)$$

On the other hand, the duration of a RF signal is relatively short even when we consider the processor's operation time (T_{OP}), MAC access time (T_{MAC}), and RF propagation time (T_{TRAV}), shown in figure 4.21. If we use a 50 MHz oscillator frequency in the microprocessor, an 8-byte RF message with a 256 kbps data transmission rate, and a 3×10^8 m/s for traveling speed, then we can estimate the approximate duration of the RF signal (T_{RF}) as follows:

$$T_{OP} = \frac{1}{50 \text{ MHz}} = 20 \text{ ns} \quad (4.16)$$

$$T_{MAC} = \frac{8 \times 8 \text{ bits}}{256 \text{ kbps}} = 250 \text{ } \mu\text{s} \quad (4.17)$$

$$T_{Trav} = \frac{50 \text{ m}}{3 \times 10^8 \text{ m/s}} = 167 \text{ ns} \quad (4.18)$$

$$T_{RF} = 2 \times (0.167 + 250) + 0.02 = 500.354 \text{ } \mu\text{s} \quad (4.19)$$

Thus, duration of the ultrasound propagation time is approximately 293 times longer than that of a RF signal. This means that the number of beacons for multi hop can

be increased to as many as 587 beacons within a single in-network domain without additional scheduling delays, and 587 RFHs can be accordingly assigned to the empty slot between the RFS and the RFH. This calculation may be quite conservative because we did not consider the operation time of the ultrasound device's microprocessor, which is generally longer than the operation time of a RF device's microprocessor due to a lower frequency.

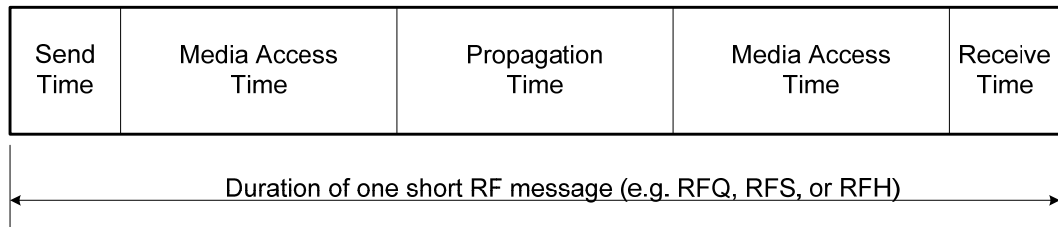


Figure 4.21: Structure of a RF message

4.4.3 Multi-Communication

A real sensor network application includes complicated network links between link and sensors, or beacons and sensors. In our application scenario, two network topologies is addressed: 1) a star network between beacon and sensors, and 2) a tree network between beacons. The main activities of signal communications in a star network are 1) sensory data communication, and 2) time of flight for estimating the distance between a beacon and a sensor. In the star network, however, a single communication between beacon and sensor is not always the case because a beacon can communicate with multiple sensors within its coverage range. Thus, we can define one-to-many communications, i.e. a beacon is linked with multiple remote nodes, as a star network. Similarly, a star network is defined as a many-to-one communication, i.e.

multiple beacons and one remote node. On the other hand, the relaying mechanism of multi hop among beacons is defined as a tree network where the RFH messages, collected in each scheduling cycle of multiple beacons, receive and forward them into adjacent beacons. This section examines the scheduling methods for multi-communications capable of implementing the collision-free scheduling in the real-world application scenario.

1) One-to-Many Communication

Since multiple radio frequency and ultrasound signals are received at one beacon simultaneously, a challenging task in this case is to schedule the sensing resource at the beacon side that avoids collision among multiple signals from/to the remote nodes. Figure 4.22 illustrates the beacon scheduling at i -th sampling cycle when N numbers of remote nodes are communicating with a beacon B_m . If we define N numbers of remote nodes as S_j , where $1 \leq j \leq N$, and if we define a beacon B_m as one of M numbers of beacons in the network domain, where $1 \leq m \leq M$, then each RFQ are stacked in a sequence order because the microprocessor in the beacon commands each instruction at the edge of the digital clock cycle. Because of this, the instruction cycle required for a digital command, each starting point at the RFQs should be shifted by the amount of the instruction delay time δ . As a consequence, each RFS and USR following each RFQ is also shifted by the instruction delay time δ .

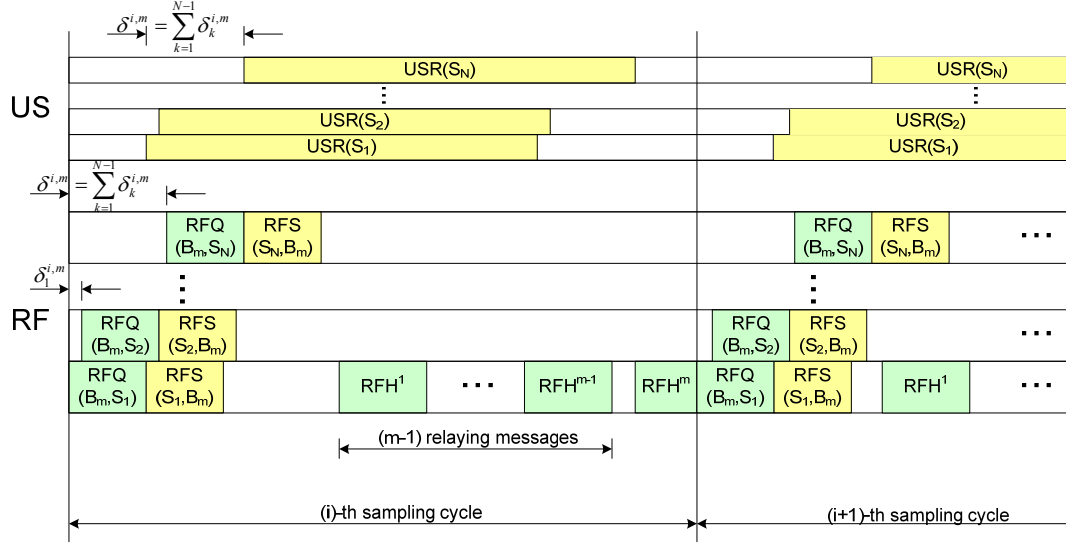


Figure 4.22: Beacon B_m scheduling of one-to-many communication

The relaying mechanism in the i -th sampling cycle starts when a N number of ultrasound signal USRs are received at beacon B_m . If the duration of the ultrasound signal $USR(S_N)$ on top of the layer equals the $\max(USR(S_j))$, where $1 \leq j \leq N$, the latest starting point of sending relay messages RFH^m in i -th sampling cycle can be delayed by the aggregated amount of each instruction time from the one in single communication approach, which can be expressed as follow:

$$\delta^{i,m} = \sum_{k=1}^{N-1} \delta_k^{i,m} \quad (4.20)$$

where, $\delta^{i,m}$ is total delay time caused by instruction cycle in m -th beacon at i -th sampling cycle and $\delta_k^{i,m}$ is a delay time in k -th layer. It is noted that if we assume the multi-hop

path of beacons as a sequential order from 1 to M, then the next hop of relaying message can be B_n .

The next consideration in this approach is that (m-1) numbers of hop messages should reside in the empty spot between the latest RFS and RFH^m. The left-over time and available number of slots of previous relay messages can be calculated using the same findings of the previous section:

$$\begin{aligned}
 T_{left-over} &= \left(\delta^{i,m} + RFQ^{i,m}(B_m, S_N) + USR^{i,m}(S_N) \right) \\
 &\quad - \left(\delta^{i,m} + RFQ^{i,m}(B_m, S_N) + RFS^{i,m}(S_N, B_m) \right) \\
 &= USR^{i,m}(S_N) - RFS^{i,m}(S_N, B_m) \tag{4.21}
 \end{aligned}$$

$$N_{hop} = \frac{T_{left-over}}{T_{RF}} = \frac{T_{left-over}}{500.354 \mu s} \tag{4.22}$$

Since the actual remote nodes within the coverage range of beacon B_m are placed randomly, $T_{left-over}$ varies according to the random probability of sensor placement. However, this research will investigate the available number of hop slots in the worst case scenario, i.e. maximum duration of ultrasound signal $USR^{i,m}(S_N)$ and $RFS^{i,m}(S_N, B_m)$. In this way, we can expect the longest cycle time of scheduling throughout the whole sensor network system.

2) Many-to-One Communication

In contrast to one-to-many communication, this case is the situation where only one remote node is interlinked with multiple beacons within its communication range. Since RFI transmitted from a remote node S_j can be reached by multiple beacons, in turn,

the beacons start sending the RF query signals to the sensor immediately after reception of RFI. Since each beacon assigns one communication link with the sensor, this communication can be considered as single communication in the beacon side. On the other hand, after advertising the sensor's RFI, the sensor eventually receives multiple RF query signals. Thus, the scheduling in the sensor side is similar to the one-to-many communication. However, the scheduling of sensor is dependent on messages received from beacons as a function of a listener, thus the actual scheduling scheme in the sensor side is not necessary.

4.4.4 Mobility Model

The mobility of the remote nodes for measuring distance is not considered static because each sensor device attached to actual construction materials move around the construction site domain. Light-weight materials could be carried by workers, but particularly bulky materials such as precasted concretes or steel membranes should utilize vehicles. In this case, we need to consider the sensor's moving speed since measuring points of remote nodes in each time step is a function of the beacons' sampling cycle. As the speed of a sensor's movement increases, uncertainty to track the exact path of the sensor's movement becomes problematic. For example, assuming that a sensor is moving at 10 km/hour ($=2.778$ m/s) around the network domain, whereas the sampling rate of the beacon's scheduling is 150 ms, then each measuring point of the moving sensor may have a 0.417 meter difference from the previous position. If the moving speed increases as much as 30 km/hour, then the difference in each sampling position may increase to 1.25 m. This means that with the sampling rate of 150 ms a beacon

cannot correctly detect the intermediate path in each 1.25 m gap. Figure 4.23 describes an example of the relationship between a sensor's moving speed and a beacon's sampling rate.

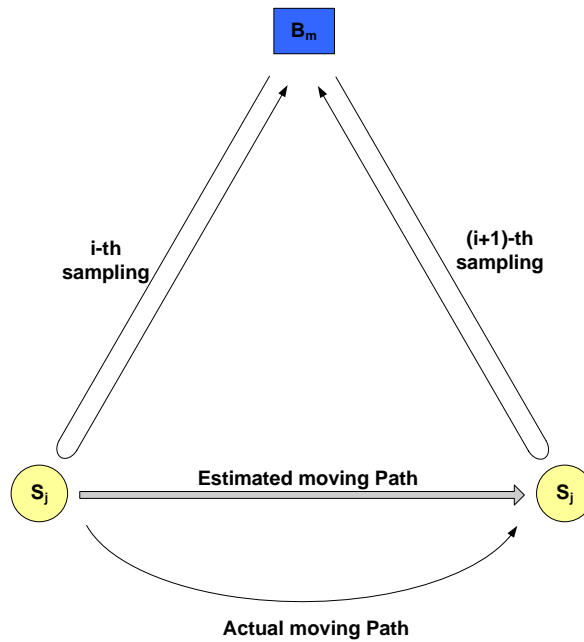


Figure 4.23: Mobility problem: errors in estimating the actual moving path when sensor's moving speed is very faster compared to beacon's sampling rate

This mobility model explains how we may track the mobile sensors at various degrees of resolution under a particular range of a beacon's sampling rate. In addition to the positioning accuracy simulated from trilateration, the degree of sampling resolution in a tracking sensor's path will also play a role in estimating the position accuracy. Based on the simulated results of a beacon's scheduling, we will investigate the tracking errors related with the sampling rate, the sensor's mobility, number of sensors and number of beacons.

4.5 Deployment Challenges in Construction Sites

Expected challenges for applying the material tracking and monitoring system in construction site can be categorized into the following two practical issues: 1) line of sight for signal propagation; and 2) battery management.

1) Line of Sight for Signal Propagation

In general, the radio frequency signal doesn't require the line of sight (LOS) issue to be communicated with the distributed remote nodes, hence it is widely accepted that the radio signal is a good candidate for wireless monitoring systems. Usually, one-way travel of ultrasound wave presented in this dissertation is able to penetrate the thin walls or objects to determine the distance without accuracy compensation. However, our localization technique may incur a limitation due to LOS issue if the obstacles that block the ultrasound wave become thick. Hence, it is necessary to configure the different setup of sensor placement for the practical use of this technique. For practical example, beacons placed in higher location, such as ones installed on top of construction light pole, might overcome the LOS problem for the tracking system of the construction materials that are laid in the complicated arrays of laydown. Or, it is possible to overcome the LOF issues if one uses multiple beacons to allow more change to get clear LOS, or if one uses RF as an extra distance detector in case of the obstruction of ultrasound wave.

2) Battery Management

Battery management is also a fundamental challenge to the application of wireless sensor network for long-term deployment framework. While power efficient technique,

such as sleep mode, can provide possible solution to better performance of battery's life-cycle, more fundamental scheme of battery management needs to be examined and developed to overcome the limitations related to the life-cycle of the battery. Our next research objectives will include the investigation of the different platform that provides an advanced architecture for a reliable and long-term power source applicable to wireless sensor framework.

4.6 Chapter Summary

This chapter introduced a new tracking architecture based on an embedded system combining the ultrasound and radio signal for better accuracy performance. The proposed methodology was examined by conducting feasibility studies on the elimination of multipath and enhancements of measurement accuracy. Adverse effect of multipath often causes the signal interferences and reduced performance of accuracy, and it could be removed by adopting the time-stamp scheme that is capable of detecting the first-arrival signal. In addition, the signal detecting mechanism associated with the instruction cycle of microprocessor was examined to advance the motivation to adopt the combination of ultrasound and radio signals in a material tracking framework. Furthermore, the feasibility simulation justified the necessity of using the proposed approach by comparing the RF-only and RF+US schemes, and a notable improvement of accuracy was observed when the RF+US scheme is deployed, due to the slow speed of ultrasound propagation. With the findings obtained from the feasibility studies, the next chapter will introduce the new development of embedded systems to verify the proposed methodology of envisioning the possible application scenarios in construction.

CHAPTER 5: PROTOTYPE EXPERIMENT FOR MATERIAL TRACKING SYSTEM

This chapter further explores the practical implementation for deployment to construction sites. Deploying the new position system presented in this dissertation offers an attractive solution in tracking construction materials. The benefits to this pilot implementation are: 1) high accuracy, 2) low-cost, and 3) robustness using the combination of ultrasound and RF. With flexible, efficient network capabilities provided by RF communication, ultrasound technology fits well with extended application scenarios on construction sites. Hardware and software architecture to design the tracking system by embedding the external ultrasound device into MICAZ in TinyOS system is examined later in this chapter. Results of distance and position estimation implemented by the proposed system are discussed in both single-hop and trilateration in order to examine measurement accuracy and system performance.

5.1 Overview

Wireless sensor modules such as MICAZ offer efficient, powerful capabilities for radio-based communications yet with low-power and low-cost architecture. General applications of sensor monitoring systems benefit from these commercialized products when sensor boards such as MTS300 or MTS310 (produced by Crossbow Technology), and well defined software are available through TinyOS communities. However, with embedded systems, tasks such as connecting external devices onto MICAZ, are challenges because the application software and hardware are solely dependant on the

purpose of the applications. For example, if a developer would like to implement a ranging device using infrared, extensive efforts are needed to configure the hardware requirements, e.g. external power supply, microcontroller's port configuration as I/O connections, control of low-level hardware components, wiring the components of the external devices, and design of data packet structure for RF communication. In addition, a developer must keep in mind that those factors in design should provide not only a compatibility with the platform of wireless sensor modules, but also a good applicability to the practical deployment strategy such as tiny size and long duration.

The presented methodology using the combination of ultrasound and RF is an innovative approach to increase accuracy and performance. Unlike radio frequency, ultrasound travels at a slow enough rate for a device to detect the first arrival signal by ignoring any associated errors caused by detection mechanisms in the instruction cycle of the microcontroller. Additionally, detecting the first arrival of a signal prevents possible multipath interference, because the mechanism for detecting a signal admits the earliest timestamp, while discarding the subsequent signals arriving in the receiver. RF signals in a beacon serves as a trigger to emit ultrasound pulses from a remote node, and is also used as a sender of the timestamp messages generated in the remote node. Four timestamps are used to measure the time-of-flight, e.g. Tx/Rx (TT1 and TR1) in a beacon and Tx/Rx (TT2 and TR2) in a remote node, and the calculation of TOF avoids any possible component of time delay caused by circuit signaling and interrupter handling inside the device, providing improved performance and accuracy. A brief illustration of the timestamping scheme is shown in figure 5.1.

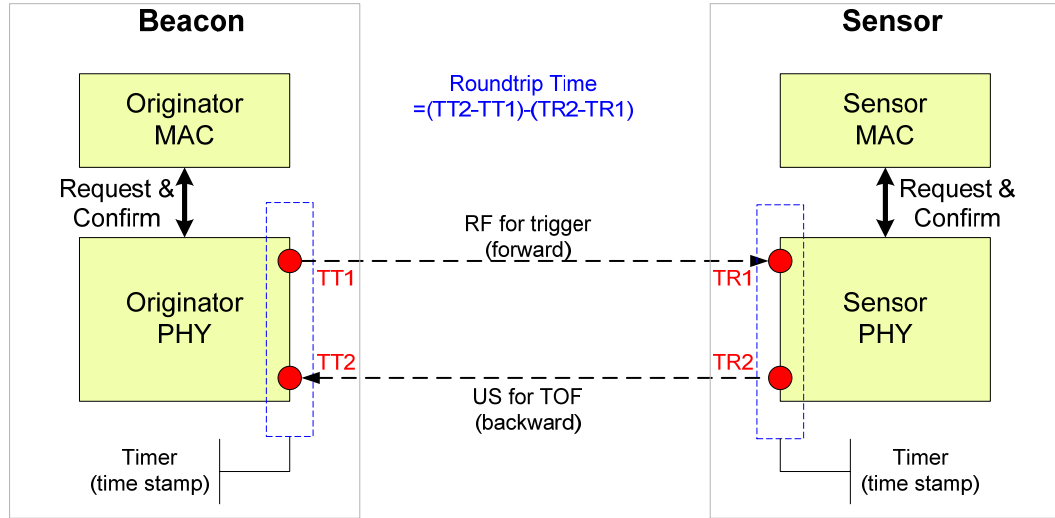


Figure 5.1: Illustration of timestamping scheme for TOF approach

5.2 Hardware Configuration

For the application of the wireless sensor network, MICAZ was used as the basic RF communicator through which data packets integral to triggering and timestamping information are transmitted. In order to implement the TOF scheme, hardware design for the external ultrasound device was developed. The external ultrasound device was then connected to the MICAZ module via an external 51-pin connector that offers input and output interface associated with ATMEGA128 microcontroller. One of each seven power ports (PW0 to PW6) and eight ADC ports (ADC0 to ADC7) gives the external device output and input controller, respectively.

5.2.1 Ultrasound Transducer

The ultrasound transducer has a variety of specifications for dominant frequency, sensitivity, and directivity that are dependent on the application purpose. 40 KHz dominant frequency is typical in small, embedded systems for ranging, distance

measuring, and object detection. The ultrasound transducer, Jameco 40TR16F, has a dominant frequency of 40 KHz with 4 KHz bandwidth, and has a physical dimension of 0.47H by 0.62D inch, which fits well with our purpose of tracking applications in construction. Ultrasound transmitters have a sound pressure level (SPL) of 119 dB at 40 ± 1 KHz, and the receiver has a minimum SPL of 65 dB at 40 ± 1 KHz, shown in figures 5.2 and 5.3, respectively.

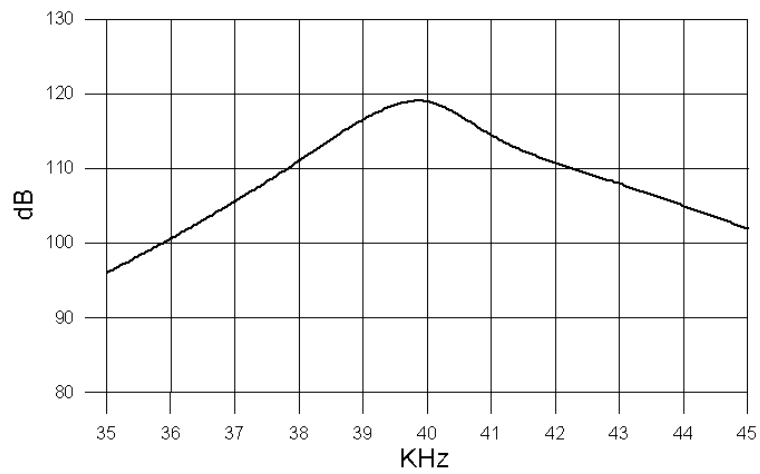


Figure 5.2: Sound pressure level at transmitter

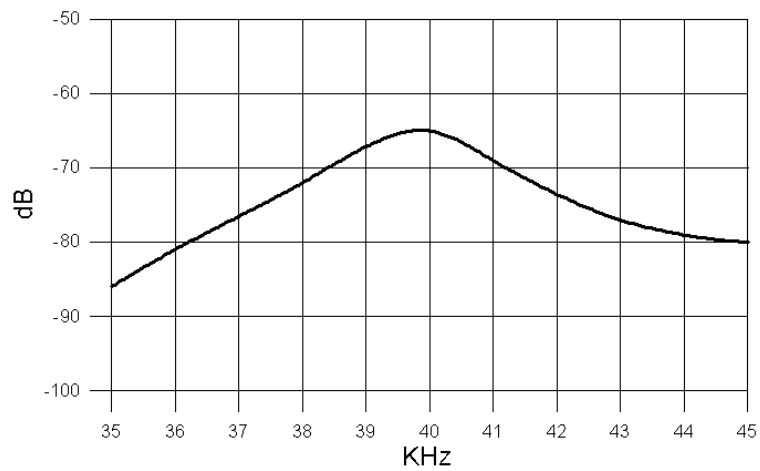


Figure 5.3: Sound pressure level at receiver

An important consideration in adopting this type of ultrasound transducer is directivity, which is defined as the radiation pattern from a source antenna that indicates how much of the total energy level from the source is radiating in a particular direction. Due to this characteristic, a radiating ultrasound signal experiences deterioration in its power strength if a receiver is not aligned directly with the source. Typically, 40TR16F has 50 degrees of directivity in which 20 dB power-losses appear at ± 50 degree, and much more power deteriorates beyond that point. The characteristic of directivity in 40TR16F is shown in figure 5.4.

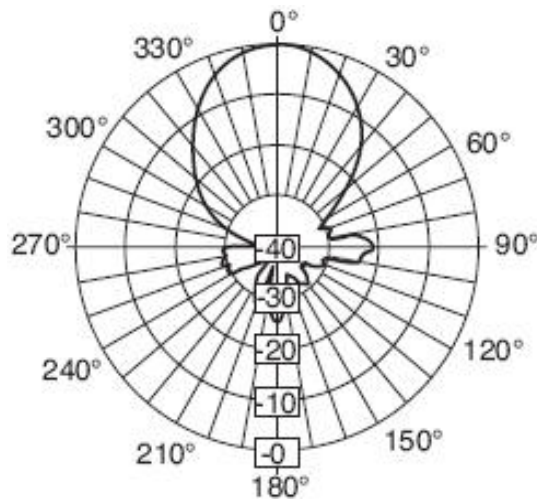


Figure 5.4: Characteristic of directivity in 40TR16F ultrasound transducer [Jameco Electronics]

5.2.2 Transmitter

For transmission, sufficient voltage is required to be driven to the ultrasound transducer. Usually, MICAZ is operated with two sets of AA batteries providing a reference voltage of 2.7 V. However, this voltage level is insufficient to assure a long

distance measuring of the ultrasound signal. As a consequence, transmitted ultrasound signals require amplification in order for a receiver to detect at the anticipated voltage level. Another key issue to consider in transmitter design is the setup of the 40 KHz voltage that is required to the ultrasound transducer. Voltage regulation can be achieved by either hardware or software. If hardware is used to control the 40 KHz voltage regulation, then it should have a circuit which uses a timer that produces a 40 KHz voltage pulse coupled with the triggering voltage input. If software is used to control the voltage regulation, the 40 KHz voltage pulse should be coded by using timer components provided in TinyOS, with a series of 40 KHz voltage pulses driven through a power input port.

Despite software implementation which allows a compact hardware design, current MICAZ provide a bit of complicated component architecture to achieve a microsecond timer, causing additional efforts in programming and wiring. For example, a default timer component, TimerC in TinyOS 1.10, provides a millisecond resolution to implement the pulse oscillation, which does not meet the purpose of generating 40 KHz voltage regulation that requires *microsecond* resolution. An available timer component for microsecond resolution is provided in the application for high frequency sampling, named `microsecondM.nc`, but additional low-level hardware control was needed to wire it to our application. Thus, this research utilized a hardware design for the 40 KHz voltage pulse generation, which eliminated further programming labor associated with microsecond timer components. Additionally, design experience indicated that the integrated circuit of timer did not significantly affect the total size of circuitry at all. However, it should be noted that hardware timers are vulnerable to environmental factors,

such as temperature and humidity, because resistors used in timer circuits often change their values under extreme conditions. As a result, the actual frequency emitted from an ultrasound transducer may not be exactly 40 KHz. In order to resolve this problem, a high performance potentiometer is used to calibrate the deviation caused by environmental factors, adjusting the frequency range as needed to the actual environmental situations.

