ID Modulation: Embedding Sensor Data in an RFID Timeseries

Joshua R. Smith[1], Bing Jiang[2], Sumit Roy[2], Matthai Philipose[1],
Kishore Sundara-Rajan[1,2], and Alexander Mamishev[2]

[1] Intel Research Seattle, 1100 NE 45th Street, Seattle, WA 98105
Joshua.r.smith@intel.com
[2] Department of Electrical Engineering, Box 352500, University of Washington,
Seattle, WA 98195

Abstract. This paper reports the first use of *ID Modulation* to embed a bitstream representing sensor information in a standards-compliant Radio Frequency Identification (RFID) channel. Like other forms of information hiding, ID Modulation embeds a new, lower bit-rate channel in a pre-existing host channel, without requiring any changes to the protocols defining the host channel. Like most other forms of information hiding, the embedded data is represented as correlations introduced into the host channel data stream. Most previous applications of information hiding have emphasized either secrecy (as in steganography) or robustness to removal (as in watermarking). The benefit of information hiding that is most important for the application reported here is backward compatibility with pre-existing standards and hardware. It has allowed us to build a new communication layer (for transmitting sensor data) on top of current RFID infrastructure.

1 Introduction

Instrumenting the physical world with networked sensors is being recognized as an intellectually and commercially important goal. [3, 5] The vision is that myriad tiny, inexpensive networked sensor units, sometimes fancifully referred to as “smart dust” [6], will make it possible to monitor environments such as buildings, civil structures, manufacturing facilities, hospitals, natural ecosystems, or the home with unprecedented detail, enabling a wide range of novel applications such as condition monitoring, situational awareness, proactive maintenance, and in-home healthcare [4,8].

However, the need to purchase, install, replace, and dispose of batteries is a severe impediment to widespread and long term deployment of sensor networks as they have been envisaged so far. To address this shortcoming of today’s sensor networks, we are developing a platform for battery-free wireless sensing called the Wireless Identification and Sensing Platform (WISP). Wisps are passive Radio Frequency Identification (RFID) tags that are augmented to support sensing in addition to identification. Like ordinary passive RFID tags, Wisps do not have an on-board power source; instead, an external reader device transmits power to the Wisp electromagnetically.

M. Barni et al. (Eds.): IH 2005, LNCS 3727, pp. 234 – 246, 2005.

In developing Wisps, an information hiding approach has allowed us to create RFID tags that have new capabilities (notably sensing), and yet are still standards compliant and readable by conventional, off-the-shelf reader hardware. We call the technique of embedding additional application-layer data in a stream of RFID reads *ID Modulation.*

2 Background: α–Wisp

The α–Wisp, a battery-free wireless one bit accelerometer, was our first RFID-based ID-sensor.[1] It implements a primitive form of ID Modulation. The α–Wisp uses mercury switches both as sensors and as modulating elements that affect the ID it returns.

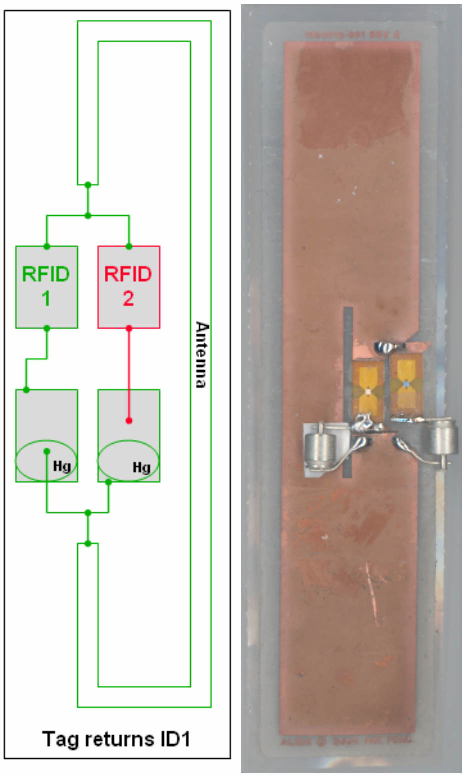


Fig. 1. Schematic diagram of α-Wisp battery-free one bit accelerometer (left); photograph of α-Wisp (right). The mercury switches are oriented anti-parallel, so that when one is open, the other is closed. In one acceleration / tilt state, RFID IC 1 is connected to the antenna, and the tag returns ID 1; in the other acceleration / tilt state, the tag returns ID 2. In α-Wisp, the mercury switches function as both sensor and modulating element. Later in the paper we split these logically distinct functions. (Figure reprinted from [1]).

