Disentangling Wireless Sensing from Mesh Networking

Thomas Schmid[†‡] Roy Shea[‡] Mani B. Srivastava[‡] Prabal Dutta[†]

[†]Computer Science & Engineering Division
University of Michigan
Ann Arbor, MI 48109
{thschmid, prabal}@umich.edu

[‡]Electrical Engineering Department
University of California, Los Angeles
Los Angeles, CA 90095
{rshea, mbs}@ucla.edu

ABSTRACT

The resource demands of today's wireless mesh networking stacks hinder the progress of low-cost, low-power wireless sensor nodes. Optimizing wireless sensors means reducing costs, increasing lifetimes, and locating sensors close to the action. Adding mesh networking functions like IP routing and forwarding increases RAM and ROM requirements and demands substantial idle listening to forward others' traffic, all of which adds cost and increases power draw. We argue that an architectural separation between *sensor* and *router*, similar to what ZigBee and traditional IP networks advocate, would allow each node class to be better optimized to the task, matched to technology trends, and aligned with deployment patterns. Although trivial to implement on current platforms, for example by turning off router advertisements in an IPv6/6LoWPAN stack, reaping the full benefits of this approach requires evolving platform designs and revisiting the link and network layers of the stack. We examine the resulting implications on the system architecture.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Design, Performance

Keywords

Sensor Networks, Energy Harvesting, System Architecture

1. INTRODUCTION

Wireless sensor networks, or *sensornets*, have long conflated the tasks of wireless sensing and mesh networking. In this paper, we argue that the compromises arising from this union, on both the physical sensor node and the logical network architecture, results in unnecessary cost and complexity, and so a cleaner separation is needed.

An influential argument for combining wireless sensing and mesh networking focused on the potential energy gains from multihop communications. Rabaey et al. observed that the energy, E_{tx}, necessary to communicate strongly relates to the distance, d, between the endpoints

$$E_{tx} \sim d^{\alpha},$$

where α is the path loss [12]. For indoor networks, α is usually between 2 and 4. This observation revealed that the transmission cost is dramatically reduced when multiple short hops are used for communications, rather than a single long hop. Sensornet researchers hoped to exploit this observation by integrating sensing and routing: dense fields of wireless sensor nodes would collect data about the environment, but individual sensors would also forward data on behalf of their neighbors.

Early protocols that leveraged mesh networking showed promise [5, 11] but the power draw of sensornet nodes still remained high [13]. While mesh networking reduced transmission costs, it created a new problem in that nodes now had to listen, often idly, to determine whether any neighbors needed their traffic forwarded. Dealing with idle listening has shaped much of the low-power wireless research agenda, and has led to duty-cycled radios, scheduled listening, preamble sampling, low-power listening, wakeup radios, and many other mechanisms. Fundamentally, however, we just traded one problem for another: idle listening, rather than transmission power, has since dominated power budgets.

Of course, mesh networking is about more than just energy efficiency. Meshing provides spectrum reuse, allowing concurrent transmissions, and improves connectivity, allowing greater data reliability. However, the argument for spatial reuse falls apart under closer examination. Typical workloads eschew data aggregation and instead require full data collection, making the sink the network bottleneck. Similarly, improving connectivity through meshing is necessary when sensors are deployed in burrows, bridges, and battlefields, where power and network is limited, and RF propagation is challenged, but less so in buildings with ample power and connectivity.

The recent emergence of small, embeddable, and batteryless energy-scavenging sensor nodes further exacerbates the demands of a mesh. How should nodes that might lose time and memory when power disappears communicate with other low-power systems? What is the impact of power intermittency on networking, if every node is also a router?

HotEmNets'10, June 28–29, 2010, Killarney, Ireland

And how much are we paying, in dollars and Joules, to support the features of a mesh that might be unnecessary? These developments raise the question of how low-power mesh networks should evolve to support energy-scavenging leaf nodes – an open area of research today.

Finally, we note that modern mesh network stacks, like IPv6/6LoWPAN and ZigBee, require significant and costly resources. The Arch Rock 6LoWPAN stack requires 23.5 kB of ROM and 3.5 kB of RAM [8] and the TI ZigBee Z-Stack requires 99 kB of ROM and 3.8 kB of RAM [14]. Although a standards-based mesh network stack on *every* node provides great deployment ease [9], it also requires more capable processors, hindering evolution to smaller, more efficient, and less costly platforms.