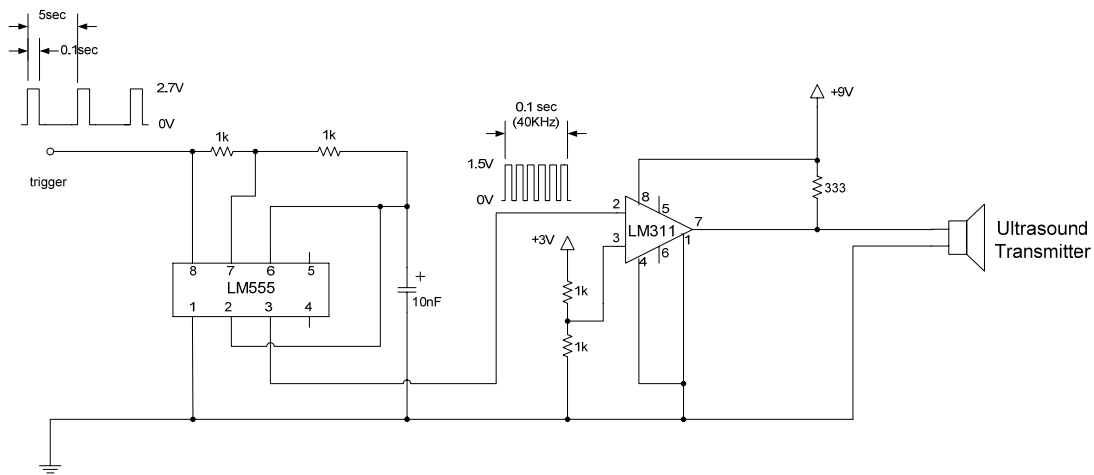


Figure 5.5: Circuit diagram for transmitter

Figure 5.5 depicts the circuit diagram used for an ultrasound transmitter, consisting of a timer and a voltage amplifier. A series of 5 second periods of voltage are blasted by the microcontroller at the level of 2.7 V amplitude, driving 0.1 second pulses set to 2.7 V for input to the hardware timer. At the level of 2.7 V amplitude with 0.1 seconds, a hardware timer, LM555, with two resistors and a capacitor, generates 25-microsecond (40 KHz) periods of voltage pulses at 1.5 V peak voltages. A series of 4000 short pulses triggered by the hardware timer are then amplified up to 9 V amplitude because this 1.5 V of transmitted signal will be attenuated down to less than a few milli-

Volt when received by the receiver circuit. Hence, a LM311 voltage comparator was used for signal amplification before transmitting the signal. The LM311 was used instead of a regular operational amplifier since it has a faster switching speed. Low-to-high and high-to-low level output response times were only 115 ns and 165 ns, respectively, which was more than enough for our application since the width of each pulse is 12.5 microsecond.

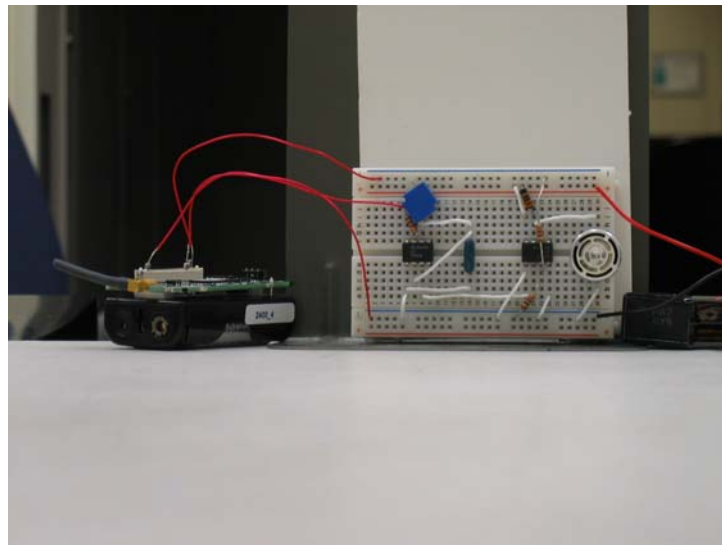


Figure 5.6: Hardware design of ultrasound transmitter connected to 51-pin connector in MICAZ

The voltage comparator compares the 3 V input pulse to a base 1.5 V. If the voltage level of the input pulse is greater than 1.5V, it outputs 9 V drawn from the power supply and drives the ultrasonic transducer, otherwise, it outputs zero V. Hence, the 3 V input pulses are amplified to 9 V pulses. The ultrasound transducer converts the input voltage into sound waves, emitting them at 40 KHz frequency. The circuit design illustrated above is implemented with breadboard in figure 5.6. While the signal shape measured at the hardware timer has a clean square pulse, the actual transmitted sound

wave from the ultrasound transducer does not have a clean square shape. This jagged noise is generated by the amplifier with a frequency four times greater than the 40 KHz of ultrasound wave. However, this noise does not affect the performance of the receiver side because the components of higher frequency (e.g. four times that of 40 KHz) is discarded by the ultrasound receiver due to its relatively small bandwidth. Output signals measured at pin number 3 of the LM311 and the ultrasound transducer are illustrated in figure 5.7.

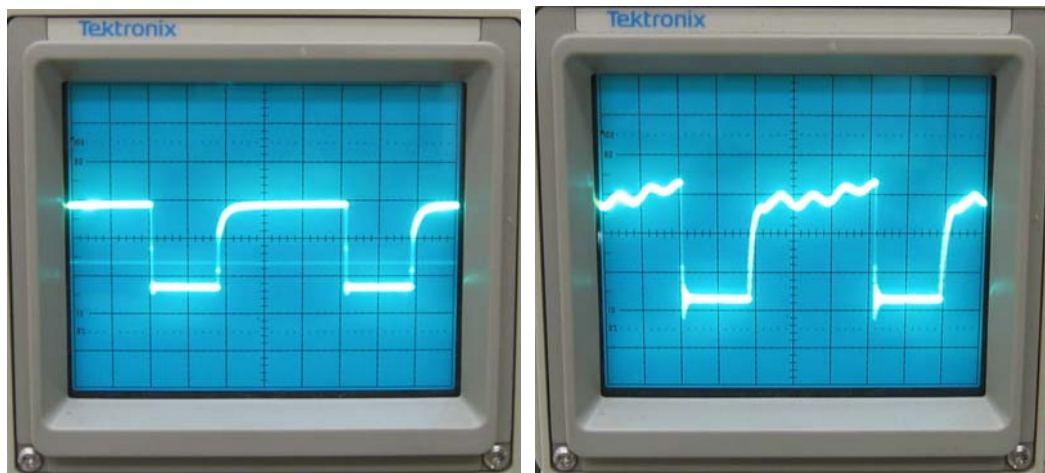


Figure 5.7: Output signal at hardware timer (left) and ultrasound transducer (right) measured by oscilloscope

5.2.3 Receiver

Usually, the received signal of raw ultrasound wave does not provide enough power to the ultrasound receiver. Depending on the ultrasound transducer, typically received voltage ranges are several milli-Volt, even at a distance of a few centimeters, and this minimal voltage is incapable of invoking the Analog-to-Digital converter (ADC) unless a sophisticated detecting algorithm is provided for accuracy. Thus, the received signal of raw ultrasound should be amplified to the allowable threshold, which is often

defined as the minimum voltage value able to be distinguished between the ultrasound signal and noise level. Ideally, zero value as the threshold can increase the measuring distance extensively, but this is not the case in practical implementation because the amplification process is often susceptible to unwanted noise. Even if the received signal is amplified to several thousand times that of the raw signal, noise level caused by the amplification is also increased, therefore, determining the lowest threshold value while ensuring a high detection rate is a prerequisite challenge in receiver design.

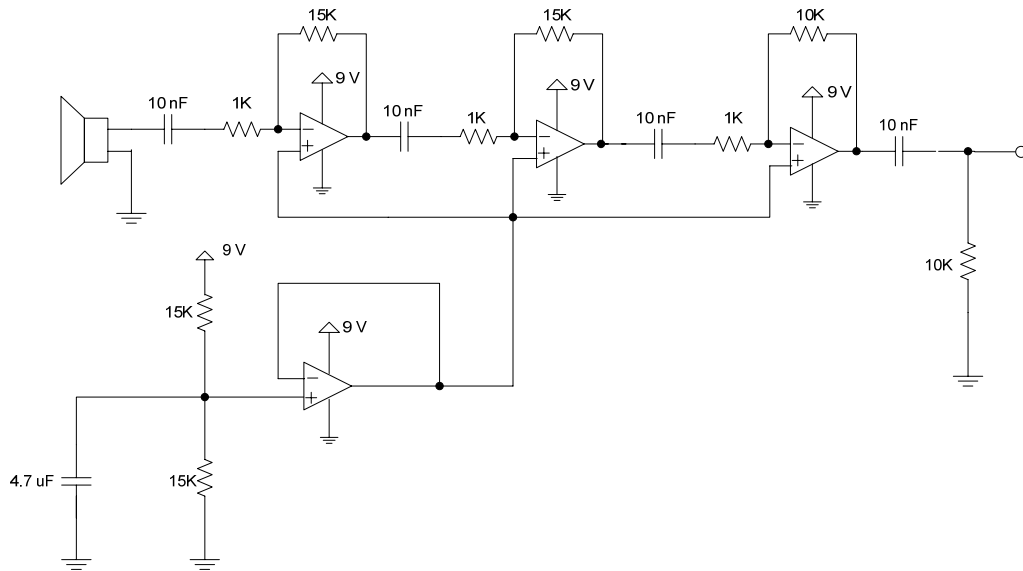


Figure 5.8: Circuit diagram of receiver

A Circuit diagram of an ultrasound receiver is illustrated in figure 5.8. A raw received ultrasound wave travels through three layers of amplification, and a LM358 OP AMP with two 15kΩ/1kΩ and 10kΩ/1kΩ, respectively, was used to achieve a gain of 2,250. The response time of the amplifier is 4 μs as shown in figure 5.9, thus switching speed may not be a big concern because it generates only about 1 mm error in distance

estimation. Three layers of amplifiers take positive input from a 9V comparator, therefore 4.5 V is fed into the positive voltage. 10 nF capacitors are placed at the output pin in order to minimize the noise level caused by the amplification. Also, output with 2,500 gains goes through a DC block circuit to achieve an AC coupled signal. Figure 5.10 demonstrates the actual hardware design and connection to MICAZ for the ultrasound receiver.

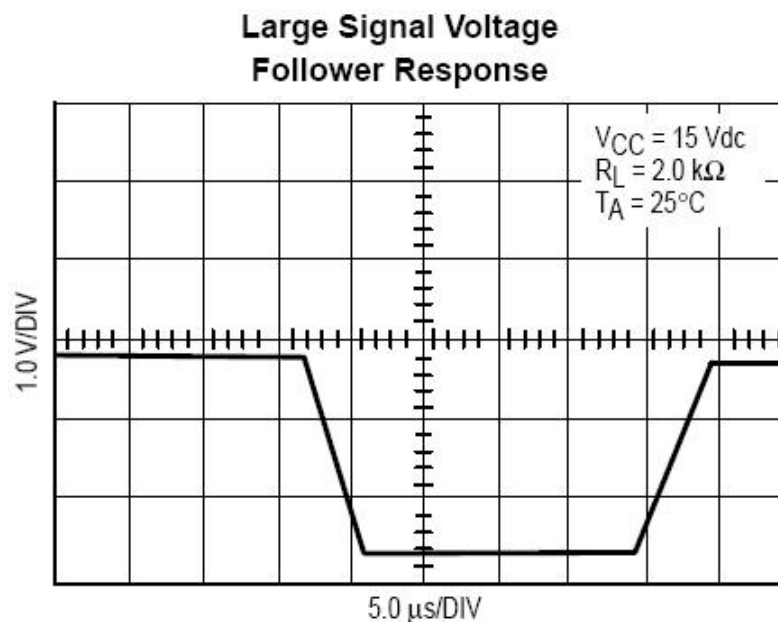


Figure 5.9: Response time of signal voltage, LM358 [Motorola]

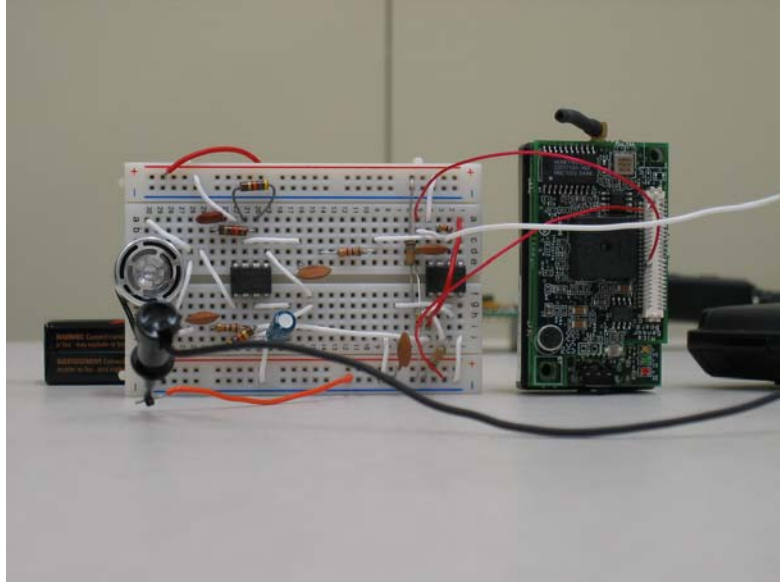


Figure 5.10: Hardware configuration of a beacon

5.2.4 Connection to MICAZ

The external sensor device requires a power supply which can be offered either by external batteries or an internal power port from MICAZ. However, most of the wireless sensor modules, including MICAZ, is supplied by 3 V external batteries, thus the external device connected to MICAZ can be provided with maximum 3 V, which is not sufficient for diverse functionalities by the external device. Also, enough power exceeding 3 V should be provided to the ultrasound transmitter connected to MICAZ, so that the ultrasound receiver has a greater chance to detect the transmitted signal, because the signal strength of the transmitted signal decays as much as the exponent of the distance between the transmitter and receiver. Accordingly, the external ultrasound device is provided with 9 V external power supplies that increase the transmitted and received power of the ultrasound signal.

Another consideration when combining the external device to the MICAZ platform is to determine the appropriate port number offered by the MICAZ interface. Typical programming events and commands are invoked throughout the external connector in which a command for triggering and timestamping events are controlled and delivered. The 51-pin external connector of MICAZ is illustrated in figure 5.11. The connector provides the ability to connect specific ports to the input or output of the external device. For instance, one of the seven PW ports is used as a signal-trigger port, so that the supply voltages can be switched on and off to drive a 0.1 second oscillating pulse, which is converted into a driving voltage of 40 KHz frequency with 0.1 second duration, at the hardware timer to invoke the ultrasound wave being transmitted. In addition to the power ports, 8-channel ADC ports are also available to receive the analog signal which was driven from the received ultrasound waves. Voltage levels measured in the passive ultrasound transducer is converted into meaningful values of voltage level upon the bit conversion, e.g. 8-bit or 10-bit ADCs in MICAZ.

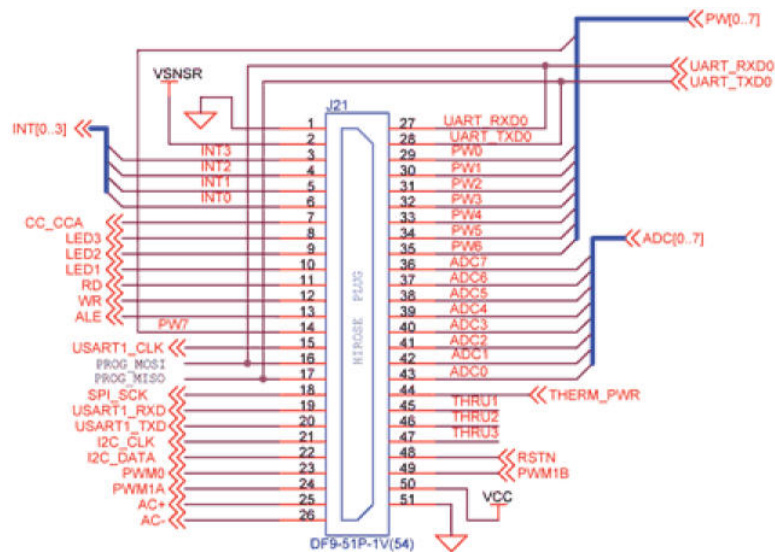


Figure 5.11: 51-pin extension connector in MICAZ [Crossbow Technology Inc.]

PW2, pin number 31 assigned as a general I/O, is used to trigger the ultrasound wave from the transmitter, where PW2 is power-on for 0.1 second and off for 4.9 seconds in each cycle, to allow the hardware timer to trigger a 40 KHz sound wave. On the other hand, the 10-bit ADC port 7, pin number 36, is assigned as a detector of a received signal, and the ADC value detected at this port is converted to digital value to invoke the timestamp for distance estimation. A threshold is then applied with the received ADC value, and command for the process of timestamp and data storage is invoked if the received voltage is over the threshold defined in the receiver code. Additionally, a reference voltage of 3 V provided by the VSNSR port at pin number 2 is supplied to a voltage comparator in the LM311, an amplifier in the transmitter side, and the GND port at pin number 51 is used in both the transmitter and the receiver.

5.3 Software Configuration

Software architecture in this dissertation consists of four components: US_TriggerM.nc, US_TxM.nc, US_BaseM.nc, and java. First, US_TriggerM.nc is applied to three beacons to perform the functionalities of RF-trigger, US-receiver, RF-receiver, and RF-relaying. Second, US_TxM.nc is developed to achieve RF-receiver, US-transmitter, and RF-transmitter. Third, US_BaseM.nc relays the received RF message containing the data packets about parent nodes, travel time of ultrasound, and distance estimation; all of which are gathered in each cycle of TOF process performed by communication between beacon and remote node. Gathered data is then forwarded to the PC platform throughout the UART interface, and a developer can display, manipulate,

and handle the received data per various application scenarios using java tools. LOF.java is used as a data logger and visual demonstrator for data representation, and displays the trace of moving activities with respect to x and y coordination, kept track of by three beacons placed at a fixed, known location.

5.3.1 Software Architecture for Remote Node

The basic functionality of the remote node developed in this dissertation is a one-way US transmission to designated beacons, and the travel time incorporated with the four different timestamps is obtained for conversion of distance measurements. In order to transmit the US signal to transmitter, the remote node should be programmed to achieve several functionalities. First, the US signal should be transmitted at the remote node only when it receives a RF trigger message coming from the beacon. The trigger message must have a node ID that distinguishes who actually sent a trigger message for US transmission, so that the remote node can return a timestamp message back to the originator with the information about RF receiving and US transmitting time needed in the TOF calculation. TR1 and TR2 are RF-receiving and US-transmitting times issued at the remote node, shown in figure 5.12.

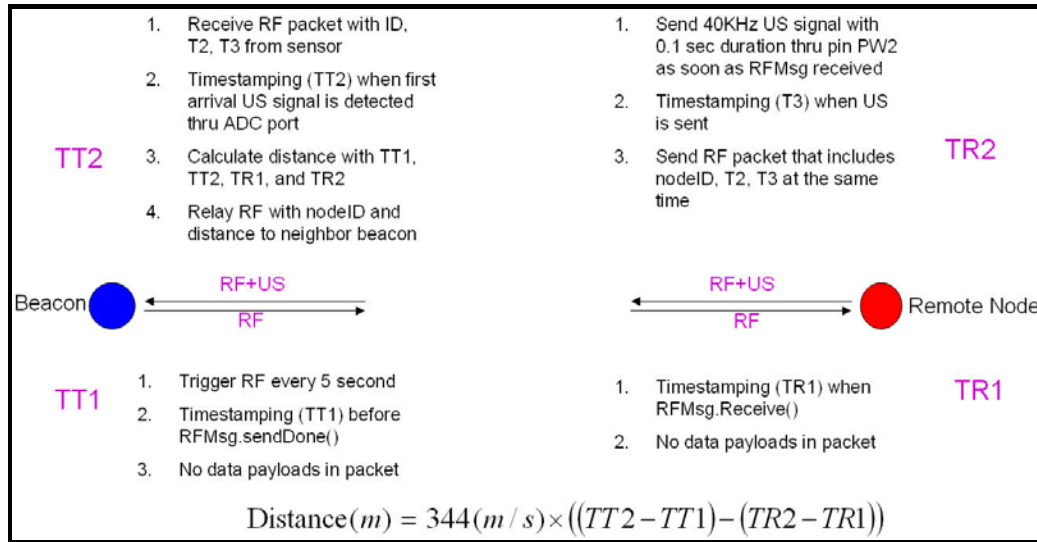


Figure 5.12: Illustration of timestamping and distance conversion

The time difference between TR1 and TR2 is then considered as a processing time to generate the US transmission. For instance, a microprocessor in the remote node checks if there is a trigger message coming from several distributed beacons, and once it receives it, the microprocessor starts the signal processing. It requests and confirms the RF packet to MAC layer, allocating the received packets to memory space. Once it confirms the RF packet as a trigger message, then the received data is assigned into the local variables for pre-defined activities in the US_TxM.nc code. These activities include: 1) flashing out the allocated memory stack generated from RF packets received, 2) calling asynchronous hardware interrupter , the system timer, to issue timestamp TR1, 3) calling the hardware controller to allow the PW2 port power-on to drive a 0.1 second duration voltage pulse, 4) calling the asynchronous hardware interrupter again to issue timestamp TR2, and 5) allocating the data variables such as the node ID and timestamps into RF packet structure for sending out the data packet to destination node. Flowchart diagram for operating US_TxM.nc is described in figure 5.13.

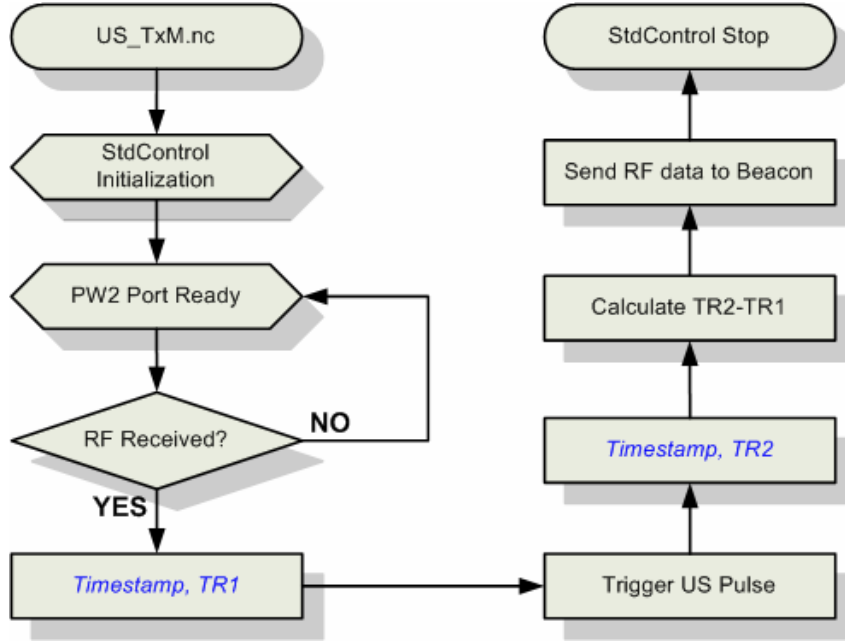


Figure 5.13: Flowchart diagram for operating US_TxM.nc

The structure of US_TxM.nc consists of the RF communication phase and the US transmission phase. As shown in figure 5.14, RFDetecMsg instance is invoked and it allocates TOS_MsgPtr, variable m, into the memory stack, defining the local variables with SendMsg pointers. As soon as the receiving event happens, a system timer is generated and assigns the timestamp value into Rtime, which is used in the later RF-transmitting process to calculate the elapsed time caused by the internal system process. Once the allocation of the local variable is complete, the system calls a local task event, USTx(), to trigger the ultrasound signal. Since the ultrasound signal is generated throughout the external device, the hardware controller should be called to set the PW2 port on and off. In TinyOS, a lower level of hardware components, such as TOSH, provides a simple and fast process, and is defined below:

TOSH_ASSIGN_PIN(VSensor, C, 2)

TOSH_MAKE_VSensor_OUTPUT()

TOSH_SET_VSensor_PIN()

TOSH_CLR_VSensor_PIN()

General I/O port PW2 is assigned at C2 as a triggering port to the hardware timer, and the VSensor, a local variable for pin assignment, overrides the name of PW2, assigns it as an output port, and drives on/off voltage generation.

```
event TOS_MsgPtr RFDetectMsg.receive(TOS_MsgPtr m) {
    struct SendMsg *pack;
    atomic {
        pack = (struct SendMsg *)m->data;
    }
    Rtime=call SysTime.getTime32();
    Destination=pack->sourceMoteID0;
    call Leds.greenOn();
    TOSH_SET_VSensor_PIN();
    post USTx();

    return m;
}
```

Figure 5.14: Example code for RFDetectMsg event

The 32-bit system timer, SysTime.getTime32(), issues a timestamp at the microsecond resolution, which is allowable to our purpose because 1 microsecond generates only 1.3 millimeter errors in the distance estimation. In the remote node, two timestamps are issued at the time when it receives a RF-trigger message from the beacon and when it transmits a US-signal to the beacon. Once an ultrasound wave is generated by the TOSH interfaces, two timestamps are carried to SendTStamp task according to the

packet structure, shown in figure 5.15. A sourceMoteID0 pointer, used in the RF receiving process issuing destination, is overridden by the address of the current local node, and is carried in the payload packet as sender's node ID to destination beacon. A TimeStampMsg event is then called with the destination address, message size, and their pointer address.

```

task void SendTStamp() {
    struct SendMsg *pack;
    atomic {
        pack = (struct SendMsg *)msg.data;
    }
    pack->sourceMoteID0 = TOS_LOCAL_ADDRESS;

    if (call TimeStampMsg.send(Destination, sizeof(struct SendMsg), &msg))
    {
        call Leds.yellowOn();
    }
}

task void USTx() {
    struct SendMsg *pack;
    Ttime=call SysTime.getTime32();
    Etime=Ttime-Rtime;
    atomic {
        pack = (struct SendMsg *)msg.data;
        pack->data[0] = Etime;
    }
    post SendTStamp();
}

event result_t TimeStampMsg.sendDone(TOS_MsgPtr sent, result_t success) {
    return SUCCESS;
}

```

Figure 5.15: Example code for US and RF transmission in a remote node

US_TxM.nc is composed of RF-receiving and RF-transmitting events that are predefined in the Active Message (AM) communication framework in TinyOS. The Original AM packet structure includes up to 29-byte payload, and the payload packet is broken down into nine variables to carry the timestamping message. In addition, the RF message packet, shown in figure 5.16, is followed by a 35-byte data packet structure

defined in a header file, TriggerMsg.h, and the variable statements. Three unsigned 8-bit sourceMoteID0, 1, and 2 are assigned for the information of sender node, and six unsigned 32-bit data is assigned to represent processing time inside the remote node. It should be noted that those variables are encapsulated in the SendMsg structure, and can avoid any confusion with the local variables when asynchronous hardware handlers are invoked during the process, and all of this process follows the concurrency model designed in TinyOS platform.