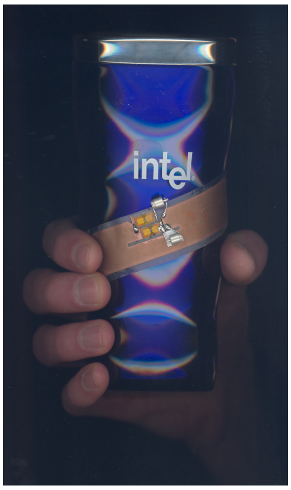


Fig. 2. The α-Wisp mounted on a coffee cup. One application of Wisp sensors is tracking human activity by object use. (Figure reprinted from [1].)

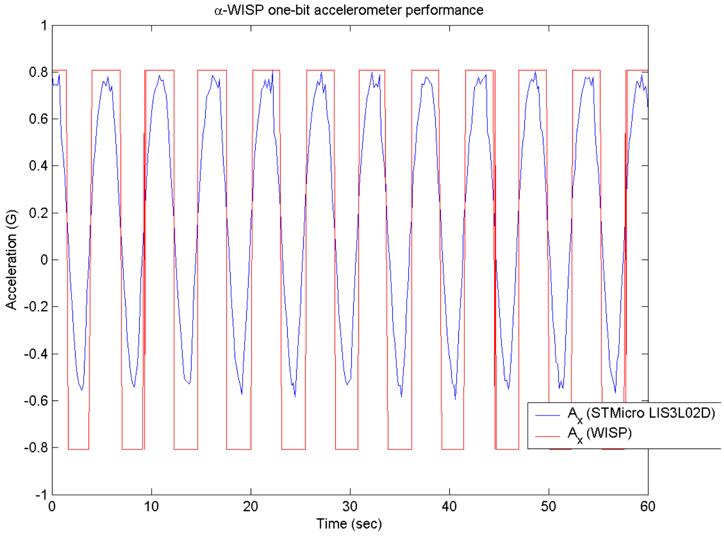


Fig. 3. α-Wisp in operation. The α-Wisp and the STMicroelectronics LIS3L02D, a conventional wired accelerometer, were mounted together and tilted periodically. The blue trace shows the x acceleration as measured by the wired accelerometer; the red trace shows the x acceleration as measured by α-Wisp, with no batteries or wires, via a conventional RFID reader. (Figure reprinted from [1].)

A mercury switch may be viewed as a simple, ultra low power inertial sensor with just one bit of dynamic range. If two mercury switches are mounted in a geometrically anti-parallel configuration, then they will in general experience acceleration that is identical in magnitude but opposite in sign. The state of the switches will therefore be anti-correlated: when one is open, the other will be closed. This enables a properly wired switch pair to function as a 2:1 multiplexer.

In the α–Wisp, geometrically anti-parallel mercury switches multiplex two RFID IC chips to one antenna. Each RFID IC is mounted in series with a mercury switch. The two IC – switch pairs are connected in parallel with one another to the antenna, as illustrated in the schematic diagram of Figure 1. (Other variants of this technique are also possible, as described in [1].) As explained above, the switch states are anti-correlated. Thus under positive acceleration (say), IC 1 will be connected to the antenna and IC 2 will be disconnected, causing the Wisp to return ID 1; under negative acceleration, the Wisp returns ID2.

To a decoder (or RFID reader, middleware system, database, warden, etc) that is unaware and uninterested in the special structure of the data stream generated by the α–Wisp, its responses look like an ordinary RFID timeseries. A decoder that is aware of the structure, however, can extract both identification and sensing data. In ordinary RFID systems, each ID is associated with a single physical object. With α-Wisp, two IDs are associated with each object. Consider the coffee cup of Figure 2, which is tagged with an α-Wisp. Seeing either ID1 or ID2 indicates that the coffee cup is present, which is what a single ordinary RFID read event conveys. But to an informed decoder, ID1 indicates that the cup is experiencing positive acceleration, and ID2 indicates negative acceleration. Seeing neither ID1 nor ID2 indicates that the coffee cup is absent.

