Acoustic Communication in Wireless Sensor Networks

Jingbin Zhang, Zhanxiang Huang, Xinyu Liu
Department of Computer Science
University of Virginia
{jz7q, zh4c, xl3t}@cs.virginia.edu

Abstract: *Acoustic communication has been widely used in undersea wireless networks. In this paper, we explore the use of acoustic communication in terrestrial wireless sensor networks. This work does not aim to replace the radio communication which is widely used in wireless sensor networks. Rather, we aim to provide an alternative way of communication among the motes in case that the radio transceiver fails to work. Also, acoustic communication can be used as an out-band communication channel in wireless sensor networks. For example, it can be used as busy tone which is used in some MAC protocols.*

1. INTRODUCTION

The ultimate goal of wireless sensor networks deployments is reliably reporting data with consuming the least amount of power [11]. Due to its efficiency and low cost, radio frequency (RF) communication is used as the primary means of communication in most sensor networks. However, in such networks, RF communication is inclined to fail due to the volatile nature of link quality or the possible hardware damage in the harsh environment. Thus, providing another form of effective communication in case that the radio transceiver fails to work is indispensable. This backup channel can be used retrieve some important data of the nodes whose radios are down.

Acoustic communication provides an attractive solution for continuous service of wireless sensor networks in above circumstances. Current prevailing sensor boards in use provide the basic hardware support for acoustic communication. For instance, the Mica sensor board has a sounder and a tone detector. The sounder is capable of making sound of certain pitch, which can be detected by the tone detector.

The objective of our work is to use the existing available hardware to provide reliable acoustic communication in wireless sensor networks. In this case, the additional cost for acoustic communication is zero in terms of extra hardware. Acoustic communication not only can be used as the backup channel to RF communication, but also can be used as busy tone in some MAC protocols [10], which further manifests its potential for acoustic communication.

To meet the requirements of reliable communication in wireless sensor networks, we define the following goals:

- Wide Usability.
- Effective coding and transmission.
- Simple implementation
- Adjustability to different applications

To achieve these goals, we develop our acoustic communication protocol based on existing hardware available in the sensor motes or sensor boards. The protocol has the similar interfaces as those used in RF communication. The design of the acoustic communication mainly contains two parts: The Acoustic physical layer design and the Acoustic MAC layer design, which can be used by the upper level applications.

This paper is organized as follows: Section 2 discusses the related work in acoustic communication. Section 3 describes the hardware we use for acoustic communication. Section 4 illustrates the packet format we use. Section 5 and Section 6 explain the design of Acoustic physical layer and Acoustic MAC layer, separately. We discuss the problems we met during the implementation in Section 7. Section 8 presents the evaluation of acoustic communication. We give our conclusion and future work in Section 9.

2. RELATED WORK

Acoustic communication plays an important role in the habitual relationships we have with any environment. In fact, acoustic communication in human beings and animals is of intensive study very recently. In wireless sensor network, acoustic communication has been applied to underwater applications such as oceanographic data collection, pollution monitoring offshore exploration, disaster prevention, etc. Several fundamental key aspects of underwater acoustic communications, including limited bandwidth, impaired channel, propagation delay, high bit error rates, limited battery power, error-prone sensors, are discussed in [8].

NEPTUNE [10] is an undersea observatory which uses acoustic communication as a wireless interface to connect instruments with moderate power and data requirements to the cabled network. The undersea observatory node includes many multi-purpose sensors and performs minimal pre-processing of sensor data on the node. Then the collected data is transmitted through underwater acoustic channels to acoustic receivers. The acoustic signal will be bandpass filtered and sampled at high rate and with as much dynamic range as possible to ensure that they are useful for marine mammal, monitoring, acoustic tomography, and ambient noise studies. Due to the technical difficulty of laying several kilometers of cable, and the power limitation of oceanographic instruments, acoustic communication provides an effective alternative to hardwired connection.

Another underwater acoustic communication system designed for wireless sensor networks, CORAL, is proposed in [14]. It uses special acoustic hardware which consists of piezo-transducers, a microcontroller-based architecture and interface circuitry. The communication in CORAL has only one direction, which goes from nodes to the base. Results of CORAL power consumption, signal strength, and noise measured at the reception point shows that WSN node operation underwater is feasible.

