Design Considerations for Distributed Microsensor Systems

Anantha Chandrakasan, Rajeevan Amirtharajah, SeongHwan Cho, James Goodman,
Gangadhar Konduri, Joanna Kulik, Wendi Rabiner, Alice Wang
Department of EECS,
Massachusetts Institute of Technology, Cambridge

ABSTRACT

Wireless distributed microsensor systems will enable the reliable monitoring and control of a variety of applications that range from medical and home security to machine diagnosis, chemical/biological detection and other military applications. The sensors have to be designed in a highly integrated fashion, optimizing across all levels of system abstraction, with the goal of minimizing energy dissipation. This paper addresses some of the key design considerations for future microsensor systems including the network protocols required for collaborative sensing and information distribution, system partitioning considering computation and communication costs, low energy electronics, power system design and energy harvesting techniques.

1. Introduction

Over the last few years, the design of micropower wireless sensor systems has gained increasing importance for a variety of civil and military applications. The Low Power Wireless Integrated Microsensors (LWIM) project has made major advances in the design of ultra low power sensor systems based on infrared, vibration and acoustic sensors [1]. They combine micropower sensor technology with a low power sensor interface, signal processing and weak-inversion RF circuits to implement entire sensor systems. An excellent overview of system applications and sensor electronics is presented in [2]. Based on these results, we will assume that low-power sensor technology and the associated interfaces are available. This paper will focus on the system design issues and circuit challenges for distributed microsensors.

There are several important differences between the wireless microsensor systems discussed here and wireless macrosensors or devices used for multimedia processing. The transmission distance of microsensors can be significantly shorter (<10m) than conventional macrosensors and handheld devices. The transmission power is hence lower, resulting in significantly different architectures for computation partitioning.

While some applications such as image sensors demand a high transmission data rate, most sensing applications will require very low data rates compared to conventional multimedia traffic and they operate in burst mode. As a result, there are several new challenges faced at the circuit level. For example, the design of a very low duty cycle radio is essential for transmitting small packets (e.g., a temperature sensor may transmit 100's of bits every few minutes). Existing radio architectures are not suitable for these very low data rates since they have significant energy overhead in powering on and off.

Collaboration is essential between sensor nodes for the fusion of data, and the organization of the nodes can be ad-hoc. While efficient protocols have been proposed to deal with ad-hoc networks, energy has not been explicitly used as a metric. The topology of sensor networks can be dynamic, and this must be considered in developing an efficient sensor architecture.

Energy scalability is also an important design consideration in these distributed sensors. The amount of resources available (e.g., battery life), the quality requirements (e.g., accuracy of sensing results), and the latency requirements can vary as a function of time. This has to be explicitly considered in the optimization of the system. For example, system-level power down can be exploited to scale quality or latency with respect to energy dissipation. At the circuit level, techniques such as dynamic voltage scaling allow the energy dissipation of a processor to be scaled with computation latency or Quality of Service.

Finally, the availability of energy from a battery or a large power source is not always a good assumption. For example, the monitoring devices can be embedded into objects where replacing the battery is very inconvenient or impossible. Energy harvesting (i.e., using ambient energy to power electronic circuits) is therefore an essential component of many wireless sensors.

2. System Partitioning

In many wireless sensor systems, it is possible to reduce power dissipation of the battery operated sensor by off-loading computation to remote base station servers that do not have energy constraints. For example, consider an image sensor that performs data compression before transmission to a high powered base station. Most video compression algorithms use some form of block-based scene motion estimation/compensation to remove the temporal correlation inherent in natural video sequences. Unfortunately these algorithms tend to be very compute intensive, resulting in significant battery drain on the sensor node.

An important observation, that can save significant power in the image sensor node, is that the motion of objects is continuous from one frame to the next in natural sequences. Thus, knowing the location of an object in a few previous frames, it is possible to predict its location in the current frame. It therefore is possible to remove the motion estimation computation from the portable encoder and perform it at the receiving base station [3]. Since the base-station only has access to the previous reconstructed frames, it must perform motion estimation on these previous frames and predict the motion vectors of the current frame from these motion vectors of the previous reconstructed frames. These predicted motion vectors are transmitted through a low-bandwidth reverse wireless link to the encoder, where the remainder of the compression algorithm is performed. This approach achieves nearly the same compression rates as encoder-based motion estimation while using over *two orders of magnitude* less computation at the encoder. In such system partitioning trade-offs, both computation and communication costs must be taken into account.

