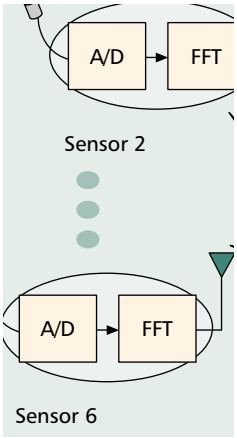
ENERGY-CENTRIC ENABLING TECHNOLOGIES FOR WIRELESS SENSOR NETWORKS

REX MIN, MANISH BHARDWAJ, SEONG-HWAN CHO, NATHAN ICKES, EUGENE SHIH, AMIT SINHA, ALICE WANG, AND ANANTHA CHANDRAKASAN,
MASSACHUSETTS INSTITUTE OF TECHNOLOGY



As a microsensor node must operate for years on a tiny battery, researchers must apply innovative system-level techniques to eliminate energy inefficiencies that would have been overlooked in the past.

ABSTRACT

Distributed networks of thousands of collaborating microsensors promise a maintenance-free, fault-tolerant platform for gathering rich multidimensional observations of the environment. Because a microsensor node must operate for years on a tiny battery, researchers must apply innovative system-level techniques to eliminate energy inefficiencies that would have been overlooked in the past. In this article we advocate two particular enablers for energy conservation: the ability to trade off performance for energy savings within the node, and collaborative processing among nodes to reduce the overall energy dissipated in the network. New levels of energy efficiency — attained through global system-level perspectives on node and network energy consumption — will enable a future where networks of hundreds, thousands, and eventually many millions of collaborating nodes are as commonplace as today's cellular phone.

DISTRIBUTED MICROSENSOR NETWORKS

Ranger Smith enters the station at the edge of his forest preserve, triggering a flurry of activity from the thousands of climate microsensors in the building. Having kept the building a trifle warm in his absence, the nano-nodes open innumerable vents to condition the hallway as he walks through. His demand to see the thermal profile of the reserve ripples across roughly 100 square miles of wilderness, waking up nearly a quarter of the million sensors that were airdropped a few years back. "Paint" on the wall receives and interpolates the data, rendering a visual picture through actuating nodes. Bad news: the thermal map indicates hot spots, and a million-node cluster is not responding. If only nano-nodes could put out fires…

While paintable "smart dust" [1] lies well within the realm of science fiction, networks of hundreds to thousands of pill-sized microsensor nodes are rapidly approaching our grasp [2, 3]. Many of the necessary components and technologies are already available. Microscopic micro-electrical mechanical system (MEMS) motion sensors are routinely fabricated on silicon. Digital circuits shrink in area constantly; a sensor node's complete data processing unit can fit on a pinhead. Entire radio transceivers, including the associated digital electronics, have been fabricated on a single chip [4]. Refinements of these enabling technologies will soon yield the form factors practical for a microsensor node.

Microsensors promise to revolutionize spatial data gathering. Driven by data aggregation — the fusion of multiple observations from different perspectives — a spatially distributed network of nodes returns a rich, high-resolution, multidimensional picture of the environment that is not possible with a single sensor or a small group of sensors. The sheer number of nodes naturally leads to the network's robustness and fault tolerance to the loss of individual nodes, making maintenance unnecessary. Since the nodes self-organize into ad hoc networks, deployment can be as easy as sprinkling nodes about the region of interest or dropping them by air, and setting up a conveniently located base station to which the nodes will relay their observations. These advantages, and the nodes' diminutive size, make sensor networks ideal for any number of inhospitable or unreachable locations where deployment is difficult, wires impractical, and maintenance impossible. The cramped confines of an appliance, facilities that produce toxic radiation or chemical vapors, lands of extreme desert or Arctic climates, and even the surface of foreign moons and planets are excellent candidates for easily deployed, maintenance-free microsensor networks.

The microsensor network is a domain with operational demands unlike any current paradigm in wireless communication. Consider a network specification for a machine monitoring application [5]. This application scenario specifies up to 12 nodes/m^2 and a maximum radio link of 10 m. Nodes are expected to process about 20 2-byte radio transmissions/s and to operate for 5–10 years with an AA battery. Practically every specification is a departure from the norms of current wireless technology. A microsensor node is the antithesis of high-bandwidth or long-range communication: node densities are higher, transmissions shorter, and data rates lower than any previous wireless sys-

1070-9916/02/$17.00 © 2002 IEEE

tem. The energy of radio transmission, normally the largest burden of the system, will likely be dwarfed by other components. As a low-data-rate and low-duty-cycle device, the node will be subject to unprecedented operational diversity, characterized by long idle periods interspersed with brief spurts of activity and communication.

Amid these unusual specifications lies the greatest challenge: the need for an unprecedented system lifetime from a wireless device. The node is a completely new breed of battery-powered device: a complete, diversely functional wireless device with the lifetime of a digital wristwatch. Adding to the challenge is the slow progress in battery energy density improvements. While the density of, say, transistors on a chip has consistently doubled every 18 months, the energy density of batteries has doubled every 5–20 years, depending on the particular chemistry, and prolonged refinement of any chemistry yields diminishing returns [6]. Moore's law simply does not apply to batteries, making energy conservation strategies essential for extending a node's lifetime.

To achieve years of lifetime in a microsensor network, energy inefficiencies in the system must be confronted and eliminated. We must take steps well beyond common low-power design practices in battery-powered wireless systems and employ innovative techniques across the system hierarchy. In the following sections we carefully characterize the energy consumption of microsensor nodes and advocate two strategies for energy conservation: the ability to trade off performance for energy savings within the node, and collaborative processing among nodes to reduce the overall energy dissipated in the network.

Microsensor Node Implementation and Characterization

The MIT project for micro-Adaptive Multidomain Power-Aware Sensors (μAMPS) is developing prototypes and technology enablers for power-aware distributed microsensor nodes. Figure 1 is our conceptual diagram of a μAMPS microsensor network. We envision a few hundred to a few thousand homogeneous nodes spread densely over a two- or three-dimensional space. The nodes are energy-constrained and immobile once placed. The nodes gather and transmit observations to a reasonably distant base station that is free from energy constraints. Within each node, an array of environmental sensors receives external stimuli from the environment, and analog-to-digital (A/D) conversion moves analog observations into the digital domain for further processing. A data processor aggregates and analyzes the observed data. Two-way communication with other nodes and a remote base station is achieved through a network processor, which packetizes and encodes the data for robustness and security, and a radio transceiver. We feel that this architecture is sufficiently general to encompass first-generation microsensor applications.

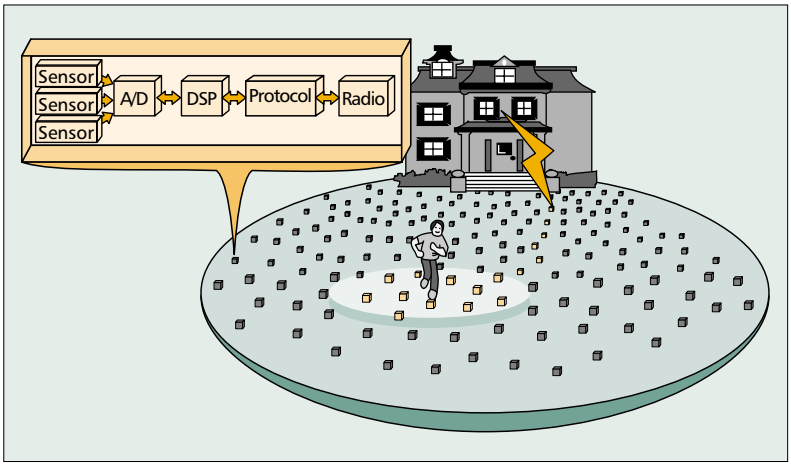


Figure 1. *The μAMPS microsensor network consists of hundreds to thousands of nodes that collaboratively gather environmental observations (footsteps of the jogger) and forward them to a remote base station (within the house). Individual nodes consist of an array of environmental sensors, A/D conversion, digital signal and network protocol processors, and a radio transceiver for two-way communication.*

Node Implementation

μAMPS-1 is our first full implementation of the μAMPS vision. The node is constructed from off-the-shelf low-power components for rapid demonstration of functionality in a series of