Different from the above, our work aims at providing a backup channel or out-band channel for RF communication. Instead of using specialized acoustic hardware for dedicated acoustic communication, our work requires no more extra hardware than existing one on most sensor boards. So it can be easily incorporated into current wireless sensor networks as a complement to RF communication, without extra cost. We also provide an efficient Acoustic MAC layer for the acoustic communication to be used directly by above network services.

3. HARDWARE

To achieve the goals outlined in Section 1, we do not resort to specialized acoustic communication hardware. Although such hardware is highly efficient, they are of high cost and not accessible to many sensor motes. So in our implementation, we take advantage of the existing hardware that has the potential to be used for acoustic communication. Figure 1 shows the structure of the Mica sensor board, which integrates a sounder and a microphone. The sounder and the microphone can be used as the transmitter and receiver, respectively. As a matter of fact, acoustic communication is not directly supported by the hardware and has to be implemented in software.

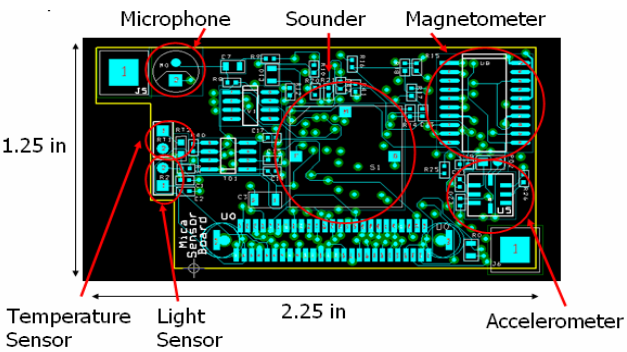

Figure 1: Mica 2 Sensor Board with an integrated sounder and Microphone.

Our acoustic communication protocol consists of two components: a transmitter and a receiver. With each Mica sensor board having the sounder and the microphone, our communication is bi-directional - each mote is able to listen as well as to talk.

A. Transmitter

The transmitter is the sounder on the Mica sensor board. The sounder is also called the “buzzer”, which is a simple fixed frequency piezoelectric resonator with drive and frequency control circuitry built into it. From our experiments, the frequency of the sound produced from the sounder is around 4.5KHz.

B. Receiver

The receiver is the hardware phase-locked loop tone detector integrated with the microphone on the Mica Sensor board. The output of the tone detector is binary. It is either 0 or 1, in which 0 means that an acoustic signal within its effective frequency range is detected and 1 means that no acoustic signal within that range is detected. The effective frequency range of the tone detector is between 4.3KHz and 5.2KHz, which matches the frequency of the chirp from the sounder.

With the binary output of the tone detector, we are unable to use Phase Shift Keying (PSK), which is a popular technique used in sonar communication to increase the bandwidth [11]. The use of PSK requires the knowledge of the received waveform. However, the tone detector only tells whether an acoustic signal within its effective frequency is detected or not, without any information about the waveform.

Although the transmitter and the receiver are separate devices, the communication is half-duplex. That is because these two devices use the same frequency and will interfere with each other if the node transmits and receives a message at the same time.

4. PACKET FORMAT

The complete packet is comprised of a payload, a Acoustic MAC layer header and an Acoustic Physical Layer header. The Acoustic Physical Layer header is comprised of the preamble, Start of Frame Delimiter (SFD) and packet length.

- *Preamble*: the preamble uses the bit sequence of 01010101….. The length of the preamble is configurable to achieve power saving. The preamble is used to indicate the arrival of a packet. A length of 4 bytes is recommended.
- *Start of Frame Delimiter (SFD)*: SFD uses bit sequence of 00001111, whose length is 1 byte. SFD is used to indicate the start of the packet contents.
- *Length*: length is the total size of the Acoustic MAC layer header including CRC, and payload.

The Acoustic MAC Layer header is comprised of the destination address, group ID, packet length and CRC.

- *Destination address*: destination address is the address of the receiver. It has 2 bytes. 0xffff is reserved for the broadcast address
- *Group ID*: group ID has 1 byte. 0xff is reserved for

the universal group.

- *Length*: length here only includes the length of the payload. It has 1 byte.
- *CRC (Cyclic Redundancy Check)*: CRC is at the end of the packet (right after the payload). It has 2 bytes. It is computed over the MAC Protocol Data Unit (MPDU). In our implementation, we use CRC-16-CCITT: $x^{16} + x^{12} + x^5 + 1$.

