Design Considerations for Energy-Efficient Radios in Wireless Microsensor Networks

EUGENE SHIH, SEONGHWAN CHO, FRED S. LEE, BENTON H. CALHOUN
AND ANANTHA CHANDRAKASAN

*Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology,
Cambridge, MA 02139-4307, USA*

Abstract. In the past few years, wireless microsensor networks have attracted a great deal of attention in the research community. This is due to the applications that will be enabled once wireless microsensor networks are in place. The design of wireless microsensor networks, however, represents a difficult challenge. Since many applications require fault-tolerant, long-term sensing, one important challenge is to design sensor networks that have long system lifetimes. Achieving long system lifetimes is difficult because sensor nodes are severely energy-constrained. In this paper, we demonstrate system-level techniques that adapt and tradeoff software and hardware parameters in response to changes in the requirements of the user, the characteristics of the underlying hardware, and the properties of the environment. By using these power-aware, system-level techniques, we are able to reduce the energy consumption of both general, adaptable systems and dedicated point systems. Moreover, given a specific set of operating conditions for a particular system, we show how energy savings of 50% can be achieved.

1. Introduction

In recent years, the idea of wireless microsensor networks has garnered a great deal of attention by researchers [1, 2]. A distributed, ad-hoc wireless microsensor network consists of hundreds to several thousands of small sensor nodes scattered throughout an area of interest. Each individual sensor contains both processing and communication elements and is designed to monitor the environment for events specified by the deployer of the network. Information about the environment is gathered by the sensors and is delivered to a central basestation where the user can extract the desired data. Because of the large number of nodes in such a network, sensors can collaborate to perform high quality sensing and form fault-tolerant sensing systems. With these advantages in mind, many applications have been proposed for distributed, wireless microsensor networks such as warehouse inventory tracking, location-sensing, machine-mounted sensing, patient monitoring, and building climate control [1, 3–5].

Applications centered around wireless microsensors are very attractive. However, before these applications will be feasible, there are a few system challenges to resolve.

First, since the number of sensors required in these applications will be large, node densities will be high (up to 20 nodes/m^3) and large quantities of data will be produced. Intelligent and efficient data management techniques will be needed. Second, unlike traditional computing applications, the rate that events occur will be rather low ($\ll$10 kbps). Traditional systems designed for high event rates may not utilize the system resources optimally. Therefore, we will need new circuits that are optimized for operating at low-duty cycles. Third, user constraints and environmental conditions, such as ambient noise and event arrival rate,

can be time-varying. Thus, we will need a system that can adapt to varying conditions.

Another challenge for wireless microsensor networks is energy. Since sensors will be small in size, the energy resources available also will be limited. Replacing batteries is not a feasible option since the number of sensors could be large and the sensors could be inaccessible. Therefore, designing systems with long lifetimes will be necessary. In order to maximize the lifetime of a system, we will need to carefully manage the energy consumption of the system. However, designing circuits and systems that consume less energy is technically challenging. For example, current commercial radio transceivers, such as those proposed for the Bluetooth standard [6], are not ideal for microsensor applications since the energy overhead of turning them on is high. Thus, innovative solutions in transceiver and protocol design are required to achieve efficient transmission. Another challenge arises due to the remote placement of the nodes and the high cost of communication. Since sensors are remotely deployed, transmitting to a central basestation has high energy cost. Thus, algorithms that favor local computation will use less energy than those that require significant communication with neighboring nodes or the basestation.

To reduce energy consumption, there are a number of strategies that can be used. For example, the use of low-power circuit design methods such as parallelization [7] and leakage control [8] can be applied to reduce the energy of computation. At the same time, tremendous energy savings can be realized if media-access control (MAC) and routing protocols carefully manage their use of communication. Finally, because of the dynamic nature of these networks, *power-aware techniques* can also be applied to reduce energy consumption over the lifetime of the network [9, 10].

In this paper, we describe specific techniques that can be used to reduce the energy consumption of the radio. In general, the energy consumption of a node is dominated by the communication energy. Therefore, reducing the energy consumed by the radio can greatly reduce overall system lifetime. Specifically, we focus on energy-efficient design techniques for the *physical and media-access control* (*MAC*) layers of the radio. We will show how to exploit the tradeoff between communication and computation at every level of our system in order to achieve a low power design. Through these power-aware, system-level techniques, we will show how energy consumption can be reduced for *any* system. In addition, given the operating conditions of

the system, including the user requirements, we show that energy savings of 50% or more can be achieved.

Before describing methods that can be used to design energy-efficient radios for wireless microsensor networks, a description of some key applications will be introduced.

1.1. Applications

Wireless microsensor networks can be used in a variety of applications. In this paper, the design choices discussed are motivated by applications for one of two different scenarios. The first application is vehicle detection and tracking. In this application, we are interested in detecting the presence of a vehicle within a region of interest and in its velocity and direction of travel. To perform vehicle tracking, hundreds to thousands of energy-constrained nodes are scattered over a fixed region. We assume that the nodes are arranged in a dense fashion. In this kind of network, inter-node data rates are quite low (≤ 1 Mbps) and packets sizes are assumed to be relatively small (<5 kbits). Furthermore, transmission distances between nodes are short (≤ 10 m). On the other hand, the distance between a node and the basestation can be as much as several kilometers. Since communication costs over such large distances is energy-draining, communication to the basestation should be avoided unless absolutely necessary. As such, data collected by the sensors should be processed locally as much as possible before being sent to a central basestation. To take advantage of locality, sensors may also form hierachical clusters. If the clusters are formed intelligently, system lifetime can be prolonged. Figure 1 shows a simple model of our vehicle tracking application.
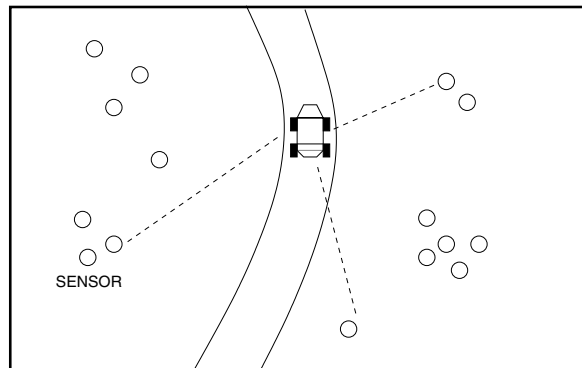

Figure 1. Vehicle tracking application.

In the later part of the paper, we will consider the design of energy-efficient MAC protocols. However, instead of looking at ad-hoc networks, we will focus on a coordinated network used for machine-mounted sensing. In this application, energy-constrained sensors communicate to a single nearby high-powered bases-tation (<10 m) where latency requirements of the data must be strictly guaranteed. Signal processing aggregation techniques cannot be easily applied since sensors monitor different properties (e.g. temperature, pressure) of the machines. Details about this particular application will be given in Section 6.

Before describing the tradeoffs available to designers of energy-efficient radios, we will first introduce the architecture of the node prototype that we have implemented for wireless sensing applications.

2. The μAMPS Wireless Sensor Node

The μAMPS (micro-Adaptive Multi-domain Power-aware Sensors) node is a wireless sensor device that allows protocols and algorithms to adapt underlying parameters of the physical hardware, including radio parameters. Figure 2 gives an overview of the architecture of the sensor node. The overall node can be broken down into different variables that define the energy consumption at each architectural block, from leakage current in the integrated circuits to the output quality and latency requirements of the user. These energy variables are exposed to the rest of the system, and thus, can be exploited at the software level to extend system lifetime and meet user constraints.

2.1. Architectural Overview

Whether for equipment monitoring, military surveillance, or medical sensing, information about the envi-

ronment must be gathered using a *sensing subsystem* consisting of sensors connected to analog-to-digital (A/D) converters. Our initial node contains an electret microphone for acoustic sensing. However, a wider variety of sensors is supported. The acoustic sensor is connected to a 12-bit A/D converter capable of converting data at a rate of 125 kilosamples per second (kSPS). In the vehicle tracking application, the required conversion rate is about 1 kSPS. An envelope detector is also included to allow ultra-low power sensing.

Once enough data is collected, the *processing subsystem* of the node can digitally process the data or the node can relay the data to a nearby node (or far-away basestation). The primary component of the data and control processing subsystem is the StrongARM SA-1110 microprocessor. Selected for its low power consumption, good performance, and static CMOS design, the SA-1110 runs at a clock speed of 59 MHz to 206 MHz. The processing subsystem also includes RAM and flash ROM for data and program storage. A multi-threaded "μ-OS" running on the SA-1110 has been customized to allow software to scale the energy consumption of the processor. Code for the algorithms and protocols are stored in ROM.