3. Energy Efficient Protocols for Networked Sensors

As sensor technologies have advanced and wireless communication has become a reality, the focus has turned from single macrosensors communicating with basestations to the problem of creating wireless networks of communicating sensors. Such sensor networks aggregate complex data to provide rich, multi-dimensional pictures of the environment that single sensors working alone cannot provide. Unlike a desktop, a battery-constrained sensor node does not have enough energy to support a full TCP/IP stack, perform complex routing algorithms, exchange data at high rates of speed, and constantly poll the system for data and routing updates. Thus, networking models that do not take limited energy into consideration are inadequate for a sensing network.

3.1 Network Routing Protocols

Recently, there have been a few papers that discuss the design of power conservative network protocols ([4]-[5]). In wireless multimedia LAN systems, all communication from an energy constrained node is through a central high-powered base-station [5]. Sensor networks typically will not have a central base-station to which all nodes can directly communicate, and thus a low-power routing protocol is needed to route the data from the sensors to the high-powered base-station. Such a “multi-hop” routing protocol is developed in [4], where the protocol is optimized for minimum power dissipation at the wireless nodes (where each node can be an information source, an information sink, and a router). Each node maintains a routing table where the cost of each path is the power dissipated in transmitting information along that path. In order to deal with the dynamic nature of the network topology, the routing tables must be updated periodically. Since each node keeps track of every other node in the system and the optimal path to that node, this multi-hop protocol enables a point-to-point communication system that is “strongly connected”. Thus every node can transmit data to the high-powered node using a minimum amount of total energy.

If the sensor network does not have access to a high-powered node, new network protocols need to be developed. Consider, for example, a network where

- All sensor nodes are identical and have the same limited energy capacity.
- The sensor with which the end-user communicates (called the user-dispatch node) varies (i.e., there is no fixed basestation).
- The end-user controls the “quality” of the output from the sensor network.
- Each sensor knows the location of its nearest neighbors with whom it can communicate.

The user-dispatch node is responsible for transmitting the aggregate sum of the sensor network data to the end-user. However, this node will change (so as not to drain the battery of a single sensor). In addition, this node will not be within communication range of all the sensors in the network. In this situation, all the sensor nodes in the network need to know the aggregate sensor data, since any sensor can become the user-dispatch at a given time. This can be accomplished by having each sensor transmit its data to all its immediate neighbors. Whenever a node receives data from one of its neighbors, it also transmits this data to the rest of its neighbors. Figure 1 shows an example of this protocol. Sensor A transmits data to Sensor B. Sensor B then combines this data with its own data and sends the aggregate data to all of its neighbors, except for Sensor A, to whom only the data for Sensor B is sent. By propagating data in this manner,

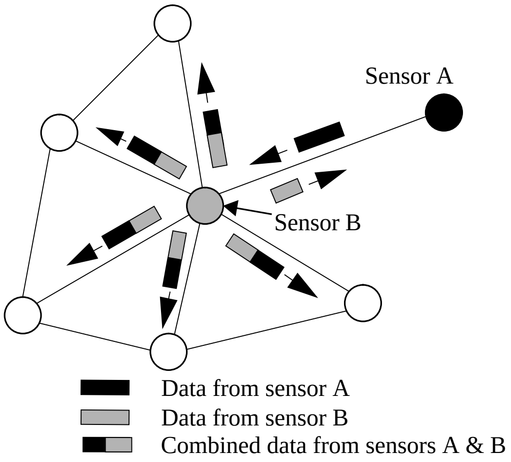

Figure 1. Example protocol for information sharing among distributed microsensors.

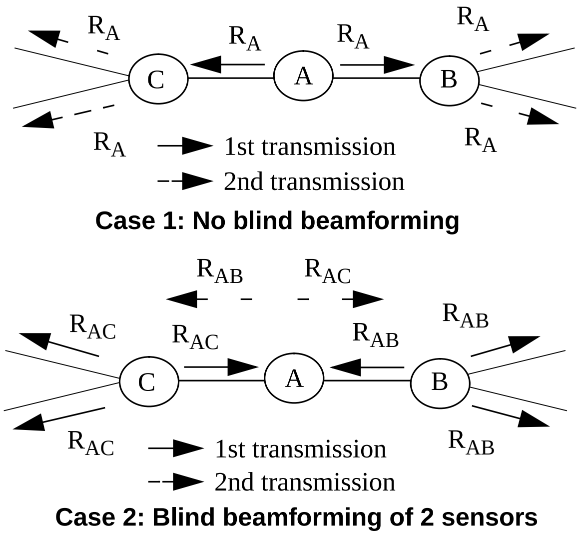

Figure 2. Data fusion architecture using beamforming.

after a certain time, every sensor in the network will have all the data of the sensor system. This system, which is a form of broadcasting, does not require routing and is extremely fault-tolerant. Since sensors can easily fail, this is an important aspect of any sensor network. This protocol also ensures that the data will reach every node with a minimum amount of latency.