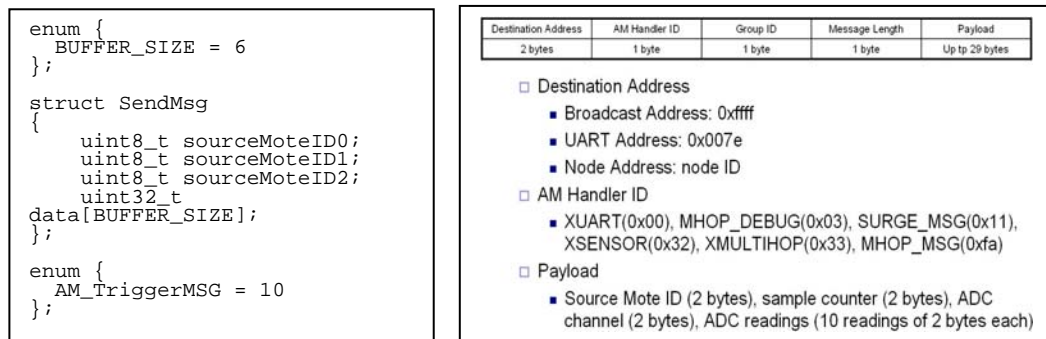


Figure 5.16: RF packet structure

5.3.2 Software Architecture for Beacon

The system architecture, for the distance estimation mentioned before, is to combine RF and US as a means of RF communication and US time-of-flight. One-way travel of ultrasound is used in order to extend the measuring range of ultrasound signal, and the one-way RF signal coming from beacons acts as a trigger to drive and make the US signal returned to the beacon. Many ultrasound ranging devices use two-way ultrasound travel to detect and measure the distance between the device and object because it is more simple and easier to implement. On the other hand, one-way travel of

ultrasound presented in this research is a smarter way to minimize the power loss over two-way travel of ultrasound signal, because most of the transmission power is significantly decreased from reflection during the time when the power is absorbed by the objects and the remaining power is returned to the originator. By decreasing the power loss, it is expected that the original sound wave of ultrasound is able to travel further, and as a consequence, the measuring distance can be extended for large-scale application domain.

A beacon consists of RF transmitter, RF receiver, and US receiver, and the messages carried by radio are communicated with RF transmitter and receiver. RF messages contain the information about triggering event flag and elapsed time taken inside the remote node for signal processing. This highlights the benefit of using the RF signal as a general communicator because it is not possible to modulate the ultrasound signal to carry any information by itself. In addition to the use of RF, the timestamping approach does not require a time synchronization process. Usually, the platform of different hardware systems has the challenge of time synchronization because the local system time of two different devices is different such that the time measurements might be shifted by the amount of local time difference. Time synchronization is one of the largest research areas and there are many available methodologies to perform the algorithm. It should be noted that the approach to time measurement in this dissertation makes utilization of the four timestamp scheme so that the time difference between two different hardware systems cancel out each other, resulting in only the elapsed times taken from the different hardware being used for the time calculations. $(TR2-TR1)$ and $(TT2-TT1)$ are the elapsed times in each device, and their difference is used to measure

the travel time from a beacon and a remote node. As a consequence, this approach can save efforts and time to implement the algorithm of time synchronization at small program size and high efficiency.

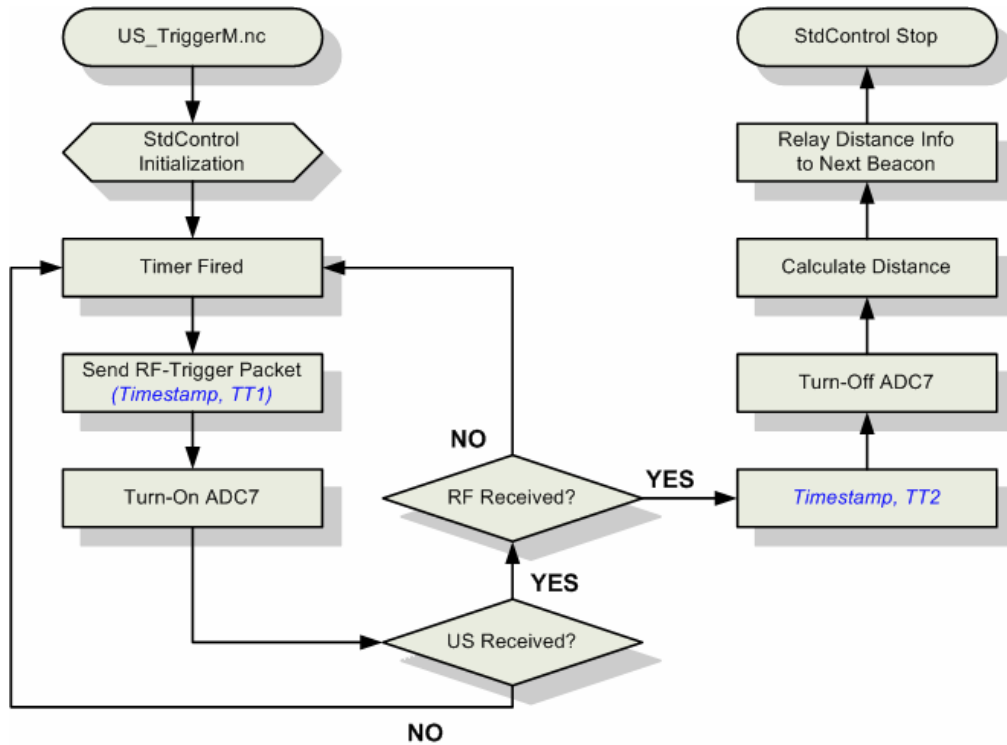


Figure 5.17: Flowchart diagram for operating US_TriggerM.nc

Figure 5.17 illustrates the brief structure of US_TriggerM.nc code. A beacon periodically sends a RF trigger signal to the remote nodes located within the communication range. At the same time, the hardware interrupter drives the ADC port to prepare the reception of US signals coming from a remote node right after the RF trigger signal is transmitted. The 10-bit ADC port 7 is then ready to detect the ultrasound signal and starts to sample the ADC measurement at tens of kSPS (kilo sampling per second) until an interrupt event for stoppage is issued. A set of macro code for the ADC sampling

rate is defined in ADCControl.nc as the parameters in which there are the lower three bits in the ADCSR register of the microprocessor. After testing the parameters of ADC sampling provided in MICAZ, it turned out that the MICAZ platform allows 30 μ s and a higher rate of TOS_ADCCSample among eight parameters. Thus, 33.3 KHz can be an instruction cycle in the microprocessor and approximately 2.5 KHz ADC sampling rate can be achieved. More detailed explanation about the determination of ADC sampling rate is discussed in the next section. Figure 5.18 describes an example code of ADC implementation.

```
module USTriggerM {
    provides interface StdControl;
    uses {
        interface ADCControl;
        interface ADC;
    }

    command result_t StdControl.start() {
        call ADCControl.setSamplingRate(3);
        return SUCCESS;
    }

    task void dataTask() {
        if (call DataMsg.send(10, sizeof(struct TriggerMsg), &msg)) {
            call ADC.getContinuousData();
        }
    }

    async event result_t ADC.dataReady( uint16_t data ) {
        if (data > 0x0078) {
            ...
        }
    }
}
```

Figure 5.18: Example code for ADC implementation

A beacon performs a US distance estimation to the remote node right after it receives a RF message from the remote node, shown in figure 5.18. Since travel time of RF is much faster than US, dataTask() issues an ADC.getContinuousData interrupter after the beacon sends a RF trigger message with a timestamp TT1 to the remote node.

Once the beacon gets the ultrasound wave from the ADC, the first arrival signal is registered, and the issuing timestamp TT2 and ADC port closes by returning FAIL flag to ADCCControl.nc. Calculated distance value and travel time of each measurement are then relayed to the next hop node with senders' node IDs, and it loops until the relaying message arrives at the base-station node. The packet structure carried in the relaying message is described in figure 5.19.

```

task void BSend() {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)msg.data;
    }
    Etime=Rtime-Ttime;
    D0=Etime-D0;
    D1=344*D0/1000;
    pack->data[0]=D0;
    pack->data[1]=D1;
    pack->data[2]=D2;
    pack->data[3]=D3;
    pack->data[4]=D4;
    pack->data[5]=D5;
    pack->sourceMoteID0 = N0;
    pack->sourceMoteID1=N1;
    pack->sourceMoteID2=N2;
    if (call BaseSend.send(destination, sizeof(struct BaseMsg), &msg)) {
        call Leds.greenOn();
    }
}

```

Figure 5.19: Example code for the RF relaying structure

5.3.3 Software Architecture for Base Station and Java

Unlike the decentralized network system, the approach for the tracking of construction material requires central control of the position information obtained from remote nodes. A decentralized system, such as GPS navigation system, addresses the end-user oriented platform that provides the users with the information of their location. Thus, the destination of the various information flows must target the users' handheld

devices such that the end-users handle and control the wireless communication for their purpose. However, the centralized system proposed in this dissertation requires a communication hub that controls and manages the information about the location of remote nodes. This is necessary in typical industry application because most of the location information of interest is not the user but the device. The communication hub, the so called base station, is collects, manipulates, and transfers the location information regarding the objects into a central server interface, designed for the purpose of applications.

Usually, a base station node in the wireless sensor network is connected to a personal computer that has larger capacities of computation and communication than the sensor itself. In the wireless sensor network, many programming applications for computing the collected data is often implemented by a java platform that provides an OS-independent framework to increase compatibility and applicability. Because of this fact, the java platform can benefit from a TinyOS architecture that provides abundant resources of application developed by java. Another advantage of PC is its communication capability because typical communication protocol in the PC includes IEEE802.11 that provides a wide accessibility throughout the world. Thus, developers can create the advanced database architecture in which web-based application can retrieve, display, and dynamically interoperate the information stored in the database.

Basic role of the base station consists of two parts: 1) gather the data packets sent wirelessly from beacons, and 2) transfer those data packets to the PC through a serial port. Flowchart diagram for operating US_BaseM.nc is illustrated in figure 5.20. The base station is a typical sensor node that is connected to MIB510 programming interface with

a serial port, and bytes of raw data are converted from/to asynchronous start-stop bit streams by a UART. The data packet structure used has 5-byte generic variables assigned by the Active Message structure and a 30-byte payload that contains information about node IDs, travel-time measurements, and distance measurements.

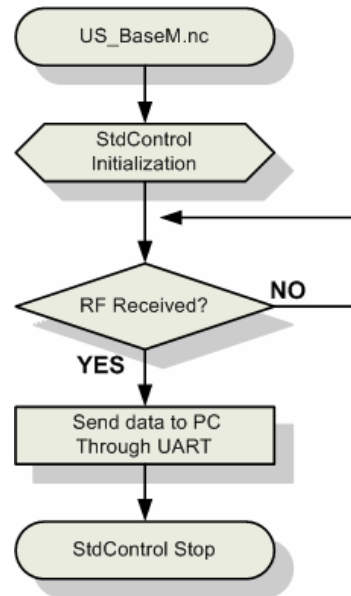


Figure 5.20: Flowchart diagram for operating US_BaseM.nc

Each stream of data received through UART is then manipulated by java application, e.g. LAF.java developed for trilateration and graphic representation. The LAF application forwards the data streams in every transmission cycle to buffers, which assigns and clears the local variables in each time for trilateration. Figure 5.21 describes the stream conversion from a byte into a decimal variable. After calculating the x-y coordination by trilateration, LAF performs a graphical display to visualize the actual location of the remote node which is updated each time the packets are received.

```

byte[] packet = reader.readPacket();

int t1 = ((packet[1] & 255) << 8) + (packet[0] & 255);
String twoChar = Integer.toString(t1,10);
if (twoChar.length() < 2) {twoChar = "0" + twoChar; }
    timeStamp = timeStamp + twoChar + " ";

for (int i=2; i<=7;i++) {
    int t2=(packet[i] & 255);
    String twoChar1 = Integer.toString(t2,10);
    if (twoChar1.length() < 2) {twoChar1 = "0" + twoChar1; }
        timeStamp = timeStamp + twoChar1 + " ";
}
for (int i=0;i<6;i++) {
    int t3=((packet[11+i*4] & 255) <<24) +((packet[10+i*4] & 255)
    <<16)+((packet[9+i*4] & 255) <<8) +(packet[8+i*4] & 255);
    String twoChar2 = Integer.toString(t3,10);
    if (twoChar2.length() < 2) {twoChar2 = "0" + twoChar2; }
        timeStamp = timeStamp + twoChar2 + " ";
}

```

Figure 5.21: Data packet conversion and variable assignments

5.4 Implementation of Distance Estimation and Trilateration

Based on the background and system architecture described in previous section, distance and position estimation are implemented in the outdoor environment. The focus of this implementation was on the distance estimation because position estimation by trilateration relies critically on the accuracy of individual distances measured from each beacon and remote node. Signal-to-noise ratio (SNR) is also important because higher SRN will generate a better performance in US signal detecting. In this section, accuracy and error distribution obtained from the combination of RF and US is examined at varying distances between beacon and remote node. Finally, position estimation using three fixed beacons and one remote node, e.g. trilateration, is implemented to illustrate the performance and measurement accuracy. This prototyping implementation and lessons learned will illustrate the possible application scenario when applied in the construction industry for tracking construction materials or valuable assets.

5.4.1 Distance Estimation

5.4.1.1 System Configuration

Individual distance estimation is a critical task with positioning systems because it gives a better chance to maximize the position accuracy when a trilateration or a multilateration scheme is applied. In order to perform the distance estimation, three MICAZs are connected to the US receiver, US transmitter, and MIB510 interface, as shown in figure 5.22. Ultrasound devices are connected throughout the external connectors provided in MICAZ, and 9V external power is supplied to the ultrasound devices. A set of MICAZ and ultrasound device is assigned as a beacon that is programmed with US_TriggerM.nc, providing the functionality of RF triggering and US/RF receiving. Another set of MICAZ and ultrasound device is assigned as a remote node for US/RF transmitting and RF receiving. As a base station node, one MICAZ is attached to the interface board for collecting and forwarding the wirelessly communicated message to the PC interface.

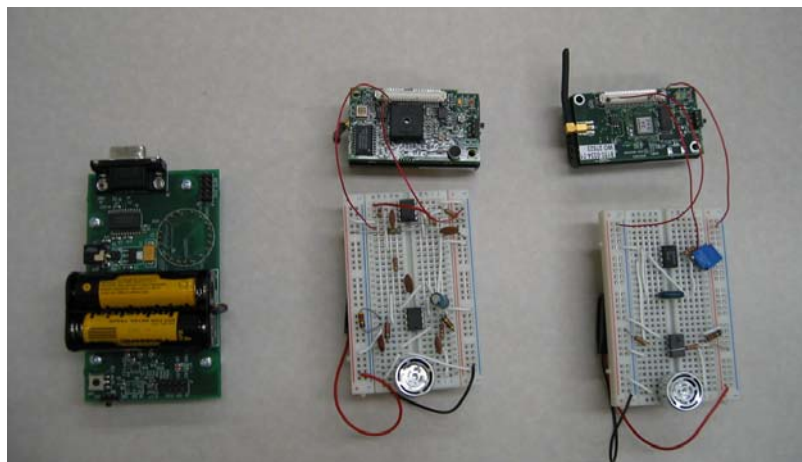


Figure 5.22: Device configuration for a MIB510 interface (left), a beacon (center), and a remote node (right)



Figure 5.23: Configuration of setup for distance estimation test

Figure 5.23 illustrates the test configuration set-up for distance estimation at the outdoor environment. A set of beacon is then placed at a fixed location and a set of remote node is placed at variable positions starting from 1 meter to 15 meters. A base station node collects and transfers the wireless message to the connected laptop. A line-of-sight is required between the beacon and remote node because ultrasound wave cannot penetrate a blocking object unless the power of the transmitted signal is sufficiently strong. This requirement may incur a limitation when this system is deployed in the practical construction site because the nature of the construction field consists of many blocking objects such as vehicles, materials, building structures, etc. However, we can increase the line-of-sight between them by placing the beacons at higher locations, such as attachment to a light pole in the field or a ceiling inside a building, which is similar to a satellite placement that allows an increased line-of-sight with GPS applications. As a

consequence, this challenge might be well resolved if a practical deployment strategy on a real construction site is examined when applied.

5.4.1.2 Signal Analysis

In order to eliminate the multipath effect of ultrasound, a signal detection scheme targets a first-arrival signal that travels the shortest path from a beacon and a remote node, and the remaining signals received after the initial one are discarded by hardware interrupter that closes the ADC port. The signal detection scheme is achieved by sequential order of signal processing in the receiver, and is followed by the RF triggering tasks controlled by the periodic timer operation. This means that ADC operation is handled by a RF trigger scheme, such that whenever the RF trigger signal is driven by a timer, the ADC is activated and ready to detect the received ultrasound signal. The ADC operation is then interrupted by a hardware handler right after the ADC issues the reception of the first arrival signal and powers off the ADC port by returning FAIL flag to the ADC controller. The process of signal detection scheme is illustrated in figure 5.24.

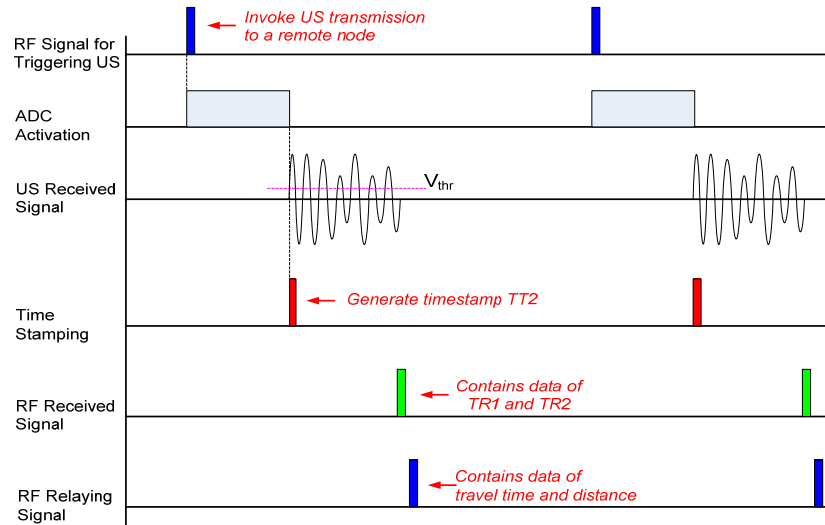


Figure 5.24: Process of signal detection scheme at receiver

The received signal is represented as voltage values that are converted by ADC. Ultrasound signal transmitted from a remote node has a 40 KHz sound pulse wave that is amplified by three layers of OP AMP, and then the ADC detects the analog signal of the pulse wave, converting it to digital format. The values of the converted signal depend on the mechanism of ADC bit-rate that determines the resolution, following the equation below:

$$V_m = V_{ref} \times \frac{ADC_Count}{ADC_FS} \quad (5.1)$$

where,

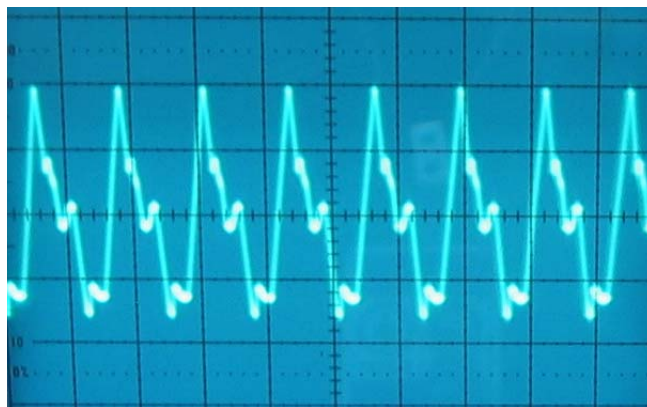
V_m : converted measurement in voltage

V_{ref} : ADC reference voltage, 2.56V

ADC_FS : ADC in full-scale, 1024 at 10-bit ADC and 256 at 8-bit ADC

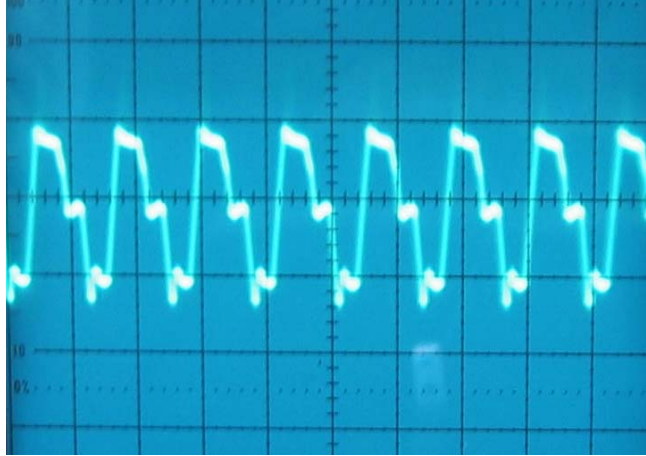
ADC_Count : data from ADC measurement at ADC port

The ADC value is then interpreted as a voltage level that gives a threshold to the signal detection algorithm. Figure 5.25 illustrates the received signal in voltage measured at the distance of 1, 5, and 10 meters from a remote node. AC-coupled 25 μ s period of received signal oscillates with peak voltage of 1.8 V at 1 meter and 0.7 V at 10 meters. It is necessary to note that the received signal does not look like a clean pulse wave such as a sinusoidal wave because we did not adopt any comparator circuit, such as a Schmitt trigger, to clean up the output driven from the amplifiers. While a Schmitt trigger provides greater stability with clean a shape in output, it can also result in peak-to-peak voltage loss that decreases the signal-to-noise ratio (SNR), which may be a challenge to determining a threshold value in signal detection algorithm. Instead, a jagged pulse wave provides a higher peak-to-peak voltage that allows increased probability to detect the signal at a higher SNR.

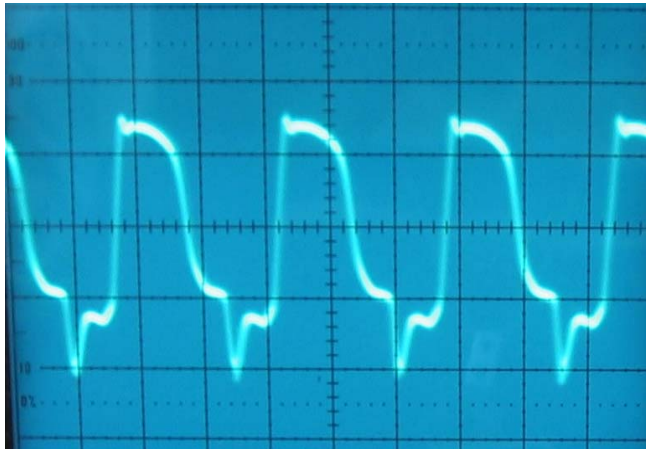


(a) Distance at 1 M (500 mV/div and 20us/div)

Figure 5.25: Screenshot of received ultrasound signal measured at 1, 5, and 10 M distance



(b) Distance at 5 M (500 mV/div and 20us/div)



(c) Distance at 10 M (200 mV/div and 10us/div)

Figure 5.25: Screenshot of received ultrasound signal measured at 1, 5, and 10 M distance (cont.)

The noise level measured at the receiver is shown in figure 5.26. Peak-to-peak noise level ranges from 0.2 V at 10 meters to 0.25 V at 1 meter, which explains the steady level of peak noise value even if the distance measurements increase. The measured noise is resulted from the process during amplification at three layers of OP AMP, and it is difficult to eliminate the noise caused by the amplification process without

improving circuit design. Instead, 10 nF capacitors are placed at each output port of the OP AMP to minimize the noise level with simple design.