In order to deliver data or control messages to neighboring nodes, the data from the StrongARM is passed to the *radio subsystem* of the node via a 16-bit memory interface. Additional protocol processing and data recovery is performed by a Xilinx FPGA. The front-end of the radio is a Bluetooth-compatible commercial single-chip transceiver [11]. The carrier frequency is generated from a 10 MHz clock oscillator and is programmable from 2.4 to 2.5 GHz. The on-chip phase-locked loop (PLL), transmitter chain, and receiver chain can be shut-off via software or hardware control for energy savings. Table 1 shows the power dissipated by the radio when in four different modes. To transmit data, an external voltage-controlled oscillator (VCO) is directly modulated. This radio architecture is simple in
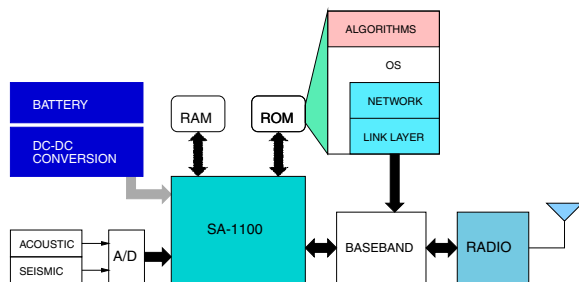

Figure 2. Architectural overview of our microsensor node.

Table 1. Power consumption of the radio in various operating modes. Since the node is based on commerical off-the-shelf technology, the power numbers can be lower. Note that supply voltage is 3.3 V.

Mode	Current (mA)	Power (mW)
Transmit active	47.6	157.1
Transmit idle	1.2	4.0
Receive active	84.4	276
Receive idle	9.3	30.7

design and allows reduced power consumption. However, the amount of data that can be transmitted continuously will be limited.

When modulating the VCO, the PLL can either be closed or open. There are advantages and disadvantages to both techniques. In open loop modulation, the PLL must first be locked to the desired carrier frequency before transmission. This will take a fixed period of time. The exact time needed for the PLL to lock is dependent on the exact implementation of the radio. During data transmission, the loop is open and the VCO is directly modulated by the baseband signal. However, since the VCO frequency will drift when the PLL is opened, a limited number of bits can be received correctly depending on the drift rate and the sensitivity of the receiver. In closed loop mode, the PLL is left on during transmission. Consequently, more power will be used during transmission. Furthermore, long sequences of ones or zeros will be distorted by the high pass nature of the PLL and hence, cause the receiver to decode the data incorrectly [12].

To receive data, the incoming signal is downconverted to 110.6 MHz and then demodulated via a quadrature tank. The radio module, with two different power amplifiers, is capable of transmitting at 1 Mbps at a range of up to 100 m. Since the radio is half-duplex the whip antenna can be shared by the transmitter and receiver.

Power for the node is provided by the *battery subsystem* via a single 3.6 V DC source with an energy capacity of approximately 1500 mAH. Switching regulators generate 3.3 V, and adjustable 0.9–2.0 V supplies from the battery. The 3.3 V supply powers all digital components on the sensor node with the exception of the processor core. The core is specially powered by a digitally adjustable switching regulator that can provide 0.9 V to 2.0 V in thirty discrete increments. The digitally adjustable voltage allows the SA-1110 to control

its own core voltage, enabling the use of a well-known technique known as dynamic voltage scaling [13, 14].

3. Radio Design Considerations

As mentioned previously, we are interested in how to design a radio for use in wireless microsensor networks. A simple high-level view of the architecture of a generic radio is shown in Fig. 3. Each of the blocks uses some amount of power during transmitting and receiving of data over the wireless channel. The blocks that are most relevant include the baseband block, the modulation/demodulation block and the power amplifier. Even with this simple model, we can propose several ways to reduce the overall energy consumption of the radio.

First, to reduce overall energy consumption, we can attempt to reduce the energy consumed by each of the blocks in isolation. In the baseband block, low-power circuit techniques and clever architectural and algorithmic techniques can be applied. In the modulation block, there are several ways to reduce energy consumption. In particular, one can change the modulation scheme used. In Section 4, the impact of M-ary modulation versus binary modulation on system energy consumption is explored. Intuitively, reducing the number of bits transmitted can reduce the duration of transmission and the energy consumption. Once a modulation scheme is chosen, various transmitter and receiver architectures can be used to implement the modulation scheme (e.g., direct, fractional-N, and heterodyne architectures). Each of these architectures have varying energy cost. Techniques can also be used to improve the energy consumption of the power amplifier. Again, circuit techniques to improve the efficiency of the power amplifier will be useful. In addition, since power amplifiers have different efficiencies depending on their
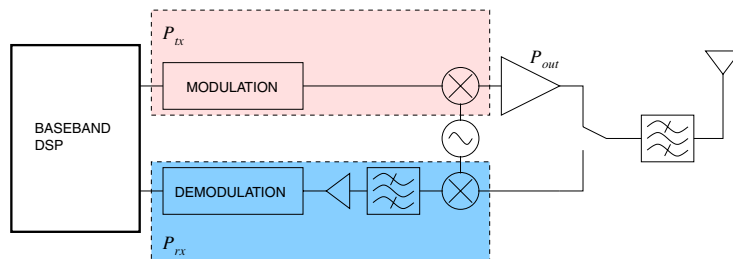

Figure 3. Simple high-level radio architecture. A typical radio consists of a baseband portion to format the data for transmission, a modulation and demodulation block, and an antenna.

operating conditions, the use of *multiple* power amplifiers can help to reduce energy consumption.

Instead of attempting to reduce energy consumption of individual blocks in isolation, designers can also tradeoff energy consumption *across* blocks. For example, in Section 5, the tradeoff between the energy used by error-control coding and the energy used by the power amplifier is considered. By using channel coding, more energy will be required in the baseband block. However, this increase will allow us to decrease the energy required by the power amplifier for a fixed bit error rate. In order to better understand the tradeoffs that can be made at each block and across blocks, we will introduce a measurement-based model of the radio.

3.1. Radio Model

One way to express the average end-to-end power consumption of a radio for use in a wireless microsensor is

$$P_{radio} = N_{tx}[P_{tx}(T_{on-tx} + T_{st}) + P_{out}T_{on-tx}]$$
$$+P_{bb-tx} + N_{rx}[P_{rx}(T_{on-rx} + T_{st})] + P_{bb-rx}$$

$$(1)$$

where $N_{tx/rx}$ is the average number of times per second that the transmitter/receiver is used, $P_{tx/rx}$ is the power

consumption of the transmitter/receiver, P_{out} is the output transmit power, $T_{on-tx/rx}$ is the transmit/receive on-time (actual data transmission/reception time), and T_{st} is the startup time of the transceiver as shown in Fig. 3. $P_{bb-tx/rx}$ represents the average power consumed by the baseband block. In general, the power consumed by the baseband block for transmission and reception will be different. Note that $N_{tx/rx}$ will largely depend on the application scenario and the media-access control (MAC) protocol being used. Also note that $T_{on-tx/rx} = L/R$, where L is the packet size in bits and R is the data rate in bits per second. In this radio model, the power amplifier is on only when communication occurs.

During the startup time, no data can be sent or received by the transceiver. This is because the internal phase-locked loop (PLL) of the transceiver must be locked to the desired carrier frequency before data can be modulated or demodulated successfully. The startup time will vary depending on the underlying implementation of the radio. In the μAMPS node, the low power transceiver has a measured startup transient of about 470 μs as shown in Fig. 4. The control input to the voltage-controlled oscillator (in volts) is plotted versus time.

It is necessary to highlight a few key points about the radio we use in our design. First, at the 2.4 GHz frequency band (as in other gigahertz bands), the power
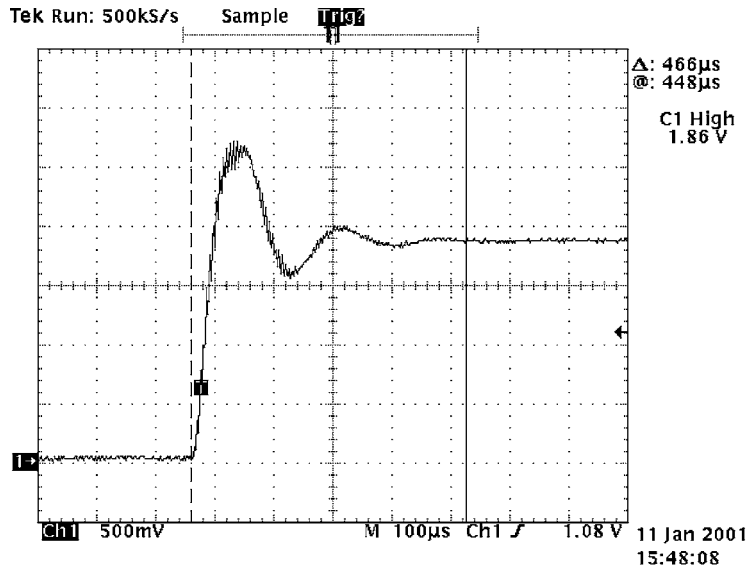

Figure 4.   Measured startup transient ($T_{st} \approx 470\ \mu$s) of a commercial low power transceiver. The control input to the VCO (in volts) is plotted versus time. When the PLL is not on, the control input to the VCO is low. Once the PLL is turned on, it takes T_{st} for the control input to settle to the right voltage.
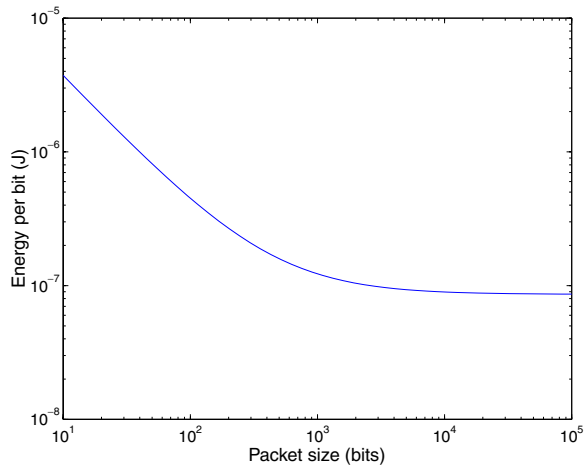
Figure 5. Effect of startup transient where $R = 1$ Mbps, $T_{st} \approx 450\ \mu$s, $P_{tx} = 81$ mW, and $P_{out} = 0$ dBm.