Assume the end-user informs a user-dispatch sensor of the desired quality. This information is propagated through the network using the above protocol. After this is complete, most nodes in the sensor network go to sleep except for a few "monitor-site" sensors that are chosen randomly and reassigned periodically. These monitor sensors run simple algorithms to detect the presence of an "event". When an event is detected, each monitor-site sensor wakes up its neighbors, who then wake up their neighbors until at some bounded time, all the sensors in the system are awake.

At this point, the network must perform data fusion to aggregate the data according to the quality metric specified by the end-user. By optimally combining data from different sensors, the SNR is increased which leads to an improvement in detection and classification. Thus the end-user can specify the number of sensors to use in order to obtain a desired quality.

One method of combining data from multiple sensors is blind beamforming. In [6], a time-domain, filter-and-sum beamformer which combines the data from randomly placed sensors is proposed. The results of simulations of this blind beamforming algorithm show that at -20 dB SNR, when the data from two sensors are combined, there is a 3.9 dB improvement in normalized mean squared error (MSE). There is a 7.7 dB improvement when the data from four sensors are combined.

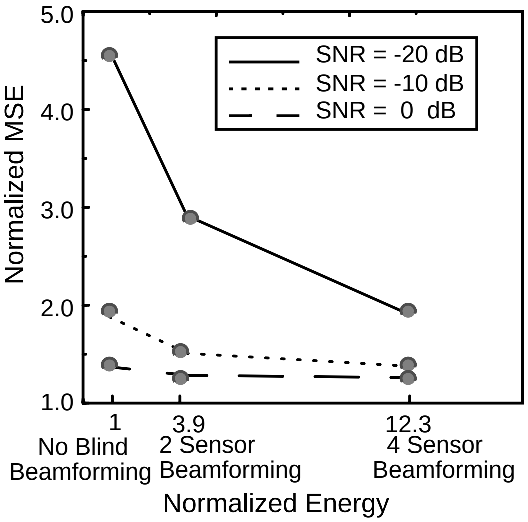

Figure 3. Energy evaluation of different beamforming configurations using a low-power StrongARM processor.

Using the example network in Figure 2, suppose that sensor A is a "monitor-site" sensor. The end-user can request any of the following scenarios from the sensors in the network:

(1) No blind beamforming (1 sensor)

(2) 2 Channel blind beamforming (2 sensors)

(3) 4 Channel blind beamforming (4 sensors)

If the end-user requests case (1), no blind beamforming is performed. When an "event" is detected by sensor A, the detection is performed locally at the sensor. Sensor A broadcasts its detection result (R_A) to its nearest neighbors (B and C), who broadcast this result to their neighbors, etc., until each sensor in the network has received R_A, as shown in Figure 2.

If the end-user requests case (2), two channel blind beamforming is performed using the data from two sensors. When an "event" is detected by A, the data is immediately transmitted to its nearest neighbors. Each of the nearest neighbors combine this data with their local data. For example, sensor A transmits its data to sensors B and C, and 2 channel blind beamforming is performed at sensor B using data from A and B and at sensor C using data from A and C. Detection is performed on the combined data, and the results (R_{AB} and R_{AC}) are then broadcast to the nearest neighbors of B and C until every sensor has received these results, as shown in Figure 2. Case (3), 4 channel blind beamforming, differs slightly from case (2) in that sensors will beamform their own data and that received from three neighboring sensors.

This example shows the trade-off between quality (measured here in terms of MSE) and energy dissipation, two important metrics in a sensor network. Figure 3 shows a plot of the normalized energy dissipated versus normalized MSE for different SNR's using the example above. The graph indicates that when the end-user requests an increase in detection performance (decrease in MSE), there is an increase in the total amount of energy needed to perform the blind beamforming algorithm. The graph also shows that as SNR decreases, increasing the number of cooperative nodes better preserves the signal characteristics.

4. Low Energy Electronics

Maximizing the battery lifetime of the wireless sensor nodes requires energy efficient design of sensor electronics to support low-energy computation (e.g., event detection and classification), information sharing among sensors, and wireless communications. Supply voltage reduction to 1V and below (through technology advances as well as circuit and architectural optimization) has clearly been the dominant technique to reduce energy in digital circuits [7]. Several dedicated signal processors have also exploited application specific properties to reduce switching events through careful selection of algorithm, architectures, and logic styles.

Clearly sensors systems need to leverage these techniques to minimize energy dissipation. However, there are several challenges in future sensor systems. First, it is desirable to design electronics that provide a knob to trade-off energy and quality/latency. Second, programmability is desirable to adapt algorithms on demand. Finally, the electronics must be optimized for very low duty cycles.