In light of the hefty energy toll that both radio transmission and idle listening place on nodes, along with the greater power draw and monetary cost of processors capable of supporting mesh network stacks, we argue that it is time to revisit the decision to shoehorn a router into every sensor in a network. A distinction between edge devices and routers is time-tested and common in many other networked domains. Traditional IP networks, for example, have routers and end hosts. Likewise, ZigBee networks have routers and end devices. The sensornet community's adherence to the principle that every node should also be a router has run its course. *It is time to disentangle wireless sensing from mesh networking.*

In this paper, we take these realities to heart and argue for removing all but the most fundamental networking capabilities from edge sensing devices in sensornet deployments. This creates a heterogeneous architecture of *wireless routers* that perform mesh networking amongst themselves and manage communication with neighboring *wireless sensors* that sense, store, process, and send, but do not forward, data. Although we use the term wireless router, this does not preclude the router from sensing itself. Rather, a *wireless router* is simply any node that forwards messages from other nodes, without any assumption or restriction on its own sensing abilities.

We evaluate the proposed architectural separation into at least two device classes by examining the impact of using a very simple asymmetric link-layer Medium Access Control (MAC) on the wireless sensors. In our proposed architecture, the more capable wireless routers connect to clusters of wireless sensors using an asymmetric wireless sensor MAC (WS-MAC), and the wireless routers network with each other using standards-based routing protocols like the IETF's ROLL [1] or the ZigBee network layer, running over a standard IEEE 802.15.4 link layer.

An asymmetric link layer provides substantial energy savings over the current state-of-the-art. The limited demands of an asymmetric MAC, coupled with a networking layer that does not route, allows wireless sensors to use inexpensive microcontrollers. The cost savings due to the simpler wireless sensors can be redirected to improving the features and performance of wireless routers.

These gains can be realized today by embracing the network topologies that the past decade of sensornet research has revealed, and that current microcontrollers already offer. The real research challenges, however, arise when we attempt to integrate the emerging class of energy-scavenging wireless sensors into the existing mesh of low-power wireless routers.

2. CONTEMPORARY ARCHITECTURES

General-purpose, standards-based wireless network stacks have emerged as an important aspect of modern sensornets, especially as microcontroller memories increase in size to accommodate the demands of routing and forwarding. Recent implementations of IPv6/6LoWPAN in wireless sensor networks [8] demonstrated low radio duty cycling and latency characteristics, achieving an average duty cycle of 0.65% and per-hop latency of 62 ms, while maintaining a compact stack size with high reception rates. This is an attractive solution for sensornet nodes, and one that offers easy integration with other IPv6 networks and features, such as auto-configuration, that ease system deployment and evolution. While we argue that the full 6LoWPAN extracts too heavy a memory and code overhead for our wireless sensors, our proposal still adopts key features of the 6LoWPAN architecture. In particular, we propose to use 6LoWPAN, or a similar network, to create the backbone mesh between routers.

We adopt the ZigBee standard's decision to differentiate end devices (which we call wireless sensors) from routers and mandate that all communications to or from such an end devices be initiated by the end device itself. Although the ZigBee standard has not achieved much traction in the academic sensornet community, the received-initiated communications that ZigBee proposes for "sleepy end devices" is architecturally essential for achieving the low power and low complexity we seek. This asymmetry allows for the extremely low duty cycles for wireless sensors needed to support intermittently-powered sensor operation. This asymmetry also results in a simpler network stack on wireless sensors, which are relieved of routing and buffering concerns.

We are not the first to differentiate sensing from other concerns. Tenet proposes a basic tasking language to control simple sensors that locally generate data that is then processed by more powerful devices with greater computing and communications resources [6]. Our proposed separation has similar motives, but we propose a lower sensing tier rather than a higher networking and processing tier. Optimizing the lowest sensing tier is critical for reducing the cost and increasing the span of sensors. Current sensornet nodes are already capable of providing an adequate mesh networking tier using today's hardware.

3. MONETARY COST OF MESHING

The radio driver and networking stack on current sensornet nodes consume a significant portion of the application memory and code footprint. For example, the radio accounts for 32.2% (5208 bytes) of ROM and 38.0% (158 bytes) of RAM used within the TinyOS 2.x Oscilloscope application. These numbers are even higher in IPv6 enabled applications. The 6LoWPAN stack developed by Arch Rock for the CC2420 radio requires 3149 bytes of ROM and 272 bytes of RAM for the radio driver, and an additional 1678 bytes of ROM and 29 bytes of RAM for media access control [8]. This does not even include the IPv6 stack, which adds an additional 7298 bytes of ROM and 2466 bytes RAM.