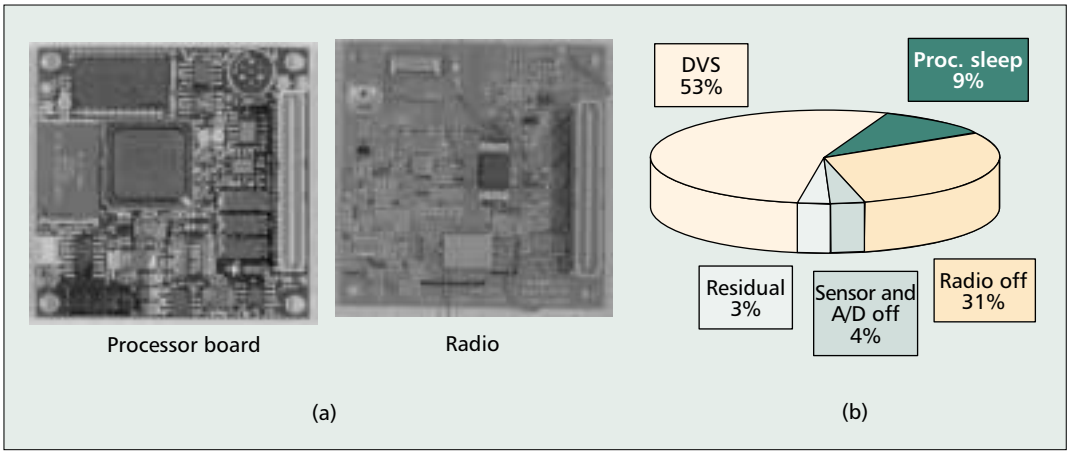


Figure 2. *a) The μAMPS-1 sensor node is implemented in stackable boards; each board is 55 mm on a side; b) energy-conserving hooks present in μAMPS-1 reduce the peak power of the node by the proportions shown here; power dissipation scales from a minimum of 27 mW to a peak of 976 mW.*

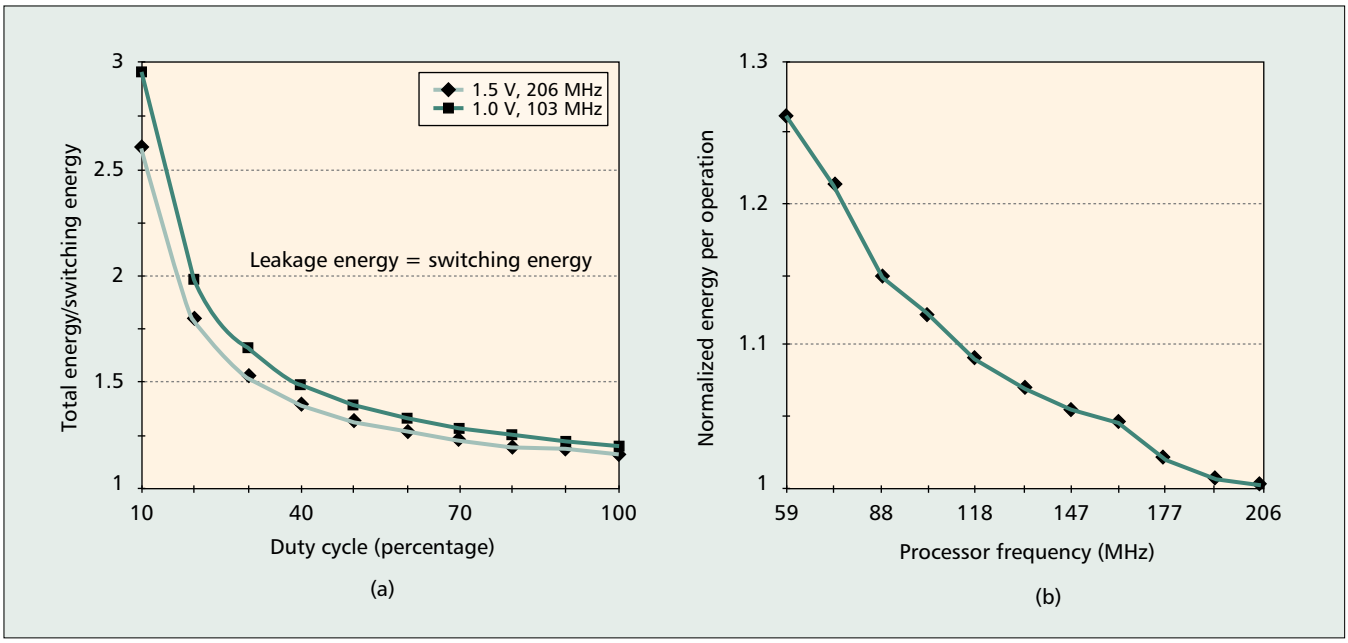


Figure 3. *a) For low duty cycles, an SA-1100 processor consumes more energy through leakage than active switching; b) running the SA-1100 at lower clock frequencies to compensate for a low duty cycle actually* increases *the total energy per operation since more leakage energy is dissipated over a longer clock period.*

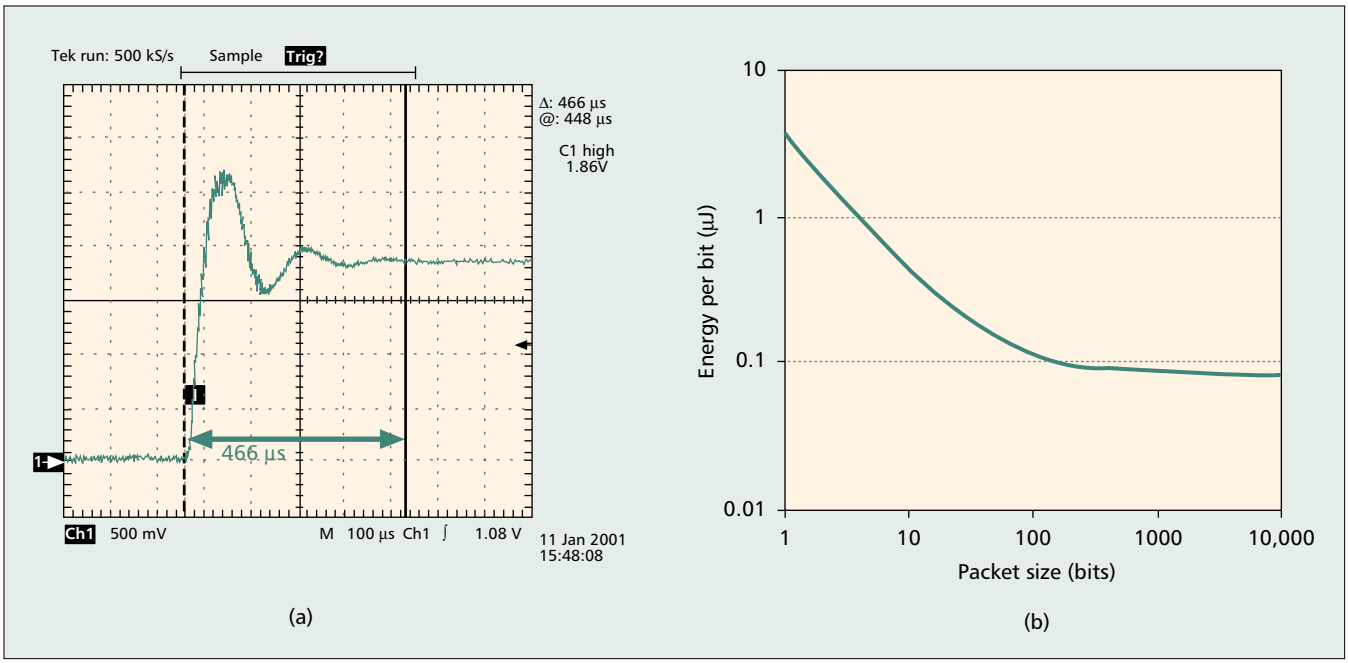
stackable 2-in square boards. Figure 2a illustrates two complete boards. µAMPS-1 contains an onboard acoustic sensor, low-power 12-bit A/D, a StrongARM SA-1110 processor for both the data and protocol processing functions, and a 2.4 GHz direct sequence spread-spectrum radio. To improve energy efficiency, the node includes custom circuitry for dynamic voltage scaling of the SA-1110 core, hooks for shutdown of each analog component, and a fully programmable link layer. The peak power dissipation of the node is just under 1 W with all components fully on. Dynamic supply voltage scaling for the SA-1110 saves up to 450 mW. Shutting down the radio saves 300 mW. Figure 2b plots the fractional contributions of these techniques to the power dissipation of µAMPS-1.