Figure 6. The μAMPS Radio. This radio is designed to stack onto the main μAMPS processor board (not shown).

consumption of the transceiver is dominated by the frequency synthesizer which generates the carrier frequency. Hence, to a first order, the data rate R will not affect the power consumption of the transceiver [15]. Second, the startup time can have a large impact on the average energy consumed per bit, E_b, since wireless sensor networks tend to communicate using short packets. In order to save power, a natural idea is to turn off the radio during idle periods. Unfortunately, when the radio is needed again, a large amount of power is spent to turn it back on; transceivers today require an initial startup time on the order of hundreds of microseconds during which large amounts of power is wasted. Given that $P_{tx} = 81$ mW and $P_{out} \approx 0$ dBm, the effect of the startup transient is shown in Fig. 5, where the energy per bit is plotted versus the packet size. We see that as packet size is reduced, the energy consumption is dominated by the startup transient and not by the active transmit time. Hence it is important to take this inefficiency into account when designing energy-efficient communication protocols (e.g., by sending larger packets). The radio parameters used here are based on a state-of-the-art commercial low power transceiver available today [11]. Note that P_{rx} for our radio is about 2 to 3 times higher than P_{tx} since more circuitry is required to receive a signal. This is true for most radios designed for short range communication. In our radio, $P_{rx} = 180$ mW. Figure 6 shows the radio used in the node.

While the model presented in (1) specifically describes the radio that is used in our node, it is sufficiently generic enough to apply to other short range

radios. Thus, we will use the model as a basis for evaluating various strategies that can be used to reduce energy consumption. First, we will show how to choose the lowest power modulation scheme given information about the underlying power consumption of the radio front-end (e.g. startup time, power of frequency synthesizer, etc.).

4. Modulation Issues

As evidenced by (1), one way to increase the energy efficiency of communication is to reduce the transmit on-time of the radio. This can be accomplished by sending multiple bits per symbol, that is, by using M-ary modulation. Using M-ary modulation, however, will increase the circuit complexity and power consumption of the radio. In addition, when M-ary modulation is used, the efficiency of the power amplifier is also reduced. This implies that more power will be needed to obtain reasonable levels of transmit output power. In this section, we provide some simple rules to help radio designers decide on an energy-efficient value for M.

The architecture of a generic binary modulation scheme is shown in Fig. 7(a), where the modulation circuitry is integrated together with the frequency synthesizer [11, 16]. To transmit data using this architecture, the VCO can be either directly or indirectly modulated.
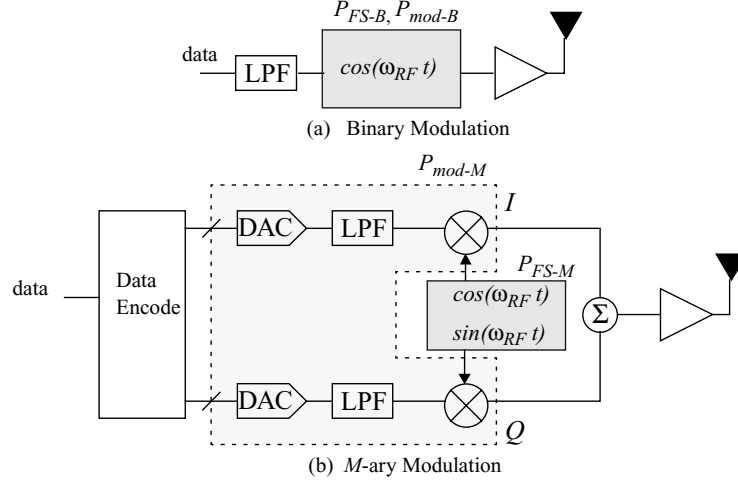
Figure 7. Binary vs. M-ary modulation.

The architecture of a radio that uses M-ary modulation is shown in Fig. 7(b). Here, the data encoder parallelizes input bits and then passes the result to a digital-to-analog converter (DAC). The analog values serve as output levels for the in-phase and quadrature components of the output signal.

The energy consumption for the binary modulation architecture can be expressed as

$$E_{bin} = (P_{mod-B} + P_{FS-B})T_{on} + P_{FS-B}T_{st}$$
$$+ P_{out-B}T_{on} \qquad (2)$$

while the energy consumption for M-ary modulation is

$$E_M = (P_{mod-M} + P_{FS-M})\frac{T_{on}}{n} + P_{FS-M}T_{st}$$
$$+ P_{out-M}\frac{T_{on}}{n}$$
$$= \frac{(\alpha P_{mod-B} + \beta P_{FS-B})T_{on}}{\log_2 M} + \beta P_{FS-B}T_{st}$$
$$+ \frac{P_{out-M}T_{on}}{\log_2 M}. \qquad (3)$$

In these equations, P_{mod-B} and P_{mod-M} represents the power consumption of the binary and M-ary modulation circuitry, P_{FS-B} and P_{FS-M} represent the power consumed by the frequency synthesizer, P_{out-B} and P_{out-M} represent the output transmit power for binary or M-ary modulation, T_{on} is the transmit on-time, and T_{st} is the startup time. As mentioned, for the same number of bits, the on-time for M-ary modulation is less

than that for binary modulation. Note that $n = \log_2 M$, the number of bits per symbol. The factors of α and β can be expressed as

$$\alpha = \frac{P_{mod-M}}{P_{mod-B}} \quad \beta = \frac{P_{FS-M}}{P_{FS-B}}.$$

Here, α represents the ratio of the power consumption of the modulation circuitry between M-ary and binary modulation, while β is the ratio of the synthesizer power between the M-ary and binary schemes. Basically these parameters represent the overhead that is added to the modulation and frequency synthesizer circuitry when one switches from a binary modulation scheme to an M-ary modulation scheme.

When we compare (2) and (3), we can see that M-ary modulation achieves a lower energy consumption when the following condition is satisfied.

$$\alpha < n\left[1 + \frac{P_{FS-B}\left[\left(1 - \frac{\beta}{n}\right)T_{on} + (1 - \beta)T_{st}\right]}{P_{mod-B}T_{on}}\right]$$
$$+ n\frac{P_{out-B}}{P_{mod-B}} - \frac{P_{out-M}}{P_{mod-B}} \qquad (4)$$
$$\approx n\left[1 + \frac{P_{FS-B}\left[\left(1 - \frac{\beta}{n}\right)T_{on} + (1 - \beta)T_{st}\right]}{P_{mod-B}T_{on}}\right] \qquad (5)$$

The last two terms of (4) can be ignored since P_{out-B} and P_{out-M} are negligible compared to the power of the frequency synthesizer since the transmission distance is short. A comparison of the energy consumption of binary modulation and M-ary modulation is
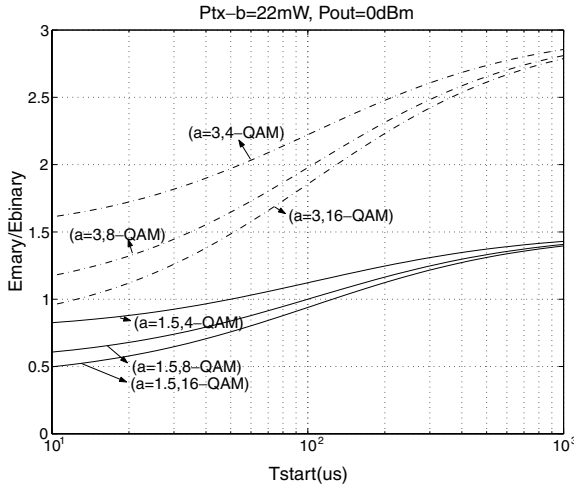
Ptx–b=22mW, Pout=0dBm

Figure 8.   Energy saving vs. overhead (α).

shown in Fig. 8. In the figure, the ratio of the energy consumption of M-ary modulation (E_M) to the energy consumption of binary modulation (E_{bin}) is plotted versus the startup time for different overhead factors ($\alpha = 1.5, 3$). We also vary M to derive different M-ary modulation schemes. For each scheme, we assume that 100 bit packets are sent at 1 Mbps. This implies that in an M-ary scheme, $1/\log_2 M$ megasymbols are sent per second and the on-time is decreased. From Fig. 8, we can see that the energy consumption for the M-ary modulation scheme is lower when α is small and M is large since the on-time (T_{on}) is shorter. We also notice that the energy savings we get from M-ary modulation depends not only on the overhead but also on the startup time. Although M-ary modulation reduces on-time and saves energy during active transmission, startup time is a cost that can limit the amount of energy savings. It can be seen that for $\alpha = 1.5$, T_{st} must be less than $40\ \mu s$ in order for the M-ary scheme to achieve lower power than the binary case. As α is increased, it becomes more difficult for the M-ary scheme to achieve lower energy than the binary scheme since T_{st} becomes a dominant factor. Therefore, we see that the startup transient plays an important role in choosing a low power modulation scheme.