The payload is comprised of an array of bytes, which are filled by the applications. The maximum length of the payload is 29, to conform to the maximum length of payload in TinyOS [12].

5. ACOUSTIC PHYSICAL LAYER

The acoustic physical layer consists of the sending module and the receiving module. The responsibilities of the sending module include encoding the bit streams, and transmitting the encoded bit streams. The responsibilities of the receiving module include sampling the channel to detect the synchronization signal, recognizing the packet preamble and header, and after that receiving the packet payload including the CRC code. The Acoustic physical layer design is based on the radio layer designs used in ad hoc networks [6][7]. We adopt some design information from CC1000 [6] and CC2420 [7]. We first show the encoding method we use, and then explain the technical details of the sending module and receiving module.

5.1. ENCODING METHOD

Acoustic link established by the hardware in Section 3 is typically not perfect, so reliable encoding should be applied to reduce the risk of communication failure.

In our acoustic physical layer, we use Manchester coding as the encoding method, because Manchester coding has the self-clocking feature, which provides a simple way of encoding arbitrary binary sequences with the clock signals embedded among the data signals [13]. Thus it prevents the loss of clock synchronization, or bit errors from low-frequency drift on poorly-equalized analog links.

In Manchester encoding, bits are represented by transitions. 0 is encoded as a low-to-high transition and 1 is encoded as high-to-low transition. A Manchester encoding of bit string 10110001101 is shown in Figure 2.

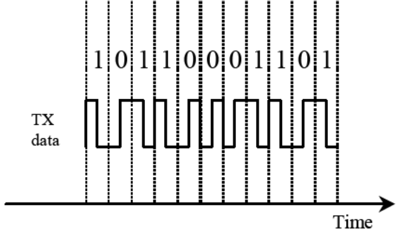

Figure 2: Manchester coding (from [6])

In the acoustic physical layer, we use the tone to represent the high value and silence to represent the low value. Thus, a bit '0' is represented by a transition from silence to tone and a bit '1' is represented by a transition from tone to silence. However, the reverse direction is not true, because a transition from silence to tone or from tone to silence can also be a clock signal. We need to deal with this on the receiver side.

Before proceeding, we first define the following two terminologies. The sending period is defined as the time interval of transmitting an encoded bit out through the sounder. Thus the bit rate is the reverse of the sending period, which is denoted in Formula (1):

$$BitRate = 1 \big/ SendingPeriod \qquad (1)$$

The receiver sampling period is defined as the time interval between two contiguous samples on the receiver side. Since we are using Manchester encoding, in which 1 encoded bit is comprised of a high value and a low value, the sampling period should be at most half of the sending period to detect both the high value and low value.

5.2. SENDING MODULE

Upon receiving a request to send a packet, the sender starts a timer with the interval of half of the sending period. This is because in each sending period, one high value and one low value are sent out, each of which lasts half of the sending period, based on the Manchester encoding. Each time the timer fires, the sender determines whether this is in the first half or the second half of a sending period. Based on that information, it decides whether to turn on or off the sounder to send out the high (tone) or low (silence) value. For example, when transmitting the bit '1', the sounder is turned on during the first half of the sending period and turned off during the second half of the sending period. Before the sender starts the transmission, it sets the state of the physical layer to BUSY state. The transmission of a packet is not interruptible. When the transmission finishes, the sender turns the sounder off and changes the state of the physical layer back to IDLE.

5.3. RECEIVING MODULE

On the receiver side, the receiver keeps sampling the channel every sampling period using the tone detector. When the receiver detects a high value which lasts for one sending period, it recognizes the clock signal and gets synchronized with the sender. Then the receiver starts interpreting the sample results synchronously, and matching them with the pattern of the preamble, 010101.... During this process, the physical layer also checks for the possible errors, such as two contiguous 0's or 1's in one sampling period. Once the preamble pattern is recognized, the receiver tries to find the start of frame delimiter (SFD), 00001111. The finite state machine shown in Figure 3 is for recognizing the SFD. The 0's and 1's on the edges are the received bits. The Unsync state denotes that the receiver is not synchronized. All the other states denote that the receiver is synchronized. State 0 represents that the preamble pattern is recognized. States from 01 to 04 and from 11 to 13 are the intermediate states of recognizing the SFD. State 14 is the final state, which denotes that the pattern of the SFD is found. After reaching the final state, the receiver starts receiving the rest of the packet.