Figure 3 shows the α-Wisp output as viewed by an informed decoder. The α-Wisp's one bit acceleration measurements were extracted by decoding the output of a conventional RFID reader according to the simple logic described above. In the figure, the α-Wisp's output is plotted against the output of a conventional wired accelerometer. The α-Wisp signal is clearly a one bit quantized version of the higher dynamic range acceleration measurement provided by the conventional accelerometer.

3 Bitstream Transmission Using ID Modulation

Transmitting a sequence of bits representing sensor data by ID Modulation is the main accomplishment in this work. As a working proof of concept, we will present a tri-axial accelerometer with one bit per axis of dynamic range that communicates its 3 bits of sensor data via ID Modulation.

3.1 Wisp Design and Implementation

Figure 4 shows a block diagram for a platform implementing bitstream ID modulation. Figure 5 is a photo of a the first working prototype. As in the original α-Wisp, this Wisp uses two RFID ICs and one RFID antenna. These original RFIDs are EPC class 1 compatible tags, the Alien Technologies ALL-9250. Unlike the original α-Wisp, this Wisp cleanly separates the sensing and modulating functions. Mercury switches are used for sensing, but no longer as modulating elements.

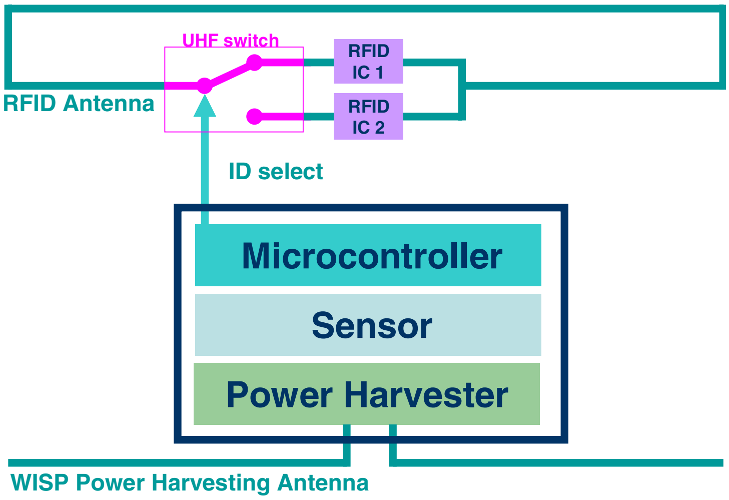


Fig. 4. Block Diagram of Wisp capable of supporting ID Modulation. The components are two RFID ICs, one RFID antenna, a GaAs switch capable of handling the 915MHz RF signal used by the base RFID system, an antenna and circuitry for harvesting power for the Wisp, a sensor, and an ultra-low power microcontroller for data acquisition and coding.

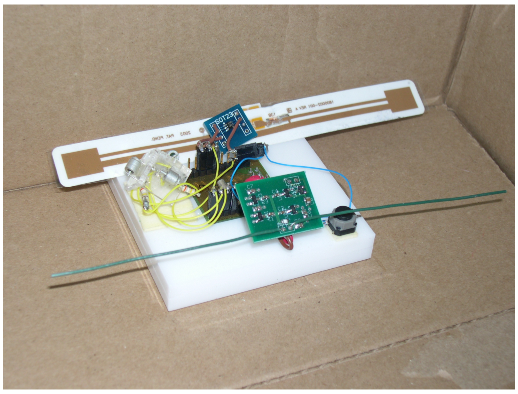


Fig. 5. Photograph of a prototype Wisp implementing the functions from the block diagram. At the top is an augmented RFID tag. The board at the bottom is our custom 915MHz power harvesting circuit. The sensor is at on the left: it is a mercury-switch-based three axis accelerometer with one bit per axis of dynamic range.