As we gain design experience with the µAMPS-1 node, we plan to integrate the functionality of the sensor node into custom integrated circuits. The next-generation µAMPS-2 node will feature custom digital signal processing (DSP) and protocol processing chips designed for power awareness from the ground up. Our custom processors will provide integrated power-aware implementations solely of the signal processing and communications operations required of the node. By removing the architectural overhead of decoding and processing general-purpose instructions, we will trade the flexibility of a general-purpose processor for a lean energy-efficient processor that supports only the functionality required of the microsensor application. From our previous experiences with such *domain-specific* processors [7], we expect at least a three-order-of-magnitude power reduction over the SA-1110. Our ultimate goal is a microsensor system on a chip, integrating MEMS sensors, A/D, data and protocol processing, and a radio transceiver onto a single die. The entire node would occupy a 1 cm cube.

Energy Consumption Characteristics

It is instructive to consider the power consumption characteristics of the node in three parts: the sensing circuitry, the digital processing, and the radio transceiver. The sensing circuitry, which consists of the environmental sensors and the A/D converter, requires energy for bias currents, as well as amplification and analog filtering. Their power dissipation is relatively constant while on, and improvements to their energy efficiency depend on increasing integration and skilled analog circuit design. In the following discussion, we consider the energies of the remaining two sections, digital computation and radio transmission, to find that their energy consumption characteristics fundamentally conflict with the operational characteristics of the microsensor node.

A node's digital processing circuits are typically used for digital signal processing of gathered data and implementation of the protocol stack. Energy consumed by digital circuits consists of *dynamic* and *static* dissipation. The dynamic energy of digital computation is the energy required to energize ("switch") parasitic capacitors on an integrated circuit from zero voltage (a digital zero) to the voltage of the power supply (a digital one). Dynamic energy is described by CV_{DD}^2, with C representing the switched capacitance and V_{DD} the supply voltage. Static dissipation originates from the undesirable leakage of current from power to ground at all times and is described by $I_0 e^{V_{DD}/nV_{\text{th}}}$, where V_{th} is the threshold voltage of the transistors, and I_0 and n are constants for the process technology. Summing the switching and leakage energy yields $E = CV_{DD}^2 + (tV_{DD})I_0 e^{V_{DD}/nV_{\text{th}}}$ for a computation that switches a total capacitance C over duration t. Note that for a constant supply voltage V_{DD}, switching energy for any given computation is independent of time, while



■ Figure 4. *a) 466 µs startup transient from the µAMPS-1 2.4 GHz radio, measured at the input to the radio's voltage controlled oscillator; b) the effects of startup energy revealed — the total radio energy per bit required to transmit packets of various sizes. In our model, the data rate is 1 Mb/s, the radio electronics consume 81 mW, and the radiated output power is 1 mW (0 dBm).*

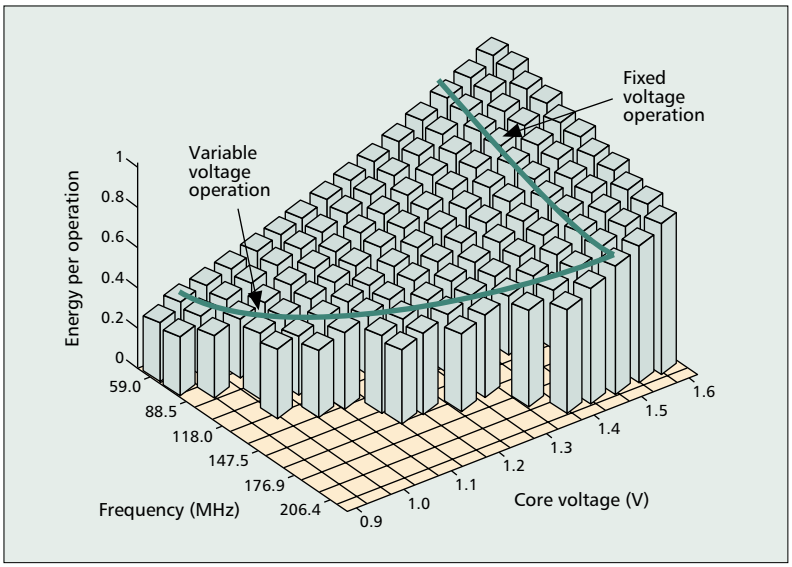
leakage energy is linear with time. While switching energy has historically exceeded leakage energy for modern complementary metal oxide semiconductor (CMOS) applications, the trend is beginning to reverse with recent semiconductor process technologies. Each new process generation (e.g., 0.18 µ or 0.13 µ) increases leakage threefold, and leakage is beginning to approach 50 percent of a digital circuit's operating power [8].

A microsensor's long idle periods impose a low duty cycle on its processor, encouraging the dominance of leakage energy. Figure 3 illustrates this concern with data gathered from the SA-1100 microprocessor. Figure 3a shows that leakage energy begins to dominate over switching energy as the processor's duty cycle is reduced. Leakage energy is proportional with time, whether or not the processor is doing useful work. As a result, slowing down the clock actually increases the amount of leakage energy within each clock period, causing the energy per operation to increase. This is illustrated in Fig. 3b. A reduction in clock frequency reduces the number of idle cycles, but leakage nonetheless remains proportional to time.

The easiest way to reduce leakage is to shut down the power supply to idle components. Microsensor nodes will no doubt benefit from this time-honored technique. For relatively simple circuits such as analog sensors, circuits can be powered up and down quickly, with no ill effect. Shutting down more complicated circuits, however, can cost time and energy overhead. For example, powering down a processor requires the preservation of its state. If the processor should be needed immediately after it has been powered down, the energy and time required to save and restore the state is wasted. There exists a hidden time and energy cost to shutting down a circuit that must be balanced with the expected duration of the shutdown. There is a break-even point at which the energy savings of shutdown equal the energy overhead of shutting down and restoring the circuit.

Nowhere is the hidden cost of shutdown greater than in the radio transceiver. The energy consumption of the radio consists of static power dissipated by the analog electronics (analogous to leakage in the digital case, except that these *bias currents* serve the useful purpose of stabilizing the radio) and the radiated RF energy. The radiated energy, which scales with transmitted distance as d^2 to d^4 depending on environmental conditions, has historically dominated radio energy. For closely packed microsensors, however, the radio electronics are of greater concern. The energy required for a complete radio transmission can be described as $P_{tx}(T_{transmit} + T_{start}) + P_{out}T_{on}$, where P_{tx} represents the power of the transmitter electronics, $T_{transmit}$ the transmit duration, T_{start} the startup time, and $P_{out}T_{on}$ the radiated energy. The startup time T_{start} is our specific concern. Today's transceivers require an initial startup time on the order of hundreds of microseconds, during which transients in the analog electronics need to settle. The radio transceiver in µAMPS-1, for instance, has a measured startup transient of 466 µs, which is illustrated in Fig 4a. Substantial amounts of energy are consumed during the startup period.

With the low data rates inherent to sensor networks, the packets transmitted by the radio between idle periods are likely to be small. Figure 4b presents the energy per transmitted bit as the length of the transmitted packet is varied. As the packet grows shorter, the energy per bit increases dramatically. The energy of active transmission is being swamped by the startup energy. Starting up the radio entails a fixed energy cost, while transmitting bits is a variable



■ Figure 5. *Measured energy consumption per operation for the SA-1100 processor, as clock frequency and voltage are varied. A lower frequency at the same voltage incurs a higher* per-operation *cost due to additional leakage energy.*

cost that is linear in the number of bits transmitted. While the radio's high bias currents require that we accept the cost of shutdown, the node ought to amortize the startup energy over more transmitted bits to reduce the energy cost *per bit* of transmission. Increased buffering between transmissions, provided that the additional latency can be tolerated, is one approach.

The Power-Aware Microsensor Node

The energy models presented above suggest that great inefficiencies can occur when the microsensor's demands are mapped onto contemporary hardware. Creative thinking is required to ensure minimal energy consumption at all times. The approach we advocate is *power awareness*, the design of circuits, architectures, and algorithms that can gracefully trade off performance and quality for energy savings. The key realization is that the node's high operational diversity will necessarily lead to energy inefficiency for a node that is optimized for any one operating point. Perhaps the nodes are accidentally deployed more closely together than expected, or stimuli are fewer and farther between, or the user chooses to tolerate a few additional milliseconds of latency. How can the node automatically take advantage of these new conditions to save additional energy? A node that cannot adapt to relaxed performance demands with energy reductions is wasting precious battery energy.

Power awareness, then, is really an awareness of the exact performance demands of the user and the environment. A power-aware system consumes just enough energy to achieve that level of performance, and not one decibel, byte, or hertz more. Power-aware systems exhibit this characteristic at all levels of the system hierarchy. Energy trade-offs are enabled at the circuit level and exploited at the algorithm level. In this section, we illustrate how we can trade off energy for a variety of performance parameters at all levels of the system hierarchy.