4.1 Energy Scalable Computing

Energy scalable computing exploits variations in the computation (e.g., precision, throughput, data statistics) to minimize the energy dissipation per input sample. Minimizing the energy dissipation per sample requires the development of architectures that can adapt to match the current requirements of the application by varying either switched capacitance per input sample, or the voltage at which the operations are performed.

To demonstrate energy scalable computing, consider an encryption processor where the level of security (i.e., quality) and energy consumed to encrypt a bit can be traded-off dynamically. The energy scalable encryption processor is based on a variable-width quadratic residue generator (QRG). The QRG is a cryptographically-secure pseudo-random bit generator that is based upon the work in [8]. The QRG operates by performing repeated modular squarings. The modular squaring is performed using an algorithm based on Takagi's iterated radix-4 algorithm [9] which requires $(\log_2 Q)/2$ iterations to compute the result $P = X \cdot Y \bmod Q$. The least significant $\log_2 \log_2 Q$ bits of each result can be extracted and used as a strong reproducible pseudo-random source for applications such as a stream cipher or key generator.

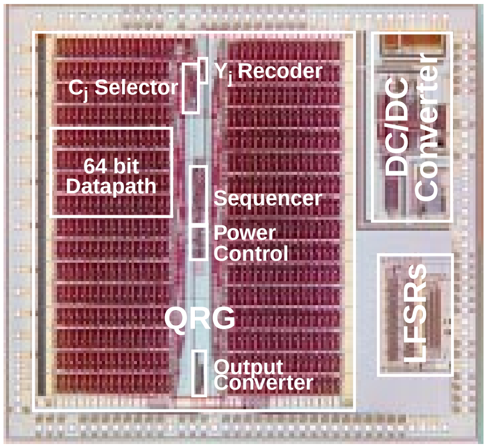

Figure 4. Encryption processor with embedded supply.

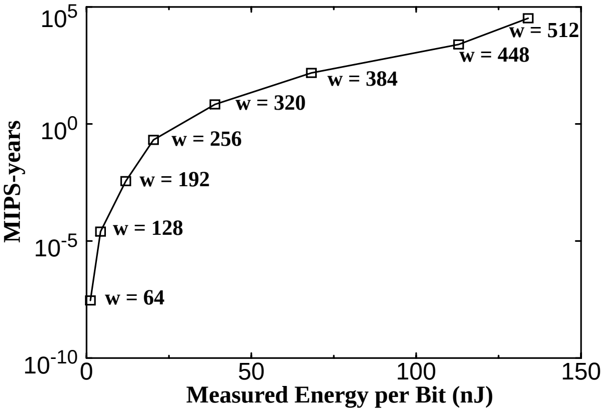

Figure 5. Security vs. Energy to encrypt a bit.

Energy scalable computing requires dynamically reconfigurable architectures that allow the energy consumption per input sample to be varied with respect to quality. In the case of the QRG the quality scales subexponentially with the modulus length, while the energy consumption scales polynomially. A fully scalable QRG architecture was developed where the width ($w = \log_2 Q$) can be reconfigured on the fly to range from 64 to 512 bits in 64 bit increments (Figure 4) [10]. The design makes extensive use of clock gating to disable unused portions of the QRG. Hence the switched capacitance of the QRG is minimized and energy scalability is achieved.

Further energy/security scalability can be achieved through the use of an adaptive supply [11]. Rather than designing a system with a static supply to meet a specific timing constraint under worst case conditions (i.e., establishing the feedback around the power converter to fix the output voltage), it is more energy efficient to allow the voltage to vary such that the timing constraints are just met at any given temperature and operating conditions; this is accomplished by establishing the feedback around a fixed processing rate or delay. In this example, when operating at a reduced width, the number of cycles required per multiplication is reduced and therefore the supply voltage can be reduced for a given throughput. The supply is varied using an embedded custom DC/DC converter. Figure 5 shows the energy /security scalability trade-off. This plot is obtained by varying both the bitwidth and supply voltage dynamically.

4.2 Low Energy Software

While the design of application specific circuits clearly results in the minimum energy solution, the flexibility provided by software is often highly desirable [12]. For software running on a processor, there are three ways in which the energy consumed by a program can be reduced: by reducing the number of operations performed, by reducing the switched capacitance of each operation, or by reducing the operating voltage. The number of operations performed to accomplish a particular task can be reduced by optimizing the code.