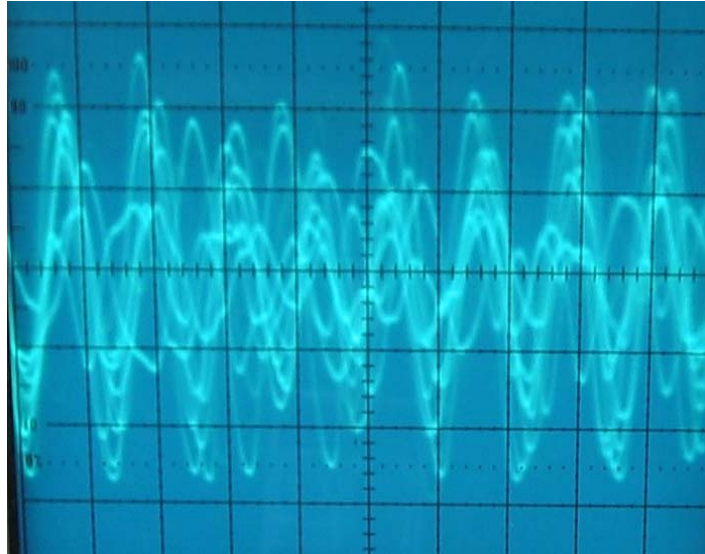


Figure 5.26: Noise level measured in ultrasound receiver (50mV/div and 20us/div)

Figure 5.27 shows the peak voltage and noise level to examine the signal-to-noise ratio. Received signal level ranges from 1.8 V at 1 meter to 0.3 V at 15 meters, and it decreases almost linearly, because the physical characteristic of ultrasound shows that the power of ultrasound propagation is inversely proportional to the square of the distance from transmitter and receiver, and the received voltage, in turn, is proportional to the inversely linear proportion to the distance. Because many signals have a very wide dynamic range, SNRs are usually expressed in terms of a logarithmic decibel scale. If the signal and the noise are measured at the same impedance, then the SNR can be obtained by calculating 20 times the base-10 logarithm of the amplitude ratio, expressed in the following equation:

$$SNR = \frac{P_{signal}}{P_{noise}} = \left(\frac{A_{signal}}{A_{noise}} \right)^2 \quad (5.2)$$

$$SNR(dB) = 10 \log_{10} \left(\frac{P_{signal}}{P_{noise}} \right) = 20 \log_{10} \left(\frac{A_{signal}}{A_{noise}} \right) \quad (5.3)$$

where,

P: average power

A: RMS amplitude

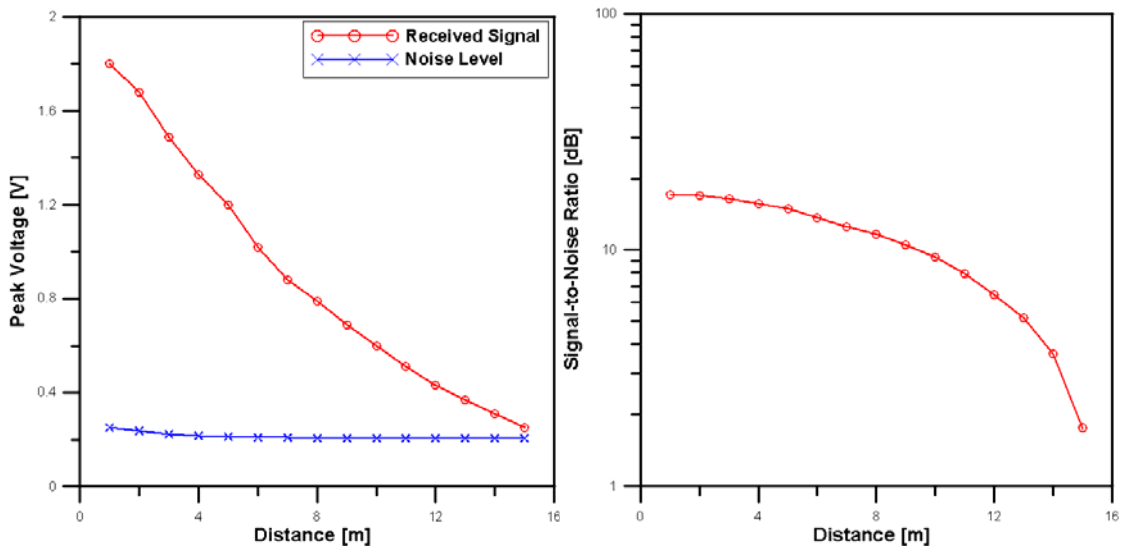


Figure 5.27: Plot of received signal level and noise level (left) and signal-to-noise ratio (SNR) in decibel

SNR measured at 1 meter distance between a beacon and a remote node indicates 17 dB, while 1.8 dB is measured at a 15 meter distance. As is mentioned before, a lower SNR results in a lower detection rate of the receiving ultrasound signal at ADC because ADC at the given threshold, with lower SNR, may be confusing as to which one is a received signal or a noise. This fact is also found in the experimental test where 60% of

ultrasound signals could not be detected at 15 meters. One simple, easy solution to be considered in order to increase the detection rate: increasing the power supply voltage at the transmitter. A higher voltage, thus an increased SNR, can be detected by the receiver at 15 meters by increasing the supply power at the transmitter. It should be noted that a decrease of detection ratio is not critical to the measurement accuracy because a detection ratio is determined by not only SNR, but also timeout value in the detection algorithm. Timeout occurs when the ADC cannot detect the received signal even after the designated numbers of cycles in pulse wave have passed through the ADC, and this scheme is a way to avoid the unwanted signal received at the ADC during the time when the ADC port is opened. The measurement accuracy is more dependent on the ADC performance, e.g. ADC sampling rate, which will be discussed later.

5.4.1.3 Analysis on Accuracy and Errors

Per aforementioned system configuration, experimentation on distance estimation was performed to obtain measurements starting from distances between 1 meter to 15 meters. Figure 5.28 shows the measurement results with 40 to 120 samplings in each location. Overall, a high density of distance measurements is obtained around the actual distance with the variation ranging from 0 cm to 20 cm. The largest variation of 50 to 70 cm occurred seven times among more than 1000 samplings, which is only around 0.7% probability. Figure 5.29 describes the percentile errors and error standard deviation measured in each location, and almost 80 percentile of measurements at each location show the error at less than 20 cm. In addition, averaged standard deviation shows 9.7 cm, which is fairly accurate in distance ranged from 1 to 15 meters.

It is worth mentioning that the error rate increases as the measured distance increased with the higher rate of timeout occurrence. The error rate relies on the signal detecting mechanism in the ADC because the high quality of ADC is not provided in MICAZ. This means that the low sampling rate of ADC may fail to detect signal values even greater than threshold. A delayed process of signal detection then causes the time shift of arrival, resulting in an error in distance estimation. If the distance goes to 15 meters, decreased signal strength lowers the SNR so that a decreased performance is expected due to failures of signal detection, which can further cause ADC timeout.

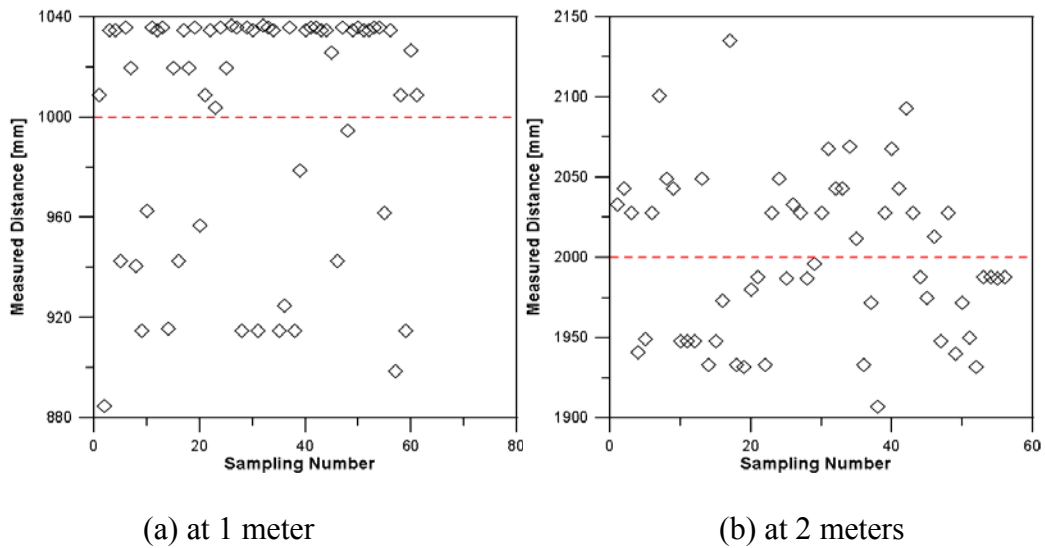
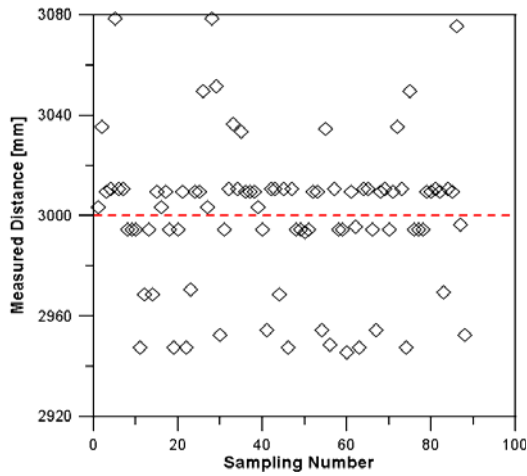
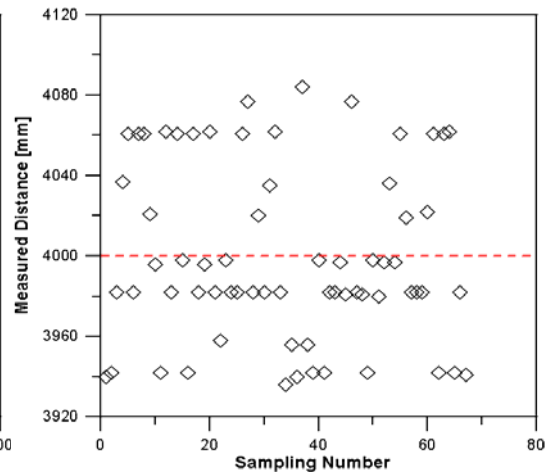


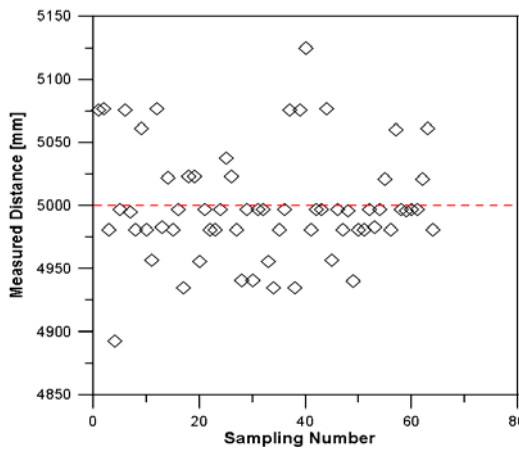
Figure 5.28: Measurements of distance that is varying from 1 to 15 meters



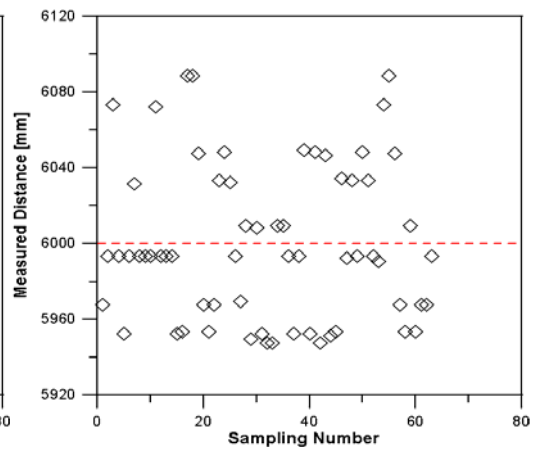
(c) at 3 meters



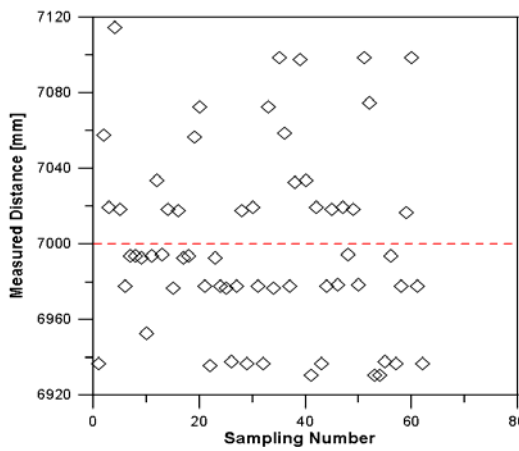
(d) at 4 meters



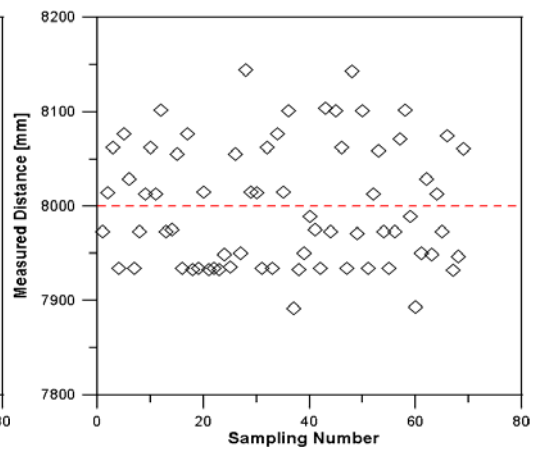
(e) at 5 meters



(f) at 6 meters

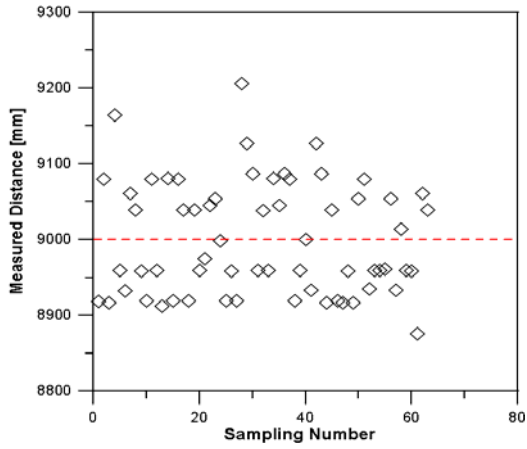


(g) at 7 meters

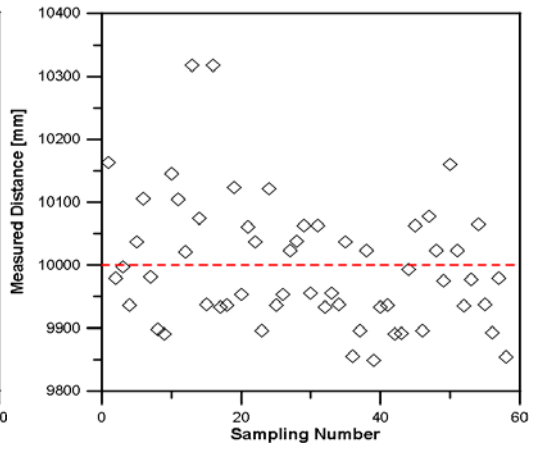


(h) at 8 meters

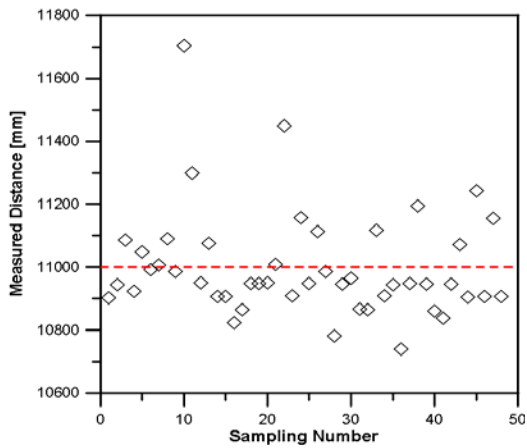
Figure 5.28: Measurements of distance that is varying from 1 to 15 meters (cont.)



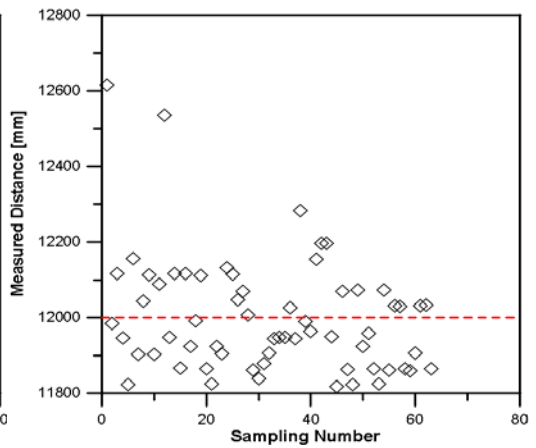
(i) at 9 meters



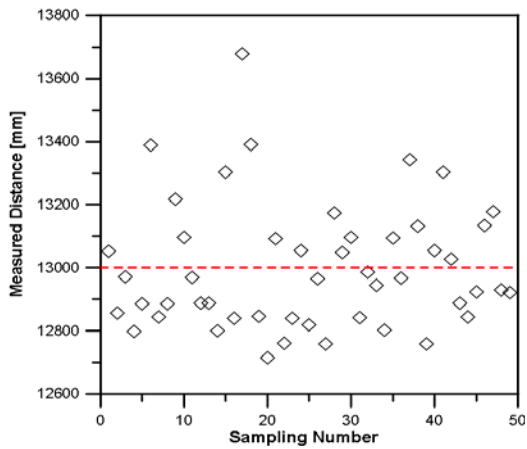
(j) at 10 meters



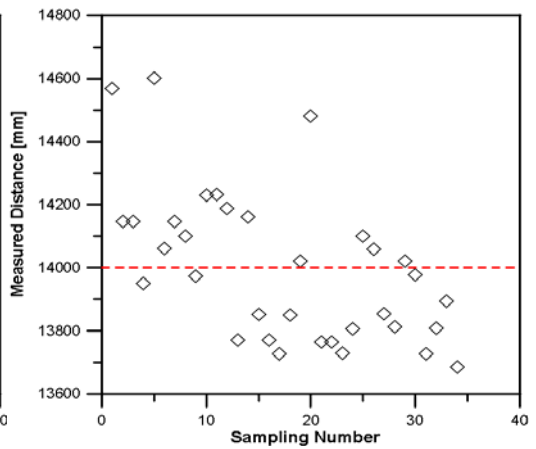
(k) at 11 meters



(l) at 12 meters

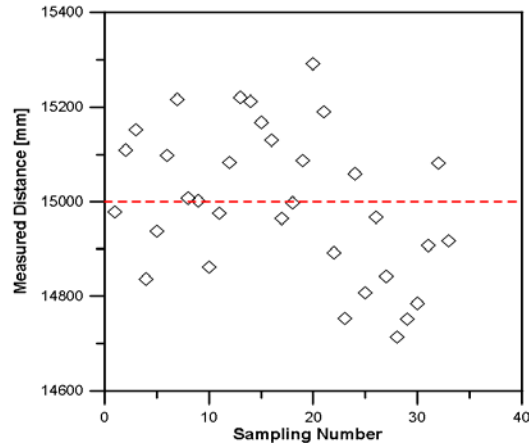


(m) at 13 meters



(n) at 14 meters

Figure 5.28: Measurements of distance that is varying from 1 to 15 meters (cont.)



(o) at 15 meters

Figure 5.28: Measurements of distance that is varying from 1 to 15 meters (cont.)

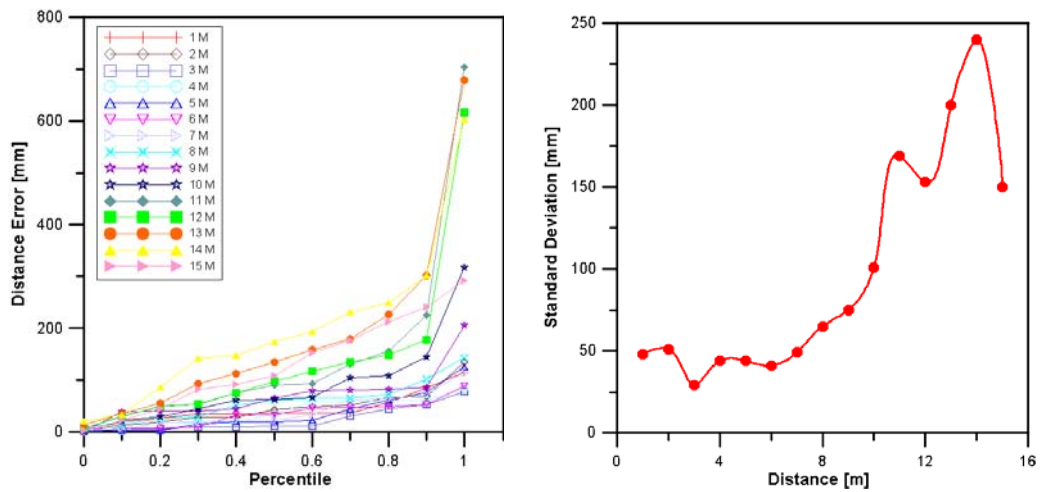


Figure 5.29: Plot of percentile errors and standard deviation

5.4.1.4 ADC Performance

The ATmega128 in MICAZ equips a 10-bit ADC port, which is connected to an 8-channel Analog Multiplexer, allowing 8 single-ended voltage inputs from pins of port F. Usually, the ADC takes 13-260 μs as a conversion time that provides up to 15 kilo

sampling per second (kBPS). A single conversion is driven by invoking a logical one to the ADC Start Conversion bit, ADSC, and a normal conversion takes 13 ADC clock cycles. Then, the first conversion after ADC takes 25 ADC clock cycles to allow the initialization of the analog circuitry. The actual sample-and-hold occurs 1.5 ADC clock cycles after starting a normal conversion and 13.5 ADC clock cycles after the start of the first conversion. The ADC data registers are used to write the results after the completion of conversion, and the ADSC is cleared simultaneously in single conversion mode. After resetting the ADSC again, the new conversion process is invoked on the first rising ADC clock edge. Figure 5.30 and 5.31 illustrate the ADC timing diagram used in MICAZ.

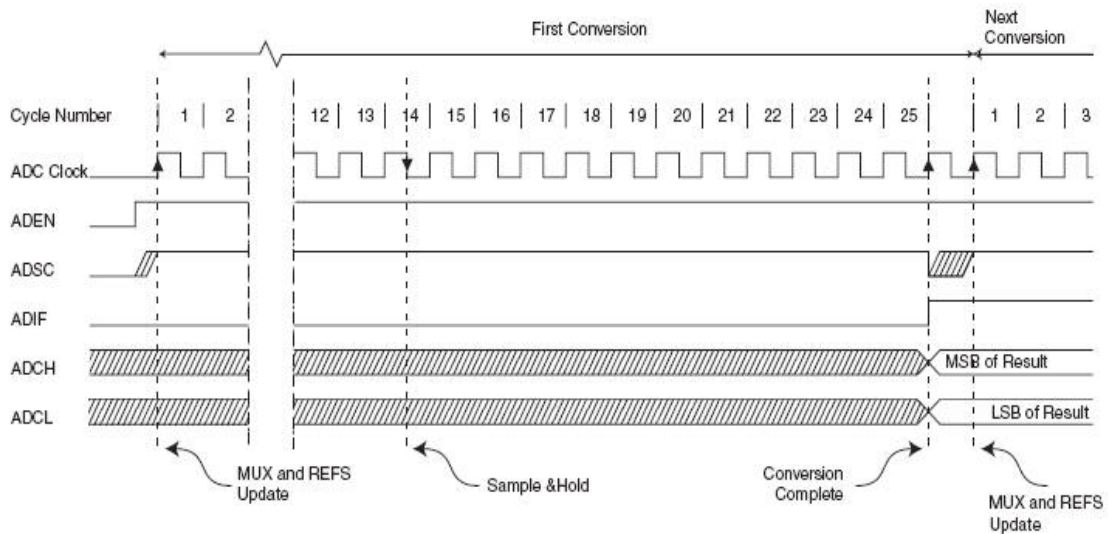


Figure 5.30: ADC timing diagram in single conversion mode (first conversion) [Atmel Corporation, 2007]

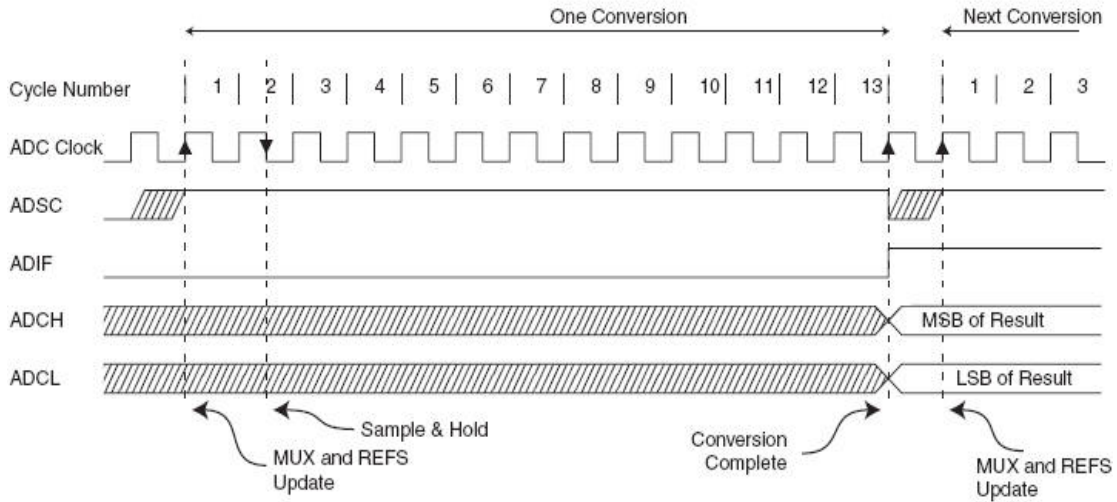


Figure 5.31: ADC timing diagram in single conversion mode (normal conversion) [Atmel Corporation, 2007]

Table 5.1 describes the ADC conversion time used in MICAZ. $30 \mu\text{s}$ is set to the ADC clock cycle in US_TriggerM.nc, and the 13 cycles required for normal conversion thus takes $390 \mu\text{s}$, which is 15.6 times that of the 40 KHz ultrasound signal ($25 \mu\text{s}$) generating a maximum 13 cm per cycle. In addition, 750 cycles of detecting period for the received signal are used in the US_TriggerM.nc as a threshold for timeout, and the maximum possible errors of distance due to timeout scheme can be up to 7.74 meters, when ADC detects the signal at the last cycle of detecting signal in the worst case. As a consequence, measurement errors in distance estimation cannot be avoidable due to the mechanism of conversion process if we adopt the internal ADC embedded in the MICAZ platform. Improved external ADC device, with higher sampling rate, will be needed to increase the performance of distance estimation. This work is beyond the scope of this dissertation, remaining as a future research topic.

Table 5.1: ADC conversion time in ATmega128 [Atmel Corporation]

| Condition | Sample & Hold | Conversion cycles |
|----------------------------------|---------------|-------------------|
| First conversion | 13.5 | 25 |
| Normal conversions, single ended | 1.5 | 13 |
| Normal conversions, diferential | 1.5/2.5 | 13/14 |

5.4.2 Trilateration

Trilateration is a basic technique, capable of deriving the coordinates of an object using three known positions, and is widely used with triangulation in typical survey technologies. This method is more efficient when three values of distance from an unknown object and three known points are given accurately. The approach implemented in this section focuses on measuring distances using known information of beacon positions rather than the triangulation approach, because more challenges exist with measuring the angles between beacons and a remote node. As a consequence, implementation of estimating the remote node's position is based on the simple, accurate method of trilateration using three beacons.