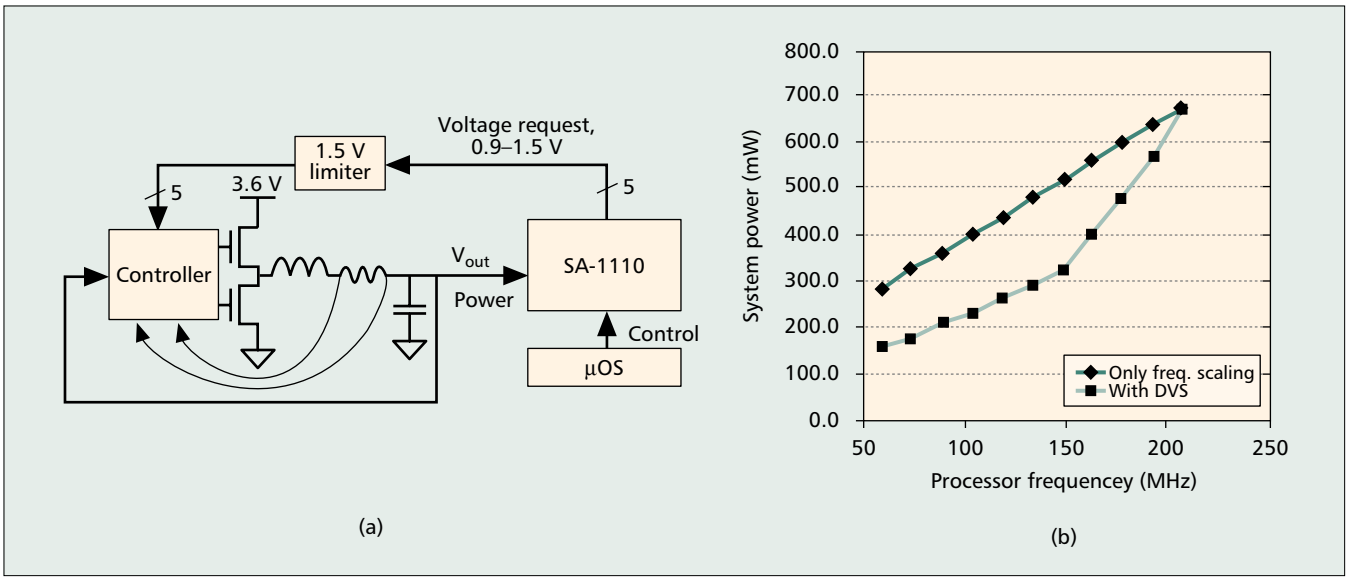
Circuits: Energy vs. Latency through Voltage Scaling

A primary performance metric for digital CMOS circuits is latency: the propagation delay of digital signals flowing through the logic. Latency can be traded for energy savings in CMOS circuits by varying the circuit's supply voltage. As discussed in an earlier section, a higher supply voltage results in a shorter propagation delay through CMOS gates, which results in a faster computation, but subjects the gates to quadratically higher switching energy and exponentially higher leakage current. A lower supply voltage thus offers dramatic energy savings at the expense of additional latency. A microsensor node's high operational diversity all but ensures that an application's average performance demands will fall well below a circuit's peak performance at maximum voltage. (Below, for instance, we discuss how scalable data processing algorithms can ease the workload presented to its processor.) Therefore, it is often desirable to trade latency for energy savings.

For a circuit with fully static CMOS logic — the most common logic family for low-power circuits — supply voltage can be adjusted on the fly. Hence, we can run a processor with a higher voltage when peak performance is demanded and scale back to a lower voltage when latency requirements are looser. In other words, computations are "stretched out" to fill the entire latency window, saving power and eliminating the need for component shutdown. µAMPS-1 supports this *dynamic voltage scaling* (DVS) technique on the SA-1110 microprocessor [9]. Figure 5 demonstrates the range of voltages and clock frequencies at which the processor can operate, under reasonable environmental conditions. A lower clock frequency, which relaxes the latency constraints on the circuits, allows the processor to operate with lower supply voltages.

DVS is implemented on the µAMPS-1 node through collaboration between the SA-1110's operating system and voltage regulator, as illustrated in Fig 6a. The voltage regulator is a programmable DC-DC converter that can provide a variable supply voltage to the SA-1110 core. The operating system, our custom adaptation of the widely used *eCOS* operating system for embedded processors, monitors processor workload and the current latency constraints required by the application. The operating system attempts to schedule the minimum clock frequency and supply voltage necessary to meet the application's latency constraints.

The energy savings enabled by DVS can be substantial. Figure 6b illustrates the energy consumed by the complete µAMPS-1 node (with radio off) as the SA-1110 clock frequency is adjusted. Enabling DVS on the SA-1110 alone reduces *total node energy* for µAMPS-1 by up to a factor of two. Since faster processing consumes more energy, DVS provides digital circuits with the ability to offer "just enough performance" to meet latency constraints with minimum energy.



■ Figure 6. *a) Dynamic voltage scaling (DVS) support aboard µAMPS-1 — the node operating system running on the SA-1110 monitors workload and latency requirements, and controls the voltage driven by the buck regulator that powers the SA-1110 core; b) DVS on the µAMPS processor alone reduces* total node energy *by nearly 50 percent. DVS enables a factor-of-four energy scalability on the node.*

Architectures: Energy vs. Bit Precision through Ensembles of Systems

CMOS circuits become complete digital systems through the collaboration of functional units, such as adders, multipliers, and memory cells. These are the fundamental building blocks of DSPs and microcontrollers for protocol stacks. We explore the power awareness of these functional units by considering a multiplication unit and its energy scalability over a new dimension of performance: the bit width of its computation.

Multiplier circuits are typically designed for a fixed maximum operand size, such as 64 b/input. In practice, however, the actual inputs to the multiplier are typically far smaller. Calculating, say, an 8-bit multiplication on a 64-bit multiplier can lead to serious energy inefficiencies due to unnecessary digital switching on the high bits. The problem size of the multiplication is a source of operational diversity, and large monolithic multiplier circuits are insufficiently power-aware.

An architectural solution to high input bit-width diversity is the incorporation of additional smaller multipliers of varying sizes, as shown in Fig 7. Incoming multiplications are routed to the smallest multiplier that can compute the result, reducing the energy overhead of unused bits. An *ensemble of point systems*, each of which is energy-efficient for a small range of inputs, takes the place of a single system whose energy consumption does not scale as gracefully with input. The size and composition of the ensemble is an optimization problem that accounts for the probabilistic distribution of the inputs and the energy overhead of routing them. Details and further examples are beyond the scope of this article but are plentiful in [10]. In short, an ensemble of systems improves power awareness for digital architectures with a modest cost in chip area. As process technologies continue to shrink digital circuits, we feel that the area trade-off is worthwhile.

Algorithms: Energy vs. Accuracy through Incremental Refinement

Now that the node's digital circuits can adapt their energy consumption to a diversity of inputs and latencies, we are motivated to design the algorithms running on these circuits to take advantage of these properties. Power-aware algorithms permit scalability in the number of operations (and hence the energy) required of the algorithm. These algorithms gracefully degrade result accuracy as the number of operations (and hence the required energy) is scaled back. In other words, the algorithms offer *incremental refinement*: the algorithm yields a reasonable approximation of the answer after a minimal number of operations, and the result grows increasingly accurate as more and more operations are performed.

Traditional computational algorithms offer an "all-or-nothing" approach to computation in which the final result is not accessible until the entire computation is complete. Fortunately, broad classes of such signal processing algorithms that do not naturally offer incremental refinement can be restructured to attain it. While techniques exist to scale computation with well-defined formal metrics for the quality of signal processing [11], a simple and intuitive approach is the *most significant first* transform which rearranges an algorithm's computations by their significance to the final result. Most significant transforms are known to exist for the fast Fourier transform (FFT), discrete cosine transform (DCT), and a variety of other linear signal processing operations. We motivate this technique in the context of beamforming, the fusion of data streams from multiple sensors into a single high-quality stream.

Finite impulse response (FIR) filtering is a common beamforming technique. A finite-length discrete-time impulse response is convolved with a digitized data stream from a

sensor. Traditional time domain FIR filtering calculates the partial sums of the convolution in numerical order: one output tap at a time, from $y[0]$ to $y[n]$. Reducing the amount of computation to the filter would result in completely uncalculated output samples toward $y[n]$. Traditional filtering methods thus yield poor energy quality scalability.