The notion of energy scalable computing using adaptive supply voltages can be exploited at the software level. The basic idea is to trade off the energy dissipation to run a program and the time required for completing the task. Low power processors (e.g., StrongARM) allow the clock frequency to be set using software control. If a longer latency is acceptable, the energy dissipation can be lowered by reducing both the frequency and the voltage dynamically. It is important to note that decreasing frequency alone does not reduce the energy consumption since voltage is fixed. In fact, energy increases as frequency is reduced at a fixed supply since the leakage energy increases as the execution time increases. Therefore, the operating voltage must be reduced to reduce energy consumption.

The impact of varying the voltage and frequency for the StrongARM-1100 is shown in Figure 6. The clock frequency of this processor can be changed by the programmer through an assembly instruction. The supply voltage can be changed using a DC-DC converter as the frequency of the clock is lowered. The measured data is shown for the QRG encryption algorithm described in the previous section. Code optimization and the technique of dynamic voltage scaling result in large energy savings. For comparison, the energy consumption of the unoptimized program to implement the QRG algorithm without voltage optimization is 432 μJ. With code optimization (i.e., through reduction of cycle count) the energy reduces to 75.6 μJ, resulting in savings of more than 80%. After dynamic voltage scaling (assuming higher latency requirements), the energy consumption was reduced to 21.25 μJ, a reduction of 95% when compared with the original code. Clearly, energy efficient programmable solutions will be important for future systems. Programmability can also be achieved through hardware reconfiguration [13]. Such approaches appear promising in providing flexibility with low energy requirements.

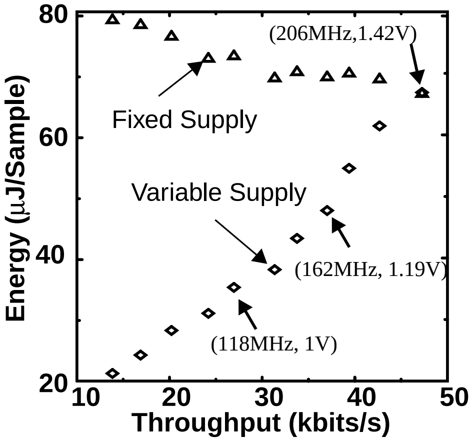

Figure 6. Measured energy dissipation per sample on the StrongARM microprocessor obtained by adaptive V_{dd}.

4.3 Low Duty Cycle Electronics

Sensor systems should be designed to deal with low duty cycles. This is true since the environmental conditions may not change rapidly (e.g., the temperature) or the sensors remain in a sleep state until some interesting event happens (detected by some simple front-end electronics). In order to maximize battery lifetime, the standby power of the computation and communication circuits must be minimized. In digital circuits, this is a major problem since low-voltage technology (i.e., low threshold devices) results in significant sub-threshold leakage. Several emerging technologies such as Multiple Threshold CMOS and substrate controlled variable threshold CMOS appear promising in controlling the leakage current during standby operation. In many sensor applications, low-threshold devices may not even be necessary since the throughput requirements are low. However, it is a major concern for high performance applications such as image sensors.

The communication module must also be designed for low duty cycle activity. For short range transmission (e.g., <10m) at GHz carrier frequencies, the power is dominated by the radio electronics (frequency synthesizer, mixers, etc.) and not the actual transmit power. Most sensors have very low data rates (<1kbs) and utilize short packet sizes (100’s of bits). In order to save power in the radio module, the electronics must be duty cycled (i.e., turned off during periods of inactivity). Unfortunately, frequency synthesizers require a significant overhead in terms of time and energy dissipation to go from the sleep state to the active state. This is not a problem for high data rate applications that utilize large packet sizes for each transmission burst. However, for short packet sizes, the transient energy during power on can be significantly higher than the energy required by the electronics during the actual transmission. Note that if latency was not an issue, the sensor data could be buffered and transmitted in a burst.

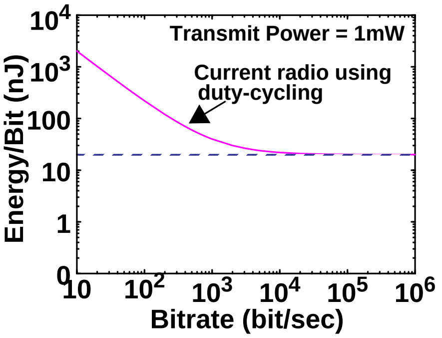

Figure 7. Energy per bit for given radio vs. bitrate.

Figure 7 shows the energy dissipation for transmitting a bit using a low-power radio capable of transmitting 1Mbs in continuous mode [14]. For lower bitrates than the peak, the radio is assumed to duty cycled. Ideally, the energy dissipated to transmit a bit should be independent of the bitrate. At low rates, however, there is significant energy overhead in turning on the radio electronics and the energy per bit increases. A typical PLL-based frequency synthesizer has a settling time on the order of milliseconds (due to its slow feedback loop) which may be much higher than the transmit time for short packets.