5.   Power-Aware Communication

In the previous section, an energy-efficient technique that applies to a specific block of Fig. 3 was shown (i.e., the modulation block). In this section, we discuss a power-aware technique that is applied across blocks. Specifically, we explore the idea of trading off the energy used by the power amplifier for the energy used by the baseband block for the energy.

5.1.   Data Reliability

As mentioned, the nodes in a wireless sensor network are severely energy-constrained. Thus, to prolong system lifetimes, circuits, algorithms and protocols that adapt the energy consumption to changing system scenarios are required. However, while reducing energy consumption, the techniques should also be aware of changes in user-specified quality requirements. Any attempt to reduce energy consumption must not compromise the quality requirements of the user.

In the design of any communication system, one parameter of interest to users is the reliability of the links between a transmitter and a receiver. Reliable data transfer can be provided either by increasing the output transmit power (P_{out}) of the radio or by adding forward error correction (FEC) to the data. With the use of FEC, we can decrease the probability of bit error (P_b) for any fixed value of the output transmit power. However, FEC will also require additional processing and thus, additional energy at the transmitter and receiver. Depending on the FEC algorithm used and the implementation of the algorithm, the additional processing may require so much power that any savings made in the reduction of the output transmit power will become negligible.

The level of reliability provided by the communication subsystem will depend on the needs of the application and user-specified constraints. In many wireless sensor networks, such as machine monitoring and vehicle detection networks, the actual data will need to be transferred with an extremely low probability of error with predictable latency.

In our application, we assume that the objects of interest will be mobile (e.g. vehicles moving) and that the nodes themselves are immobile. Given that the carrier frequency is near 2.4 GHz and the data rate is 1 Mbps, we can assume that the coherence time of the channel is not much larger than the signaling time or one period of the data rate (1 μs). Given this scenario, we can assume that nodes communicate over a frequency non-selective, slow Rayleigh fading channel with additive white Gaussian noise. This is a reasonable channel model to use for communication at 2.4 GHz where line-of-sight communication is not always possible.

Consider one node transmitting data to another over such a channel using the radio described in Section 3.1. The radio presented uses non-coherent binary frequency-shift keying (FSK) as the modulation scheme. For comparison purposes, the best achievable probability of error using raw, non-coherent binary FSK over a slowly fading Rayleigh channel will be presented. Let P_e be a function of the received *energy per bit to noise power ratio* ($\gamma_{b,rx}$).

In general, $\gamma_{b,rx} = \alpha^2(E_b/N_0)$, where α is a random variable for a fading channel. It is shown in [17] that the probability of error using non-coherent, orthogonal binary FSK is $P_e = \frac{1}{2+\bar{\gamma}_{b,rx}}$, where $\bar{\gamma}_{b,rx}$ is the average $\gamma_{b,rx}$. Unfortunately, this does not directly tell us the amount of transmit power P_{out} that must be used in order to get a certain probability of error. In order to determine P_e as a function of P_{out}, we must consider the implementation of the radio. In general, one can convert $\gamma_{b,rx}$ to P_{out} using

$$\left(\frac{E_b}{N_0}\right)_{rx} = \frac{P_{out}}{P_{loss}\overline{\alpha^2}} \cdot \frac{1}{WN_{th}N_{rx}} \qquad (6)$$

where P_{loss} represents the large-scale path loss, $\overline{\alpha^2}$ is the average attenuation factor due to fading, W is the signal bandwidth, N_{th} is the thermal noise and N_{rx} is the noise contributed by the receiver circuitry known as the noise figure. In general, $P_{loss} \propto \frac{1}{4\pi d^k}$, $2 \leq k \leq 4$.

A conservative estimate for $P_{loss}\overline{\alpha}$ is about 70 dB at 10 m [18, 19]. With a signal bandwidth of $W = 1$ MHz, $N_{th} = -174$ dBm/Hz and $N_{rx} \approx 10$ dB, we find that $P_{out} = E_b/N_0 - 34$ dBm assuming a data rate of 1 Mbps. This equation can be used to find the transmit power needed to obtain a certain average E_b/N_0. The uncoded curve in Fig. 9 shows the probability of error plotted against the output power of the transmitter.

Since using a power amplifier alone is highly inefficient, forward error correction (FEC) can be applied to the data to decrease the probability of error. Many types of error-correcting codes can be used to improve the probability of bit error. However, we will only consider convolutional codes with base coding rates of $R_c = 1/2$ and punctured derivatives. For a frequency non-selective, Rayleigh fading channel, a bound on the P_e can be determined by applying

$$P_e < \frac{1}{k} \sum_{d=d_{\text{free}}}^{\infty} \beta_d P(d).$$

Here d represents the Hamming distance between some path in the trellis decoder and the all-zero path, the co-
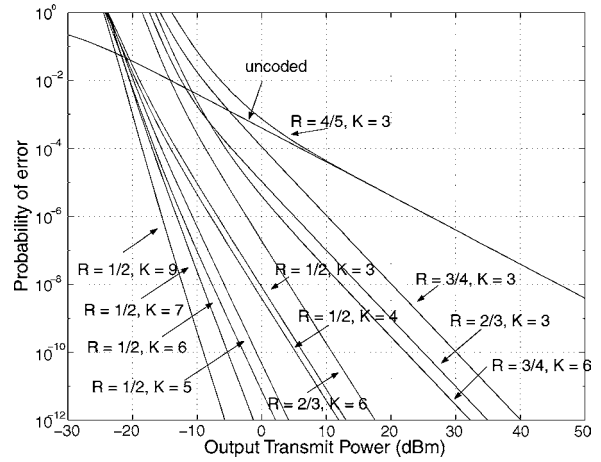

Figure 9. The probability of error of different rate convolutional codes plotted versus the transmit power for the radio described in Section 3.1. $P_{loss} = 70$ dB, $N_{rx} = 10$ dB, and $R = 1$ Mbps.

efficients $\{\beta_d\}$ can be obtained from the expansion of the first derivative of the transfer function, $P(d)$ is the first-event error probability, and d_{free} is the minimum free distance [17]. Figure 9 shows the P_e for codes with varying rates R_c and constraint lengths K. The constraint length is essentially the maximum length of the encoder or the number of previous outputs that affect the current encoder output. Note that the probabilities shown assumes the use of a hard decision Viterbi decoder at the receiver. We see that greater redundancy (lower rate) or more memory (higher constraint length) lowers the output power for a given P_e. From this perspective, coding should *always* be used.

5.2. Energy Consumption of Coding

As shown, the use of FEC can decrease the transmit power. However, the additional processing required will increase the energy of computation. Depending on the underlying architecture, energy cost can be significant. Additional processing energy, denoted by E_{dsp} must be expended in order to encode and decode the data. Additional energy cost will be incurred during the communication of the message since encoding a bit stream will increase the size of the packet by approximately $1/R_c$, thereby increasing T_{on} and the radio energy required to transmit a packet. If we denote the energy to encode as $E_{dsp}^{(e)}$ and decode data as $E_{dsp}^{(d)}$, then the total energy cost of the communication can be
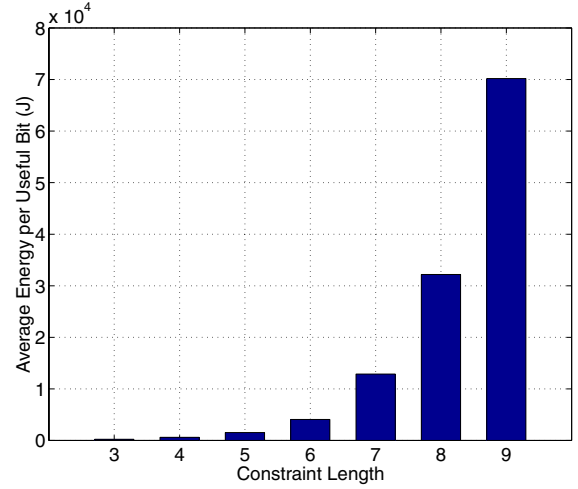
derived from (1) as

$$E = P_{tx}(T_{on-tx} + T_{st}) + P_{out}T_{on-tx} + E_{dsp}^{(e)}$$
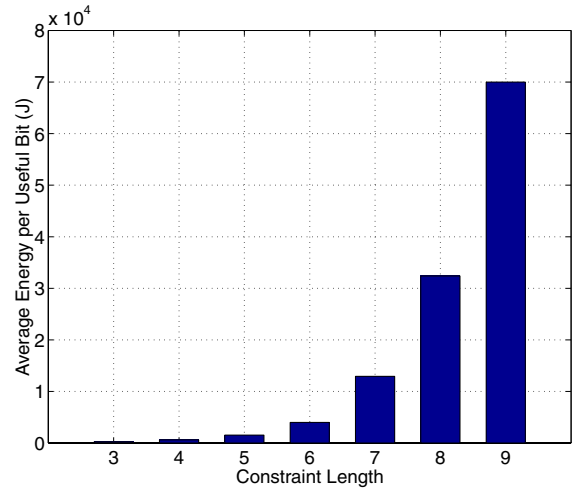$$+ P_{rx}(T_{on-rx} + T_{st}) + E_{dsp}^{(d)} \qquad (7)$$