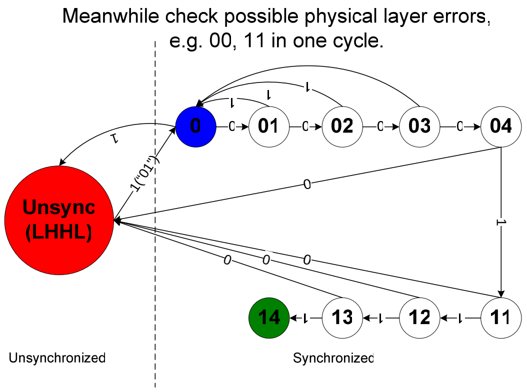

Figure 3: Finite State Machine for Recognizing the Packet Preamble and Header

In our current implementation, we choose the sampling period as half of the sending period. This choice may introduce some recognition errors, such as the unpredictable sampling results when the receiver samples at the transitions. We will discuss its impact in Section 7.

6. ACOUSTIC MAC LAYER

To facilitate the use of our acoustic layer communication, we develop a light-weight, configurable MAC layer. The Acoustic MAC layer design adopts the principle of Carrier Sense Media Access (CSMA) [4].

In RF communication, the capability of snooping on traffic over the broadcast medium is crucial for extracting information about surroundings and preventing cycles [11]. This kind of snooping can also be applied to the acoustic communication. In our implementation, we use the tone detector to snoop on the acoustic signals within its effective frequency range. If it detects some acoustic signals, it sets the channel status BUSY. Else, it sets the channel status IDLE. The Acoustic MAC layer decides whether a packet can be transmitted based on the channel status. If the channel status is BUSY, the packet is randomly backed off to avoid possible collisions. If the channel status is IDLE, the Acoustic MAC layer can call the Acoustic physical layer to transmit the packet.

Another function of the Acoustic MAC layer which is not fully implemented is link layer acknowledgements. We aim at providing an automatic ACK after a node receives a package, if ACK is enabled. The ACK is not a packet; rather it is a certain bit sequence, which is one byte long, to save energy. This kind of implementation is widely used in RF communication [6][7], which is initiated mainly by the hardware rather than software.

Similar to [11], we provide a set of interfaces that allow network services to dynamically configure the Acoustic MAC. Figure 4 shows the interfaces used to enable or disable acknowledgements, set the preamble length, and set the sampling frequency.

The configurable mechanism presented here can lead to the optimization of the applications. For example, the ability of changing the preamble length dynamically allows us to save power. In RF communication, the radio can go into doze mode periodically to save power, as long as the preamble length is long enough to cover the sleeping period and wake up the intended receiver. The same technique can also be applied here. We can set a long preamble length and let the tone detector sleep and periodically to save power.

```
interface AcousticMacLayer {
    command result_t MACEnableAck();
    command result_t MACDisableAck();
    command result_t MACSetPreambleLength
    ( uint8_t length ) ;
    command result_t MACGetPreambleLength
    ( uint8_t length );
    command result_t MACSetSamplingFrequency
    ( uint16_t frequency ) ;
}
```

Figure 4: Interfaces for MAC reconfiguration.

Other features can also be added to the current implementation of Acoustic MAC layer. For example, we can add collision detection into the current implementation by using the tone detector.

7. DISCUSSION

During our implementation and experiments, we find several problems which complicate the implementation. In this section, we discuss these problems and provide some solutions to these problems.

7.1 RECEIVER SAMPLING PERIOD

One of the goals in our design of the acoustic physical layer is to provide flexibility, which allows the user to specify parameters to accommodate different applications. However, there exist constraints for these parameters, especially for the receiver sampling period, which is critical in receiving the acoustic signals and translating them into bits correctly.

Under perfect conditions, which mean that there are no noise and circuit fault, the results of sampling always reflect the actual acoustic signal when the sampling takes place between transitions. However, if a sampling happens during a transition, the result is unpredictable. As shown in Figure 5, if the receiver samples at time t1, t2, and t3, the results from these samples are unpredictable. The sampling results can be high, high, and low, or low, low, and high, or any other combinations of low and high. In this case, if the receiver sampling period is set to half of the sending period, the receiver is unable to find the correct sequence of the high and low values. The packet is prone to be lost.