The modulation (or multiplexing) is now under electronic control, by the NEC UPG152TA SPDT GaAs switch, which is capable of switching high frequency signals (up to 2.5GHz), and offers low insertion loss.[9]

The heart (or brain) of the Wisp is a TI MSP430F1121 ultra low power microcontroller, which in its lowest power operating mode draws only 160µA at 1MHz and 2.2V, in standby mode only 0.7µA, and in off mode (with RAM retention), just 0.1µA.[7]

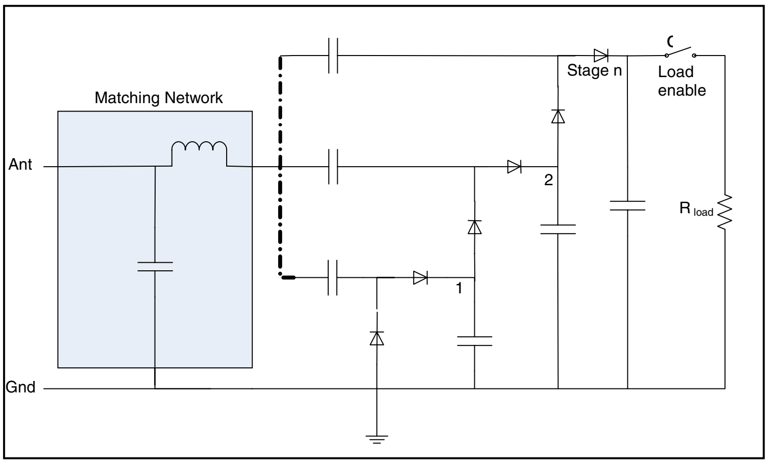


Fig. 6. The power harvesting circuit. The network highlighted in the rectangle matches the impedance of the antenna to that of the power harvesting circuit. This matching network is followed by n stages of voltage doubling. The Wisp described in this paper uses 4 stages of voltage doubling. The resistor at the right represents the load. The switch (labeled reset) indicates that the load is not always present, as the microcontroller may sleep for power conservation purposes.

Our custom-designed power harvesting unit appears at the bottom of the photo in Figure 5. Figure 6 is a schematic for the power harvesting unit. The unit consists of 4 cascaded voltage doublers, based on Agilent Technologies HSMS-2852 zero bias Schottky diodes. The size of the final filter capacitor determines how smooth the output power is, how quickly the unit powers up from zero, and how long the unit can operate in the absence of new power. At one extreme, we have experimented with using super capacitors on the order of 1F, which are able to store enough energy to enable usage models in which sensing can occur outside the field of view of the readers for substantial periods of time. A trade off is that the time to charge up the capacitor (from a current-constrained power harvester) becomes comparably long.

The sensor appears to the left of the development board. It consists of three orthogonally-mounted mercury switches. The orthogonal fixture for the mercury switches was fabricated by laser cutting acrylic. Each mercury switch is in series with a 150K resistor to ground. A "sensor enable" output pin on the microcontroller applies Vcc (the positive supply voltage) to the three switches. Three of the microcontroller's input pins are connected to the nodes between the resistors and switches. When a switch is closed, this node is pulled high; when the switch is open, it is pulled low.

3.2 Coding

For the initial experiments reported here, we have used a very simple coding scheme. The Wisp transmits packets that begin with a known synchronization sequence, and then codes the three bits using an interleaved repetition code.

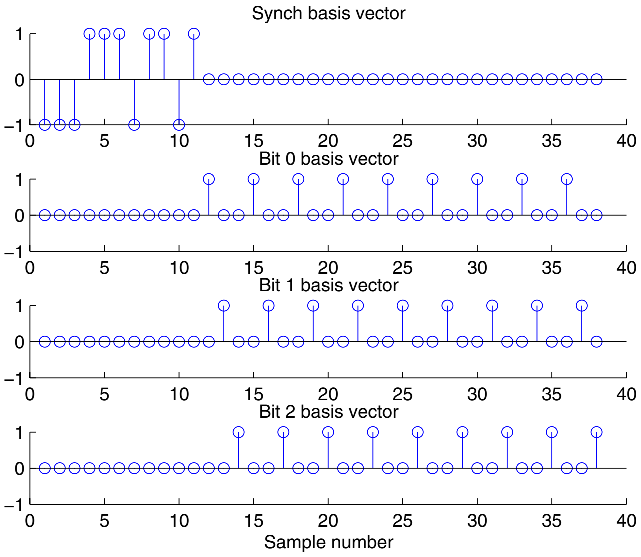


Fig. 7. Basis vectors used to encode and decode Wisp communications: Length 11 Barker Code (top subplot) for synchronization; interleaved length 9 repetition codes (subplots 2-4) for data bits.

The Wisp packets begin with a length-11 Barker sequence. The autocorrelation of a Barker sequence has the ideal "thumbtack" property for synchronization: it is sharply peaked at synchronization, and quickly becomes flat away from synchronization.[10, 2]. The Barker code is illustrated in the top subplot of Figure 7.