Intuitively, the partial sums that are largest in magnitude, and therefore the largest contributors to the final result, arise from samples of the impulse response that have the highest magnitude. Hence, sorting the coefficients by magnitude and computing the partial sums derived from the largest taps first will produce a reasonable approximation of the final result from just a few taps of the impulse response. Figure 8a shows an example beamforming FIR filter response with the five most significant taps highlighted. Figure 8b compares the output from the full filter with the approximation generated by just five well-chosen taps. Since the energy consumed by the filter is proportional to the number of taps convolved, processing the most significant taps first achieves a graceful degradation of quality for energy savings. The power awareness of FIR filtering has been enhanced with very little overhead: on the StrongARM processor, we measured the energy required for sorting the taps of a 128-tap filter to be 4 percent of the energy required to produce a single sample of output.

Most significant first transforms are also applicable on a signal-by-signal basis. A node that wishes to aggregate signals received from multiple nodes can sort the incoming signals by their signal-to-noise ratio (SNR). When energy is constrained, the node aggregates the highest SNR signals first to ensure that the lowest SNR signals are discarded when the computation is scaled back. Utilizing the scalable versions of essential microsensor signal processing algorithms ensures that the algorithm will be able to provide a graceful degradation of quality under energy constraints.

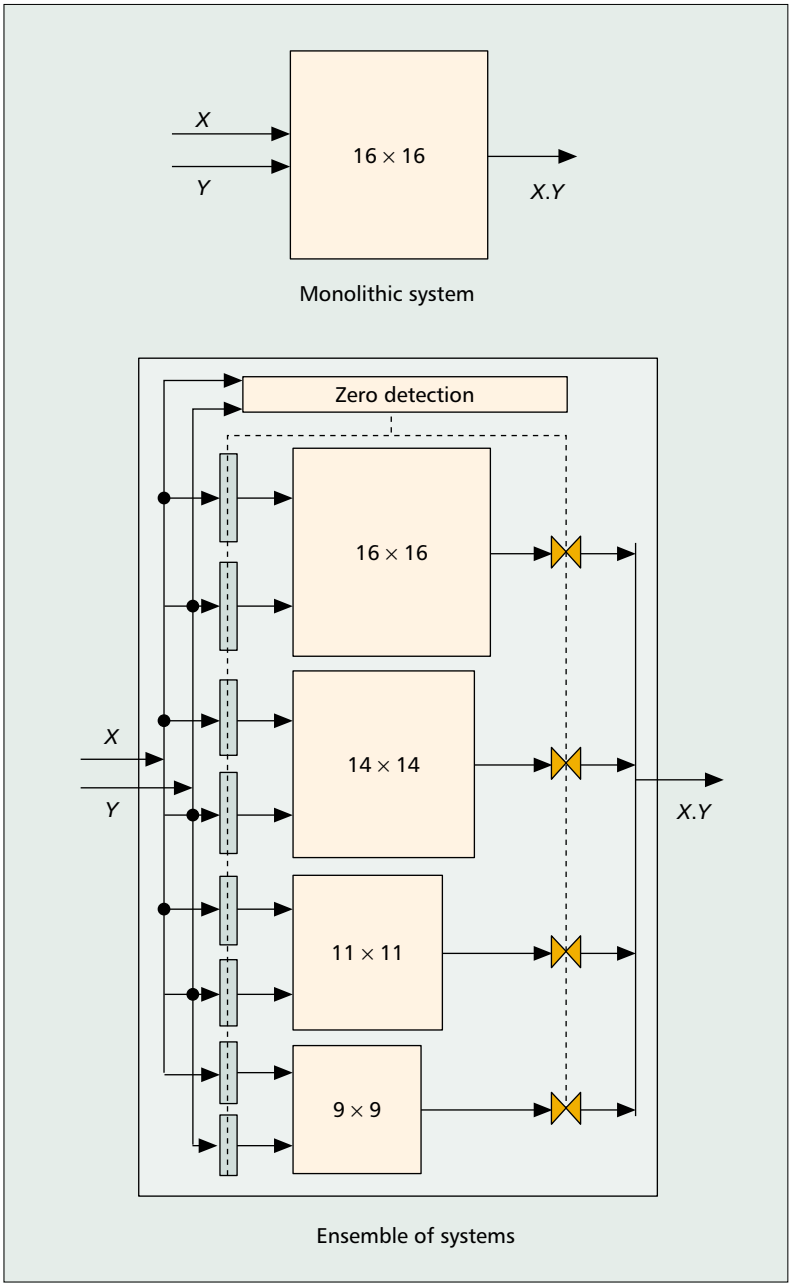


Figure 7. *A monolithic multiplier (top) is less power-aware than an ensemble of multipliers (below). The ensemble routes each pair of incoming operands to the smallest multiplier that can compute the result to take advantage of the lower energy consumption of the smaller circuit. On a series of operands typical of speech processing applications, the ensemble consumes 57 percent less power than the monolithic multiplier [10].*

Radio Link: Energy vs. Bit Error Rate through Energy-Agile Coding

In our final example of enhancing power awareness, we explore the trade-off between energy consumption and the performance of the node's radio link. The performance metric we employ is link reliability, measured by the link's bit error probability. In the case of a periodic data gathering network, this metric can be loosely viewed as the user's tolerance to missed samples, as bit errors lead to packet errors, causing lost data samples.

The energy vs. link quality trade-off here is a particularly interesting one, for there are two completely different ways to scale the link's bit error rate (BER). These options are summarized in Fig. 9a. The physical layer solution is to increase the transmit power of the radio, resulting in a higher SNR for the transmission. The link layer solution is to envelop the transmitted data in a stronger error-correcting code that is more resistant to poor link quality. The use of a more error-resistant code requires more computation energy in the network. In the case of convolutional codes, which are appropriate for microsensor networks, the energy required by the Viterbi decoder for the code increases exponentially with the code's constraint length.

To explore these two dimensions, we considered a family of convolutional codes of varying rate by puncturing a rate-1/2 base code. For each code, we computed the radio transmission power required to achieve a range of bit error probabilities for a sample path loss of 70 dB. Detailed energy models were generated from a custom CMOS layout of a network processor,

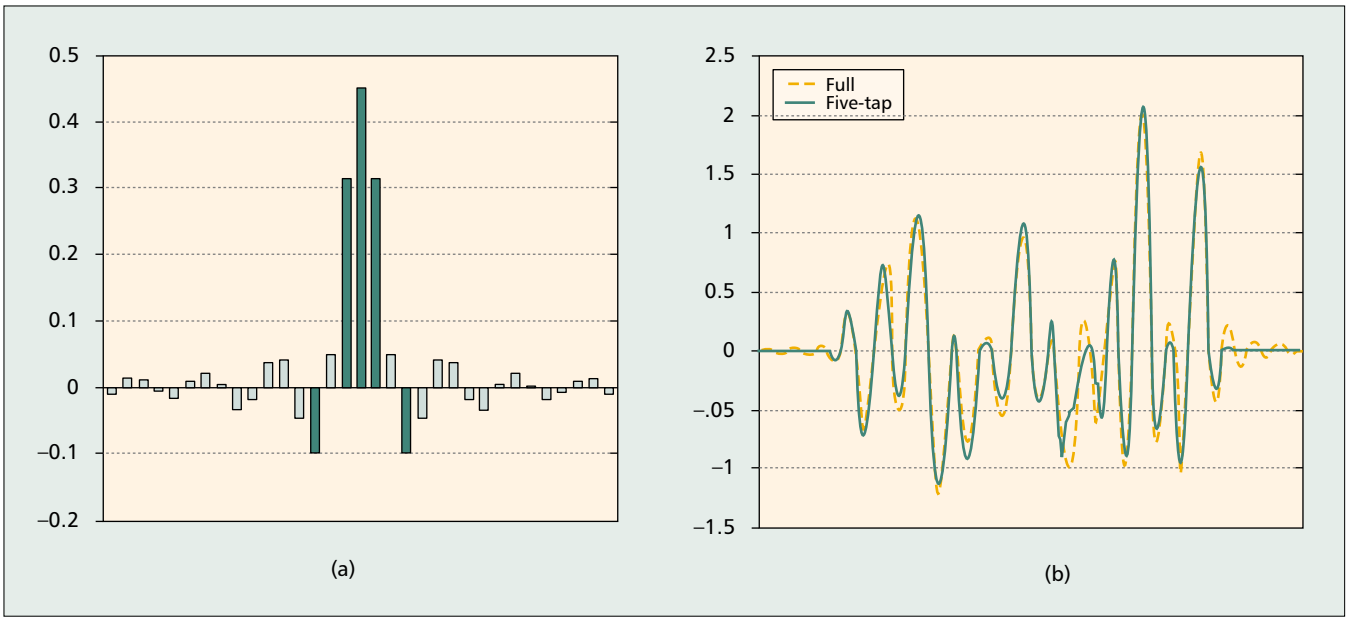


Figure 8. *a) Finite-impulse response with the five* most significant *taps highlighted; b) a signal is filtered through both the full FIR filter and an abbreviated filter consisting of the five previously highlighted taps. Even a few of the most significant taps provide a close approximation to the full result.*

the μAMPS-1 radio electronics, and transmission path loss. Summing the energies of coding, decoding, and transmission from these models yields the total energy per bit vs. the probability of error for each code under consideration. Figure 9b plots these curves.