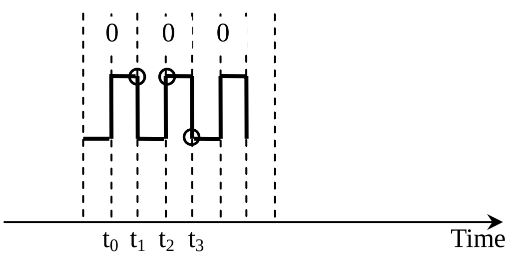

Figure 5: Receiver sampling during transitions taking place at t_1, t_2, and t_3

The error happens because only one sample is taken each half of the sending period. When the sample is taken at the transition, the receiver is unable to decide whether the result in this half of the sending period is tone or silence. In order to correctly recover the acoustic signals from the sender, we need to increase the number of samples each half of the sending period. In this case, if one sample happens to be taken at the transition, the nearby samples can be used to decide whether the result in this half of the sending period is high or low.

Here we provide the theoretical high bound of the ratio of the receiver sampling period and the sending period. Once the used ratio is below the high bound, the receiver can always correctly recover the acoustic signals, no matter where the first sample is taken, assuming perfect conditions. Assume the sending period is T, and the receiver sample period is S. Within the time interval T/2, the receiver can get maximally [T/(2*S)+1] times of the same value; within the time interval T, the minimal times of the same value are (T/S -1). So in order to distinguish a signal length of T/2 and T, the receiver sampling period S should satisfy:

$$\left[T \big/ (2 \times S) + 1 \right] < (T \big/ S - 1) \Rightarrow S < T \big/ 4 \qquad (2)$$

We concluded that the receiver should sample at a period less than a quarter of the sending period to guarantee the correctness. In such cases, the receiver can always recognize the acoustic signal correctly even though the sampling may happen at the transition time. We leave the implementation of this scheme as our future work.

7.2 SOUNDER DELAY

In the design of acoustic communication, we assume that the hardware starts to work immediately upon receiving a command. However, this is not always the case. As mentioned in section 3, the acoustic hardware used in our acoustic communication is far from perfect. In fact, we found that there is a delay between the time when the sounder is turned on and the time when the sounder actually produces the sound. This delay is approximately 2-3 milliseconds. On the other hand, the sounder becomes silent as soon as it is turned off. As a result, the length of the tone becomes shorter due to the effects of sounder delay, which is shown in Figure 6. If the receiver sampling happens at the delay interval, it will result in a false negative.

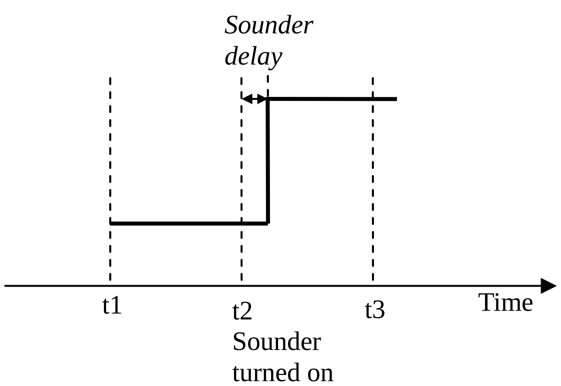

Figure 6: Sounder delay

There are basically two solutions to this problem. First, the sounder can be turned on in advance to compensate for the sounder delay, while the turn-off time remains the same. In this solution, the sounder delay should be precisely measured to ensure the correctness of receiver sampling results. The second solution is to decrease the receiver sampling period. In this way, even though the receiver gets incorrect sampling results during the delay interval, it can still make correct translations based on the previous and following sampling results.

7.3 SATURATION IN TONE DETECTOR

As stated in [17], the tone detector is saturated if the sound signals are strong or the gain value is not well adjusted. Once the tone detector is saturated, it is unresponsive to the incoming acoustic signals. During our experiments, we also find that the tone detector is possible to be saturated, even though the sound from the sounder is weak. The saturation happens when the sender and the receiver are within short distance, such as 0.2m, and the receiver uses a high gain value, such as 100. Recall that, the gain value in the tone detector ranges from 1 to 127. A higher gain value means that the tone detector is more sensitive and is easier to be saturated.

Based on the consideration of communication range, it is better to use a higher gain value which allows longer communication range. However, a higher gain value results in the possibility of saturation if the sender and the receiver are near each other. A better solution is to use dynamic gain value adjustment like Three Phase Adjustment used in [17], which is left as our future work.