Given this model, we can then derive the average energy to transmit, receive, encode and decode each information bit. If R_c is the code rate and L is the packet length transmitted, then the number of information bits is $L' \approx LR_c$. Thus, the energy per useful bit is $E_b = E/L'$.

In general, for convolutional codes, the energy required to encode data is negligible. However, performing Viterbi decoding on a StrongARM using C is energy-intensive. We have measured the energy per useful bit required to decode 1/2 and 1/3-rate convolutional codes with varying constraint length on the StrongARM. The results are shown in Fig. 10. Two observations can be derived from these graphs. First, the energy consumption scales exponentially with the constraint length. This is expected since the number of states in the trellis increases exponentially with constraint length. Another observation to make is that the energy consumption seems to be independent of the coding rate. This is reasonable since the rate only affects the number of bits sent over the transmission. A lower rate code does not necessarily increase the computational energy since the number of states in the Viterbi decoder is unaffected. In addition, the cost of reading the data from memory is dominated by updating the survivor path registers in the Viterbi algorithm. The size of the registers is proportional to the constraint length and is not determined by the rate. Therefore, given two convolutional codes $\mathcal{C}_1$ and $\mathcal{C}_2$ both with constraint lengths K, where $R_{\mathcal{C}_1} < R_{\mathcal{C}_2}$, the per bit energy to decode $\mathcal{C}_1$ and $\mathcal{C}_2$ is the same even though more bits are transmitted when using $\mathcal{C}_1$.

Given the data in Fig. 10, we can now determine which convolutional code to use to minimize the energy consumed by communication for a given probability of error. In Fig. 11, the total energy per information bit E_b is plotted against P_b. Figure 11 shows that the energy per bit using no coding is lower than that for coding for $P_b > 10^{-5}$. The reason for this result is that the energy of computation, i.e. decoding, dominates the energy used by the radio for high probabilities of error. For example, assuming the model described in (7) and $P_{out} = 0$ dBm, the communication energy to transmit and receive per useful bit for an $R_c = 1/2$ code is 85 nJ/bit. On the other hand, the energy to decode an

(a) Rate 1/2

(b) Rate 1/3

Figure 10. (a) Decoding energy per useful bit for $R_c = 1/2$ codes with $K = 3$ to 9 and (b) decoding energy per useful bit for $R_c = 1/3$ codes with $K = 3$ to 9.

$R_c = 1/2$, $K = 3$ code on the SA-1100 is measured to be 2200 nJ per bit.

At lower probabilities of error, the power amplifier energy begins to dominate. At these ranges, codes with greater redundancy have better performance. These results imply that coding the data is not always the right thing to do if energy-efficiency is a criterion. One may suspect that this result is due to the inefficiency of the StrongARM in performing error correction coding. However, we will show that this result will hold even if a more efficient implementation of the Viterbi decoder is used. (Note that the x-axis of the graph extends below
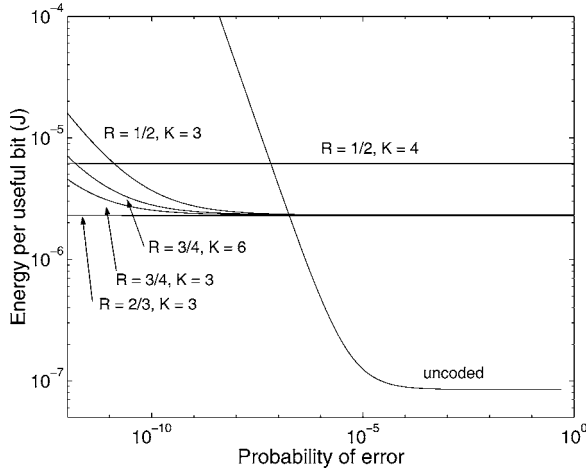
Figure 11. Energy per useful bit plotted versus P_b of an uncoded signal and a few convolutional codes with varying rates and constraint lengths ($P_{loss} = 70$ dB, $N_{rx} = 10$ dB, $R = 1$ Mbps). The number of information bits is 10000.

Figure 12. Measured decoding energy per useful bit for $R_c = 1/2$ codes with $K = 3$ to 7 using our synthesized VLSI implementation.

Figure 13. The energy per useful bit plotted against P_b using no coding and various convolutional codes ($P_{loss} = 70$ dB, $N_{rx} = 10$ dB, $R = 1$ Mbps). The number of information bits is 10000.

10^{-9}. At such low P_b, these results are likely invalid. They are shown so that the different codes can be distinguished.)

Since the use of the StrongARM to perform Viterbi decoding is energy inefficient, a dedicated integrated circuit solution to perform decoding is preferred. To explore the power characteristics of dedicated Viterbi decoders, we implemented 1/2-rate decoders with different constraint lengths and synthesized them using 0.18 μm CMOS TSMC ASIC technology. Our designs are fully parallel implementations of the Viterbi algorithm where a separate add-compare-select (ACS) unit is used for each state. Using Synopsys Power Compiler, we estimated the energy per bit used by our designs during the decoding of 20000 bits. Figure 12 shows the energy per bit for various constraint lengths. Using our implementation, in addition to our radio model, we determined the minimum energy code to use for a given probability of error. In Fig. 13, the energy per useful bit is plotted against P_b. From the graph, one can see that the communication/computation scheme to use will be dependent on the probability of error desired at the receiver. For $P_b > 10^{-4}$, no coding should be used. This is due to the fact that the transceiver power ($P_{tx/rx}$) is dominant at high probabilities of error. Since coding the data will increase the on-time (T_{on}) of the transceiver, using coding will increase the overall energy per useful bit. At high P_b, the required transmit power is not very high and the energy to perform decoding is small. Thus, the transceiver energy is dominant. Since coding
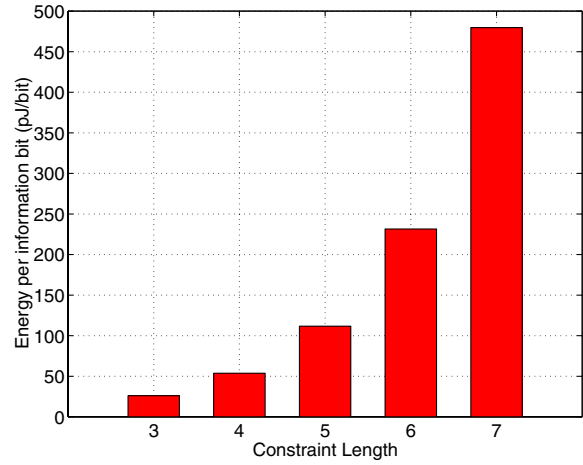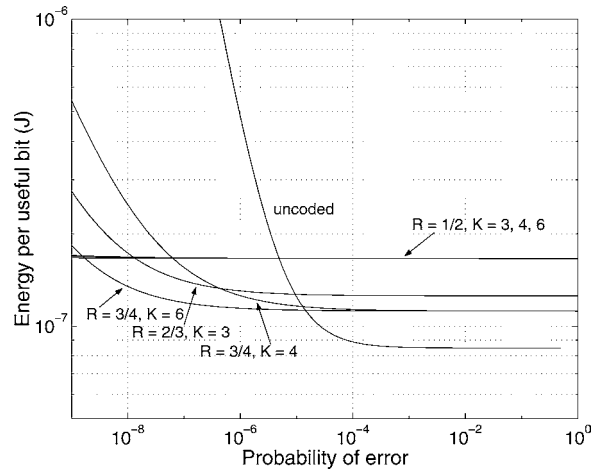
increases the number of bits to transmit, the energy with coding is greater. Note that once $P_b < 10^{-5}$, the overall communication energy with coding is smaller since the energy of the power amplifier (P_{out}) will begin to dominate. Figure 13 reinforces the idea that coding the data may not be the best solution if energy-efficiency is a criteria. Indeed, the coding strategy is highly dependent on the desired output quality of the user.