For these initial experiments, we have used an interleaved repetition code of rate 1/9. To improve robustness to burst errors, we have interleaved the codewords for each bit, i.e., we transmit bit 0, bit 1, and bit 2, and then repeat these 3 bits nine times. Subplots 2-4 of Figure 7 illustrates the coding of the sensor data. The figure shows stem plots of the basis functions corresponding to the synchronization signal, and for bit 0 – bit 2. Interleaving (as opposed to transmitting nine repetitions of bit 0, followed by nine repetitions of bit 1, and so on), should provide robustness to burst errors: sufficiently short burst errors will partially degrade all three bits, instead of completely erasing any single bit.

Each packet consists of 38 chips: 11 for the synchronization code, and 27 for the data. We used a chip time of 0.25s, thus the time to transmit a complete packet is 9.5s for an information bit rate of 0.3 bps.

3.3 Decoding

Decoding was performed by a host PC connected to an Alien Technologies 9RE-0001 Nanoscanner RFID reader. We implemented four simultaneous correlators in software, one for the synch signal, and one each for the data bits. Figure 7 shows the vectors that the correlators are convolving with the RFID timeseries. Because the synch pulse and the data pulses do not overlap, the target correlation sequences used by each correlator are non-overlapping. At the moment that a peak is detected in the synchronization signal (i.e. in the output of the correlator that is targeting the Barker sequence), the values on the other correlators are thresholded to determine the values for the data bits. (Even though the features that the correlators are seeking are offset from one another in time, this is handled automatically. The correlators are "loaded" with target sequences that represent the relative temporal positions of the features of interest. That is why we simply look a the correlator outputs at the moment the synch pulse is detected.) Decoding results will be presented in the next section.

4 Results

The antenna of the RFID reader was placed approximately one foot from the Wisp. For all the results and performance discussion in this paper, the Wisp was operated from harvested power.

Figure 8 illustrates ID Modulation from the Wisp's perspective. The top trace in each of the three oscilloscope screen shots shows the power supply voltage. The other two traces, which are microcontroller output pins under software control, serve as a differential signal that controls the GaAs switch. (Line 1 is set high and line 2 low to set the switch in its first state; line 1 and line 2 are reversed to place the switch in its second state).

The figure shows that the power supply voltage fluctuates dramatically, despite the final bypass capacitor. This is due to variations in the reader output. The reader signal has significant time-domain structure, such as quiet periods and amplitude modulation for downstream communication, that affect the harvested power levels. Also, the reader does not operate at a fixed frequency of 915MHz; instead it hops among 30 frequencies in the band spanning 902MHz to 928MHz. Using a network analyzer as a controlled frequency source, we verified that the Wisp's operating voltage was maximized at a drive frequency of 915MHz, and dropped about 7% (from 700mV to 650mV, for a fixed test antenna geometry) as the source was tuned down to 902MHz or up to 928MHz.

Figure 9 shows the outputs for all 4 correlators in operation for one minute. A peak in the synchronization signal is clearly visible with a period of 9.5 s as expected. Figure 10 shows a "zoomed in" look at the same trace. Beneath the synchronization pulse, one can see that the red (dashed) trace is positive, and the other two traces are negative. This is the correct result, as the red (dashed) trace represents bit 0, as the information being sent is bit 2 = 0, bit 1 = 0, and bit 0 = 1.

4.1 Performance and Characterization

The latest version of the three axis by one bit Wisp system, programmed to wake up every 150mS and toggle its ID if any sensor value has changed, draws only 7µA. The ID Modulation experiments reported here used an earlier version of the board that contained a voltage regulator. In the experiments, this earlier platform consumed 300uA to 500uA, depending on range to the reader. At higher voltages, more current was consumed.