8. EVALUATION

We implement our acoustic communication protocols in TinyOS [12] to evaluate its efficacy. Table 1 compares the code size of our acoustic communication protocols, B-MAC [15] and S-MAC [16]. Recall that, the acoustic communication protocol not only includes the implementation of Acoustic MAC layer, but also includes the implementation of Acoustic physical layer, while both B-MAC and S-MAC only includes the implementation of the MAC layer. From Table 1, we can see that the acoustic communication protocol is efficiently implemented.

Protocol	ROM	RAM
Acoustic Communication	4876	268
B-MAC w/LPL & ACK	4386	172
SMAC	6274	516

Table 1: A comparison of the size of acoustic communication protocols, B-MAC and S-MAC.

Based on our implementation, we evaluate the efficacy of acoustic communication on Mica2 motes. In our experiments, the length of the payload is set to 15 bytes. So the length of the MPDU is 21 bytes. All of the experiments are conducted in an indoor environment with carpet on the floor.

We first conduct the experiments by varying the sending periods. In this set of experiments, we put two motes in a room which are 1.5 feet away. One mote is the sender and the other is the receiver. For each sending period, the sender sends 100 packets to the receiver. The receiver records the number of the packets received correctly to compute the packet deliver rate.

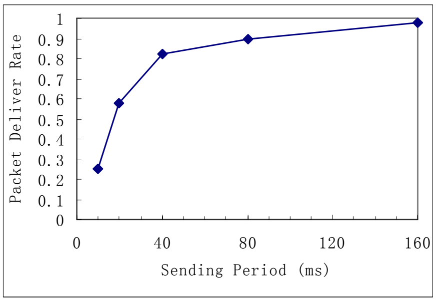

Figure 7: Packet deliver rate *vs* sending period

Figure 7 shows the different packet deliver rates when the sampling rate changes. As shown in the figure, the packet deliver rate decreases as the sending period decreases. It results from the following two reasons. First, in our current implementation, the receiver sampling period is half of the sending period. However, as illustrated in Section 7.1, the receiver has the possibility to sample at the transition between the tone and the silence. When the receiver happens to sample at the transition, the output of the tone detector is highly unpredictable. In this case, the packet is prone to be lost. When the sending period is smaller, the chance for the receiver to sample at the transition is higher. Second, as illustrated in Section 7.2, there is delay between the time when the sounder is turned on and the time when the sound comes out. When the sending period becomes smaller, the time difference between the duration of the tone and the duration of the silence is bigger. In this case, the receiver has higher possibility to sample the silence while it is supposed to be a tone.

Then we conduct the experiments by varying the communication distances and the gain values of the tone detector in the receiver side. In this set of experiments, the sending period is set to 80ms and two gain values of 115 and 60 are used. For each setting, 100 packets are sent to compute the packet deliver rate.

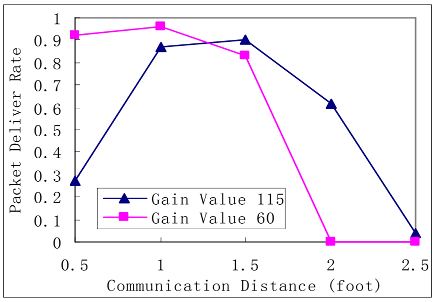

Figure 8: Packet deliver rate vs communication distance with gain value 115 and 60.

Figure 8 shows the different packet deliver rates when the communication distance is changed. Overall, the packet deliver rate decreases as the communication distance increases. However, when the gain value is set to 115, the packet deliver rate decreases dramatically when the communication distance is within 1 foot. The reason for this abnormal behavior is the saturation problem stated in Section 7.3. When the communication distance is short and the receiver uses a large gain value, the tone detector is inclined to be saturated, even though the acoustic signals from the sounder is weak. Another observation in Figure 8 is that a gain value of 115 has a longer communication range than a gain value of 60. That is because a higher gain value makes the tone detector more sensitive which is able to detect weaker acoustic signals.

From Figure 8, we can see that the communication range is quite short. It is about 2 feet. The main reason for the short communication range is the limited ability of the sounder. If the sound from the sounder is powerful enough, we expect a much longer communication range. Another reason for the short communication range is that our experiments are conducted in an indoor environment with carpet on the floor. Based on our observation, the communication range of acoustic communication tends to be larger when the floor does not have the carpet, because the carpet is able to absorb the energy of the acoustic signals effectively. The communication range of acoustic communication can reach up to 3 meters if the floor does not have the carpet. However, because we do not find such a place which is not covered with the carpet and with no people, to do a complete evaluation, we are unable to show the corresponding results here.