While these figures are small compared to those for handheld, laptop, or desktop class machines, they are nonetheless substantial for microcontrollers. Nearly every major sensornet node used the largest memory and code available in a microcontroller at the time of its design, and many seri-

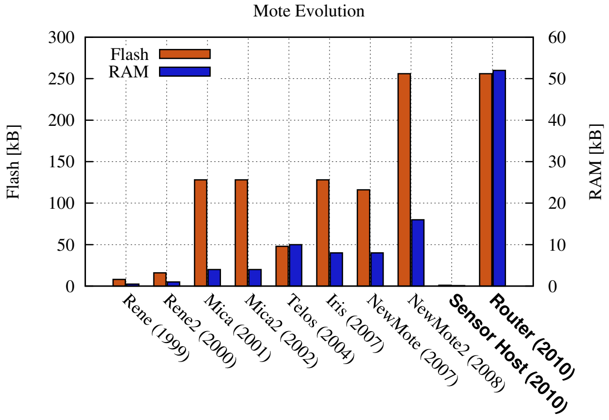

Figure 1: Node resource trends over the past decade and our proposed split into two new distinct classes of devices: wireless sensors and wireless routers. NewMote and NewMote2 are hypothetical platforms based on microcontrollers that would have been chosen had they been available at design time for the Epic Core [3]. Wireless sensors require fewer resources than a decade-old node, while wireless routers require more than today's typical nodes.

ous applications quickly exhausted the hardware capabilities. Figure 1 illustrates the growth in processor complexity observed in sensor node platforms over the past 10 years. Much of this increase has been motivated by ROM and RAM pressures originating from large network stacks. The insatiable demand for memory and code can only be addressed, we believe, through a different architectural decomposition.

Figure 2 illustrates the monetary cost of a range of Texas Instrument MSP430 microcontrollers as a function of available RAM and flash size. Early sensornet researchers argued for sensor nodes that would cost under \$1 by 2005 [7, 12]. Examining microcontrollers that are within a factor of two of this overdue target reveals an upper limit of 10 kB flash and 512 B of RAM – a limit substantially smaller than either 6LoWPAN or ZigBee require. The least expensive microcontrollers in the MSP430 line to support the larger 6LoWPAN and ZigBee stacks are the recently released F5xxx family that start at a base cost of \$4, four times the \$1 target platform cost.

Moore's Law predicts a doubling of transistor density every two years while Bell's Law predicts the emergence of a new computing class every decade. According to Moore's Law, powerful wireless routers (e.g. based on the ARM Cortex-M3) will soon be available for the cost of today's sensor nodes while Bell's Law suggests that a new tier of inexpensive sensors will emerge (e.g. energy-scavenging wireless sensors). But in addition to monetary cost, there is also energy cost. By removing idle listening and mesh networking from wireless sensors, we can use less costly microcontrollers and also significantly improve energy efficiency, asymptotically approaching sampling power-proportional energy consumption.[1]

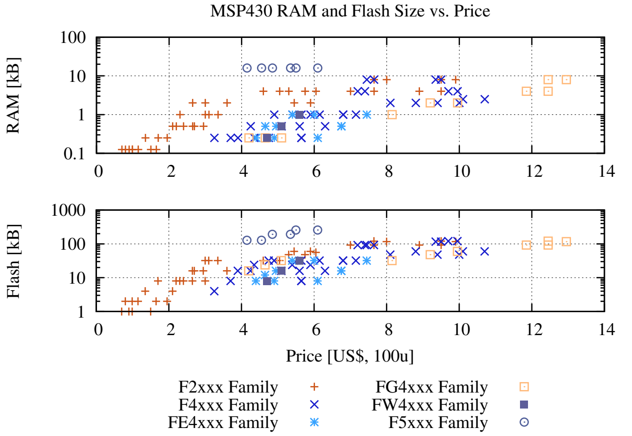

Figure 2: The cost of different Texas Instrument MSP430 microcontroller configurations as a function of the RAM (top) and flash (bottom) sizes. The data were obtained from TI's website in July 2009 for 100 piece pricing.

4. ENERGY COST OF MESHING

The energy consumed by idle listening in sensor nodes becomes significant for low duty-cycle systems. However, idle listening is essential in order to receive messages from other nodes that need to be forwarded along to the next hop. Therefore, we propose an asymmetric system with very simple wireless sensors sitting alongside capable routers. The advantage of this asymmetry is that low-power wireless sensors that need not pay the cost of idle listening and can be positioned close to the physical phenomena that needs to be sensed while more capable routers can be positioned in environments with greater energy scavenging or mains power potential.

This asymmetry demands a new MAC design to exploit the capabilities of our proposed tiers, and aggressively minimize the complexity and radio usage of the lowest tier nodes. Many common sensornet design problems disappear as a result of developing the MAC to function in the constrained setting of wireless sensors.