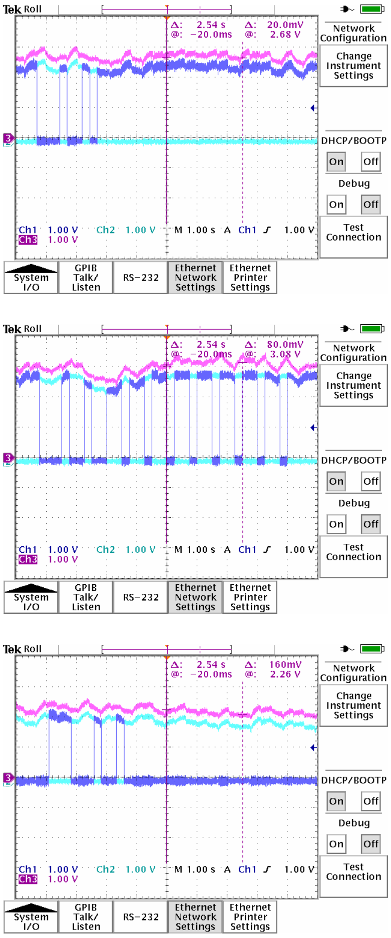


Fig. 8. Modulation viewed from the Wisp. The top trace in each figure shows the Wisp's power supply voltage. The other two traces represent the differential control signal for the multiplexer: when the darker trace voltage (input 1 to the multiplexer) is greater than the lighter trace voltage (input 2 to the multiplexer) RFID IC 1 is connected to the antenna and IC 2 is disabled; when the darker trace voltage is less than the lighter trace voltage, RFID IC 2 is enabled. The top figure corresponds to data 000; the middle figure is 001; and the bottom figure is 111. The Barker sequence is easily visible in the bottom image. The Barker sequence is also visible in the top and middle images, although the first, positive pulse in the Barker sequence is partially off the screen on the left side in these two images. The structure of the interleaved length 9 repetition code is easily visible in the middle figure.

The MSP430 requires at least 1.8V to operate. It appears that the voltage constraint, not the power constraint, is the factor that typically prevents Wisp microcontroller from operating. However, there is another factor that limits the operating range of the system. The GaAs switch negatively affects the range of the

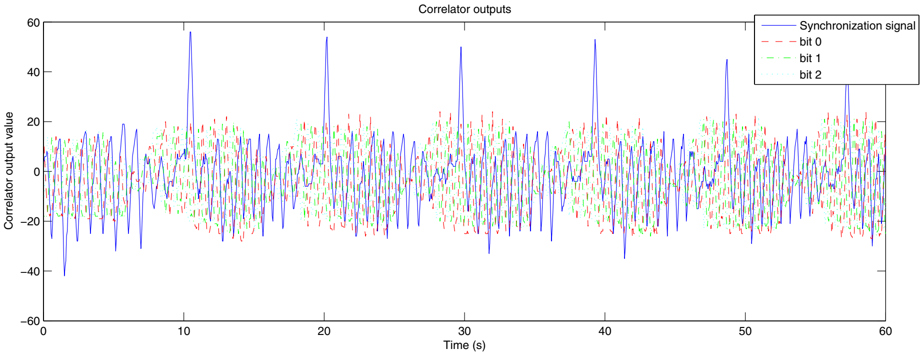


Fig. 9. Demodulated signals at the host. The RFID timeseries is feed into four simultaneous correlators. The signal that is seen spiking approximately every 10 seconds is the correlation with a length 11 Barker sequence used as a synch pulse. When a peak is detected on the synch signal, the values of the other correlators are thresholded to determine the data bit values.

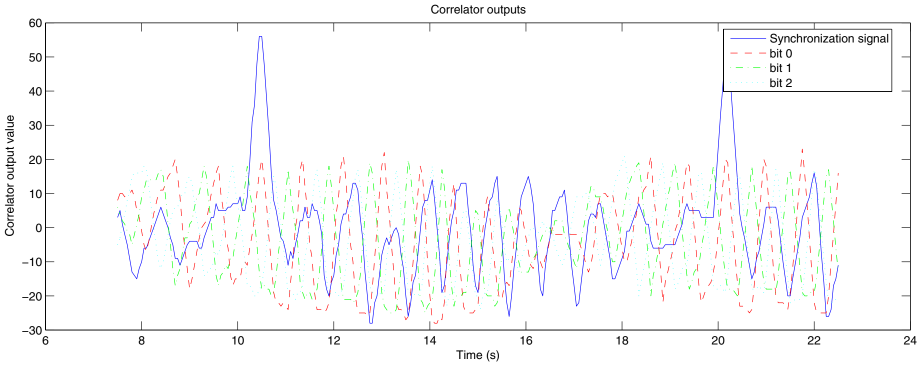


Fig. 10. Zoomed in view of demodulated signals at the host. To decode the data, the signals corresponding to the data bits are thresholded when there is a peak in the synch signal. In this case, the data value is 001 (bit 2 = 0, bit 1 = 0, bit 0 = 1).