5.3. *Variable vs. Fixed Error Correction Strategies*

Given the information in Fig. 13, the strategy which gives the least energy dissipation is the one that uses

Table 2. Coding strategy to use for different ranges of P_b (I).

Range of P_b	Strategy
$P_b > 10^{-5}$	Uncoded
$10^{-9} < P_b < 10^{-5}$	$R_c = 3/4, K = 6$
$P_b < 10^{-9}$	$R_c = 1/2, K = 3$

Table 3. Coding strategy to use for different ranges of P_b (II).

Range of P_b	Strategy
$P_b > 10^{-5}$	Uncoded
$10^{-6} < P_b < 10^{-5}$	$R_c = 2/3, K = 3$
$10^{-8} < P_b < 10^{-6}$	$R_c = 3/4, K = 3$
$P_b < 10^{-9}$	$R_c = 1/2, K = 3$

Table 4. Energy expended during transmission of $N = 500$ packets of $L = 10000$ information bits for four different error correction strategies.

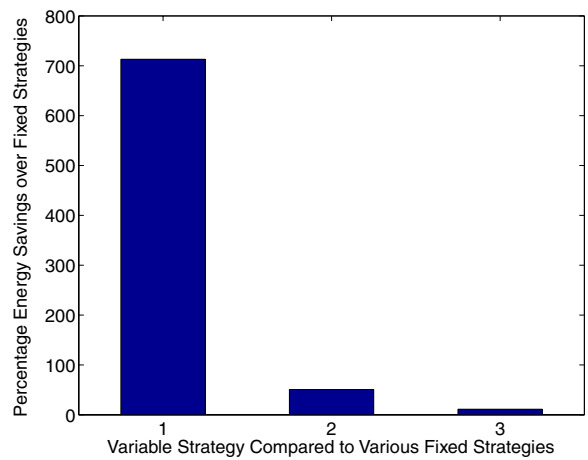| Strategy	Energy (J)
Variable	0.55
No coding	4.5
Fixed $R = 1/2, K = 3$	0.83
Fixed $R = 3/4, K = 3$	0.61

Figure 14. Energy Savings of Variable Strategy versus Various Fixed Strategies. The first bar represents the energy saved by the variable scheme over the fixed, no coding scheme. The second bar represents the energy saved by the variable scheme over the $R = 1/2, K = 3$ scheme, while the last bar represents the energy saved by the variable scheme over the $R = 3/4, K = 3$ scheme. The energy savings are 713%, 51%, and 11% respectively.

different codes depending on the desired probability of error. Table 2 shows the strategy to use for different ranges of P_b given that the codes available have a base rate of $R_c = 1/2$. This strategy, while providing the minimal energy per useful bit for the range of probabilities of error under consideration, has the disadvantage that codes of varying constraint length are required. As a result, various Viterbi implementations would be required. If area is a constraint, a single rate-adaptive Viterbi decoder with fixed constraint length could be used. Table 3 shows the strategy to use for different range of P_b given that the codes available are based on a $R_c = 1/2, K = 3$ convolutional code.

While it is clear that the variable strategy will produce the minimal energy per useful bit given a desired probability of error, the overall energy savings during communication is not clear. In other words, what is the total energy savings obtained by using this strategy over a fixed strategy? Figure 14 shows the percentage energy savings of the variable ECC strategy versus three fixed strategies. We simulated the transmission of $N = 500$ packets, each of $L = 10000$ information bits long using the four different strategies. The first strategy uses the variable coding strategy proposed in Table 3. The second strategy uses no coding. The third strategy employs a fixed 1/2-rate, $K = 3$ code, while the fourth strategy uses a fixed 3/4-rate, $K = 3$ code. In the simulation, we assume that the user changes the desired probability of error during the transmission of the $N = 400$ packets. Every one hundred packets, the probability of bit error requirements changes. For the purposes of the simulation, we assume that the user

requires bit error rates of $10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$ and 10^{-7}. Table 4 shows the total energy expended by each of the four schemes during transmission of five hundred packets of $L = 10000$ information bits.

6. Low Power MAC Protocols

In Sections 4 and 5, the energy-efficient techniques focus on how to decrease the energy consumption of the communication on a per node basis. In this section, we take a more global view and examine how to design an energy-efficient media-access control (MAC) protocol for a coordinated network. Such a network is used in machine-monitoring applications as described in Section 1.1. Recall that the main characteristic of these networks is that they are asymmetric, *coordinated*

Table 5. Machine monitoring specification.

Cell density	<300 in 5 m × 5 m
	<3000 in 100 m × 100 m
Range of link	<10 m
Message rate ($L = 2$ bytes)	Average: 20 msgs/s
	Maximum: 100 msgs/s
	Minimum: 2 msgs/s
Error rate and latency	10^{-6} after 5 ms
	10^{-9} after 10 ms
	10^{-12} after 15 ms
System lifetime	5 years
Frequency band	2.400–2.4835 GHz (ISM)

sensor networks where clusters or cells are formed around high-powered basestations. Table 5 summarizes the application specifications.

6.1.   Determining N_{rx} and N_{tx}

In (1), recall that N_{rx} and N_{tx} are factors that relate to the underlying MAC protocol and intended application. In this application, these parameters depend largely on the latency requirement specified by the user. Given these requirements, we derive the best parameters to use for a low power MAC protocol for a single cell where a high-powered basestation gathers data from the sensors. We assume that the power consumed by the baseband blocks during communication is negligible. Moreover, we only consider low power transmitters where the power consumption of the transmitter is dominated by the frequency synthesizer and is not affected by the data rate [12, 15].

First, we examine a few candidate MAC protocols. For the purposes of this discussion, we limit our choice of MAC protocols to time division multiple access (TDMA) and frequency division multiple access (FDMA) schemes. For this application, other more complex multi-access schemes may not be appropriate. In particular, on-demand schemes (e.g. CSMA-CA) that require handshaking may increase the latency of the data, and detrimentally affect energy.

In a TDMA scheme, in each slot, the full bandwidth of the channel is dedicated to a single sensor node for communication purposes. Thus, the signal bandwidth per node is equal to the available bandwidth and sensors can transmit at the highest data rate. Since the transmit on-time (T_{on-tx}) of the radio model described in (1), is inversely proportional to the signal bandwidth, the
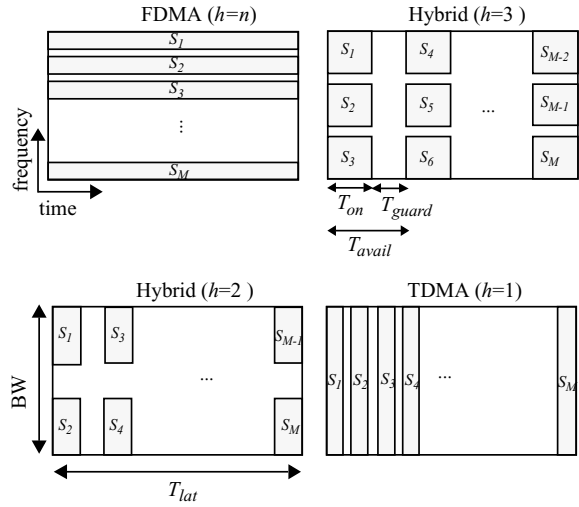

Figure 15. Multiple access methods.

T_{on-tx} is minimized in TDMA schemes. On the other hand, in an FDMA scheme, the signal bandwidth (total available bandwidth divided by number of nodes) is minimal. Thus, T_{on-tx} is at its maximum. A hybrid scheme involving both TDMA and FDMA (TDM-FDM) is also possible. In a TDM-FDM scheme, both time and frequency are divided into available transmission slots. Figure 15 illustrates each of the different multiple-access schemes considered, where a shaded area indicates a valid transmission slot for sensor S_i.

In the schemes where TDM is employed, note that a downlink signal from the basestation to the sensors is required to maintain time synchronization among the nodes in the network. Due to the finite error among each sensor's reference clock, the basestation must send out *synchronization* packets (SYNCs) to avoid collisions among transmitted packets. Hence, the receiver circuitry of each sensor must be activated periodically to receive the SYNC signals. As explained in Section 3.1, the receiver uses more power than the transmitter. Thus, we need to reduce the average number of times the receiver is active. The number of times the receiver needs to be active (N_{rx}) will depend on T_{guard}, the minimum time difference between two time slots in the same frequency band, as shown in Fig. 15. During T_{guard}, no sensor is scheduled to transmit any data. Thus, a larger guard time will reduce the probability of packet collisions and thus, reduce the frequency of SYNC signals and N_{rx}.