9. CONCLUSION AND FUTURE WORK

The acoustic communication in this paper provides a useful communication alternative in wireless sensor networks when RF communication fails. Further, it can be used as out-band channel, such as busy tone in some MAC protocols [10]. By using acoustic hardware integrated in most sensor boards, this work can be applied to many kinds of sensor networks. By making communication adjustable, our acoustic communication can be tuned to satisfy specific requirements of different applications. Besides acoustic physical layer, we also provide a simple acoustic MAC layer designed for acoustic communication. Issues including receiver sampling period, sounder delay and saturation in tone detector are discussed. Finally, our evaluation shows that the acoustic communication works well when the communication distance is short.

As future work, we are looking forward to the solutions for dynamic gain value adjustment to deal with saturation problem of the tone detector. Also, we are interested in the solution to deal with the delay in the sounder.

10. REFERENCE

[1] IEEE 802.11 Working Group, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications,” 1997.

[2] IEEE 802.11a Working Group, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications – Amendment 1: High-speed Physical Layer in the 5 GHz band,” 1999.

[3] IEEE std. 802.15.4, “Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs),” 2003.

[4] IEEE Std 802.3ae™, “Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications,” 2002.
[5] http://en.wikipedia.org/wiki/Cyclic_redundancy_check
[6] Chipcon CC1000 data sheet. http://www.chipcon.com/files/CC1000_Data_Sheet_2_2.pdf
[7] Chipcon CC2420 data sheet. http://www.chipcon.com/files/CC2420_Data_Sheet_1_1.pdf.
[8] I. F. Akyildiz, D. Pompili, T. Melodia, “Underwater Acoustic Sensor Networks: Research Challenges," Elsevier's Journal of Ad Hoc Networks, Vol. 3, Issue 3, pp. 257-279.
[9] G.A. Brigham and L.H. Royster, “Present Status in the Design of Flextensional Underwater Acoustic Transcucers,”, J. Acoust. Soc. Am. 46, 92(1969).
[10] J. Deng and Z. Haas, “Dual Busy Tone Multiple Access (DBTMA): A New Medium Access Control for Packet Radio Networks,” in IEEE ICUPC, October 1998.
[11] L. Freitag, M. Stojanovic, M. Grund, S. Singh, “Acoustic Communications for Regional Undersea Observatories,” in Proceedings of Oceanology International 2002.
[12] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler, “TinyOS: An Operating System for Wireless Sensor Networks,” In Ambient Intellligence. Springer-Verlag, 2004.
[13] T. V. Muoi, “Receiver Design for Digital Fiber Optic Transmission Systems Using Manchester (biphase) Coding,” IEEE Trans. Commun., vol. COM-31, pp. 608–619, 1983.
[14] S. Pandya, J. Engel, J. Chen, Z. Fan, and C. Liu, "CORAL: Miniature Acoustic Communication Subsystem Architecture for Underwater Wireless Sensor Networks," The 4th IEEE International Conference on Sensors, Irvine, California, 31 Oct- 1 Nov, 2005.
[15] J. Polastre, J. Hill, D. Culler, “Versatile Low Power Media Access for Wireless Sensor Networks,” in Proceedings of Sensys, 2004.
[16] W. Ye, J. Heidemann, and D. Estrin, “An Energy-Efficient Mac Protocol for Wireless Sensor Networks, ” In Proceedings of InforCom, 2002.
[17] J. Zhang, T. Yan, J.A. Stankovic, S.H. Son, “Thunder: Towards Practical, Zero Cost Acoustic Localization for Outdoor Wireless Sensor Networks,” In submission.

WHO DOES WHAT

Jingbin Zhang: finish the overall design and create the design document; take the lead in the implementation of the acoustic MAC layer; conduct the complete evaluation of acoustic communication; take part in the writing of the project report.

Zhanxiang Huang: take part in the design of acoustic physical layer; take the lead in the implementation of the acoustic physical layer; take part in the implementation of the acoustic MAC layer; take part in the writing of the project report.

Xinyu Liu: take part in the design of acoustic physical layer; take part in the implementation of the acoustic physical layer; take the lead in the writing of the project report.