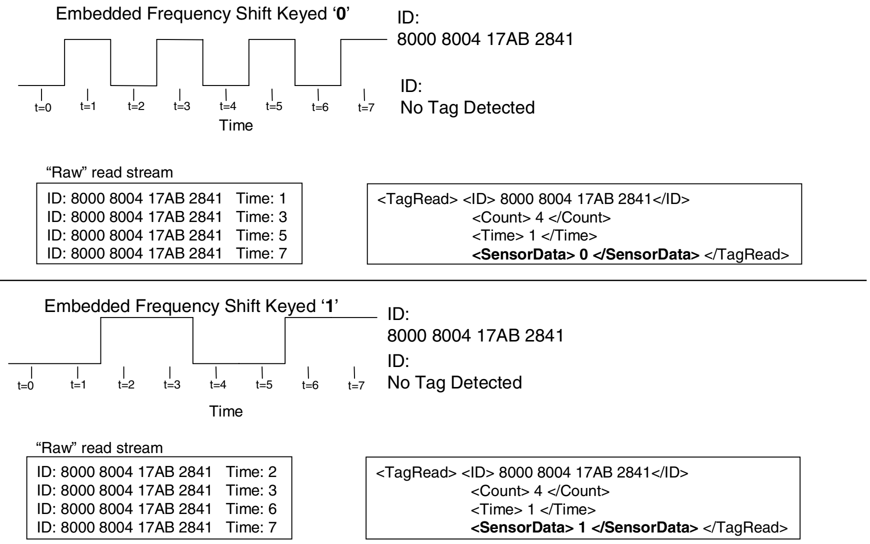
base RFID system, so currently there is a range at which the WISP harvests enough power to run, and yet cannot be read by the reader. The full system can operate at distance range of 1 to 4 feet. From 4 to 5 feet, the microcontroller runs, but the IDs cannot be read. Beyond 5 feet, the microcontroller is not able to run.

The current end-to-end communication rate is about 0.3 bps. The underlying channel can be viewed as an erasure channel, since the probability of reading “No ID” when ID 1 or ID 2 are present is so much higher than the probability of reading ID 2 when ID 1 was transmitted (a substitution error). Thus the channel can be modeled as an erasure channel. The capacity of the erasure channel is well known to be 1-BER bits per use, where BER is the bit error rate for the underlying channel symbols. In the case of the Wisp, a BER = 0.13 was typical. This corresponds to a capacity of 0.87 bits per channel use. At 20 reads per second (a typical value using the Alien reader's fast Verify command), the theoretical capacity of the embedded sensor channel is 17.4 bits per second. Certainly it should be possible to make use of more intelligent coding schemes, in order to more closely approach the theoretical optimum.

Many of the key performance metrics of the Wisp described in this paper, such as rate and range, can be improved through various optimizations and engineering measures, a process we are currently engaged in.

5 Extensions and Future Work

Figure 11 shows a version of the ID Modulation scheme that uses just a single ID for communication. Rather than using ID1 and ID2 as channel symbols, ID1 and "no ID" would be used. The higher level channel model will no longer be an erasure channel, since a failure to read will correspond to a symbol substitution. This scheme will have a higher bit error rate, and therefore lower throughput. However, it has practical benefits as well: the one ID scheme offers an increased level of compatibility with legacy databases and business processes, since most of these systems assume that there is just one ID per object. It is also nice that no ID space is sacrificed in order to



```
ID: 8000 8004 17AB 2841   Time: 1
ID: 8000 8004 17AB 2841   Time: 3
ID: 8000 8004 17AB 2841   Time: 5
ID: 8000 8004 17AB 2841   Time: 7
```

```
<TagRead> <ID> 8000 8004 17AB 2841</ID>
          <Count> 4 </Count>
          <Time> 1 </Time>
          <SensorData> 0 </SensorData> </TagRead>
```

```
ID: 8000 8004 17AB 2841   Time: 2
ID: 8000 8004 17AB 2841   Time: 3
ID: 8000 8004 17AB 2841   Time: 6
ID: 8000 8004 17AB 2841   Time: 7
```

```
<TagRead> <ID> 8000 8004 17AB 2841</ID>
          <Count> 4 </Count>
          <Time> 1 </Time>
          <SensorData> 1 </SensorData> </TagRead>
```