To simplify the design, and minimize radio usage, on bottom tier devices, our proposed wireless sensor MAC (WS-MAC) only supports initiation of data transfers from the wireless sensors themselves. Wireless sensors can initiate transfer at any time by sending a data packet or polling for incoming data. The router is responsible for listening for and managing these requests as they appear. This can be accomplished by the router in a power efficient manner through a combination of sampled listening and scheduling to emulate an always on link.

Wireless sensors do not support connections initiated by the router or other sensors, so they never need to spend energy idling their radio in receive mode. This provides a significant reduction in average power draw, since passive listening for transmissions in even the most recent communication protocols [8] consumes a significant fraction of the power budget. Messages destined to a wireless sensor are placed in a message queue on a neighboring router. Upon reception of a poll or data packet from a wireless sensor with pending receive messages, the router instructs the wireless sensor to stay awake and receive data using a pending bit in the ACK, similar to what was previously proposed and now

---

[1] By sampling power-proportional, we mean that the communication energy is proportional to the sensor sampling rate, i.e., at 0 Hz a wireless sensor consumes $\sim$ zero energy.

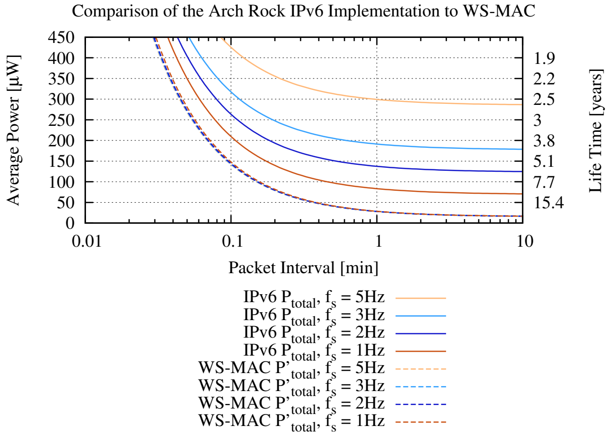

Figure 3: Average power draw and duty-cycling comparison of Arch Rock's 6LoWPAN implementation and WS-MAC. WS-MAC offers substantially longer life by transferring the idle listening burden to wireless routers that are better positioned to capitalize on local energy sources.

supported in hardware [2].

Following is a sample calculation based on the Arch Rock 6LoWPAN implementation that shows the impact that removing idle listening has on the average power draw. A 6LoWPAN leaf node that does not receive any messages has an average power draw of

$$P_{total} = P_{sleep} + P_{listen} + P_{tx}.$$

Because of idle listening, P_{listen} contains

$$P_{listen} = f_{sample} \cdot E_{sample},$$

where f_{sample} is the channel sample frequency and E_{sample} the channel sample energy cost. In the optimal case the leaf node knows the polling schedule of the next hop, and thus only unicast messages have to be sent. Assuming these unicast messages are sent at a rate of f_{txu}, the transmit power becomes

$$P_{tx} = f_{txu}\left(E_{tx} + E_{cb} + E_{cd}\left(2 + \frac{f_{\Delta}}{f_{txu}}\right)\right),$$

where E_{tx} is the transmit energy, E_{cb} is the chirp baseline, and E_{cd} the chirp delta energy (see [8] for details).

With our proposed WS-MAC, a wireless sensor does not need to listen and the average power draw becomes

$$P'_{total} = P_{sleep} + f_{txu} \cdot \left(E_{tx} + E_{cb} + E_{cd} \cdot \frac{1}{f_{sample}}\right).$$

The effect of this difference in average power draw is shown in Figure 3, calculated using the energy costs from [8] and summarized in Table 1. WS-MAC approaches the theoretic optimum, only consuming energy if a message actually has to be sent. This allows WS-MAC to have a 90% energy reduction for a polling interval of 5 Hz, or a 60% reduction at a polling interval of 1 Hz. Translated into node lifetime, the WS-MAC node will run more than 15 years at a data message interval of 30 seconds, while the regular IPv6 node will last for 2.5 years at a 5 Hz polling interval and the same data message rate, or 7.5 years at a 1 Hz polling interval.

While this asymmetric protocol places a disproportionate burden on routers, required scheduling and radio duty cycling are well within the router's computational capabilities and power constraints. More importantly, this architecture provides significant flexibility in deployment. The router node can now be carefully placed in a region of better energy harvesting potential or near wired power sources and wireless sensors can be placed to optimize sensing of the underlying phenomena of interest.