If two slots in the same frequency band are separated by T_{guard}, it will take T_{guard}/δ seconds for these two packets to collide, where δ is the percent difference

between the two sensors' clocks. Hence the sensors must be resynchronized at least δ/T_{guard} number of times every second. In other words, the average number of times the receiver is active per second, $N_{rx} = \delta/T_{guard}$. Assuming that the total slot time available $T_{avail} = T_{on} + T_{guard}$, we can derive a formula relating N_{rx} to T_{lat}, the latency requirement of the transmitted packet, as follows,

$$N_{rx} = \frac{\delta}{T_{guard}} = \frac{\delta}{T_{avail} - T_{on}}$$
$$= \frac{\delta}{\left(\frac{T_{lat}}{M} - \frac{L}{W}\right)h} \tag{8}$$

where W is the available bandwidth, L is the length of the data packet in bits, T_{lat} is the latency requirement of the transmitted packet, h is the number of channels in the given band W, and M is the number of sensors. Here we have assumed that the data rate R equals the signal bandwidth. Hence, $T_{on} = \frac{L}{R} = \frac{L}{W/h}$. From (8), we see that as the number of channels decreases, the guard time becomes larger and N_{rx} is reduced. It is also apparent that the advantage of ideal FDMA is that a receiver at the sensor is not needed (i.e., as $T_{guard} \rightarrow \infty$, $N_{rx} \rightarrow 0$).

From (1) and (8), we can determine an analytical formula to find h_{opt}, the number of channels which gives the lowest power consumption:

$$h_{opt} = \sqrt{\frac{\delta P_{rx}(T_{on-rx} + T_{st})}{\left(\frac{T_{lat}}{N_{cell}} - \frac{L}{W}\right)N_{tx}(P_{tx} + P_{out})\frac{L}{W}}}$$
$$\propto \sqrt{\frac{P_{rx}}{N_{tx}P_{tx}}}. \tag{9}$$

Clearly, we see that h_{opt} is determined by the ratio of the power consumption of the transmitter to the receiver. As expected, if the receiver consumes less power, a TDMA scheme is favored. On the other hand, if the receiver uses more power, FDMA is more appropriate.

An example of the previous analysis is performed in a scenario where a sensor node on average sends twenty 100-bit packets/s ($N_{tx} = 20$ times/s, $L = 100$ bits) and the latency requirement is 5 ms ($T_{lat} = 5$ ms). Also, we assume that $W = 10$ MHz and the number of sensors in a cell is $M = 300$. The resulting average power consumption is plotted in Figs. 16 and 17 where the horizontal axis represents the number of channels available ($h = 1$: TDMA, $h = 300$: FDMA) and the vertical axis is the average power consumption.
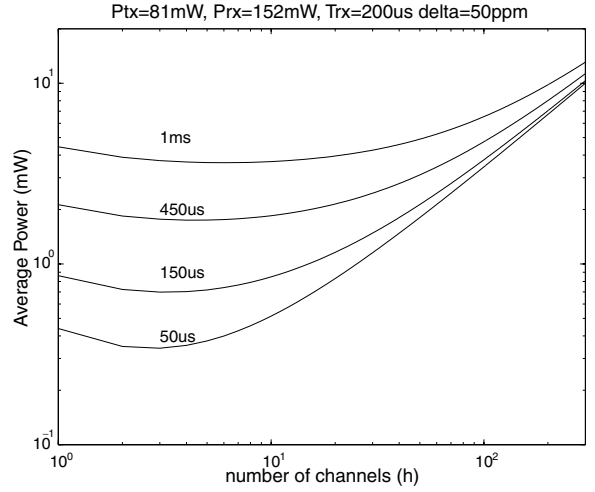

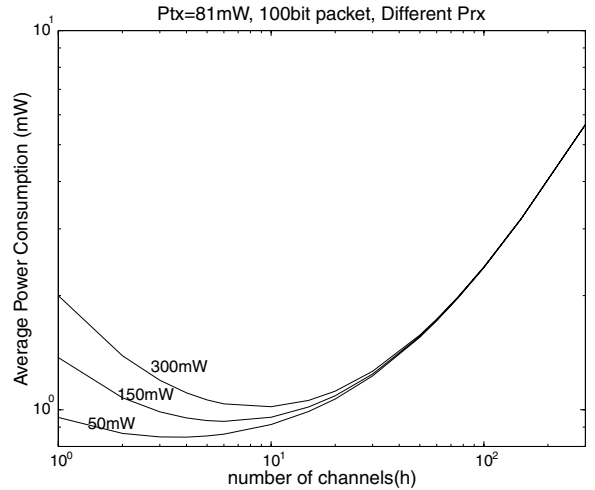*Figure 16.* Energy with different T_{st}.

Figure 17. Energy with different E_{rx}/E_{tx}.

In Fig. 16, the average power consumption is plotted for various startup times ($T_{st} = 50$ μs to 1 ms). We can see that the average power reaches a minimum value when a hybrid TDM-FDM scheme is used. The variation in power consumption for different h gets smaller as the T_{st} is increased since the overall power consumption is dominated by the startup time. In Fig. 17, we can see how the power consumption curve will vary if different radio receivers are used. That is, we vary P_{rx} while maintaining a constant P_{tx}. We see from this figure that h_{opt} increases as receiver power increases. Notice that despite the fact that a TDMA scheme will have the minimum transmit on-time, the TDMA scheme does not achieve the lowest

power. Indeed, as the number of channels is reduced, the guard time decreases. This implies that more synchronization is needed and thus, the receiver power starts consuming a large portion of the total power.

7. Related Work

The rise in interest in low power sensing devices is a direct result of recent advances in integrated circuit and MEMS technology. Many research groups are exploring the issues related to the design of nodes for deployment in wireless sensor networks. The WINS [20] and PicoRadio [5] projects are seeking to integrate sensing, signal processing, and radio elements onto a single integrated circuit. Meanwhile, researchers involved in SmartDust [3] aim to design particle-sized nodes for wide-area distributed sensing.

The growing demand in wireless devices has also spawned research into the design of low-power integrated radio transceivers. Many designs try to improve the performance of transceivers using both architectural and circuit techniques. In [15], low power high data rate modulation is achieved by applying a pre-emphasis filter before the modulated data to overcome the frequency synthesizer's narrow bandwidth. Furthermore, the addition of calibration circuitry in [21] enables higher level modulation with low cost.

Network protocols for wireless microsensor networks, such as *directed diffusion* [22] and *LEACH* [23], are also beginning to emerge. In directed diffusion, routes are dynamically formed as data is sensed. Initially, routes called gradients that link *sources* of interesting data to *sinks* are formed. Through data aggregation techniques, caching, and reinforcement messages, the appropriate link is dynamically selected from the candidates. Links are created only when data of interest is sensed. Theoretically, less energy will be consumed. LEACH is a protocol that uses hierarchy to reduce the data collected by the sensors before sending it on to a central base station. Reducing the data that needs to be sent helps make LEACH more energy efficient.

While research into energy-efficient protocols for sensor networks is relatively new, many energy-efficient network protocols for other wireless networks have been presented. In [24–26], techniques and metrics to evaluate and design energy-efficient routing and MAC protocols for wireless networks are presented. Detailed energy-efficient strategies for traditional wireless networks are proposed in [27, 28]. In [27], an energy-saving TDMA scheduling algorithm, that allows the network size to grow, is proposed. In [28], an energy-efficient hybrid CDMA/TDMA scheme is suggested. In this scheme, the network traffic is scheduled based not only on traffic priority but also on the battery status at each node. Both schemes do not base their model on measurements of a real radio.

Energy-efficient protocols that adapt transmit output power and/or error correction control parameters are explored by a number of researchers [29–32]. In [29], the authors use an adaptive radio designed for wireless multimedia communications over ATM as a model. In that paper, frame length and forward-error correction parameters are adapted to lower energy consumption of the radio and improve throughput as conditions of the channel change. A similar study is performed by [31] in the context of a cellular-style network, but the output transmit power is also considered. In [32], an energy-efficient protocol that adjusts both RF transmit power and error control strategy is examined for 802.11 wireless LANs. The authors of [30] offer an in-depth study of the error process and then introduce a probing ARQ-type scheme that is designed for energy-constrained devices.

8. Conclusion

The emergence of wireless microsensors offer commercial and military users the opportunity to deploy and implement many exciting applications. However, central to the success of these applications is the design of ultra low-power sensor nodes and ultra low-power networks. The energy usage by all parts of the system—from the network to the sensors themselves—must be minimized. In this paper, design techniques to reduce the energy consumption of communication was discussed. Using a measurement-based model of our radio, we introduced low-power techniques and methods that trade off computation and communication parameters in order to reduce energy consumption.

In general, we showed that the underlying characteristics of the radio such as the startup time, transceiver power, and output power should dictate the modulation, MAC, and error-control strategy to use if energy is a concern. Specifically, in choosing a modulation scheme, using complex hardware will increase the energy consumed by the radio. Front-end radio architectures with long startup times will also increase energy consumption as it may dominate the actual data transmission time. In choosing a MAC scheme, the amount

of multiplexing in the time and frequency domain to use should take into account the power consumed by the receiver, transmitter, and synthesizer. For channel coding, given a varying bit error rate, one can best utilize the available energy on the node by adapting the error-control coding to changing conditions. Indeed, we showed that energy savings of 50% is attainable by using an adaptable scheme over a fixed scheme.