The graphs illustrate that the most energy-efficient coding strategy changes as the BER varies. When high error rates are tolerable, total energy is minimized by shutting down the Viterbi decoder and using no error-correcting code at all. The required transmission energy to attain the desired BER is already low without coding, and any computational energy for error correction would actually dominate the total energy consumption. As the BER demands grow more stringent, the need for additional transmit power grows, and it becomes advantageous to expend more energy for stronger error-correcting codes. These results are for a single transmission distance representing a 70 dB path loss. Different transmission distances would uniformly raise or lower the radio transmission energies for each code, resulting in a

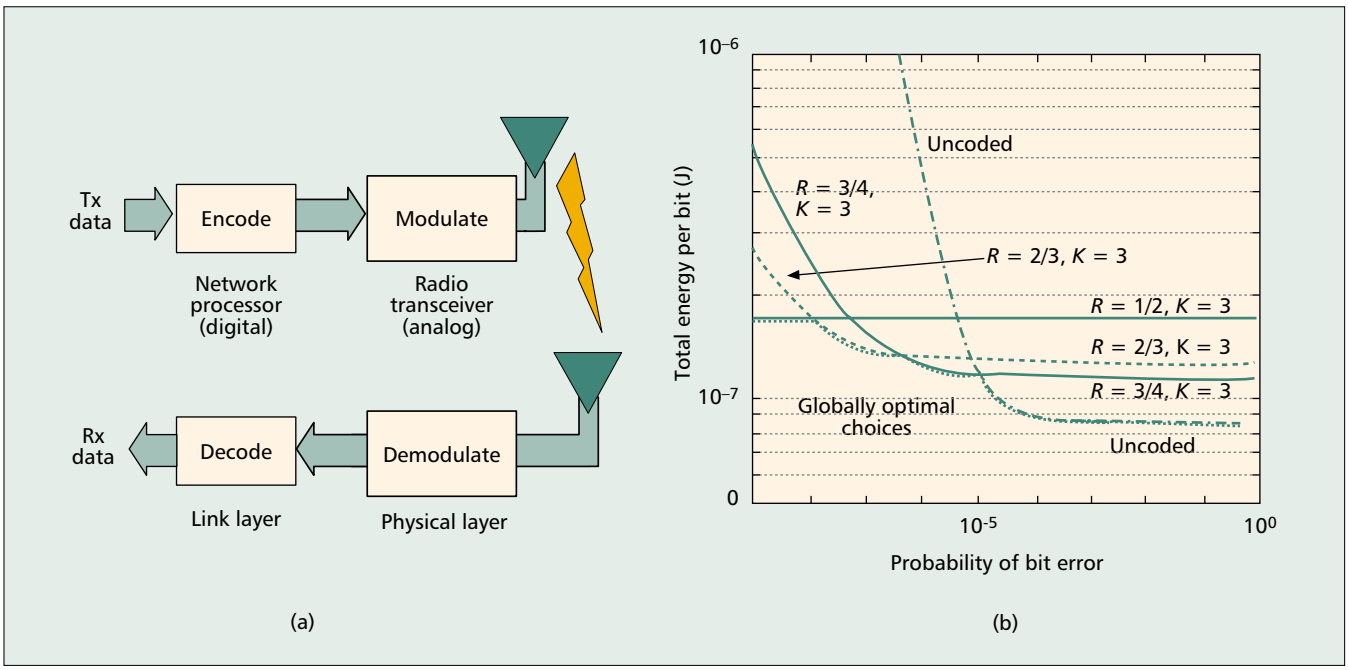


Figure 9. *a) Energy for communication is dissipated at the network processor, which encodes and decodes the packets, and the analog radio transceiver; b) energy required to transmit 10 kb of data at 1 Mb/s over a 70 dB path loss, with various coding schemes. The globally optimal coding scheme, which is highlighted above, dynamically selects stronger codes as the tolerable bit error decreases.*

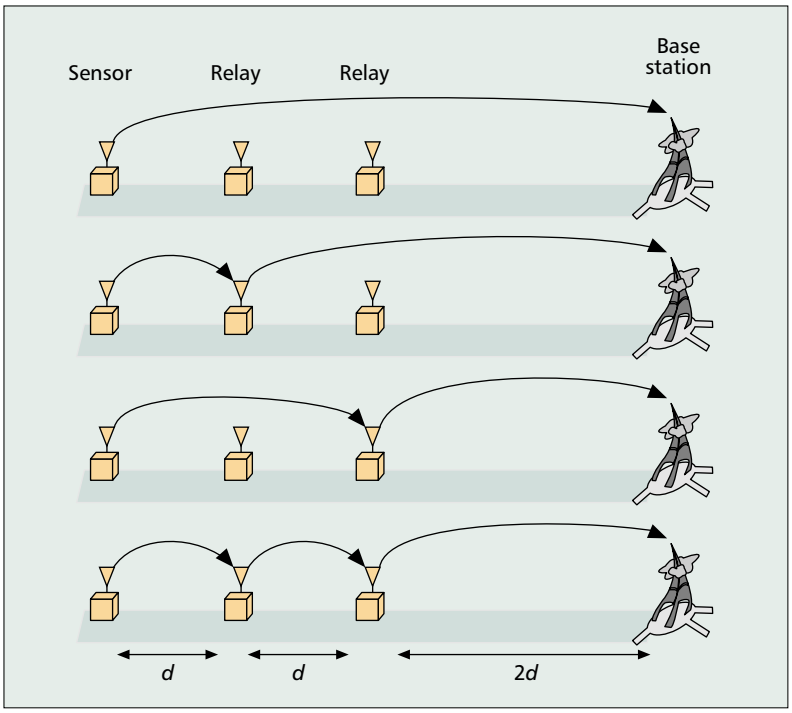


Figure 10. *In a collinear three-node network, we consider four potential multihop routes from the most distant sensor node to the base station. The minimum energy routing strategy over the lifetime of the network is a temporal rotation among these roles, in a ratio determined by the nodes' energy consumption characteristics.*

shift in the breakpoints where different coding schemes (or no coding at all) become optimal. In short, a maximally energy-quality scalable link layer requires a network processor that can adjust its coding scheme *on the fly* as the quality demands on the link change.

Networking 1000 Nodes

In the previous section we encouraged power-aware design at all levels of the system hierarchy, from CMOS circuits to signal processing algorithms. A complete system-wide view of the node that considers the interactions among all levels of the system hierarchy is essential for minimizing the total energy consumption of the node. Extending this view to encompass the entire network, the interactions among multiple nodes determine the *total energy consumption of the network*. In other words, the microsensor network can be seen as a spatial distribution of discrete chunks of energy, and the *total network energy* is characterized by the sum and spatial variance of these chunks. With microsensors' unprecedented reliance on collaboration, the concept of network energy is a novel but natural view. In this section we demonstrate techniques for embedding network energy reduction into the natural collaboration among nodes.

Propagating data from the observer nodes to the base station is an energy-intensive operation. Direct wireless communication between all nodes and a potentially distant base station is infeasible. For starters, direct communication would squander precious wireless bandwidth, since frequency reuse would not be possible. From the system energy point of view, there are two inefficiencies: a great deal of redundant data is being transmitted, and all nodes are transmitting over the long wireless link to the base station. Collaborative routing protocols can reduce both inefficiencies.