Low data rate sensor systems will require innovation in low power frequency synthesizers that have a fast transient response. One approach to reduce the lock time of the PLL frequency synthesizer is to force an initial value on the VCO input voltage [15]. The PLL output frequency will lock faster if the starting point is near the desired value. Figure 8 shows such a proposed architecture. When the PLL is locked, the VCO input voltage is sampled and stored. When the synthesizer is turned on again for the next transmission, the D/A output forces an initial value on the VCO and the PLL will achieve faster lock.

Fractional-N synthesizer architectures in which phase/frequency modulation is achieved by dithering the divide value in PLL can be used to eliminate high power mixer circuits. In order to overcome the limitation of the PLL bandwidth for high rate transmission, a digital compensation filter can be used [14]. Direct open loop modulation of the VCO as shown in Figure 8 is an alternate approach to overcome the bandwidth limitations of the PLL and achieve further power reductions[16]. During transmission, the feedback loop is open and the VCO is directly modulated.

The advantages of open loop modulation come with some drawbacks. This includes frequency stability which may require high Q external components, resulting in lower integration. Fortunately, the transmit period is very short in our case due to duty cycling and therefore we can expect smaller variations.

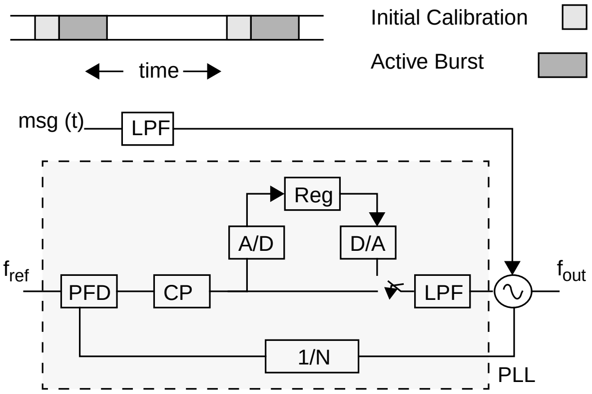

Figure 8. Direct VCO Modulation Architecture.

5. Energy Harvesting

Based on continued advances in power management techniques, it is projected that the power consumption of future low to medium throughput DSPs will be scaled to 10’s to 100’s of µW. At these low power levels, an interesting question arises: can we use ambient energy sources to power electronic systems? A circuit powered by ambient sources has a potentially infinite lifetime, as long as the source persists. In long-lived sensor embedded systems where battery replacement is difficult, generating power from ambient sources becomes imperative.

Various schemes have been proposed to eliminate the need for batteries in a portable digital system [17]. Basically, energy that exists in the environment of the device is converted by a transducer into electrical form which can be used by a circuit to perform useful work. The sources of ambient energy available to the system depend on the application. The most familiar source is solar power, often used in commercial electronic calculators. Other examples include other types of electromagnetic fields (used in RF powered ID tags, inductively powered smart cards, or noninvasive pacemaker battery recharging), thermal gradients, fluid flow, and mechanical vibration. Other proposals include powering electronic devices through harnessing energy produced by the human body [18] or the action of gravitational fields [19].

Table 1 lists potential power output for a wide variety of energy sources. Starner [18] models the power available from directly converting the energy of footsteps by inserting a piezoelectric transducer in the heel of a shoe. A direct transduction technique like this has the potential to generate large amounts of power, on the order of 5 W. The usable energy, of course, will be significantly lower. Photovoltaic cells are the most popular transducer for converting ambient energy. Advances in solar cell technology has pushed efficiency toward 20%. Assuming a typical incident power density for light of 100 mW/cm^2, this yields 20 mW for a 1 cm^2 array. Besides light, other types of electromagnetic fields have been proposed as energy sources. Magnetic fields coupled using an on-chip inductor have been shown to generate 1.5 mW of power, enough to power circuitry for a telephone card [20]. RF ID tags have been demonstrated that power their internal circuitry from the interrogating field [21]. The field coupling for these systems is much weaker than for the phone cards, but their circuit power requirements are also much lower (approximately 5 μW). Two examples of power generation using mechanical vibration are shown here. The first uses a macroscopic generator coupled to vibrations produced by human walking and leads to a power output of 400 μW. This approach produces much less energy than a direct conversion, but is a more portable and convenient solution which is still adequate for a large variety of electronics. A MEMS transducer approach which, when coupled to a much higher frequency vibration source, produces 100 μW of power.