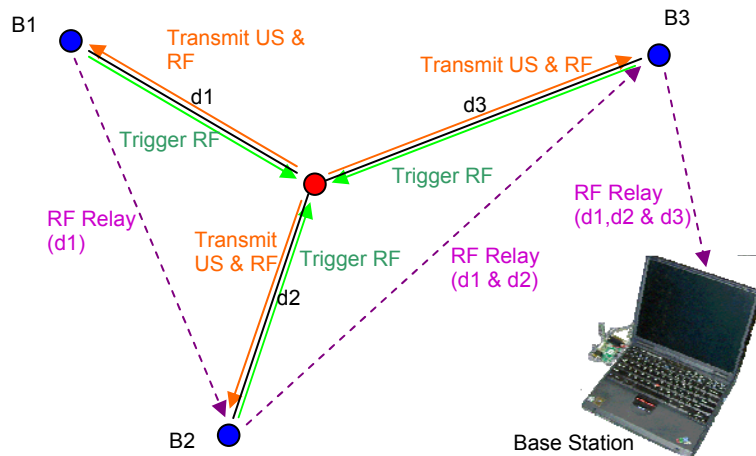


Figure 5.32: System configuration of trilateration and RF relaying scheme to determine the position of a remote node

Figure 5.32 illustrates the system configuration for trilateration in determining the position of a remote node, demonstrating a possible deployment scenario for a practical application, such as material tracking on a construction site. Distance estimation for d_1 , d_2 , and d_3 is achieved by the approach expressed in the previous section using RF and US. However, the communication between each beacon is achieved by a RF relaying scheme in which the information obtained from each periodic set of distance measurement is carried by radio packet. Finally, the RF relaying message arrives at the base station node that collects the information regarding the distances d_1 , d_2 , and d_3 , and their recipients, e.g. beacon ID, in every relaying cycle. The trilateration is performed through a laptop that is connected with the base station node, with the remote node updated every five seconds. Figure 5.33 shows the ultrasound devices which are externally connected to MICAZ wireless modules.

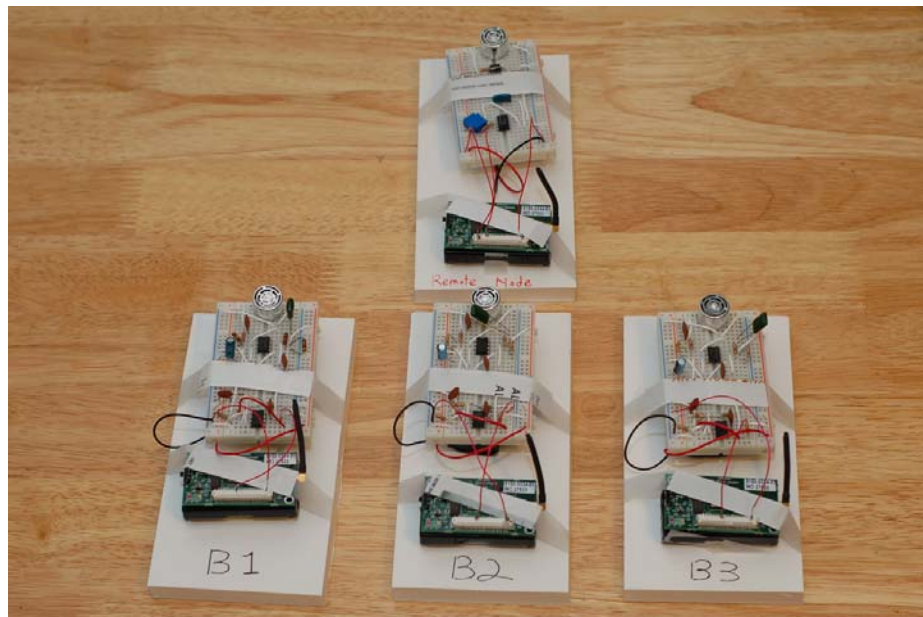


Figure 5.33: Hardware configuration of three beacons and a remote node

For the visual representation, we developed a java application that displays the position of the remote node in every relaying cycle. Serial connection between the base node and laptop enables data packets to be forwarded to a java platform that implements the calculations of the trilateration. The java application presents the position updates by drawing a red dot representing the measured position of the remote node. Figure 5.34 provides a screenshot of the java application illustrating the visual representation of three fixed beacons and a remote node.

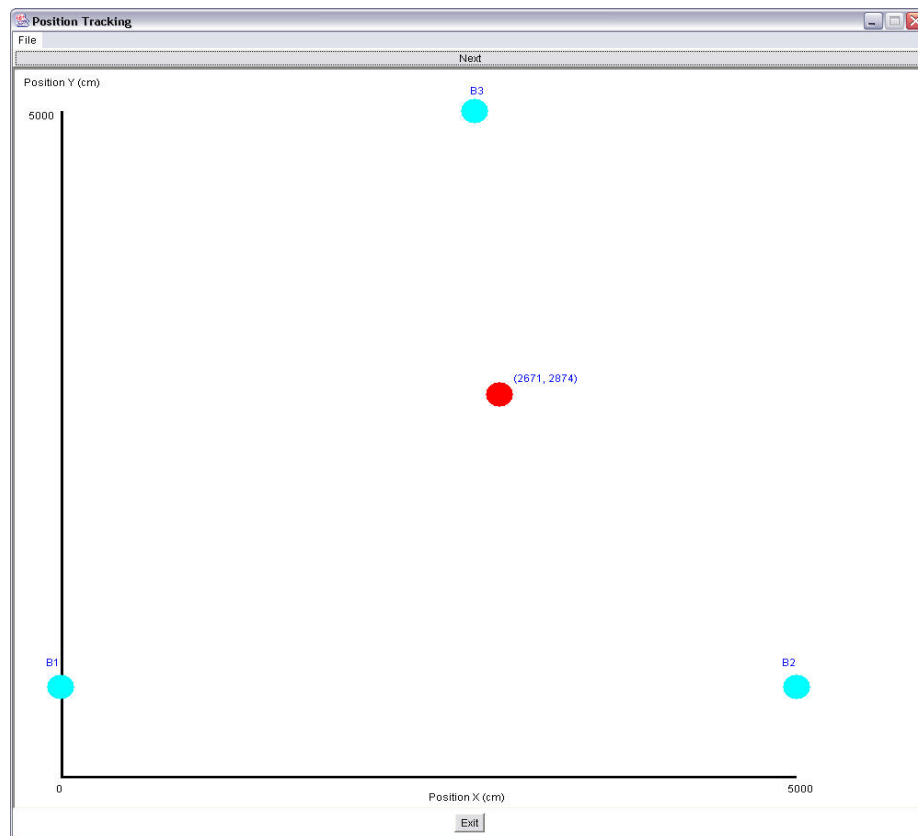


Figure 5.34: Java application for visual representation of beacons and a remote node

In order to implement the accuracy performance, three beacons are positioned along with the triangular shape, with 5 meter distance from each other, and a triangular

path with a 1.3 meter length used for measuring the position of the remote node that is circling the triangular path. The position of the remote node is measured and tracked at each vertex point of the triangular path every 5 seconds until it circles the path 30 times. The measured path and actual path is shown in figure 5.35, where the trace of blue line represents the measured position and red line represents the actual path. The measured position of the remote node shows the dense population around the actual vertex points, with estimated deviation of position ranging from zero to 133 mm with respect to the radius. Figure 5.36 describes the error estimation where average deviation of position is measured ranging from 43 mm to 47 mm in radius, and maximum errors are measured with 85 mm at B1, 133 mm at B2, and 76 mm at B3 in radius. Standard deviation also shows accurate measurements ranging from 18 mm to 29 mm in radius.

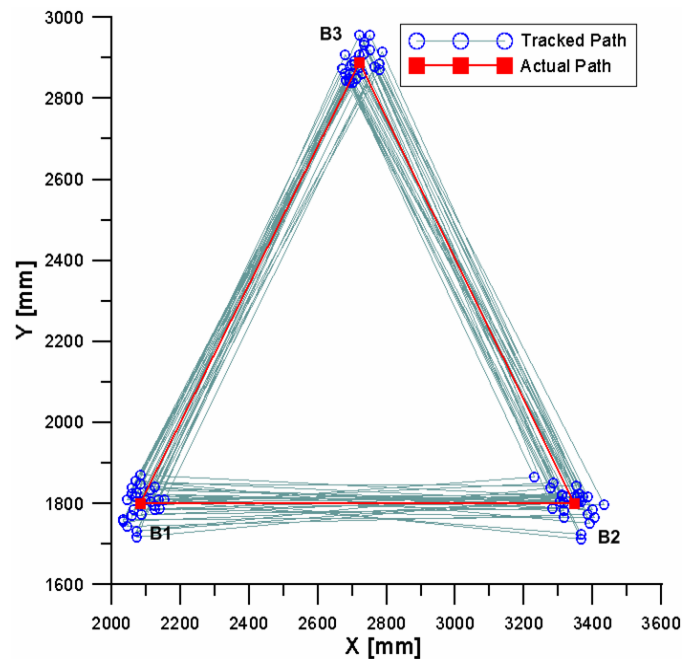


Figure 5.35: Measuring the position of a remote node that turns around a triangular path with 1.3 meter from each vertex

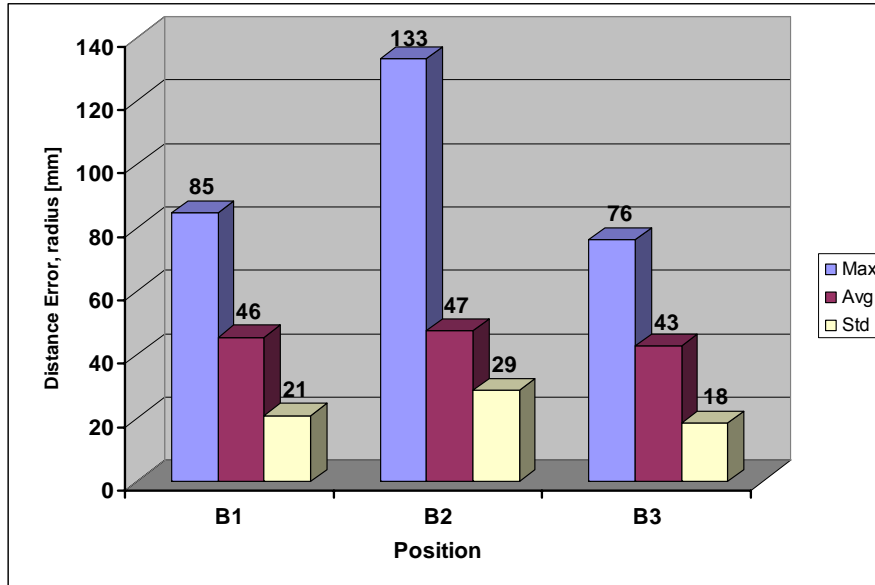


Figure 5.36: Distance errors in radius (unit: mm)

Overall, the results of position estimation provide notable accuracy that could draw interest and motivation for possible deployment in industrial practices, because less than 13 cm may give a quite allowable resolution of accuracy in many large-scale construction sites, such as tracking of construction materials, equipment, assets, and labors. As a result, the methodology for distance and position measurement demonstrated in this dissertation provides a prototyping illustration for possible application of tracking construction components with better accuracy and advanced networking capability. One key note that should be mentioned after this prototyping experiment, however, is the performance associated with the signal detecting rate. As the distance between a beacon and a remote node increases, the signal detecting rate decreases due to low signal strength, resulting in the ADC timeout. Because of this fact, the actual signal detection of ultrasound wave was achieved in almost every two relaying cycles, and the low detection rate could cause delays when a prompt tracking application,

such as a path-finding robot system, is considered. Although, a low detecting rate is not a critical factor that affects the measurement accuracy, it is necessary to design the improved circuit and software architecture for better performance and applicability; which is a topic for further research in near future.

5.5 Chapter Summary

Feasibility studies described in Chapter 4 drove to developing a new tracking framework based on the combination of radio and ultrasound signals. Based on the methodology presented, embedded systems including hardware and software have been developed. An external ultrasound device was designed to support the added functionality of distance and position estimation because the current technology of wireless sensor modules provide limited features, such as wireless communication and general sensing capability. Those generic functionalities in the modules, e.g. MICAZ, cannot perform the localization scheme presented in this dissertation, such that it is necessary to embed the external ultrasound device with the MICAZ platform to function as a tracking system for possible applications in positioning framework. Detailed hardware and software architecture have been examined to make the embedded system realized for the purpose of construction material tracking. After conducting the lab test with the system, field experiments have been implemented with respect to distance and position estimation. The experimental results showed notable improvement of measurement accuracy and networking flexibility, indicating that the positioning accuracy ranges from a mean of 4.3 to 4.7 cm among 30 samplings. Individual distance estimation indicated that 90 percentile of measurements among more than 1,000

samplings ranges from 3 to 25 cm up to a ranging distance of 15 meters. Unlike the proximity approach, trilateration method gives the geographic coordinates of mobile objects with increased accuracy. In addition, position accuracy with less than 5 cm resolution was superior to many GPS technologies, in which an average 10 cm error variation has been observed in even differential GPS (DGPS). However, some limitations may arise in the issues that include the signal detecting rate and ADC timeout. The challenges associated with the performance limitations caused a small variation of accuracy and a limited range of distance. Future research should be conducted to improve the performance on the signal detection mechanism, and to provide a better alternative for practical applications of materials tracking in construction sites.

CHAPTER 6: COST-BENEFIT ANALYSIS

Previous chapters explored technological solutions to current problems observed in construction material tracking practices, and a new approach with embedded wireless sensor architecture was proposed to illustrate a deployment scenario for the application. In order to verify the application feasibility, both analytical and experimental studies were implemented using the proposed embedded system, developed for distance and position estimation; key features of tracking construction materials. Performance and accuracy analysis of the embedded sensor system showed that the proposed system architecture provides increased performance in accuracy and networking capabilities that could bring a possible solution over the traditional practices of material tracking systems. Based on the findings, this chapter examines the expected cost benefits associated with the deployment of this technology in actual construction areas by addressing both quantitative and qualitative approaches. Finally, possible deployment scenarios with the proposed embedded system are examined to illustrate solutions to various aspects of the problems with many construction processes.

6.1 Introduction

Deployment of a new technology to specific applications requires an extensive effort to obtain the best solution among all possible alternatives. Usually, the technical feasibility does not always cope well with the anticipated cost-benefits because setting the obtainable profitability from the deployment requires much more consideration than one may expect. Traditional consideration for cost-benefit typically includes financial costs and return-on-investment after the deployment of new technology. However,

today's advancement in information technology makes it difficult to estimate the cost-benefit from only financial costs and ROI. This is because the expected benefits are also governed by technological trends in the market place, interoperability with different software/hardware platforms, and immeasurable values of profit from the use of the technology.

The demands of today's industry are often observed in the current situation where different functionalities of technology in construction material tracking often have different benefits, directly or indirectly, by their deployment. This observation implies that the challenges are on the direct comparison among several alternatives that may incur the inconsistency to generalize their expected benefits. For example, Wi-Fi (IEEE802.11) may be the best choice where internet protocol can take an advantage to use a big storage of database server, to share the information among project stakeholders, or to access to the information sources at remote places, but it is not a good candidate for steel structure crews working with Wi-Fi compatible handheld devices on a high-rise building. Another example includes the RFID technology that may be an excellent candidate for manufacturing tracking and scanning, vehicle scanning in a parking gates, and airport security systems. Despite excellence in scanning and tracking capability, it is not well fit to large scale positioning systems using proximity method, such as tracking systems on construction sites, asset tracking in military applications, and positioning systems requiring accuracy and scalability due to the inflexible networking feature of RFID readers.

The development of new technologies may not always meet the necessities and demands to solve the specific problems observed in the construction sites. Most

technology developments are targeted to the generalized products because of the vendor's marketing strategy to increase opportunities for the newly developed applications. Sometimes a technology deployed may be too heavy to be applied to a construction area with respect to their functionalities. Often the construction applications deploy technology that was developed for slightly different purposes, resulting in improper use of the technology and forcing it to be customized for a specific application with wasted time and efforts. As a consequence, civil, construction, and building engineers have the responsibility of making decisions regarding the prior requirements that address the correct direction of the deployment strategy. They are the only personnel who can solve the real problems of current industrial practices, diagnose the limitations of previous technologies used, present the actual needs and capacities a new technology may provide, and determine the alternatives amongst various innovations for the problems.

The problem stated in current practices of construction material tracking has led to the development of the combination technique involving ultrasound and radio signals. In previous chapters, detailed methodology and system architecture of the proposed tracking techniques were examined, and the feasibility study and prototype implementation with experiments illustrated the advantages of the proposed scheme in accuracy and performance. With these findings and success from the implementation, expected costs and benefits will be discussed quantitatively and qualitatively to present the advantages and disadvantages during the deployment stage.

6.2 Cost and Benefit Analysis

Cost and benefit analysis is a common framework to evaluate the benefits and drawbacks associate with a particular project. Usually, cost-benefit analysis includes

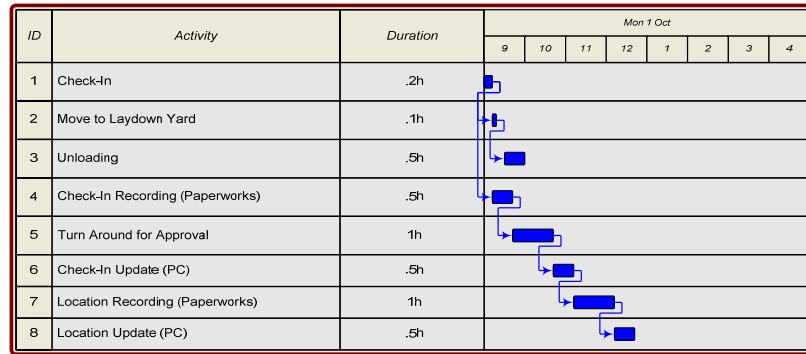
three principal approaches: 1) the quantitative approach, 2) the qualitative approach, and 3) a combination of both quantitative and qualitative approaches. Quantitative approaches are drawn from the rationalistic approaches that are exploited by data collection and analysis. On the other hand, qualitative approaches often use the interpretive philosophical view and inductive thought to determine the relationships [Amjad, 2004].

The combination of the qualitative and quantitative approaches provides an excellent illustration of when the expected benefits, such as those in information technology, often result from both measurable and immeasurable factors. Advantages with financial costs may not represent the greatest benefit from the technology deployment, because a large fraction of users will also consider other critical factors such as ease of use, comfort of use, compatibility to other systems, and technological trends; all of which reside in the qualitative criteria. The approach described in this chapter is based on the combination approach that can provide an illustration of the expected benefits during the deployment of new technology. The comparison is performed on the implementation cost among the manual process and the sensor-based processes (including RFID and the embedded system). For the quantitative benefits, labor-hour and its cost savings associated with information handling and process are compared with three different methodologies for material tracking practices based on a process cycle calculation. For qualitative benefits, intangible advantage is compared with the use of a sensor-based scenarios and traditional practices in construction material tracking, and we examine the possible, invisible benefit obtainable from the deployment of the technology.

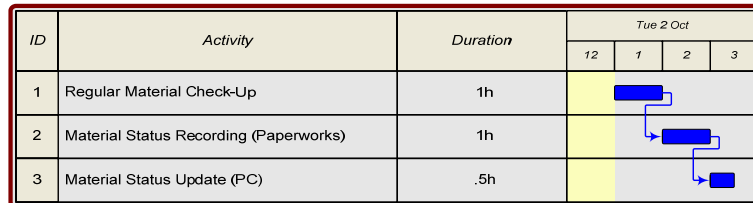
6.2.1 Quantitative Approaches

Quantitative approaches to cost benefit analysis involves the use of explicit monetary values and processing times associated with field material management. However, many material management systems in construction fields cannot represent definite values because actual lengths of processing time in field material management differ between sites, type of projects, construction time, and physical locations. Also, the difficulties with determining required labor associated with material tracking pose a challenge in defining the actual process time. Accordingly, the method of expert interviews with various construction & engineering companies would be a good source to determine the monetary values and process time of material tracking implementation because expert experiences would provide a critical reference for the process information. By summarizing these interviews, field material tracking could then be represented by cyclic processes consisting of check-in cycle, daily checkup cycle, and installation cycle. Although actual material management includes complicated activities, such as requests for quotation (RFQ), bid documents, purchase orders (PO), material delivery, materials receiving report (MRR), accounting documents, payments [Lee et al., 2004], the quantitative approach in this chapter defines the processes involved from material receiving to installation where the wireless sensor framework for construction material tracking functions as a tool for data collection and updates that replace the manual processes.

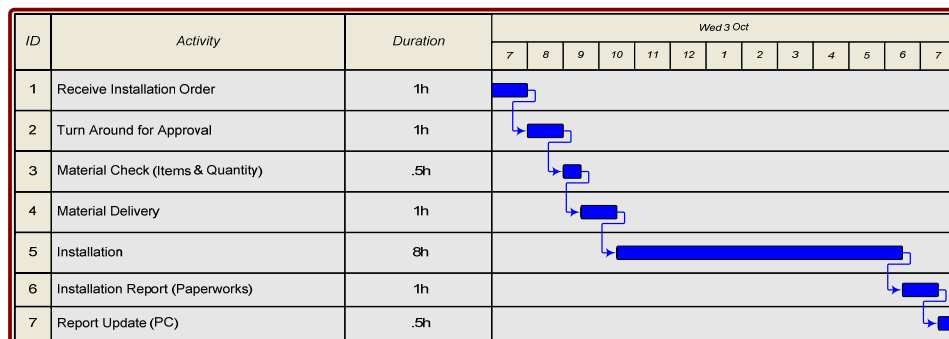
Scenario-Based Tracking Simulation



(a) Check-in cycle related to the materials reception and unloading



(b) Daily checkup cycle related to the daily check-up the status of materials



(c) Installation cycle related to delivery and installation process

Figure 6.1: Three cyclic procedures associated with the material tracking and handling

The three cyclic processes, shown in figure 6.1, represent materials management processes at a simulated construction site. It is assumed that the listed processes include

only in-site activities and preprocesses such as RFQ, manufacturer's handling, shipping, etc. are not considered. Installation activities, that initiate RFQ, are performed during the designated timeframe succeeded by the material PO and delivery, and materials tracking process in site is often followed by a repetitive cycle. As a consequence, information about total material quantities and their installation processes are determined, it is then possible to estimate the material delivery cycle for partial or entire project duration.

The break-down of activities in each cycle represents the typical procedure responsible for material tracking labor in most construction projects. The procedure is initiated at material check-in and completed at report update after installation. During each cycle, the activity duration is expressed as the number of labor hours per cycle. While the actual numbers for activity durations in real construction projects differ among project type, time, location, and personnel, it is possible to obtain the generalized number of the durations since they are verified by experts from different areas of the construction industries.

The findings from interviews of more than twenty experts described the detailed activity procedures, duration, resources, and crew's responsibility associated with the materials management in a typical construction project. Figure 6.1 illustrates the three process cycles with break-down activities from check-in to installation, and the unit labor hours were obtained by comparing the manual-based process and sensor-based process (including both RFID and embedded systems). It is reasonable to divide the cycle into three parts because the timeframe required in each cycle is different. For example, the check-in cycle copes with the delivery cycle, the time by which available trucks take to carry the materials from a manufacturer's site to a construction site. Furthermore, the

delivery schedule is followed by the installation schedule and material PO, in which the required amount of materials are determined. The daily checkup cycle represents the cycle required for a material crew to daily check the material storage about the material status, items, quantity, and locations. This daily check-up is typical material management conducted in almost every construction site because information about the material status is important when prompt actions for hunting or installation of materials are needed in a specific situation. The daily checkup cycle is determined by the time required for the materials to be stored between delivery and installation, so enough but not too long a period of time should be allocated to avoid any process delays in a limited laydown yard. The installation cycle represents the activities that are incurred when the installation order is placed, and is determined by the installation schedule per each day.

The unit labor hours required to conduct each cycle and the comparison between manual and sensor based methods are shown in table 6.1. It is important to note that activities such as reporting, paperworks, data updates, and daily check-ups are considered labor-intensive tasks that may be removed when a sensor-based method is applied. The rationale behind this is that a RFID or embedded sensor system could automate material check-in, recording the material status, paperworks for approvals, and data updates without labor activities. As a consequence, the deducted amount of working hours from a sensor-based method represents the savings of unit labor hour in each cycle associated with the material and information handling. According to the figures, labor hour in check-in, daily checkup, and installation cycle could be improved by savings of 2.7 hr/cycle, 3 hr/cycle, and 3 hr/cycle, respectively.