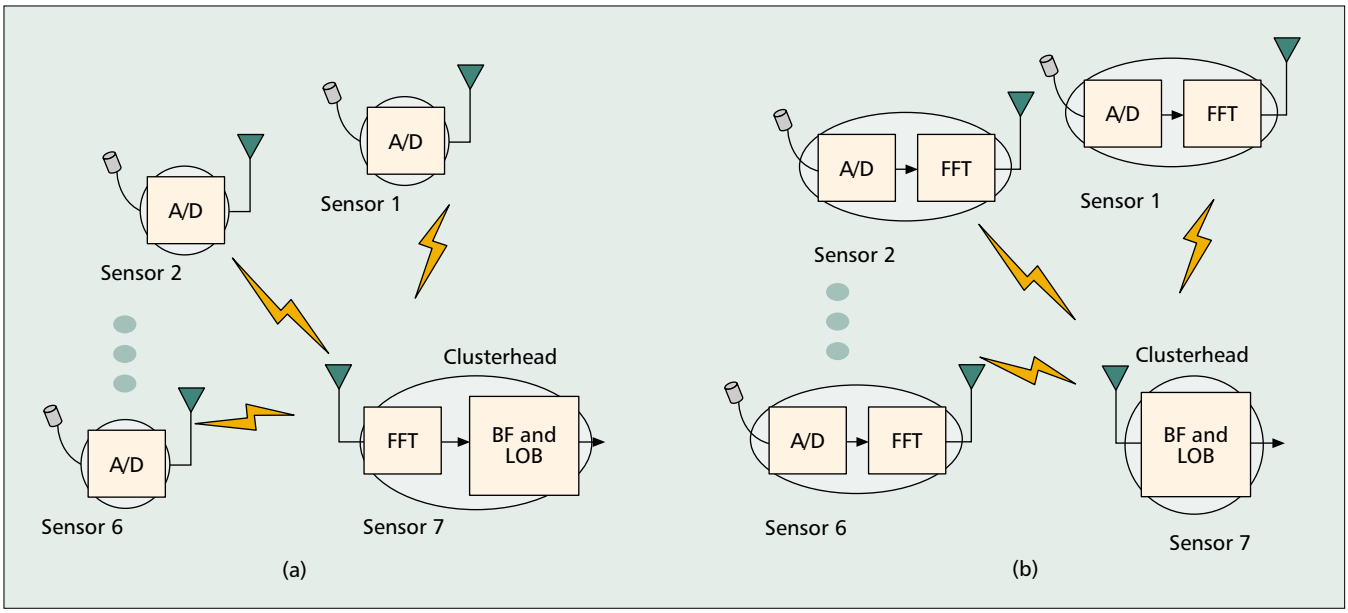
Multihop routing is a step in the right direction. In multihop, intermediate nodes act as routers for others' data. Observed data from the farthest corners of the network propagate from node to node in a series of short wireless hops, eventually reaching the base station. The r^2–r^4 path losses of wireless communication dictate that many small hops will consume less transmission energy than a single long-distance link, assuming that processing overhead is minimal.

In a naïve multihop scheme, all data destined for the base station will flow through the nodes nearest to the base station, causing these nodes to die quickly. The burden of making the final long hop to the base station shifts to ever farther nodes as the closest ones continue to die like a receding tide. Although we have reduced total system energy, the hotspots in energy consumption near the base station that lead to uneven spatial energy consumption are symptomatic of global inefficiency, which leads to shortened network lifetime.

Maximizing the lifetime of a multihop network requires careful consideration of each node's role in the process. We choose the collinear three-node network depicted in Fig. 10 as a motivating example. In a three-node network, there are four useful routes from the farthest node to the base station: direct transmission, a single hop through one or the other of the intermediate nodes, or hops through both nodes. We seek the routing strategy that is optimal-the one that maximizes system lifetime.

It turns out that no single route, by itself, is an optimal route over the entire life of the network. The optimal solution actually involves a rotation of roles over time. This result is obtained by considering each possible route in a linear program for maximizing the network lifetime under the constraints of a positive lifetime value and fixed battery energy for each node. This technique is discussed in detail and generalized in [12]. The final numerical result depends heavily on the node energy models that quantify the trade-off between the path loss of transmission and the startup and static power dissipation of the routing nodes. For the scenario of Fig. 10, the optimal role assignment using energy models from [12] suggests sending 24.5 percent of packets through each of the one-hop routes, and the remaining 51 percent through the two-hop route. (Direct transmission is never used.) Rotating the nodes' roles is an effective collaborative technique for prolonging lifetime.

Cluster-based routing protocols are an alternative to multihop with special advantages for microsensor networks. The Low Energy Adaptive Clustering Hierarchy (LEACH) protocol serves as our example. LEACH is a cluster-based protocol in which groups of adjacent nodes organize themselves into *ad hoc* clusters. One node in each cluster, the *clusterhead*, receives data from the other nodes in the cluster and forwards the combined data from the entire



■ Figure 11. *Two ways to partition a computation over a cluster of nodes: a) all digital computation (FFT, beamforming, and line-of-bearing detection) is performed at the clusterhead; b) computation is distributed between the clusterhead and observer nodes. The second system partition utilizes the network's inherent parallelism to relax global latency deadlines.*

cluster to the base station in a single long-distance transmission. New clusterheads (and new clusters as well) are chosen at periodic intervals to rotate this energy-draining role around the network [13].

Clusterheads are an ideal place to aggregate data from the cluster's nodes. Data aggregation is the fusion of multiple streams of correlated data inputs into a single high-quality output. A class of algorithms known as *beamforming* algorithms can perform this aggregation, either with or without knowledge of the nodes' locations [14]. Since environmental observations from adjacent nodes in the cluster are likely to be highly correlated, aggregation is an essential collaborative tool to reduce the number of bits transmitted over wireless links.

We conclude this section with an example of networkwide energy minimization on power-aware nodes. In a collaborative effort with the Army Research Laboratory, we have implemented the ARL processing algorithms to track the bearing of moving targets on the µAMPS hardware. A seven-sensor cluster takes acoustic observations of the target. The observations are each processed with an FFT and beamformed together in the frequency domain.

We seek to implement this processing flow over the sensor network in a way that minimizes total network energy, under a fixed latency constraint on the processing. While the beamforming of the six data streams must necessarily take place at the clusterhead, the FFT operations can be processed at either the clusterhead or the observer nodes. In the former case, the clusterhead receives six time-domain observations and performs six FFTs. In the latter case, each observer node performs a single FFT on its data stream and transmits the frequency domain result to the cluster head. Figure 11 illustrates these two possible system partitions.

Although only the distribution of computation, and not its amount, has changed, the latter distribution of computation halves the power required for computation on SA-1100-based nodes. The constraint on latency requires that we operate the clusterhead's SA-1100 processor at its full operating voltage, in order to compute six FFTs and beamform the results in time. Distributing the FFTs over the six observers reduces the workload on the clusterhead and allows the observer nodes to work in parallel. With less computation to complete in the same amount of time, the clusterhead and observers can operate their processors at a lower voltage. From our experiments, the result is a total computational energy of 3.4 mJ across the nodes vs. 6.2 mJ for the inferior partition. Taking advantage of the network's inherent parallelism, we have distributed a global latency constraint across the entire cluster to relax local latency constraints at the clusterhead, and allowed the power-aware nodes to take advantage of the additional allowable latency. In short, collaboration among the nodes can achieve more than just high-quality data: collaboration can reduce and distribute overall system energy.

Conclusion and Future Directions

A microsensor network that can gather and transmit data for years demands nodes that operate at energy efficiencies unheard of in today's wireless systems. Sensor nodes must take advantage of operational diversity, such as the long periods of idle time between interesting events, by gracefully scaling back energy consumption. The user must precisely define the network's performance requirements using metrics ranging from latency to accuracy to reliability so that the network performs just enough computation to meet the user's specific demands, and no more. The network must consider itself as a single entity, where collaborative

Only with such careful attention to the details of energy consumption at every point in the design process can we expect to see a thousand-node microsensor network that can deliver years of continuous service.

communication protocols remove redundancies in computation and communication, and maintain an even spatial distribution of energy. Only with such careful attention to the details of energy consumption at every point in the design process can we expect to see a 1000-node microsensor network that can deliver years of continuous service.

We have focused our discussion on the hardware and algorithmic enablers for energy-efficient microsensor networks. The final step in the design hierarchy, the design of an application programming interface (API) and development tools that will bring the functionality of the network into the hands of users, is an emerging field of research. We envision that an ideal node API would expose the power-aware operation of the node without sacrificing the abstraction of low-level functionality. The API would enable an application to shut down or throttle the performance of each hardware component on the node. Top-level API calls directed at the network as a single entity would allow quality and performance to be set and dynamically adjusted, allowing the network to manage global energy consumption through energy-quality trade-offs. Power-aware signal processing routines would be available as a library and be callable with ceilings on their latency or energy consumption.

Deploying software for a 1000-node microsensor network requires advances in network simulation. A simulator and development environment for microsensor networks must enable programmers to profile the performance and energy efficiency of their software and hardware over a variety of operating conditions. Energy profiling requires the creation of accurate energy models for the cost of computation and communication on the nodes. Simulating 1000-node and larger networks will require new techniques in high-speed simulation. Simulation results published in the literature rarely exceed several hundred nodes due to the limited performance of existing tools. Perhaps the simulator could locate and cache redundant computations across nodes, or offer a trade-off between result precision and speed.