Energy Source	Transducer	Power
Walking (Direct Conversion)	Piezoelectric	5 W
Solar	Photovoltaic Cell	20 mW
Magnetic Field	Coil	1.5 mW
Walking (Vibration)	Discrete Moving Coil	400 μW
High Frequency Vibration	MEMS Moving Coil	100 μW
RF Field	Antenna	5 μW

Table 1. Examples of Ambient Energy Sources

5.1 Vibration Based Power Generation Example

One particular approach to using ambient energy sources for power involves transduction of mechanical vibration to electrical energy. A generator based on transducing mechanical vibrations has some distinct advantages: it can be enclosed and protected from the outside environment, it functions in a constant temperature field, and it can be activated by a person. It is particularly suited for machine mounted sensors, where the vibration of the machinery provides the power, or body area sensors, where the movement of the human body generates vibrations that can be used as a power source.

Figure 9 is a detailed block diagram of our self-powered system [22]. This approach involves transduction of mechanical vibration to electrical energy. A moving coil generator is used which consists of a mass attached to a spring, which is attached to a rigid housing. The generator and rectifier subsystem is shown at the top. Transformer X1 (with a 1:10 turns ratio) converts the output voltage of the generator V_{gen} to a higher voltage that can be rectified by the half-wave rectifier formed by diode D1 and capacitor C1. Note that with proper electromechanical design, the transformer can be eliminated. Voltage V_{in} is the time-varying input voltage to the regulator.

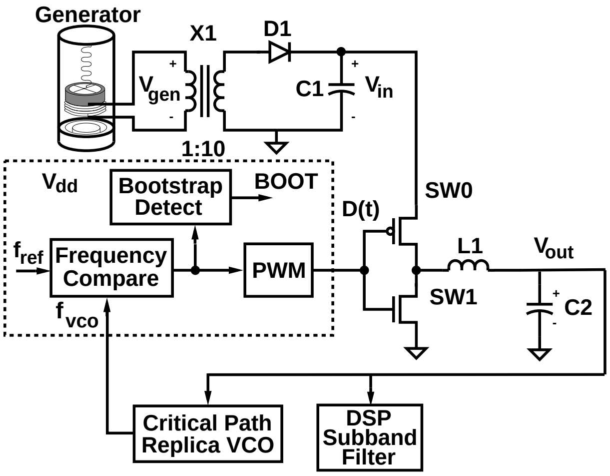

Figure 9. Vibration based self-powered system.

The regulator consists of five main subsystems: a VCO, frequency comparator, pulse-width modulated (PWM) waveform generator, bootstrap detection circuit, and a Buck converter. An external input provides the performance constraint: the rate at which the load circuit must produce results. The load circuit in this case is an 8 bit FIR filter. The DSP rate command is delivered in the form of a clock, f_{ref}, the rate at which new input samples are fed to the DSP. Its period therefore corresponds to the total delay between valid output samples of the DSP. To achieve the lowest possible power consumption, the converter downconverts V_{in} to the lowest voltage at which the DSP can run and still produce correct results at the rate set by f_{ref}.

An important difference between the self-powered system and a battery-powered system is that the former requires a backup power source. This is necessary since at startup the voltage regulator must derive its power from some source and the generator output is too uncontrolled to be used. The source could be a very small battery or a previously charged large capacitor, but it need not provide much energy since it is only used during the startup transient of the system. The bootstrap detect block switches the controller to V_{out} when the output voltage is deemed stable. One excitation of the generator produces 23ms of DSP operations, corresponding to 2,340 *free* operations. The clock rate is adjustable from 500kHz to 1MHz for the supply voltage varying from 0.9V to 1.1V. Figure 10 shows the closed loop regulation of the generator output.

The real promise of self-powered systems will be fulfilled when a micromachined generator can be integrated with a regulator circuit and a processor on the same die.

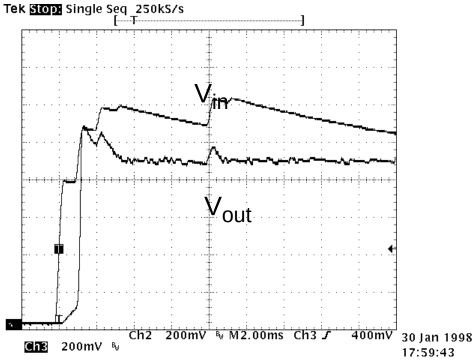

Figure 10. Closed loop regulation of generator output.

This low cost and compact implementation will have a significant impact on the future of embedded sensors. With continued advances in power management, self-powered systems should find more numerous and more impressive applications.

6. Conclusion

Low power distributed microsenors will enable a wide variety of civil and military applications. There is a rich set of research problems associated with distributed microsensors that require very different solutions than traditional macrosensors and multimedia devices. Energy dissipation, scalability, and latency must all be considered in designing network protocols for collaboration and information sharing, system partitioning, and low power electonics. Energy harvesting techniques that eliminate the need for battery source and provide “infinite” lifetime will become critical as the size of sensor systems grows.