Table 6.1: Estimation of unit working hour in three process cycles

| Category | Manual | | | | Wireless Sensor | | | |
|---------------------|------------------------------------|----------|---------------|-------------------------|--|------------|--------------|--------------------|
| | Activity | Duration | Resources | Function | Activity | Duration | Resources | Function |
| Check-In Cycle | Check-In | 0.2 | Crew | Reception | <i>Check-In</i> | | Sensors | Reception |
| | Move to Laydown | 0.1 | Truck | | Move to Laydown | 0.1 | Truck | |
| | Unloading | 0.5 | Crane | | Unloading | 0.5 | Crane | |
| | Check-In Recording | 0.5 | Crew | Counting and Item Check | <i>Check-In Recording</i> | | Sensors | Automatic Record |
| | Turn Around for Approval | 1 | Crew, Manager | Approval Signature | Turn Around for Approval | 1 | Manager, PC | Approval Signature |
| | Material Reception Update | 0.5 | Crew | Typing to PC | <i>Material Reception Update</i> | | Sensors | Automatic Record |
| | Paperwork for location information | 1 | Crew | Update Location Log | <i>Paperwork for location information</i> | | Sensors | Automatic Record |
| | Material Location Update | 0.5 | Crew | Typing to PC | <i>Material Location Update</i> | | Sensors | Automatic Record |
| | Subtotal | 4.3 | | | Subtotal | 1.6 | | |
| | | | | | Working Hour Saving per Cycle | 2.7 | | |
| Daily Checkup Cycle | Daily Material Checkup | 1.5 | Crew | Counting and Item Check | <i>Regular Material Checkup</i> | | Sensors | Automatic Record |
| | Paperwork for Material Status | 1 | Crew | Update Status Log | <i>Paperwork for Material Status</i> | | Sensors | Automatic Record |
| | Status Update | 0.5 | Crew | Typing to PC | <i>Status Update</i> | | Sensors | Automatic Record |
| | | Subtotal | 3 | | | Subtotal | 0 | |
| | | | | | Working Hour Saving per Cycle | 3 | | |
| Installation Cycle | Installation Order Received | 0.5 | Crew | Order Reception | <i>Installation Order Received</i> | | Sensors | Automatic Record |
| | Turn Around for Approval | 1 | Crew, Manager | Approval Signature | Turn Around for Approval | 1 | Manager, PC | Approval Signature |
| | Material Check for Items & Qnt. | 1 | Crew | Counting and Item Check | <i>Material Check for Items & Qnt.</i> | | Sensors | Automatic Record |
| | Material Delivery | 1 | Truck, Crane | | Material Delivery | 1 | Truck, Crane | |
| | Installation | 8 | Crane | | Installation | 8 | Crane | |
| | Installation Report | 1 | Crew | Update Installation Log | <i>Installation Report</i> | | Sensors | Automatic Record |
| | Data Update | 0.5 | Crew | Typing to PC | <i>Data Update</i> | | Sensors | Automatic Record |
| | Subtotal | 13 | | | Subtotal | 10 | | |
| | | | | | Working Hour Saving per Cycle | 3 | | |

With the findings from the savings of the unit labor hour, the quantitative approach for the cost benefit analysis was conducted on an example construction project. A State of Massachusetts investigation determined that a highway bridge in Boston, MA built in 1970 required superstructure replacement after structural evaluation. The bridge had a stringer/multi-beam steel girder structural type with 6 lanes of traffic, and a total span length and a span width of 310.3 meters and 24.4 meters, respectively. Current steel girder and concrete decks should be replaced by a steel girder and fiber-reinforced polymer (FRP) deck after preliminary design of the superstructure. The overview of the bridge is described in table 6.2.

Table 6.2: Overview of a bridge in the State of Massachusetts requiring superstructure replacement

| Bridge Description | |
|-----------------------|---|
| Structural Type | Stringer/Multi-beam Steel Girder Bridge |
| Service on Bridge | Highway |
| Service Under Bridge | Highway-waterway |
| Traffic Lanes | 6 |
| Daily Traffic | 70770 |
| Total Span Length (m) | 310.3 |
| Span Width (m) | 24.4 |
| Operating Rate | 53.5 tons |
| Number of Spans | 9 |
| Girder Material | Steel |
| Deck Material | Concrete |

Quantity take-off for the superstructure replacement was conducted based on the bridge drawings. It should be noted that this approach used the typical bulk materials, e.g. girder, decks, drainage pipes, and bracing structure, for the sensor-based materials

tracking application, even though the entire materials may include tiny entities, e.g. bolts, expansion joints materials, cements, connection rings, etc. Since the installation of those bulk materials represents between 70-90% of the total project activities, specifically in bridge superstructure replacement, such small and less critical materials were excluded from this calculation. An I-shape steel girder with 35 meter length and varying height of 1.3 to 1.8 meters is used, and I-shape steel beams with 3 meter length and a height of 0.5 meters were used as lateral bracing structures. The total bridge length of 310.3 meters with 9 spans required a 35-meter I-shape girder, so a total of 79 girders including contingency rate were used. For the lateral bracing steel beam, 485 steel beams were used. For durability and a cost-efficient method, a pre-cast FRP deck of 7-meter length with 24-meter width was scheduled to be fabricated from a manufacturer, and a FRP deck installation required 50 decks including contingency rate. Finally, D30 with 6-meter length PVC pipes were used for drainage, and installed beneath each rail, requiring 683 pipes. Quantity take-off summary of the typical bulk materials for the superstructure replacement project is presented in the table 6.3.

Table 6.3: Quantity Take-Off of major bulk materials for superstructure replacement

| Item | Dimension (m) | Quantity* | Unit |
|----------------------|------------------|-----------|------|
| I-shape Steel Girder | 35L×1.3H to 1.8H | 79 | EA |
| Cross steel bean | 3L×0.5H | 485 | EA |
| FRP Deck | 24W×7L | 50 | EA |
| Drainage Pipe | D30×6 | 683 | EA |

*quantity includes 10% contingency

Table 6.4: Cycle estimation for typical bulk materials

| Item | Check-In Cycle | Installation Cycle |
|----------------------|---|---------------------|
| I-shape Steel Girder | $79/(2 \text{ EA} \times 1 \text{ day})=40$ | 9 spans/1 SPD*=9 |
| Cross steel beam | $486/(10\text{EA} \times 1 \text{ day})=49$ | 9 spans/0.28 SPD=32 |
| FRP Deck | $50/(1\text{EA} \times 1 \text{ day})=50$ | 9 spans/1 SPD=9 |
| Drainage Pipe | $683/(50\text{EA} \times 1 \text{ day})=14$ | 9 spans/1 SPD=9 |
| Total Cycle | 153 | 57 |

* SPD: span per day

The cycle estimation was performed by considering each individual cycle of Check-in, Daily Checkup, and Installation. The check-in cycle may be calculated based on the delivery schedule for specific activities, and the numbers of cycle were obtained based on data in table 6.4. Because installation activities were implemented by individual span construction, a one-day duration was assumed for them. Accordingly, 8-girders with 49-lateral beams (7-beams at 5-meter length), 5-FRP bridge decks, and 12-PVC pipes were installed in individual span construction.

Cost Analysis

Three project durations of 3, 12, and 24 months were considered to compare the savings of labor associated with the material and information handling among three different methods, e.g. manual, RFID-based, and embedded tracking frameworks. It should be mentioned that device purchases were considered in implementation cost, because the infrastructure for a sensor-based network system should be executed before actual implementation. Thus, a 50-by-50 meter laydown yard was selected from a total size of 100-by-50 meter yard to store the bulk materials, and it was assumed that the

RFID and the embedded system had the same capacity, including the same tracking function and coverage range for consistent comparison between them. A 10-meter coverage range was applied to both sensor-based methods, and a total of 16 beacons were placed to configure the network for tracking remote nodes attached to the bulk materials. Since the actual delivery follows the installation schedule, total 57 cycles of installation requires an installation order that determines the actual number of remote nodes to be tracked. This fact explains why the entire bulk materials cannot be stored in the limited laydown yard, so the actual materials is delivered according to the installation cycle requiring just 57 remote sensors for tracking. Those remote nodes may be detached and reattached whenever new bulk materials are checked in.

Total implementation costs of the three different tracking schemes are described in figure 6.2. Current market places for device purchases require RFID tags costing 10 cents each, and a UHF (900 MHz) or HF (13.56 MHz) RFID reader with 10-meter measuring distance costs from \$1,000 to \$2,500 depending on the capacity of the device [Mohamadi, 2004]. On the other hand, current MICAZ wireless sensor modules cost \$91, including some extras for US devices. It is hard to estimate the purchase price for the embedded systems because the proposed system with a US device is still in the research stage. It can be assumed that the cost of an external US device is included in the cost of MICAZ wireless sensor modules. Based on the information regarding the purchase cost, implementation cost comparison shows that RFID readers rank high in the cost of a device purchase, around \$16,000, and an embedded system requires about \$6,600 for device purchase.

Unlike the sensor-based tracking system, the manual process does not require the device purchase. However, unit labor cost increases at the manual-based material tracking stage. The implementation cost for a manual process is \$22,000 for a 3-month project, \$34,000 for a 12-month project, and \$48,000 for a 24-month project. On the other hand, the implementation costs for sensor-based material tracking remain the same at the level of \$11,000, even if the project duration increases. This is because sensor-based implementation automates the labor tasks required in the manual process, and there is no increase in labor hour under the sensor-based method. For example, recording, paperwork, check-up, order reception, and data updates are automated by the sensor-based schemes, and related labors are removed from the information handling tasks.

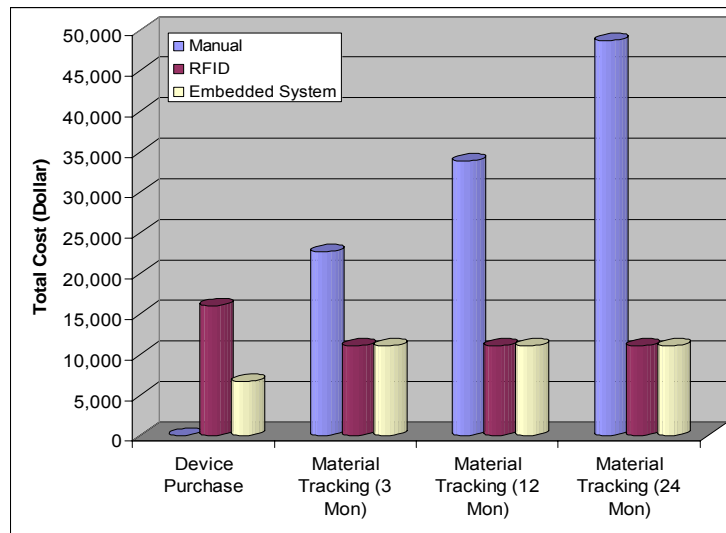


Figure 6.2: Total implementation cost² for field materials tracking compared with three alternatives

² Based on the median hourly wages for bridge construction labor: \$13.55 [Occupational Outlook Handbook, 2007]

Even if the initial cost of a device purchase requires a high budget for planning, gradual increase in labor costs associated with material handling is getting ahead of the implementation costs of a sensor-based tracking system. The labor cost increase is proportional to the entire project duration, because labor intensive tasks such as information handling impacts the project budget. Furthermore, the high cost of a RFID reader influences an adverse effect on deciding the adoption of RFID-based tracking system even though the implementation cost equals that of the embedded tracking system. Figure 6.3 illustrates the comparison between manual or RFID vs. embedded system in terms of total implementation costs, including device purchase. As the duration of the project increases, huge savings in implementation cost may be observed in the comparison of manual and embedded system frameworks. Implementation cost savings of the embedded system increase as much as the purchase cost difference between RFID and the embedded system framework. This is due to the fact that anticipated capacities are assumed as the same in the sensor-based implementation.

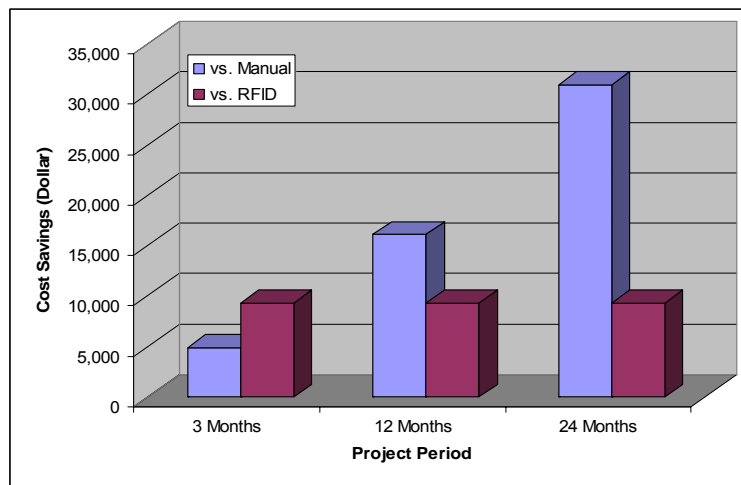


Figure 6.3: Implementation cost savings due to the deployment of embedded sensor-based tracking methods

Sometimes, direct comparison of implementation costs does not represent the actual project situation because the conditions of construction tasks vary according to many different situations. This fact often causes a challenge to quantify the cost savings and expected cost-benefit, but it is helpful to review the unit labor cost savings based on the approach presented in this section. For a more general expression, the comparison between manual or RFID vs. an embedded system is obtained by the fractional rate of percentage, which is shown in figure 6.4. This result shows that labor hours associated with the material tracking and information handling may be saved up to 64% if the current manual process is replaced by the embedded system framework. When RFID is compared with the embedded system framework, 35% savings are expected due to the initial cost of device purchase. If a large construction domain is considered, the difference between a RFID and an embedded system will increase, assuming that both devices have the same capacities in tracking the construction materials.

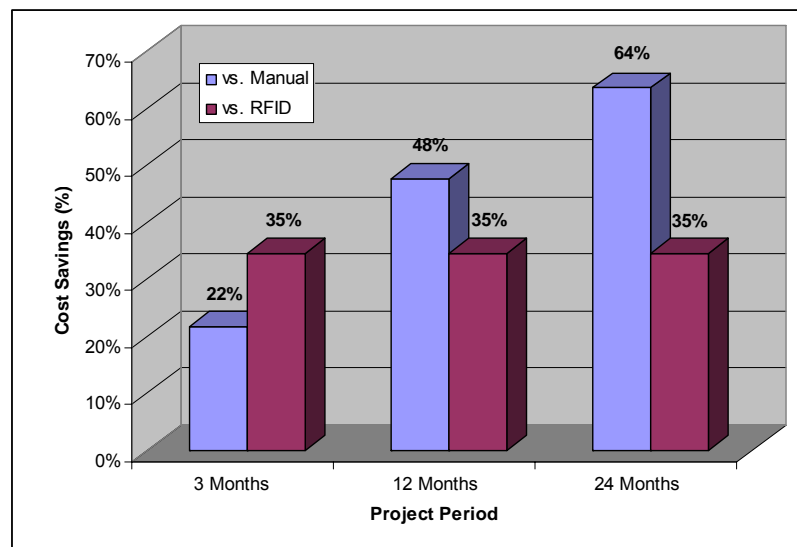


Figure 6.4: Unit labor cost savings in percent due to the deployment of sensor-based tracking methods

6.2.2 Qualitative Approach

Unseen, but tangible benefits by adopting the sensor-based material tracking play a critical role when decision makers plan a specific construction project. These comprehensive benefits, from the use of sensor-based framework may be discovered in many aspects of project practices. First of all, the most important, intuitive benefit may be the efficiency of communication. A wireless sensor-based framework provides an automated way of information communication not only in the field of data acquisition, but also improved interoperability to the central management systems when they are connected. Prompt updates of field information regarding material status, locations, quantity, etc. along with the wireless communication make it easy to manage and control the process of construction field by connecting it with the decision tools. More importantly, a particular situation where material tracking associated with the critical path activities can benefit from the automated process of installation order/confirmation and of status check-up with minimizing the delays. Today's advancement of communication technologies give a better opportunity to wireless sensor framework that could reside in a certain networking domain with different communication platforms, such as handheld devices, portable internet devices, and GPS platforms. Such kinds of communication platforms could increase their functionalities when combined with the wireless sensor network by improving the current practices of field information acquisition system.

Ease of use, less effort for documentation, cycle time reduction in approvals, and efficient utilization of labor force will also be the qualitative benefits. The aforementioned survey shows that significant time and effort will be saved during the process of location hunting, recording, updating, and paperwork in actual construction

practices. The quantitative approach makes it possible to infer the amount of unit labor savings when the sensor-based framework is adopted. Increased efficiency in those processes can also affect the improvement of the associated tasks, such as data retrieval, data representation, and data manipulation if the system dynamically interacts with the management systems. This fact may also contribute to the comfort and convenience to many end-users, which is hard to estimate the quantitative benefits from the use of sensor-based frameworks. For example, one can easily think of the expected benefit of internet technology, and everyone can agree that the internet provides comfort, convenience, fast and easy way of communication, information sharing, and work processing. At the same time, it is not easy to quantify the benefits from the use of the internet because diverse factors, such as user's age, gender, location, purpose of use, organization, etc., need to be factored into calculating the amount of benefits.

If a sensor-based framework is connected with a database system, benefits can be expected through reduction of the volume of historical archives. The costs associated with the document handling and the storage space could be saved by automating the traditional paperwork required with the construction projects. As the number of projects that a construction company has accumulated increases, extensive efforts are required for document handling and management. Additional expenses for the use of printers and copy machines may be saved by an automated data management framework as well.

Finally, a resource management system can benefit from the sensor-based framework. Most materials management systems are intertwined with other project management modules. For example, material procurement modules can be shared with the material tracking system, enabling prompt decisions on the material RFQ and PO.

Another example shows that an accounting system could be associated with the field information acquisition module in the management system where prompt updates of field information increases the efficiency associated with paper-based document handling. In the world of fast changing information technology, the framework for consistent management in work processes may be realized by an “everywhere” sensor network and the potential of the sensor network promises to reorganize standards and coordination toward the new paradigm and trends of project management. At the same time, new construction industry developments endowed with the emerging technology provides the new understanding of the implications that may deliver the efficiency and effectiveness of construction projects.

Table 6.5: Qualitative benefits by adopting a sensor-based tracking system

| Category | Qualitative Benefits |
|---------------------|---|
| Communication | <ul style="list-style-type: none"> ▪ Improved data acquisition ▪ Interoperability to management systems ▪ Prompt updates of field information |
| Operation | <ul style="list-style-type: none"> ▪ Easy of use ▪ Less effort for documentation ▪ Process time reduction ▪ Efficient utilization of labor force |
| Paperwork | <ul style="list-style-type: none"> ▪ Reduced volume of historical archives ▪ Cost savings on document handling and storage space ▪ Reduced use of papers |
| Resource Management | <ul style="list-style-type: none"> ▪ Improvement of procurement & accounting systems ▪ Enhanced information sharing ▪ Increased ownership and responsibility |

6.3 Possible Application Scenarios

The emerging wireless sensor technology promises to enable enhanced tracking and monitoring of construction components in field construction sites. Wireless data communication between the sensors opens up a range of possibilities because of the ease

and the low cost by which the sensors can be deployed. In terms of tracking capability, accurate determination of mobile entities gives a better opportunity to ranging applications, in many of construction projects, through a flexible networking scheme. With these sensing and positioning possibilities, one can quickly begin to think of the various scenarios in which such information could be used.

Construction asset tracking and monitoring could be a good candidate for the deployment strategy. As the physical size and complexity of construction projects become large, there will be increased demands to locate the construction assets including high-cost equipment, tools, and machines as well as construction materials. Limited size of laydown yards and increased numbers of construction assets imply that a wireless sensor-based tracking system helps to locate them when they are needed, or even when needed to be stored securely for a long time. Sensing capability in the MICAZ platform can also be utilized, together with the ultrasound devices that are capable of measuring environmental stimuli such as temperature, humidity, and chemical substances. For example, a humidity sensor can be attached to the bulk of a cement bag or a steel beam to sense the level of humidity in order to avoid the hardening or corrosion caused by the water in a humid environment. Other examples are demonstrated in the PVC pipe where a temperature sensor is placed to detect the temperature variance to avoid melting or any defects caused by high temperature, especially in the hot summer, allowing a field manager, with the next step of preparation, to mitigate the adverse phenomenon. The deployment benefit of this possible scenario is expected to provide not only the method of tracking the construction components but also practical way of wireless monitoring on the construction site.

In addition to construction asset tracking and monitoring, the presented tracking scheme may be deployed to various applications for safety monitoring. One of the leading causes of death to construction workers is falls on construction sites. Falling off at the perimeter of a high-rise, open floor to the ground, and falling over a precipice where backhoes dig the ground for foundation work, are examples of falls that occur on construction sites. A safety system with the mobile tracking devices attached to the construction workers can trigger a warning when the workers are in hazardous locations or acting in an unsafe manner [Kim, et al., 2007]. Safety of workers from mechanical equipment is also an important issue in the mitigation of construction accidents. Even if a driver controls a backhoe in a skillful manner, it could be under dangerous situation where the field workers, concentrating on their specific tasks, are placed within the working range of the backhoe. A safety warning system can alert workers located in the working range of the backhoe if the presented positioning system measures the exact distance between workers and backhoes.

A safety warning system is important in many construction sites, and traditional safety systems have used either CCTV or GPS. Unlike the GPS tracking system, the presented framework, based on the combination of RF and US, gives a better applicability with flexible, easy networking configuration that expands the networking domain into a large scale, even in an indoor system. This is because of the fact that most of GPS tracking systems require a line-of-sight between objects and satellites, and applicability will be limited when indoor safety tracking is required inside a building. Enhancements with GPS can be found specifically in the differential GPS (DGPS) that uses a network of fixed ground based reference stations to broadcast the differences

between the positions, measured by satellite and the known fixed positions. Despite the fact that the DGPS provides improved features that enable tracking of the object even in an indoor environment, extensive efforts, including additional implementation costs, should be made on the networking configuration.

Further application scenarios may be identified for the framework of real-time information collection, in which construction activities associated with the information about material, workers, and equipment, could be updated to, and collaborated with, project management systems. These include web-based project management systems (PMS), 4D visualization, project scheduling packages, or enterprise resource planning (ERP) systems. The information flow diagram illustrated in figure 6.5 envisions a possible information system for collaboration with the project management tools based on the identification of the functional dependencies of each event associated with the construction activities. This information system can suggest the detailed design of the sensor, server, and application layer to provide motivation for automated construction environments, utilizing the advanced technologies of sensor and network.

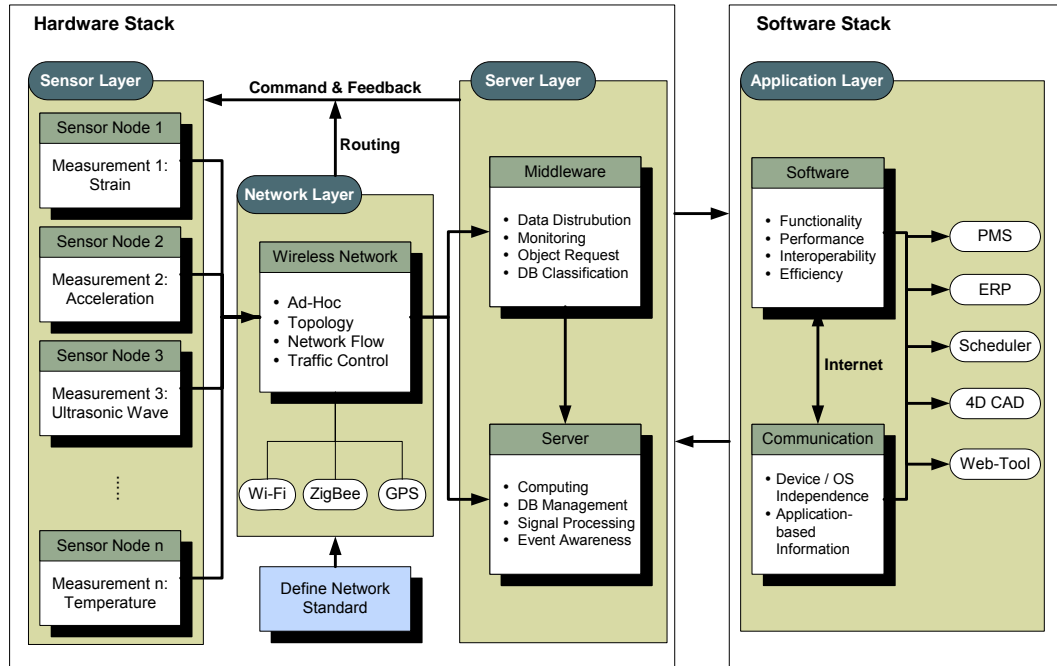


Figure 6.5: Information flow diagram for possible collaboration with project management systems

6.4 Chapter Summary

This chapter examined both quantitative and qualitative approaches to illustrate the decision criteria with the selection of new technology. Justification of the possible deployment scenarios is with the comparison of implementation costs among three alternative practices: manual, RFID, and embedded system. Upon the selection of a real bridge construction project, it was possible to examine the expected time savings of unit labor associated with the construction material tracking practices. Cost benefit associated with the deployment of emerging technology could save up to 64% of unit labor hours in a 2-year bridge construction project. In addition, by deploying the wireless sensor technology, current practice of construction material tracking could be improved to an efficient way of data communications, resource management, document management,

and labor utilization. With the benefits described in this chapter, it is expected that the presented tracking framework can be extended into diverse application areas, such as construction asset tracking, safety monitoring for workers, and further interoperability among different technologies in project management systems, such as ERP systems, web-based PMS, and n-D visualization tools.

CHAPTER 7: CONCLUSIONS AND FUTURE STUDIES

7.1 Summary

A recent National Institute of Standards and Technology study indicates that tracking and monitoring of bulk materials constitute approximately 2% of an entire construction budget. Other studies show that supervisors spend 30% to 50% of their time manually inserting data to manage resources. In terms of cost savings, inefficiencies related to the manual operations with field data in current tracking systems are becoming an evermore important issue as the size and scale of construction projects increase. Although many challenges in the construction industry are unique, improved methodologies that would increase the efficiency of material tracking in construction have not emerged because there has been a lack of interdisciplinary research activities among different areas of expertise, failure to cooperation efforts among them, and inefficient use of emerging technologies. This study attempted to create a framework for integrating the latest innovations in wireless sensor network that automate tracking and monitoring construction assets, e.g. equipment, materials, and labor in construction sites. This research constitutes one of the few studies to incorporate emerging information and sensor network technologies with the construction industry, which has been slow to migrate away from legacy processes.

The presented research works introduce a new prototype framework of an automated tracking system that will address the needed shift from the time- and labor-intensive legacy systems to sensor- and network-based collaboration and communication systems for construction processes. Software and hardware architecture for the new tracking system was developed using the combination of ultrasound and radio signal. By

embedding the external ultrasound device with a MICAZ platform, improvements to networking flexibility and wireless communication was observed over the previous technologies used in the construction material tracking systems. At the same time, a notable performance of positioning accuracy was obtained from the experimental analysis. The results of single distance estimation showed that measurement errors indicated an average 10 cm, with a measuring distance of 15 meters.

By implementing the trilateration, the results of position estimation showed an excellent accuracy performance indicating an average 5 cm. The results obtained from the prototype implementation, using a combination of ultrasound and radio signal showed a better performance in terms of accuracy when compared with the proximity approach in RFID applications. Cost benefit analysis based on quantitative approach implied that the presented framework can save the implementation cost of material tracking by up to 64 percent in a typical construction project. In addition to cost-benefits, intangible and comprehensive benefits from the use of sensor-based tracking system can be expected efficiently in communication, labor utilization, document management, and resource management.