While the techniques described here focus on the design of a radio for wireless sensor networks, the design methodology presented could be used in the context of other applications. Through the use of power-aware, system-level techniques, one should be able to design an energy-efficient system for use in any other time-varying environment or application.

Acknowledgments

References

1. D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next Century Challenges: Scalable Coordination in Sensor Networks," in *Proceedings of the Fifth Annual International Conference on Mobile Computing and Networking (MobiCom '99)*, Aug. 1999, pp. 263–270.

2. K. Bult et al., "Low Power Systems for Wireless Microsensors," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '96)*, Aug. 1996, pp. 17–21.

3. J. Kahn, R. Katz, and K. Pister, "Next Century Challenges: Mobile Networking for Smart Dust," in *Proceedings of the Fifth Annual International Conference on Mobile Computing and Networking (MobiCom '99)*, Aug. 1999, pp. 271–278.

4. N. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," in *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCom '00)*, Aug. 2000, pp. 32–43.

5. J. Rabaey et al., "PicoRadio Supports Ad Hoc-Ultra-Low Power Wireless Networking," in *Computer*, July 2000, pp. 42–48.

6. L. Nord and J. Haartsen, *The Bluetooth Radio Specification and The Bluetooth Baseband Specification*. Bluetooth, 1999–2000.

7. A.P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-Power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 4, 1992, pp. 473–484.

8. J. Kao, "Subthreshold Leakage Control Techniques for Low Power Digital Circuits," Ph.D. Thesis, Massachusetts Institute of Technology, 2001.

9. M. Bhardwaj, R. Min, and A. Chandrakasan, "Power-Aware Systems," in *Proceedings of the 34th Asilomar Conference on Signals, Systems, and Computers*, Nov. 2000.

10. M. Bhardwaj, R. Min, and A. Chandrakasan, "Quantifying and Enhancing Power-Awareness in VLSI Systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 6, 2001, pp. 757–772.

11. *LMX3162 Evaluation Notes and Datasheet*, April 1999.

12. S. Cho and A. Chandrakasan, "A 6.5 GHz CMOS FSK Modulator for Wireless Sensor Applications," in *IEEE Symposium on VLSI Circuits Digest of Technical Papers*, 2002, pp. 182–185,

13. A. Klaiber, *The Technology Behind Crusoe Processors*. Transmeta Corporation. http://www.transmeta.com, Jan. 2000.

14. Intel XScale Microarchitecture. http://developer.intel.com/design/intelxscale/, 2000–2001.

15. M. Perrott, T. Tewksbury, and C. Sodini, "27 mW CMOS Fractional-N Synthesizer/Modulator IC," in *ISSCC Digest of Technical Papers*, Feb. 1997, pp. 366–367.

16. N. Filiol, T. Riley, C. Plett, and M. Copeland, "An Agile ISM Band Frequency Synthesizer with built-in GMSK Data Modulation," in *IEEE Journal of Solid-State Circuits*, vol. 33, July 1998, pp. 998–1008.

17. J. Proakis, *Digital Communications*, 4th edn. McGraw-Hill, New York City, New York, 2000.

18. D. Akerberg, "Properties of a TDMA Pico Cellular Office Communication System," in *IEEE Proceedings of GLOBECOM*, 1998, pp. 1343–1349.

19. G. Janssen and R. Prasad, "Propagation Measurements in an Indoor Radio Environment at 2.4 GHz, 4.75 GHz and 11.5 GHz," in *Proceedings of the VTS Conference on Frontiers of Technology*, 1992, pp. 617–620.

20. G. Asada et al., "Wireless Integrated Network Sensors: Low Power Systems on a Chip," in *Proc. ESSCIRC '98*, 1998.

21. D. McMahill, "Automatic Calibration of Modulated Fractional-N Frequency Synthesizers," Ph.D. Thesis, Massachusetts Institute of Technology, 2001.

22. C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCom '00)*, Aug. 2000, pp. 56–67.

23. W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proc. HICSS 2000*, Jan. 2000.

24. J.-H. Chang and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks," in *Proceedings of IEEE INFOCOM (INFOCOM '00)*, March 2000, pp. 22–31.

25. S. Singh, M. Woo, and C. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," in *Proceedings of the Fourth Annual International Conference on Mobile Computing and Networking (MobiCom '98)*, Oct. 1998, pp. 181–190.

26. V. Rodoplu and T.H. Meng, "Minimum Energy Mobile Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, 1999, pp. 1333–1344.

27. J.-H. Ju and V. Li, "TDMA Scheduling Design of Multihop Packet Radio Networks Based on Latin Squares," in *Proceedings of IEEE INFOCOM (INFOCOM '99)*, 1999, pp. 187–193.
28. S. Kishore, J.-C. Chen, K. Sivalingam, and P. Agrawal, "A Battery Power Level Aware MAC Protocol for CDMA Wireless Networks," in *IEEE Internatinal Conference on Universal Personal Communications*, 1998, pp. 967–971.
29. P. Lettieri and M.B. Srivastava, "Adaptive Frame Length Control for Improving Wireless Link Throughput, Range, and Energy Efficiency," in *Proceedings of IEEE INFOCOM (INFOCOM '98)*, March 1998, pp. 564–571.
30. M. Zorzi and R. Rao, "Error Control and Energy Consumption in Communications for Nomadic Computing," *IEEE Transactions on Computers*, March 1997.
31. B. Narendran et al., "Evaluation of an Adaptive Power and Error Control Algorithm for Wireless Systems," in *IEEE ICC '97*, 1997.
32. J.-P. Ebert and A. Wolisz, "Combined Tuning of RF Power and Medium Access Control for WLANs," in *Proceedings of the Sixth International Workshop on Mobile Multimedia and Communications (MoMUC '99)*, 1999.

Eugene Shih received the B.S. degree in Computer Engineering from the University of Washington in 1998 and an S.M. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology in 2001. Currently, he is a Ph.D. candidate in Computer Science at the MIT. Shih's research interests include wireless sensor networks, ubiquitous computing, and medical sensor systems.
eishih@mit.edu

SeongHwan Cho received the B.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST) in 1995 and the S.M. and Ph.D. degrees from the Department of Electrical Engineering and Computer Science from the Massachusetts Institute of Technology in 1997 and 2002, respectively. He joined Engim Inc. in 2002, where he was involved in data converters, PLL and VCO design. Since 2003, he has been with the military of South Korea working as a public service agent. His research interests include low power circuits and systems for wireless sensors.
chosta@mtl.mit.edu

Fred S. Lee received the B.S.E.E. and M. Eng. degrees from the Massachusetts Institute of Technology, Cambridge, MA in June, 2002, where he is currently working toward the Ph.D. degree in electrical engineering. His research interests include ultra-wideband radio transceiver architectures, feedback circuits, low power mixed-signal circuits, and analog RF circuit design.
fslee@mtl.mit.edu

Benton H. Calhoun received the B.S. degree in electrical engineering with a concentration in computer science from the University of Virginia, Charlottesville, Virginia, in 2000. He received the M.S. degree from the Massachusetts Institute of Technology, Cambridge, Massachusetts, in 2002. He is currently pursuing a Ph.D. at the Massachusetts Institute of Technology. His research interests include leakage reduction, sensor networks, and energy-efficient circuits.
bcalhoun@mtl.mit.edu

Anantha P. Chandrakasan received the B.S., M.S. and Ph.D. degrees in Electrical Engineering and Computer Sciences from

the University of California, Berkeley, in 1989, 1990, and 1994 respectively. Since September 1994, he has been with the Massachusetts Institute of Technology, Cambridge, where he is currently a Professor of Electrical Engineering and Computer Science.

He has received several awards including the 1993 IEEE Communications Society's Best Tutorial Paper Award, the IEEE Electron Devices Society's 1997 Paul Rappaport Award for the Best Paper in an EDS publication during 1997, and the 1999 Design Automation Conference Design Contest Award.

His research interests include low-power digital integrated circuit design, wireless microsensors, ultra-wideband radios, and emerging technologies. He is a co-author of *Low Power Digital CMOS Design* (Kluwer Academic Publishers, 1995) and *Digital Integrated Circuits* (Prentice-Hall, 2003, 2nd edition). He is also a co-editor of *Low Power CMOS Design* (IEEE Press, 1998) and *Design of High-Performance Microprocessor Circuits* (IEEE Press, 2000).

He has served as a technical program co-chair for the 1997 International Symposium on Low-power Electronics and Design (ISLPED), VLSI Design '98, and the 1998 IEEE Workshop on Signal Processing Systems. He was the Signal Processing Sub-committee Chair for ISSCC 1999–2001, the Program Vice-Chair for ISSCC 2002, the Program Chair for ISSCC 2003. He was an Associate Editor for the IEEE Journal of Solid-State Circuits from 1998 to 2001. He serves on the SSCS AdCom. He is the Technology Directions Chair for ISSCC 2004.

anantha@mtl.mit.edu