Acknowledgments

This work is sponsored by DARPA, ARL Advanced Sensors Federated Lab program, and an NSF Career Development Award MIP-9501995. Wendi Rabiner is supported by a Kodak Fellowship and Alice Wang is supported by a Lucent GRPW Fellowship.

References

[1] K. Bult, et al., “Low Power Systems for Wireless Microsensors,” IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 17-21, August 1996.

[2] G. Asada, M. Dong, T.S. Lin, F. Newberg, G. Pottie, W. J. Kaiser, H. O. Marcy, “Wireless Integrated Network Sensors: Low Power Systems on a Chip,” IEEE ESSIRC, 1998.

[3] W. Rabiner and A. Chandrakasan, “Network-Driven Motion Estimation for Wireless Video Terminals,” IEEE Transactions on Circuits and Systems for Video Technology, pp. 644-653, August 1997.

[4] T. Meng, R. Volkan, “Distributed Network Protocols for Wireless Communication,” IEEE ISCAS, May 1998.

[5] P. Agrawal, “Energy Conservation Design Techniques for Mobile Wireless VLSI Systems,” IEEE Computer Society Workshop on VLSI, pp. 34-39, April 1998.

[6] K. Yao, R.E. Hudson, C.W. Reed, D. Chen, T.L. Lung, and F. Lorenzelli, “Array Signal Processing for a Wireless MEM sensor Network”, pp.11-20, IEEE Workshop on Signal Processing Systems: Design and Implementation, October 1998.

[7] A. Chandrakasan, R. Brodersen, Low Power CMOS Design, IEEE Press, 1998.

[8] L. Blum, M. Blum, and M. Shub, “A Simple Unpredictable Pseudo-Random Number Generator,” SIAM Journal on Computing, v. 15, n. 2, pp. 364-383, 1986.

[9] N. Takagi, “A Radix-4 Modular Multiplication Hardware Algorithm for Modular Exponentiation”, IEEE Transactions on Computers, pp. 949 - 956, August 1992.

[10] J. Goodman, A. Dancy, A. Chandrakasan, "An Energy/Security Scalable Encryption Processor Using an Embedded Variable Voltage DC/DC Converter," IEEE Journal of Solid-state Circuits, pp. 1799-1809, Nov. 1998.

[11] V. Gutnik, A.P. Chandrakasan, "Embedded Power Supply for Low-Power DSP," IEEE Trans. on VLSI Systems, pp. 425-435, December 1997.

[12] V. Tiwari, S. Malik, A. Wolfe, “Power Analysis of Embedded Software: A First Step Towards Software Power Minimization”, IEEE Trans. on VLSI Systems, pp. 437-445, December 1994.

[13] A. Abnous, J. Rabaey, “Ultra-Low-Power Domain-Specific Multimedia Processors,” IEEE VLSI Signal Processing Workshop, San Francisco, October 1996.

[14] M. Perrott, T. Tewksbury, C. Sodini, “27 mW CMOS Fractional-N Synthesizer/Modulator IC”, IEEE ISSCC, pp. 366-367. 1997.

[15] T. Iritani, K. Junzo, O.Takahiro, “Fast convergence synthesizer with initial phase difference”, Electronics and Communications in Japan, Part1, Vol. 78, No.7, 1995.

[16] S. Heinen, et al., “A 2.7V 2.5GHz Bipolar chipset for digital communication”, pp. 306-307, IEEE ISSCC 1997.

[17] C.B. Williams and R.B. Yates, ”Analysis of a Micro-Electric Generator for Microsystems”, Transducers ‘95/Euro-Sensors IX, 1995.

[18] T. Starner, “Human Powered Wearable Computing”, IBM Systems Journal, vol. 35, no. 3 & 4, pp. 618-629, 1996.

[19] M. Hayakawa, “Electronic Wristwatch with Generator”, U.S. Patent No. 5,001,685, March 1991.

[20] J. Bouvier, Y. Thorigne, S. Abou Hussein, M. Revillet, and P. Senn,“A Smart Card CMOS Circuit with Magnetic Power and Communications Interface”, IEEE ISSCC, pp. 296-297, 1997.

[21] D. Friedman, H. Heinreich, and D-W. Duan,“A Low-power CMOS Integrated Circuit for Field-Powered Radio Frequency Identification Tags”, IEEE ISSCC, pp. 294-295, 1997.

[22] R. Amirtharajah, A. Chandrakasan, "Self-Powered Signal Processing Using Vibration Based Power Generation," IEEE Journal of Solid-state Circuits, pp. 687-695, May 1998.