The presented work described a system that can be effectively used in a range of applications for tracking and monitoring purposes and presents a practical use of existing wireless sensor technologies in construction applications. The cost of such hardware decreases rapidly, thereby permitting large numbers of application scenarios on construction sites with improved energy consumption, hardware performance, durability, and safety.

7.2 Recommendations for Future Research

With current wireless technology, a great challenge arises because of the level of expertise needed to fully make use of the hardware. The most sophisticated hardware has emerged from university laboratories and often requires advanced knowledge of embedded programming to achieve the level of performance desired. Such knowledge is not common among the civil and construction engineers that are often tasked to use the devices on construction sites or in buildings. An added complication lies in the fact that much of the work involves proprietary programming methods, so it is difficult to develop a standardized method of setting up a sensor network. Hardware manufacturers have typically focused their attention on developing the means to transmit radio signals efficiently through a network and have sold systems with simple software to display readings from these sensors. If a construction engineer would like to utilize a different type of sensor on these boards or use the hardware with a different application, significant efforts would be needed to reprogram the hardware in order to process the sensor's signal and to send that data over the airwaves.

Lessons learned during the implementation presented in this dissertation lie in the fact that actual performance of the embedded system generated more challenges than anticipated. Even though the estimation of distance and coordination resulted in excellent resolution and accuracy, the system performance needs to improve. For example, a relatively low signal-to-noise ratio often caused the failure of signal detection as the measurement distance increased. In addition, a low-quality internal ADC attached on the MICAZ board did not provide a good sampling performance with a 40 KHz frequency ultrasound wave, and the low sampling performance caused an ADC timeout

that is defined in the programming code of beacons. The aforementioned downsides to the embedded system are attributed to the selection of the ultrasound transducer that has low transmission power and limited angle of propagating direction. At the same time, it is necessary to improve the circuit design to reduce noise level during the amplification process and to enhance the software design in order to increase the performance of received signal detection and ADC control.

With the findings from the implementation and the results, future research works will likely be focused on enhancements of the signal detecting algorithm and improved circuit design for better performance. In addition to the efforts with performance improvement, future research will also likely exploit the scheduling algorithm to provide a tracking system with multiple numbers of beacons and remote nodes to realize the scalable tracking framework applicable to practical construction applications. Based on the development of improved tracking systems, more cost benefit analysis should be conducted by deploying the tracking system to actual construction projects in order to quantify the actual cost and labor savings.

APPENDICES

1. US_Trigger1

<triggerMsg.h>

```
enum {
    BUFFER_SIZE = 6
};

struct TriggerMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

struct BaseMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

enum {
    AM_TriggerMSG = 10
};
```

<UStTrigger.nc>

```
includes TriggerMsg;

configuration UStTrigger {
}

implementation {
    components Main, UStTriggerM, TimerC, SysTimeC, LedsC,
    ADCC,GenericComm as Comm;
    Main.StdControl -> UStTriggerM.StdControl;
    Main.StdControl -> TimerC.StdControl;

    UStTriggerM.Timer -> TimerC.Timer[unique("Timer")];
    UStTriggerM.SysTime -> SysTimeC.SysTime;
    UStTriggerM.Leds -> LedsC;
    UStTriggerM.CommControl -> Comm;
    UStTriggerM.DataMsg -> Comm.SendMsg[AM_TriggerMSG];
    UStTriggerM.BaseSend -> Comm.SendMsg[AM_TriggerMSG];
    UStTriggerM.RStampMsg -> Comm.ReceiveMsg[AM_TriggerMSG];

    UStTriggerM.ADC -> ADCC.ADC[7];
    UStTriggerM.ADCCControl -> ADCC;
}
```

<UStTriggerM.nc>

```
/**
 * @author Won S. Jang
 */
includes TriggerMsg;

module UStTriggerM {
    provides interface StdControl;
    uses {
        interface Timer;
        interface SysTime;
        interface Leds;
        interface StdControl as CommControl;
        interface SendMsg as DataMsg;
        interface SendMsg as BaseSend;
        interface ReceiveMsg as RStampMsg;
        interface ADCControl;
        interface ADC;
    }
}

implementation
{
    TOS_Msg msg;
    uint8_t N0;
    uint8_t N1;
    uint8_t N2;
    uint32_t Ttime;
    uint32_t Rtime;
    uint32_t Etime;
    uint32_t D0;
    uint32_t D1;
    uint32_t D2;
    uint32_t D3;
    uint32_t D4;
    uint32_t D5;
    uint16_t readingNumber;
    uint16_t count;
    uint16_t count1;
    uint16_t count2;
    uint32_t timer_rate;

    command result_t StdControl.init() {
        call ADCControl.init();
        call Leds.init();
        call Leds.yellowOff(); call Leds.redOff(); call Leds.greenOff();
        call CommControl.init();
        D0=0;D1=0;D2=0;D3=0;D4=0;D5=0;
        timer_rate=5000;
        return SUCCESS;
    }

    command result_t StdControl.start() {
        call ADCControl.setSamplingRate(3); // not respond from 0 to 2 -->
        3 is 30us sampling rate (=32kHz)
    }
}
```

```

        call CommControl.start();
        call Timer.start(TIMER_REPEAT, timer_rate);
return SUCCESS;
}

command result_t StdControl.stop() {
    call Timer.stop();
    call CommControl.stop();
    return SUCCESS;
}

task void dataTask() {
    struct TriggerMsg *pack;
    atomic {
        pack = (struct TriggerMsg *)msg.data;
    }

    pack->sourceMoteID0 = TOS_LOCAL_ADDRESS;N1 = 0;N2 = 0;
    pack->data[0]=0;pack->data[1]=0;pack->data[2]=0;
    pack->data[3]=0;pack->data[4]=0;pack->data[5]=0;

    if (call DataMsg.send(10, sizeof(struct TriggerMsg),
        &msg))
    {
        call Leds.yellowOn();
        count2=0;
        call ADC.getContinuousData();
    }
}

event result_t DataMsg.sendDone(TOS_MsgPtr sent, result_t success) {
    Ttime=call SysTime.getTime32();
    return SUCCESS;
}

event result_t Timer.fired() {
    return post dataTask();
}

event TOS_MsgPtr RStampMsg.receive(TOS_MsgPtr m) {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)m->data;
    }
    if (pack->sourceMoteID0==10) {
        D0=pack->data[0];
        call Leds.yellowOff();
    }
    return m;
}

task void BSend() {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)msg.data;
    }
    Etime=Rtime-Ttime;
}

```

```

D0=Etime-D0;
D1=344*D0/1000;

pack->data[0]=D0;
pack->data[1]=D1;
pack->data[2]=0;
pack->data[3]=0;
pack->data[4]=0;
pack->data[5]=0;
pack->sourceMoteID0 = 1;
pack->sourceMoteID1=2;
pack->sourceMoteID2=3;

    if (call BaseSend.send(2, sizeof(struct BaseMsg),
                           &msg))
    {
        call Leds.greenOn();
    }
}

async event result_t ADC.dataReady( uint16_t data ) {
    if (data > 0x0078 && count==0) { //0x00b4=180mV
        count1++;
        if (count1 < 2) {
            Rtime=call SysTime.getTime32();
            post BSend();
            call Leds.redOn();
            count=1;
            return FAIL;
        }
        else {
            count1=0;
            return FAIL;
        }
    }
}

else {
    count2++;
    if (count2 > 0x02ee) {
        count2 = 0;
        call Leds.redOff();
        return FAIL;
    }
    call Leds.redOff();
}
return SUCCESS;
}

event result_t BaseSend.sendDone(TOS_MsgPtr sent, result_t success) {
    count=0;
    count2=0;
    call Leds.greenOff();
    call Leds.redOff();
}

```



```

        return SUCCESS;
    }
}

```

2. US_Trigger2

<TriggerMsg.h>

```

enum {
    BUFFER_SIZE = 6
};

struct TriggerMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

struct BaseMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

enum {
    AM_TriggerMSG = 10
};

```

<UStigger.nc>

```

includes TriggerMsg;

configuration UStigger {
}
implementation {
    components Main, UStiggerM, TimerC, SysTimeC, LedsC,
    ADCC,GenericComm as Comm;
    Main.StdControl -> UStiggerM.StdControl;
    Main.StdControl -> TimerC.StdControl;

    UStiggerM.Timer -> TimerC.Timer[unique("Timer")];
    UStiggerM.SysTime -> SysTimeC.SysTime;
    UStiggerM.Leds -> LedsC;
    UStiggerM.CommControl -> Comm;
    UStiggerM.DataMsg -> Comm.SendMsg[AM_TriggerMSG];
    UStiggerM.BaseSend -> Comm.SendMsg[AM_TriggerMSG];
    UStiggerM.RStampMsg -> Comm.ReceiveMsg[AM_TriggerMSG];

    UStiggerM.ADC -> ADCC.ADC[7];
    UStiggerM.ADCCControl -> ADCC;
}

```

<UStTriggerM.nc>

```
/**
 * @author Won S. Jang
 */
includes TriggerMsg;

module UStTriggerM {
    provides interface StdControl;
    uses {
        interface Timer;
        interface SysTime;
        interface Leds;
        interface StdControl as CommControl;
        interface SendMsg as DataMsg;
        interface SendMsg as BaseSend;
        interface ReceiveMsg as RStampMsg;
        interface ADCControl;
        interface ADC;
    }
}

implementation
{
    TOS_Msg msg;
    uint8_t N0;
    uint8_t N1;
    uint8_t N2;
    uint32_t Ttime;
    uint32_t Rtime;
    uint32_t Etime;
    uint32_t D0;
    uint32_t D1;
    uint32_t D2;
    uint32_t D3;
    uint32_t D4;
    uint32_t D5;
    uint16_t readingNumber;
    uint16_t count;
    uint16_t count1;
    uint16_t count2;
    uint32_t timer_rate;

    command result_t StdControl.init() {
        call ADCControl.init();
        call Leds.init();
        call Leds.yellowOff(); call Leds.redOff(); call Leds.greenOff();
        call CommControl.init();
        D0=0;D1=0;D2=0;D3=0;D4=0;D5=0;
        timer_rate=5000;
        return SUCCESS;
    }

    command result_t StdControl.start() {
        call ADCControl.setSamplingRate(3);
        call CommControl.start();
    }
}
```

```

    return SUCCESS;
}

command result_t StdControl.stop() {
    call Timer.stop();
    call CommControl.stop();
    return SUCCESS;
}

task void dataTask() {
    struct TriggerMsg *pack;
    atomic {
        pack = (struct TriggerMsg *)msg.data;
    }

    pack->sourceMoteID0 = TOS_LOCAL_ADDRESS;N1 = 0;N2 = 0;
    pack->data[0]=0;pack->data[1]=0;pack->data[2]=0;
    pack->data[3]=0;pack->data[4]=0;pack->data[5]=0;

    if (call DataMsg.send(10, sizeof(struct TriggerMsg),
        &msg))
    {
        call Leds.yellowOn();
        count2=0;
        call ADC.getContinuousData();
    }
}

event result_t DataMsg.sendDone(TOS_MsgPtr sent, result_t success) {
    Ttime=call SysTime.getTime32();
    return SUCCESS;
}

event TOS_MsgPtr RStampMsg.receive(TOS_MsgPtr m) {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)m->data;
    }
    if (pack->sourceMoteID0==10) {
        D2=pack->data[0];
        call Leds.yellowOff();
    }
    if (pack->sourceMoteID0==1) {
        N0=pack->sourceMoteID0;
        D0=pack->data[0];
        D1=pack->data[1];
        call Timer.start(TIMER_ONE_SHOT, 800);
    }
    return m;
}

event result_t Timer.fired() {
    return post dataTask();
}

```

```

task void BSend() {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)msg.data;
    }

    Etime=Rtime-Ttime;
    D2=Etime-D2;
    D3=344*D2/1000;

    pack->data[0]=D0;
    pack->data[1]=D1;
    pack->data[2]=D2;
    pack->data[3]=D3;
    pack->data[4]=0;
    pack->data[5]=0;
    pack->sourceMoteID0 = 1;
    pack->sourceMoteID1=2;
    pack->sourceMoteID2=3;

    if (call BaseSend.send(3, sizeof(struct BaseMsg),
        &msg))
    {
        call Leds.greenOn();
    }
}

async event result_t ADC.dataReady( uint16_t data ) {
    if (data > 0x0078 && count==0) {
        count1++;
        if (count1 < 2) {
            Rtime=call SysTime.getTime32();
            post BSend();
            call Leds.redOn();
            count=1;
            return FAIL;
        }
        else {
            count1=0;
            return FAIL;
        }
    }

    else {
        count2++;
        if (count2 > 0x02ee) {
            count2 = 0;
            call Leds.redOff();
            return FAIL;
        }
        call Leds.redOff();
    }
    return SUCCESS;
}

```

```

event result_t BaseSend.sendDone(TOS_MsgPtr sent, result_t success) {
    count=0;
    count2=0;
    call Leds.greenOff();
    call Leds.redOff();
    return SUCCESS;
}
}

```

3. US_Trigger3

<TriggerMsg.h>

```

enum {
    BUFFER_SIZE = 6
};

struct TriggerMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

struct BaseMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

enum {
    AM_TriggerMSG = 10
};

```

<USTRigger3.nc>

```

includes TriggerMsg;

configuration USTRigger {
}

implementation {
    components Main, USTRiggerM, TimerC, SysTimeC, LedsC,
    ADCC,GenericComm as Comm;
    Main.StdControl -> USTRiggerM.StdControl;
    Main.StdControl -> TimerC.StdControl;

    USTRiggerM.Timer -> TimerC.Timer[unique("Timer")];
    USTRiggerM.SysTime -> SysTimeC.SysTime;
    USTRiggerM.Leds -> LedsC;
    USTRiggerM.CommControl -> Comm;
    USTRiggerM.DataMsg -> Comm.SendMsg[AM_TriggerMSG];
    USTRiggerM.BaseSend -> Comm.SendMsg[AM_TriggerMSG];
    USTRiggerM.RStampMsg -> Comm.ReceiveMsg[AM_TriggerMSG];
}

```

```

    USTriggerM.ADC -> ADCC.ADC[7];
    USTriggerM.ADCCControl -> ADCC;
}

<USTriggerM.nc>
/**
 * @author Won S. Jang
 */
includes TriggerMsg;

module USTriggerM {
    provides interface StdControl;
    uses {
        interface Timer;
            interface SysTime;
            interface Leds;
        interface StdControl as CommControl;
        interface SendMsg as DataMsg;
        interface SendMsg as BaseSend;
            interface ReceiveMsg as RStampMsg;
            interface ADCCControl;
            interface ADC;
    }
}

implementation
{
    TOS_Msg msg;
    uint8_t N0;
    uint8_t N1;
    uint8_t N2;
    uint32_t Ttime;
    uint32_t Rtime;
    uint32_t Etime;
    uint32_t D0;
    uint32_t D1;
    uint32_t D2;
    uint32_t D3;
    uint32_t D4;
    uint32_t D5;
    uint16_t readingNumber;
    uint16_t count;
    uint16_t count1;
    uint16_t count2;
    uint32_t timer_rate;

    command result_t StdControl.init() {
        call ADCCControl.init();
        call Leds.init();
        call Leds.yellowOff(); call Leds.redOff(); call Leds.greenOff();
        call CommControl.init();
        D0=0;D1=0;D2=0;D3=0;D4=0;D5=0;
        timer_rate=5000;
        return SUCCESS;
    }
}

```

```

command result_t StdControl.start() {
    call ADCControl.setSamplingRate(3);
    call CommControl.start();
    return SUCCESS;
}

command result_t StdControl.stop() {
    call Timer.stop();
    call CommControl.stop();
    return SUCCESS;
}

task void dataTask() {
    struct TriggerMsg *pack;
    atomic {
        pack = (struct TriggerMsg *)msg.data;
    }

    pack->sourceMoteID0 = TOS_LOCAL_ADDRESS;N1 = 0;N2 = 0;
    pack->data[0]=0;pack->data[1]=0;pack->data[2]=0;
    pack->data[3]=0;pack->data[4]=0;pack->data[5]=0;

    if (call DataMsg.send(10, sizeof(struct TriggerMsg),
        &msg))
    {
        call Leds.yellowOn();
        count2=0;
        call ADC.getContinuousData();
    }
}

event result_t DataMsg.sendDone(TOS_MsgPtr sent, result_t success) {
    Ttime=call SysTime.getTime32();
    return SUCCESS;
}

event TOS_MsgPtr RStampMsg.receive(TOS_MsgPtr m) {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)m->data;
    }

    if (pack->sourceMoteID0==10) {
        D4=pack->data[0];
        call Leds.yellowOff();
    }

    if (pack->sourceMoteID1==2) {
        N0=pack->sourceMoteID0;
        N1=pack->sourceMoteID1;
        D0=pack->data[0];
        D1=pack->data[1];
        D2=pack->data[2];
        D3=pack->data[3];
        call Timer.start(TIMER_ONE_SHOT, 800);
    }
}

```

```

    return m;
}

event result_t Timer.fired() {
    return post dataTask();
}

task void BSend() {
    struct BaseMsg *pack;
    atomic {
        pack = (struct BaseMsg *)msg.data;
    }

    Etime=Rtime-Ttime;
    D4=Etime-D4;
    D5=344*D4/1000;

    pack->data[0]=D0;
    pack->data[1]=D1;
    pack->data[2]=D2;
    pack->data[3]=D3;
    pack->data[4]=D4;
    pack->data[5]=D5;
    pack->sourceMoteID0 = 1;
    pack->sourceMoteID1=2;
    pack->sourceMoteID2=3;

    if (call BaseSend.send(0, sizeof(struct BaseMsg),
        &msg))
    {
        call Leds.greenOn();
    }
}

async event result_t ADC.dataReady( uint16_t data ) {
    if (data > 0x0078 && count==0) {
        count1++;
        if (count1 < 2) {
            Rtime=call SysTime.getTime32();
            post BSend();
            call Leds.redOn();
            count=1;
            return FAIL;
        }
        else {
            count1=0;
            return FAIL;
        }
    }

    else {
        count2++;
        if (count2 > 0x02ee) {
            count2 = 0;

```



```

        call Leds.redOff();
        return FAIL;
    }
    call Leds.redOff();
}
return SUCCESS;
}

event result_t BaseSend.sendDone(TOS_MsgPtr sent, result_t success) {
    count=0;
    count2=0;
    call Leds.greenOff();
    call Leds.redOff();
    return SUCCESS;
}
}

```

4. US_Tx

<DetectMsg.h>

```

enum {
    BUFFER_SIZE = 6
};

struct TstampMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

enum {
    AM_DetectMSG = 10
};

```

<US_Tx.nc>

```

includes DetectMsg;
configuration US_Tx {
}
implementation {
    components Main, US_TxM, LedsC, TimerC, SysTimeC, GenericComm as
Comm;
    Main.StdControl -> US_TxM.StdControl;
    US_TxM.SysTime -> SysTimeC.SysTime;
    US_TxM.Leds -> LedsC;
    US_TxM.CommControl -> Comm;
    US_TxM.TimeStampMsg -> Comm.SendMsg[AM_DetectMSG];
    US_TxM.RFDetectMsg -> Comm.ReceiveMsg[AM_DetectMSG];
    US_TxM.Timer -> TimerC.Timer[unique("Timer")];
}

```

```

<US_TxM.nc>
includes DetectMsg;

module US_TxM {
    provides interface StdControl;
    uses {
        interface Leds;
        interface StdControl as CommControl;
        interface SysTime;
        interface SendMsg as TimeStampMsg;
        interface ReceiveMsg as RFDetectMsg;
        interface Timer;
    }
}
implementation {
    TOSH_ASSIGN_PIN(REDLED, A, 2);
    TOSH_ASSIGN_PIN(VSensor, C, 2); //(PW2)
    int i;
    TOS_Msg msg;
    uint8_t Destination;
    uint32_t Ttime;
    uint32_t Rtime;
    uint32_t Etime;

    command result_t StdControl.init() {
        TOSH_MAKE_REDLED_OUTPUT();
        TOSH_MAKE_VSensor_OUTPUT();
        return call CommControl.init();
    }

    command result_t StdControl.start() {
        return call CommControl.start();
    }

    command result_t StdControl.stop() {
        return call CommControl.stop();
    }

    task void SendTStamp() {
        struct TstampMsg *pack;
        atomic {
            pack = (struct TstampMsg *)msg.data;
        }
        pack->sourceMoteID0 = TOS_LOCAL_ADDRESS;

        if (call TimeStampMsg.send(Destination, sizeof(struct TstampMsg),
            &msg))
        {
            call Leds.yellowOn();
        }
    }

    task void USTx() {
        struct TstampMsg *pack;
        Ttime=call SysTime.getTime32();
        Etime=Ttime-Rtime;
    }
}

```

```

    atomic {
        pack = (struct TstampMsg *)msg.data;
        pack->data[0] = Etime;
    }
    post SendTStamp();
    call Timer.start(TIMER_ONE_SHOT, 500);

}

event result_t Timer.fired() {
    TOSH_CLR_VSensor_PIN();
    TOSH_SET_REDLED_PIN();
    call Leds.greenOff();
    return SUCCESS;
}

event result_t TimeStampMsg.sendDone(TOS_MsgPtr sent, result_t success)
{
    call Leds.yellowOff();
    return SUCCESS;
}

event TOS_MsgPtr RFDetectMsg.receive(TOS_MsgPtr m) {
    struct TstampMsg *pack;
    atomic {
        pack = (struct TstampMsg *)m->data;
    }
    Rtime=call SysTime.getTime32();
    Destination=pack->sourceMoteID0;
    call Leds.greenOn();
    TOSH_SET_VSensor_PIN();
    TOSH_CLR_REDLED_PIN();
    post USTx();

    return m;
}
}

```

5. US_Base

<UMsg.h>

```

enum {
    BUFFER_SIZE = 6
};

```

```

struct UartMsg
{
    uint8_t sourceMoteID0;
    uint8_t sourceMoteID1;
    uint8_t sourceMoteID2;
    uint32_t data[BUFFER_SIZE];
};

```

```

enum {
    AM_TriggerMSG = 10
};

<USBBase.nc>
includes UMsg;

configuration USBBase {
}
implementation {
    components Main, USBBaseM, TimerC, LedsC, GenericComm as Comm,
    UARTComm as UComm;
    Main.StdControl -> USBBaseM.StdControl;
    Main.StdControl -> TimerC.StdControl;

    USBBaseM.StampTimer -> TimerC.Timer[unique("Timer")];
    USBBaseM.Leds -> LedsC;
    USBBaseM.CommControl -> Comm;
    USBBaseM.UartSend -> UComm.SendMsg[AM_TriggerMSG];
    USBBaseM.DataMsg -> Comm.ReceiveMsg[AM_TriggerMSG];
}

<USBBaseM.nc>
/**
 * @author Won S. Jang
 */
includes UMsg;

module USBBaseM {
    provides interface StdControl;
    uses {
        interface Timer as StampTimer;
        interface Leds;
        interface StdControl as CommControl;
        interface SendMsg as UartSend;
        interface ReceiveMsg as DataMsg;
    }
}
implementation
{
    TOS_Msg msg;
    uint8_t N0;
    uint8_t N1;
    uint8_t N2;
    uint32_t Ttime;
    uint32_t Rtime;
    uint32_t Etime;
    uint32_t D0;
    uint32_t D1;
    uint32_t D2;
    uint32_t D3;
    uint32_t D4;
    uint32_t D5;

    command result_t StdControl.init() {
        call Leds.init();
    }
}

```

```

    call Leds.yellowOff(); call Leds.redOff(); call Leds.greenOff();
    call CommControl.init();

    dbg(DBG_BOOT, "USttrigger initialized\n");
    return SUCCESS;
}

command result_t StdControl.start() {
    call CommControl.start();
    return SUCCESS;
}

command result_t StdControl.stop() {
    call CommControl.stop();
    return SUCCESS;
}

event TOS_MsgPtr DataMsg.receive(TOS_MsgPtr m) {
    struct UartMsg *pack;
    atomic {
        pack = (struct UartMsg *)m->data;
    }

    N0=pack->sourceMoteID0;
    N1=pack->sourceMoteID1;
    N2=pack->sourceMoteID2;
    D0=pack->data[0];
    D1=pack->data[1];
    D2=pack->data[2];
    D3=pack->data[3];
    D4=pack->data[4];
    D5=pack->data[5];
    call StampTimer.start(TIMER_ONE_SHOT, 100);

    return m;
}

task void USend() {
    struct UartMsg *pack;
    atomic {
        pack = (struct UartMsg *)msg.data;
    }

    pack->data[0]=D0;
    pack->data[1]=D1;
    pack->data[2]=D2;
    pack->data[3]=D3;
    pack->data[4]=D4;
    pack->data[5]=D5;
    pack->sourceMoteID0 = N0;
    pack->sourceMoteID1=N1;
    pack->sourceMoteID2=N2;

    if (call UartSend.send(TOS_UART_ADDR, sizeof(struct UartMsg),
        &msg))

```

```

    {
        call Leds.greenToggle();
    }
}

event result_t StampTimer.fired() {
    post USend();
    call Leds.redToggle();
    return SUCCESS;
}

event result_t UartSend.sendDone(TOS_MsgPtr sent, result_t success) {
    return SUCCESS;
}
}
}

```

6. Java Visualization

<LAF.java>

// \$Id: LAF.java,v 1.0 2007/07/31 Won S. Jang Exp \$

```

package net.tinyos.test;

import java.io.*;
import net.tinyos.packet.*;
import net.tinyos.util.*;
import net.tinyos.message.*;
import net.tinyos.test.*;
import java.awt.*;
import java.awt.event.*;
import java.lang.Math.*;
import java.lang.Integer.*;

public class LAF {
    public static void main(String args[]) throws IOException {
        int readingNo;
        readingNo=0;
        if (args.length > 0) {
            System.err.println("usage: java net.tinyos.tools.LAF");
            System.exit(2);
        }

        PacketSource reader = BuildSource.makePacketSource();
        if (reader == null) {
            System.err.println("Invalid packet forwarding (check your
MOTE setting)");
            System.exit(2);
        }

        PointDrawFrame pdf = new PointDrawFrame("Position Tracking");
        pdf.madeFrame();
    }
}