Fig. 11. ID Modulation using a single ID. In this example, a zero is encoded via an ID presence / absence transition frequency of 1.0 per unit time; a one is encoded as a transition frequency of 0.5 transitions per unit time. The achievable communication rate is lower using just a single ID, but this technique is more backward compatible with legacy databases and business processes, since these typically assume just one ID per physical object. It is also notable that this method does not require sacrificing any ID space to encode sensor data.

encode additional data. The single ID hiding scheme emphasizes that the additional sensor data is not encoded in the ID bits, but rather in the time structure of the RFID read stream. RFID systems typically treat ID reads as independent events. Our system encodes information in correlations among multiple events, and decodes the data by processing these events jointly. This extension will decrease sensor data communication rate, and increase backward compatibility.

Another direction we plan to take this work in the future will increase data throughput, but offer a lower degree of backward compatibility. The schemes reported in this paper encode less than one bit of sensor data per RFID read event, and rely on channel coding techniques to achieve robustness. The alternative approach that we will experiment with next is to use custom hardware to more properly implement RFID communication protocols. This will enable us to set all the ID bits to arbitrary values, which will allow us to encode more than one sensor bit per RFID read packet. This allows a larger number of sensor bits to be communicated per read event, while moving much further from the "one ID, one object" model.

An important issue that must be addressed in the future for any of the communication techniques discussed so far is that of combining the ID modulation-based sensing functionality with anti-collision protocols. The EPC specification requires support of certain primitives to perform anti-collision, but does not specify in detail how these primitives be used to support anti-collision. Those details are left to the RFID reader implementer. We believe that it is possible to create collision resolution protocols that support both ID Modulation and anti-collision, while maintaining compatibility with the EPC standard. These investigations may also suggest directions for the design of future protocols that can support the competing functions of identification (anti-collision) and sensor data communication, which both require the same scare resource, access to the reader channel.

6 Conclusion

It appears that Wisps can help deliver on the promise of sensor networks, by eliminating batteries in appropriate deployment scenarios. By using the fundamental principles of information hiding, we have been able to add a new capability---sensing---to RFID tags, while maintaining backward compatibility and preserving infrastructure investments. Most applications of information hiding thus far have focused on secrecy (steganography) or robustness to removal (watermarking). This paper highlights *standards extension* as another major benefit that information hiding can provide.

References

1. M. Philipose, J.R. Smith, B. Jiang, A. Mamishev, S. Roy, K. Sundara-Rajan "Battery-free Wireless Identification and Sensing." IEEE Pervasive Computing Magazine, Vol 4, No 1. pp. 10-18.
2. R.H. Barker, "Group Synchronizing of Binary Digital Sequences." In Communication Theory. London: Butterworth, pp. 273-287, 1953. As cited in [10].

3. D.E. Culler, H. Mulder, “Smart Sensors to Network the World,” *Scientific American*, June 2004, pp. 85-91.
4. E. Dishman “Inventing Wellness Systems for Aging in Place,” Computer, Vol 37, no. 5, 2004, pp. 34-41.
5. R. Poor, B. Hodges “Wireless Networks for Industrial Systems.” Whitepaper available from http://www.ember.com/resources/whitepapers/request.html
6. J. M. Kahn, R. H. Katz, K. S. J. Pister. "Next century challenges: mobile networking for "Smart Dust", MobiCom 1999, pp. 271-278.
7. MSP430C11x1, MSP430F11x1A Mixed signal microcontroller, http://focus.ti.com/lit/ds/symlink/msp430f1121a.pdf
8. M. Philipose, K. Fishkin, D. Patterson, M. Perkowitz, D. Hahnel, D. Fox, and H. Kautz Inferring Activities from Interactions with Objects. IEEE Pervasive Computing Magazine, Vol 3, Issue 4, pp. 50-57.
9. NEC UPG152TA L,S Band SPDT GaAs MMIC Switch, http://www.qsl.net/n9zia/wireless/pdf/u152ta.pdf
10. E.W. Weisstein et al. “Barker Code.” From MathWorld. http://mathworld.wolfram.com/BarkerCode.html