The deployment of the first successful sensor network will be an exciting and revolutionary milestone. From there, the research opportunities only become more interesting. One possible next step is a node with infinite lifetime. Since nodes are essentially sensing energy in the environment, why not harvest it for operation as well? A "sensor" that efficiently transduces environmental energy into useful electrical energy is an energy harvester. With the refinement of energy harvesting techniques that can gather useful energy from vibrations, blasts of radio energy, and the like, self-powered circuitry is a very real possibility. Energy harvesting schemes developed in the laboratory have generated 10 μW of power from mechanical vibrations, already enough for low-frequency DSP [15]. With continuing advances in energy harvesting and improvements in node integration, a batteryless infinite-lifetime sensor network is possible.

As we hinted in the introduction, ultra-dense sensor networks are also a logical next step. As silicon circuits continue to shrink, the physical size of the nodes themselves will shrink as well. As node form factors shrink and researchers become comfortable with network protocols for 1000-node networks, networks with several hundred to several thousand nodes per square meter will begin to appear. Such a dense network would offer a new level of parallelism and fault tolerance … not to mention trivially small radio transmission energies.

It is inevitable that wireless microsensor networks will mature from laboratory curiosities to networks of millions of nodes, deployed through paintbrushes, misters, injections, and aircraft. So perhaps it is not far-fetched to envision that the wireless microsensor network will be the true enabler for ubiquitous computing: the availability of computational power that is taken for granted anywhere, at any time. To be truly imperceptible, technology must be omnipresent. And in Ranger Smith's forest preserve, teeming with many millions of nano-nodes, it is.

Acknowledgments

The authors would like to thank Ben Calhoun, Fred Lee, and Theodoros Konstantakopoulos for their help in designing the μAMPS-1 node.

This research is sponsored by the Defense Advanced Research Project Agency (DARPA) Power Aware Computing/Communication Program and the Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-00-2-0551; and by the Army Research Laboratory (ARL) Collaborative Technology Alliance, through BAE Systems, Inc. subcontract RK7854. Rex Min is funded by an NDSEG Fellowship, Manish Bhardwaj is funded by an IBM Fellowship, and Alice Wang is funded by a Lucent Graduate Research Fellowship.

References

[1] B. Warneke, B. Atwood, and K. S. J. Pister, "Smart Dust Mote Forerunners," *Proc. 14th IEEE Int'l. Conf. MEMS*, Jan. 2001, pp. 357–60.
[2] J. Rabaey *et al.*, "PicoRadio Supports Ad Hoc Ultra-low Power Wireless Networking," *Comp.*, vol. 33, no. 7, July 2000, pp. 42–48.
[3] G. Asada *et al.*, "Wireless Integrated Network Sensors: Low Power Systems on a Chip," *Proc. 1998 ESSCIRC*, Sept. 1998.
[4] F. Op't Eynde *et al.*, "A Fully-integrated Single-chip SOC for Bluetooth," *Proc. ISSCC 2001*, Feb. 2001, pp. 196–97, 446.
[5] A. Wang *et al.*, "Energy Efficient Modulation and MAC for Asymmetric RF Microsensor Systems," *Proc. ISLPED 2001*, Aug. 2001.
[6] R. Powers, "Advances and Trends in Primary and Small Secondary Batteries," *IEEE Aero. Elec. Sys. Mag.*, vol. 9, no. 4, Apr. 1994 pp. 32–6.
[7] J. Goodman, "An Energy-efficient IEEE 1363-based Reconfigurable Public-key Cryptography Processor," *IEEE ISSCC 2001*, Feb. 2001, pp. 330–31, 461.
[8] V. De and S. Borkar, "Technology and Design Challenges for Low Power and High Performance," *Proc. ISLPED 1999*, Aug. 1999, pp. 163–68.
[9] R. Min, T. Furrer, and A. Chandrakasan, "Dynamic Voltage Scaling Techniques for Distributed Microsensor Networks," *IEEE Comp. Soc. Wksp. VLSI*, Apr. 2000, pp. 43–46.
[10] M. Bhardwaj, R. Min, and A. Chandrakasan, "Quantifying and Enhancing Power Awareness of VLSI Systems," to appear in *IEEE Trans. VLSI Systems*, 2001.
[11] S. H. Nawab *et al.*, "Approximate Signal Processing," *Journal VLSI Signal Proc. Sys. Signal, Image Video Tech.*, vol. 15, no. 1/2, Jan. 1997, pp. 177–200.
[12] M. Bhardwaj, "Power-Aware Systems," Master's thesis, MIT, May 2001, pp. 75–95.

[13] W. Rabiner Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Routing Protocols for Wireless Microsensor Networks," *Hawaii Int'l Conf. Sys. Sciences (HICSS '00)*, Jan. 2000, pp. 3005–14.
[14] K. Yao *et al.*, "Blind Beamforming on a Randomly Distributed Sensor Array System," *IEEE JSAC*, vol. 16, no.8, Oct. 1998, pp. 1555–67.
[15] R. Amirtharajah *et al.*, "A Micropower Programmable DSP Powered Using a MEMS-based Vibration-to-electric Energy Converter," *Proc. ISSCC 2000*, Feb. 2000, pp. 362–3, 469.

BIOGRAPHIES

REX MIN (rmin@mtl.edu) received his S.B. and M.Eng. in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT) in 1998 and 1999, respectively, and continues toward a Ph.D. at MIT. His research interests center on the power-aware design of hardware and software for large-scale embedded systems, with an eye toward emerging wireless applications. He is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi, and is a National Defense Science and Engineering Graduate (NDSEG) Fellow.

MANISH BHARDWAJ (manish@mtl.mit.edu) received a B.A.Sc. (honors) degree in computer engineering from Nanyang Technological University, Singapore in 1997 and an S.M. degree in electrical engineering and computer science from MIT in 2001. From 1997 to 1999 he worked as a digital integrated circuits designer and later as a DSP architect with the microelectronics research and development arm of Siemens Semiconductors (now Infineon Technologies). He is currently working for Engim Inc. based in Acton, Massachusetts, on a leave of absence from the Ph.D. program at MIT.

SEONG-HWAN CHO (chosta@mtl.mit.edu) received a B.S. degree from the Korea Advanced Institute of Science and Technology in 1995, and S.M. and Ph.D. degrees from MIT in 1997 and 2002, where he worked on an energy-efficient RF communication system for microsensor networks. He is currently with Engim Inc., where he is investigating analog circuit techniques for RF and IF filters.

NATHAN ICKES (nickes@mtl.mit.edu) received his S.B. and M.Eng. in electrical engineering and computer science from MIT in 2001 and 2002, respectively. Presently a Ph.D. candidate at MIT, his research interests include low-power and power-aware computing architectures and operating systems.

EUGENE SHIH (eugene@mtl.mit.edu) is a third-year graduate student at MIT. He has a B.S. degree in computer engineering from the University of Washington, Seattle (1998) and an S.M. degree in electrical engineering and computer science from MIT (2001). His research interests include mobile computing, wireless communication, and ubiquitous computing systems.

AMIT SINHA (sinha@mtl.mit.edu) received his S.M. and Ph.D. degrees in electrical engineering and computer science from MIT in 1999 and 2001, respectively. He received his B.Tech. in electrical engineering from the Indian Institute of Technology, Delhi, where he was awarded the President of India Gold Medal for graduating summa cum laude. His research interests include low-power systems and software design with emphasis on wireless systems. He is currently involved with Engim, a wireless broadband startup.

ALICE WANG (aliwang@mtl.mit.edu) received her S.B. and M.Eng. in electrical engineering from MIT in 1997 and 1998, respectively. She is currently at MIT working toward her Ph.D. in the design of an ultra-low-power subthreshold digital signal processor for wireless sensors. Her research interests include energy-efficient implementation of signal processing algorithms, low-power DSPs for wireless systems, and subthreshold circuit design. She was a Lucent Technologies Fellow and is currently an Intel Fellow.

ANANTHA CHANDRAKASAN (anantha@mtl.mit.edu) received his Ph.D. degree in E.E.C.S. from U.C. Berkeley in 1994. He is an associate professor of electrical engineering and computer science at MIT. He has received numerous awards and has served on the technical program committees of various IEEE and ACM conferences. He is technical program chair for ISSCC 2003. His research interests include the energy-efficient implementation of DSPs and wireless microsensor networks.

Perhaps it is not far fetched to envision that the wireless microsensor network will be the true enabler for ubiquitous computing: the availability of computational power that is taken for granted anywhere, at any time.