```

```

int x1=0;
int y1=1340;
int z1=0;
int x2=5000;
int y2=1340;
int z2=0;
int x3=2500;
int y3=5000;
int z3=0;

try {
    reader.open(PrintStreamMessenger.err);

    for (;;) {

        String timeStamp = "";
        byte[] packet = reader.readPacket();

        int t1 = ((packet[1] & 255) << 8) + (packet[0] & 255);
        String twoChar = Integer.toString(t1,10);
        if (twoChar.length() < 2) {twoChar = "0" + twoChar; }
        timeStamp = timeStamp + twoChar + " ";

        for (int i=2; i<=7;i++) {
            int t2=(packet[i] & 255);
            String twoChar1 = Integer.toString(t2,10);
            if (twoChar1.length() < 2) {twoChar1 = "0" + twoChar1; }
            timeStamp = timeStamp + twoChar1 + " ";
        }
        for (int i=0;i<6;i++) {
            int t3=((packet[11+i*4] & 255) <<24) +((packet[10+i*4] &
255) <<16)+((packet[9+i*4] & 255) <<8) +(packet[8+i*4] & 255);
            String twoChar2 = Integer.toString(t3,10);
            if (twoChar2.length() < 2) {twoChar2 = "0" + twoChar2; }
            timeStamp = timeStamp + twoChar2 + " ";
        }

        int r1=((packet[15] & 255) <<24) +((packet[14] & 255)
<<16)+((packet[13] & 255) <<8) +(packet[12] & 255);
        int r2=((packet[23] & 255) <<24) +((packet[22] & 255)
<<16)+((packet[21] & 255) <<8) +(packet[20] & 255);
        int r3=((packet[31] & 255) <<24) +((packet[30] & 255)
<<16)+((packet[29] & 255) <<8) +(packet[28] & 255);

        float tmpXa=(float) (-y2*Math.pow(x3,2)-y2*Math.pow(y3,2)-
y1*Math.pow(z2,2)+y1*Math.pow(x3,2)+y1*Math.pow(y3,2)+y1*Math.pow(z3,2)
-y1*Math.pow(r3,2)+Math.pow(z1,2)*y2-
Math.pow(z1,2)*y3+Math.pow(y2,2)*y3+Math.pow(y1,2)*y2-
Math.pow(y1,2)*y3-Math.pow(r2,2)*y3-
Math.pow(r1,2)*y2+Math.pow(r1,2)*y3+Math.pow(x2,2)*y3+Math.pow(z2,2)*y3
-y1*Math.pow(y2,2)+y1*Math.pow(r2,2)-y1*Math.pow(x2,2)-
y2*Math.pow(z3,2)+Math.pow(x1,2)*y2-
Math.pow(x1,2)*y3+y2*Math.pow(r3,2));
        float tmpXb=(float) (-x2*y1-x1*y3+x2*y3-x3*y2+x3*y1+x1*y2);
        float tmpX=(float) 1/2*tmpXa/tmpXb;

```

```

        float tmpYa=(float) (-x1*Math.pow(z2,2)-x3*Math.pow(r2,2)-
x2*Math.pow(r1,2)+x3*Math.pow(y2,2)-
x3*Math.pow(y1,2)+x3*Math.pow(r1,2)+x2*Math.pow(z1,2)-
x3*Math.pow(z1,2)+x2*Math.pow(x1,2)+x1*Math.pow(y3,2)-
x2*Math.pow(z3,2)+x3*Math.pow(z2,2)+x2*Math.pow(r3,2)+x2*Math.pow(y1,2)
+x1*Math.pow(x3,2)-x2*Math.pow(y3,2)-
x1*Math.pow(r3,2)+x3*Math.pow(x2,2)+x1*Math.pow(z3,2)-
x3*Math.pow(x1,2)-x2*Math.pow(x3,2)-
x1*Math.pow(x2,2)+x1*Math.pow(r2,2)-x1*Math.pow(y2,2));
        float tmpYb=(float) (-x2*y1-x1*y3+x2*y3-x3*y2+x3*y1+x1*y2);
        float tmpY=(float) -1/2*tmpYa/tmpYb;

        int X = (int) tmpX;
        int Y = (int) tmpY;

        String rawX = Integer.toString((X));
        String rawY = Integer.toString((Y));

        readingNo++;

        pdf.PXY(rawX, rawY);

        System.out.print(X+" "+Y+"\t");
        System.out.println();
        System.out.flush();
    }
    }
    catch (IOException e) {
        System.err.println("Error on " + reader.getName() + ": " + e);
    }
}
}
}

```

```

class PointDrawFrame extends Frame implements ActionListener {

    MenuBar mb;
    Menu m;
    MenuItem mi1;
    MenuItem mi2;

    FileDialog fd;

    Panel np;
    Panel sp;

    Button drawButton;
    Button exitButton;
    Screen sc;

    ScrollPane scp;

    PointDrawFrame(String str) {
        super(str);
        sc = new Screen();
    }
}

```



```

}

public void madeFrame() {
    mb = new MenuBar();
    m = new Menu("File");
    mi1 = new MenuItem("Load");
    mi2 = new MenuItem("Exit");

    setMenuBar(mb);
    mb.add(m);
    m.add(mi1);
    m.add(mi2);

    fd = new FileDialog(this, "Load", 0);

    np = new Panel();
    sp = new Panel();

    drawButton = new Button("Next");
    exitButton = new Button("Exit");

    scp = new ScrollPane();
    scp.add(sc);

    np.setLayout(new GridLayout(1, 2, 0, 0));
    np.add(drawButton);
    //np.add(deleteButton);

    sp.add(exitButton);

    setLayout(new BorderLayout());
    add("North", np);
    add("Center", scp);
    add("South", sp);

    setSize(1000, 1000);
    setResizable(false);
    setVisible(true);

    m.addActionListener(this);
    drawButton.addActionListener(this);
    exitButton.addActionListener(this);
    addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent we) {
            dispose();
            System.exit(0);
        }
    });
}

public void actionPerformed(ActionEvent ae) {
    String selectedButton = ae.getActionCommand();
    if(selectedButton.equals("Exit")) {
        dispose();
        System.exit(0);
    }
}

```

```

    }
}

public void PXY(String x, String y) {

    int RawX = new Integer(x).intValue();
    int RawY = new Integer(y).intValue();

}
}

class Screen extends Canvas {
    int x;
    int y;
    int axisX;
    int axisY;
    Color c;

    public void drawPoint(int x, int y) {
        System.out.println("X: " + x + "Y: " + y);

        axisX=x;
        axisY=y;
        c = Color.red;
        this.x = (8*x/100+100);
        this.y = (850-8*y/100);

        System.out.println("Draw");

        repaint();

    }
    public void paint(Graphics g) {
        if (x==0 && y==0) {c = Color.white;}
        System.out.println("Paint");

        g.setColor(c);
        g.fillOval(x-15, y-15, 30, 30);
        g.setColor(Color.blue);
        g.drawString("(" + axisX + ", " + axisY + ")", x+15, y-15);
        g.setColor(Color.black);
        g.fillRect(50, 50, 3, 800); // y axis
        g.fillRect(50, 850, 800, 3); //x axis

        g.drawString("Position X (cm)",450,880);
        g.drawString("Position Y (cm)",10,20);
        g.drawString("10000",840,870);
        g.drawString("0",45,870);
        g.drawString("10000",15,60);

        g.setColor(Color.cyan);
        g.fillOval (35,728,30,30);
        g.fillOval (835,728,30,30);

```

```
g.fillOval (485,35,30,30);

g.setColor(Color.blue);
g.drawString("B1",34,718);
g.drawString("B2",834,718);
g.drawString("B3",495,30);
}
}
```

BIBLIOGRAPHY

Abell, P. (2004), "Why Buyers & Sellers Will Want RFID," presented at NACDS MarketPlace, ePC Group, June 24, 2004, available online at: <http://meetings.nacds.org>, accessed at August 23, 2006.

ABI Research, "The Market for RFID Readers," Technical Report from ABI Research 2005, Document code RR-RFR available online at: http://www.abiresearch.com/products/market_research/RFID_Readers, accessed at September 5, 2007.

Active Badge, AT&T Laboratories Cambridge, available online at: <http://www.cl.cam.ac.uk/research/dtg/attachive/thebadge.html>, accessed on June 24, 2007.

Active Bat, AT&T Laboratories Cambridge, available online at: <http://www.cl.cam.ac.uk/research/dtg/attachive/bat/flatbatinside.jpg>, accessed on June 24, 2007.

Adams, J. T. (2005), "ZigBee Wireless Technology and the IEEE 802.15.4 Radio – Enabling Simple Wireless," *Texas Wireless Symposium 2005*, University of Texas, October 26-28.

Amjad, A. A. (2004), "Cost Benefit Analysis for construction Projects," *IEP-SAC Journal 2004-2005, Institute of Engineering Pakistan*, pp. 85-90.

Ascension Technology Corp, MotionStar, Burlington, VT, available online at: <http://www.ascension-tech.com/products/motionstar.php>, accessed in May 2007.

Atmel Corporation, Atmega128 datasheet, available online at: <http://www.atmel.com>, accessed in September 2007.

Bahal, P. and Padmanabhan, V. (2000), "RADAR: An In-Building RF-based User Location and Tracking System," *Proceedings of IEEE INFOCOM*, Tel-Aviv, Israel, March 2000.

Baker, N. (2005), "ZigBee and Bluetooth - Strengths and Weaknesses for Industrial Applications," *Computing & Control Engineering Journal, IEEE*, Vol. 16, No. 2, pp. 20-25.

Balakrishnan, H., Baliga, R., Curtis, D., Goraczko, M., Miu, A., Priyantha, B., Smith, A., Steele, K., Teller, S. and Wang, K. (2003), "Lessons from Developing and Deploying the Cricket Indoor Location System," MIT Computer Science and Artificial Intelligence Laboratory (CSAIL), available online at: <http://nms.lcs.mit.edu/cricket/>, November 2003, (preprint).

Bell, Lansford C. and McCullouch, Bob G. (1988), "Bar Code Applications in Construction," *Journal of Construction Engineering and Management, ASCE*, Vol. 114, No. 2, pp. 263-278.

Bell, Lansford C. and Stukhart, George (1986), "Attributes of Materials Management Systems," *Journal of Construction Engineering and Management, ASCE*, Vol. 112, No. 1, pp. 14-21.

Bell, Lansford, C. and Stukhart, George (1987), "Costs and Benefits of Materials Management Systems," *Journal of Construction Engineering and Management, ASCE*, Vol. 113, No. 3, pp. 222-234.

Bernold, Leonhard (1990), "Testing Bar-Code Technology in Construction Environment," *Journal of Construction Engineering and Management, ASCE*, Vol. 116, No. 4, pp. 643-655.

Chen, Zhen, Li, Heng and Wong, Conrad T.C. (2002), "An Application of Bar-Code System for Reducing Construction Wastes," *Automation in Construction, Elsevier*, Vol. 11, No. 5, pp. 521-533.

Cheng, Min-Yuan and Chen, Jiann-Chyun (2002), "Integrating Barcode and GIS for Monitoring Construction Process," *Automation in Construction, Elsevier*, Vol. 11, No. 1, pp. 23-33.

Cheok, G. S., Lipman, R. R., Witzgall, C., Bernal, J. and Stone, W. C. (2000), *NIST Construction Automation Program No. 4: Non-Intrusive Scanning Technology for*

Construction Status Determination, NISTIR 6457, BFRl, NIST, January, 2000, Gaithersburg, MD.

Crossbow Technology Inc., MPR-MIB Series Users Manual, June 2007, PN: 7430-0021-08, available online at: http://www.xbow.com/Support/Support_pdf_files/MPR-MIB_Series_Users_Manual.pdf, accessed in September, 2007.

Crossbow Technology, Inc., Wireless Sensor Modules, available online at: <http://www.xbow.com>, access in September, 2007.

Ding, G. and Sahinoglu, Z. (2005), "Reliable Broadcast in ZigBee Networks," *Proceedings of Sensor and Ad Hoc Communications and Networks, IEEE SECON 2005*, Santa Clara, CA, September 26-29, pp. 510-520.

Echeverry, Diego and Beltrán, Alfredo (1997), "Bar-Code Control of Construction Field Personnel and Construction Materials," *Proceedings of Computing in Civil Engineering, Fourth Congress held in conjunction with A/E/C Systems '97, ASCE*, Philadelphia, PA, June 16-18, pp. 341-347

Egan, D. (2005), "The Emergence of ZigBee in Building Automation and Industrial Control," *Computing & Control Engineering Journal, IEEE*, Vol. 16, No. 2, pp. 14-19.

Elnahrawy, E., Li, X., and Martin, R. M. (2004), "The Limits of Localization using Signal Strength: A Comparative Study," *Proceedings of Sensor and Ad-Hoc Communications and Networks Conference 2004 (SECON'04)*, October 2004, Santa Clara, CA.

FlexiPannel Ltd., "ZigBee for Applications Developers," October 2005, available online at: <http://www.flexipanel.com>, accessed at August 24, 2006.

Formoso, C. T. and Revelo, V. H. (1999), "Improving the Materials Supply System in Small-Sized Building Firms," *Automation in Construction, Elsevier*, Vol. 8, No. 6, pp. 663-670.

Futcher, K. G. (2001), "User Survey on a WAN Portfolio M.I.S. Used for Portfolio/Project Management in Hong Kong," *Proceedings of IT in Construction in Africa, International Council for Research and Innovation in Building Construction (W78:2001)*, Mpumalanga, South Africa, May 30-June 1, pp. (44) 1-14.

Garcia, Camilo, Huebschman, Ryan, Abraham, Dulcy M. and Bullock, Darcy M. (2006), "Using GPS to Measure the Impact of Construction Activities on Rural Interstates," *Journal of Construction Engineering and Management, ASCE*, Vol. 132, No. 5, pp. 508-515.

Garmin Ltd., What is GPS?, available online at: <http://garmin.com/aboutGPS>, accessed in September, 2006.

Goodrum, P. M., McLaren, M. A. and Durfee, A. (2006), "The Application of Active Radio Frequency Identification Technology for Tool Tracking on Construction Job Sites," *Automation in Construction, Elsevier*, Vol. 15, No. 3, pp. 292-302.

Goodrum, P. M., McLaren, M. A., and Durfee, A. (2006), "The Application of Active Radio Frequency Identification Technology for Tool Tracking on Construction Job Sites," *Automation in Construction, Elsevier*, Vol. 15, No. 3, pp. 292-302.

Harter, Andy, Hopper, Andy, Steggles, Pete, Ward, Andy and Webster, Paul (1999), "The Anatomy of a Context-Aware Application," *Proceedings of 5th International Conference on Mobile Computing and Networking (MOBICOM'99)*, ACM Press, Seattle, WA, August 15-20, pp. 59-68.

Hatami, A. (2006), *Application of Channel Modeling for Indoor Localization Using TOA and RSS*, Dissertation, Dept. of Electrical and Computer Engineering, Worcester Polytechnic Institute, May 2006.

Henry, P. S. (2002), "Is Wi-Fi in Your Future?" *Proceedings of INFOCOM 2002*, New York, NY, June 23-27.

Hewlett-Packard (2002), *Understanding Wi-Fi™*, White Paper, January 2002, available online at: <http://www.hp.com>, accessed at August 15, 2007.

Hightower, Jeffrey and Borriello, Gaetano (2001), "Location Systems for Ubiquitous Computing," *IEEE Computer*, Vol. 34, No. 8, pp. 57-66

Hill, J. L. (2003), *System Architecture for Wireless Sensor Networks*, Ph.D. Dissertation in Computer science, University of California, Berkeley, CA.

IEEE 802 LAN/MAN Standards Committee, available online at: <http://grouper.ieee.org/groups/802/index.html>, accessed at August 24, 2007.

IEEE Standards (2003), *IEEE standards 802.15.4 – part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs)*. LAN/MAN Standards Committee of IEEE Computer Society, IEEE Press, New York, NY, USA, October 2003.

Jameco Electronics, 40TR16F Ultrasound Transducer datasheet, available online at <http://www.jameco.com>, accessed in June, 2007.

Jaselskis, E. J. and El-Misalami, T. (2003), "Implementing Radio Frequency Identification in the Construction Process," *Journal of Construction Engineering and Management*, ASCE, Vol. 129, No. 6, pp. 680-688.

Jiang, Yi and Li, Shuo (2002), "Measuring and Analyzing Vehicle Position and Speed Data at Work Zones Using Global Positioning Systems," *ITE Journal, Institute of Transportation Engineers*, Vol. 72, No. 3, pp. 48-53.

Kahn, J. M. Kahn, Katz, R. H. Katz, and Pister, K. S. J. Pister, (1999), "Next Century Challenges: Mobile Networking for "Smart Dust","" *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Seattle, Washington, August 15-19, pp. 271-278.

Kalman, R. E. (1960), "A New Approach to Linear Filtering and Prediction Problems," *Transaction of the ASME—Journal of Basic Engineering*, Vol. 82, pp. 35-45.

Kim, J.-H., Lee, U.-K., Cho, H.-H., and Kang, K.-I. (2007), "Development of a Hybrid Device Based on Infrared and Ultrasonic Sensors for Human Resource Management," 24th International Symposium on Automation & Robotics in Construction (ISARC 2007), Kochi, India, September 19-21, pp. 111-115.

Kim, S. H, Lee, S. W, and Lee, Y. S. (2000), *Development of Evaluation Procedure for Vibrating and Displacement Serviceability of Bridge*, Technical Report, Korea Infrastructure Safety and Technology Corporation, Korea.

Lee, S., Arif, A. U., and Jang, H. (2004), "Quantified Benefit of Implementing Enterprise Resource Planning Through Process Simulation," *Canadian Journal of Civil Engineering, NRC Canada*, Vol. 31, No. 2, pp. 263-271

Levis, P. and Culler, D. (2002), "Moté: A Tiny Virtual Machine for Sensor Networks," *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, October 6-9, San Jose, CA, pp. 85-95.

Li, H., Chen, Z., Yong, L. and Kong, S.C.W. (2005), "Application of Integrated GPS and GIS Technology for Reducing Construction Waste and Improving Construction Efficiency," *Automation in Construction, Elsevier*, Vol. 14, No. 3, pp. 323-331.

Lorincz, K. and Welsh, M. (2005), "Motetrack: A Robust, Decentralized Approach to RF-based Location Tracking," *Proceedings of the International Workshop on Location- and Context-Awareness (LoCA '05)*, Munich, Germany, May 12-13, 2005.

Lu, M., Chen, W. and Chan, W. H. (2004), "Discussion of "Building Project Model Support for Automated Labor Monitoring"," *Journal of Computing in Civil Engineering, ASCE*, Vol. 18, No. 4, pp. 381-383

Lymberopoulos, D., Lindsey, Q., and Savvides (2006), A., *An Empirical Analysis of Radio Signal Strength Variability in IEEE 802.15.4 Network using Monopole Antennas*,

Technical Report 050501, Embedded Networks and Applications Lab (ENALAB), Yale University, New Haven, CT

MICA2DOT, Wireless Microsensor Mote. Document Part Number: 6020-0043-03, available online at: http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/6020-0043-01_A_MICA2DOT.pdf, accessed at September, 2007.

McCullouch, B. (1997), "Automating Field Data Collection on Construction Organizations," *Proceedings of Construction Congress V, ASCE*, Minneapolis, Minnesota, October 5-7, pp. 957-963.

Mohamadi, F. (2004), "A Miniature Reader/Active Tag Streamlines Supply Chain Management," *Tx/Rx Technology, RFDesign Magazine*, pp. 26-37.

Moteiv Corporation, Datasheet, available online at: <http://www.moteiv.com/products/docs/tmote-sky-datasheet.pdf>, accessed in June 2007.

Motorola, Inc., LM358 datasheet, available online at http://www.tranzistoare.ro/datasheets/166/49945_DS.pdf, accessed in June, 2007.

Murphy, K., "Kalman Filter Toolbox for Matlab," Computer Science and Artificial Intelligence Lab., MIT, 1998.

Navon, R. and Berkovich, O. (2005), "Development and On-Site Evaluation of an Automated Materials Management and control Model," *Journal of Construction Engineering and Management, ASCE*, Vol. 131, No. 12, pp. 1328-1336

Occupational Outlook Handbook, Construction Laborers, Bureau of Labor Statistics, U.S. Department of Labor, available online at: <http://www.bls.gov.oco>, accessed in September, 2007

Oloufa, A. A., Ikeda, M. and Oda, H. (2003), "Situational awareness of construction equipment using GPS, wireless and web technologies," *Automation in Construction, Elsevier*, Vol. 12, No. 6, pp. 737-748

Pahlavan, K. and Levesque, A. (2005), *Wireless Information Networks*, 2nd ed., John Wiley & Sons.

Peyret, F., Bétaille, D. and Hintzy, G. (2000), "High-Precision Application of GPS in the Field of Real-Time Equipment Positioning," *Automation in Construction, Elsevier*, Vol. 9, No. 3, pp. 299-314.

Priyantha, Nissanka B., Chakraborty, Anit and Balakrishnan, Hari (2000), "The Cricket Location-Support System," *Proceedings of 6th International Conference on Mobile Computing and Networking (MOBICOM2000)*, ACM Press, Boston MA, August 6-11, pp. 32-43.

Rappaport, T. S. (2002), *Wireless Communications: Principles and Practice*, Second Edition, Prentice Hall PTR.

Rasdorf, William J. and Herbert, Mark J. (1990), "Bar Coding in Construction Engineering," *Journal of Construction Engineering and Management*, ASCE, Vol. 116, No. 2, pp. 261-280.

Rauch, H.E., Tung, F., and Striebel, C.T. (1965), "Maximum Likelihood Estimates of Linear dynamic Systems," *American Institute of Aeronautics and Astronautics (AIAA) Journal*, AIAA, Vol. 3, No. 8, pp. 1445-1450.

RFID Journal, How Much Does an RFID Tag Cost Today, The Cost of RFID Equipment, available online at <http://www.rfidjournal.com/faq/20/85>, accessed at July 11, 2007.

Roberts, S. (2002), *As RFID Vendors Prepare for Take-Off, Analysts Strike a Note of Caution*, Frontline Solutions, White Paper, February 2002.

Römer, K., (2005), *Time Synchronization and Localization in Sensor Network*, Ph.D. Dissertation, Swiss Federal Institute of Technology Zurich.

Saidi, Kamel S., Lytle, Alan M. and Stone, William C. (2003), "Report of the NIST Workshop on Data Exchange Standards at the Construction Job Site," *Proceedings of*

20th International Symposium on Automation and Robotics in Construction (ISARC2003),
The Future Site, September 21-25, Eindhoven, The Netherlands, pp. 617-622

Savi Technologies, *Active and Passive RFID and Selecting the Right Active Frequency*,
White Paper, available online at: <http://www.autoid.org>, accessed at August 23, 2006.

Schneider, M. (2003), *Radio frequency identification (RFID) technology and its applications in the commercial construction industry*, M.S. Dissertation, Civil Engineering Department, University of Kentucky, 2003.

Skibniewski, M. and Jang, W.-S. (2006), "Ubiquitous Computing: Object Tracking and Monitoring in Construction Processes Utilizing ZigBee Networks," *Proceedings of 23rd International Symposium on Automation and Robotics in Construction (ISARC)*, October 3-5, 2006, Tokyo, Japan, pp. 287-292.

Smart Dust Mode, available online at: <http://www.electronicproducts.com>, accessed at August 23, 2007.

Song, J., Ergen, E., Haas, C. T., Akinci, B. and Caldas, C. (2006), "Automating the Task of Tracking the Delivery and Receipt of Fabricated Pipe Spools in Industrial Projects," *Automation in Construction*, Elsevier, Vol. 15, No. 2, pp. 166-177.

Texas Instrument, 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver, CC2420 datasheet, SWRS041B, available online at: <http://focus.ti.com/lit/ds/symlink/cc2420.pdf>, accessed in September 2007

The business Roundtable (1982), *Modern Management Systems*, Construction Industry Cost Effectiveness Report A-6, November

Thomas, H. Randolph, Sanvido, Victor E. and Sanders, Steve R. (1989), "Impact of Material Management on Productivity-A Case Study," *Journal of Construction Engineering and Management*, ASCE, Vol. 115, No. 3, pp. 370-384

TinyOS, An Open-Source OS for the Networked Sensor Regime, available online at: <http://www.tinyos.net>, accessed in September, 2007.

Umetani, T., Arai, T., Mae, Y., Inoue, K. and Maeda, J. -I. (2006), "Construction Automation Based on Parts and Packets Unification," *Automation in Construction*, Elsevier, Vol. 15, No. 6, pp. 777-784.

Want, Roy, Hopper, Andy, Falcão, Veronica and Gibbons, Jonathan (1992), "The Active Badge Location System," *ACM Transactions on Information System*, Association for Computing Machinery, Vol. 10, No. 1, pp. 91-102.

Weiser, M. (1993), "Hot Topics: Ubiquitous Computing," *IEEE Computer*, Vol. 26, No. 3, October.

Welch, G. and Bishop, G. (1995), *An Introduction to the Kalman Filter*, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, USA. TR95-041.

Wi-Fi, available online at: <http://www.wi-fi.org>, accessed at August 16, 2007.

Yagi, J., Arai, E. and Arai, T. (2005), "Parts and packets unification radio frequency identification (RFID) application for construction," *Automation in Construction*, Elsevier, Vol. 14, No. 4, pp. 477-490.

Yedavalli, K., Krishnamachari, B., Ravula, S. and Srinivasan, B. (2005), "Ecolocation: A Technique for RF Based Localization in Wireless Sensor Networks," *Proceedings of Information Processing in Sensor Networks (IPSN '05)*, Los Angeles, CA, April 2005

Zheng, J. and Lee, M. J. (2004), "A Comprehensive Performance Study of IEEE 802.15.4," *Sensor Network Operations*, IEEE Press, Wiley Interscience, Chapter 4, pp. 218-237.

ZigBee Alliance (2005), *ZigBee Specification*, ZigBee Document 05347r06, June 27, 2005, San Ramon, CA.