Primitive	Cost	Primitive	Cost
E_{sample}	54 μJ	E_{cd}	46 μJ
E_{tx}	630 μJ	P_{sleep}	15.3 μW
E_{cb}	119 μJ	f_{Δ}	±20 ppm

Table 1: Cost per primitive. This data is from [8].

5. BATTERY POWERED MESH, WITH ENERGY SCAVENGING LEAVES

The energy demands of message forwarding and idle listening can be significant, if not prohibitively high, for an emerging class of sensor nodes. We expect that in the near future, ultra-low power sensors will run forever (or at least for times not bound by the initial power store) by scavenging energy from their environments. Unfortunately, scavenged energy is often only intermittently available, so these new devices will need to deal with periods without power, and even deep sleep might not be supported by scarce energy resources, leading to possible state and synchronization loss.

While conventional thinking might discourage the use of such intermittently powered systems, this need not always be the case. In some situations the energy harvesting mechanism can be directly linked with the observed phenomena and thus energy is guaranteed to be available when the node must sense. Examples of such systems are piezoelectric powered door sensors, electric power meters that scavenge energy from the EM field of current flow, and HVAC air flow meters that scavenge energy from vibration or small wind turbines.

WS-MAC provides a natural way to deal with intermittently powered wireless sensors. Sensors using WS-MAC need not wake up on regular intervals and they need not worry about losing routing information should they completely lose power. Rather, these sensors can simply wake up and send their data when power is available.

One disadvantage of our proposed highly asymmetric WS-MAC protocol is the possible latency of messages going to the wireless sensor. If a wireless sensor aggressively duty cycles, then messages could take seconds, if not minutes, before they reach the device. Fortunately, in most applications, low latency is only important for messages from the sensor to a fusion center, since usually only delay tolerant re-configurations flow back to the sensors. One notable exception is actuator and control systems that require a tight control loop. In such systems, WS-MAC can still use the channel polling strategy at the cost of increased average power.

6. MESH NETWORKING OVERHEAD

One could argue that separating nodes into routers and sensors will complicate deployments since one must now ensure an interconnected router backbone with wireless sensors located within reach of the backbone. Yet this architecture has already succeeded in other domains, such as the large scale commercial WiFi solution provided by Meraki's cloud mesh networking infrastructure [4]. In the Meraki network,

Network	# Nodes	E[# Routers]
ACMe[9]	50	13
Motelab[15] Floor 1	56	4
Motelab Floor 2	74	4
Motelab Floor 3	54	4

Table 2: This table shows the number of routers necessary to connect all the nodes in the particular testbed. Only a few routers are needed, even for moderate-sized networks, suggesting that optimizing the sensors makes economical and architectural sense.

wireless access points automatically build a mesh network with each other while end computers connect as leaf devices. Meraki showed that this type of infrastructure can be successful, and that once the mesh backbone exists, adding leaf nodes becomes trivial.

Sensor networks can be deployed in a very similar manner. While dense in today's deployments, only a few key routing nodes are necessary to create a connected mesh network. We analysed the network graph of two large sensor network deployments, the ACMe deployment at UC Berkeley [9] spanning several floors of the Computer Science building, and the Motelab testbed at Harvard [15]. We analyzed each of the three floors individually, since no floor-to-floor connectivity graphs are available. Table 2 shows the number of total nodes within the network and the number of routing nodes necessary to interconnect all the nodes. The data shows that we vastly over provision sensor network nodes by adding mesh-network capabilities into each and everyone of them.

6.1 Engineering Considerations

The reduced cost and high ratio of wireless sensors to routers provides monetary savings that can be applied to more expensive and capable router hardware. For example, one can extend routers with low noise amplifiers, power amplifiers, or antenna diversity using directional, sectorized, or constellation antennas to vastly reduce path loss and improve connectivity [10]. Routers can also use multiple radios, providing the potential for heterogeneous wireless sensors capable of choosing cheaper, low-power radios as used in the mesh network infrastructure.

Since the routers are not closely tied to the sensed phenomena, they can be installed in locations with rich energy harvesting potential, supplied a bulky energy source, or even attached directly to a source of mains power. While the wireless sensors are still restricted in placement due to their coupling with the sensed phenomena, the small processors they use are more energy-efficient given the low duty cycles observed in typical sensornet deployments for sampling slowly-changing phenomena.

7. CONCLUSION

We propose to create a new tier of parsimonious, cold-blooded wireless sensors that are free from the shackles of idle listening and mesh networking. Relieved of these burdens, the nodes can achieve sampling power-proportionality on memory and code resources smaller than those seen in a decade. This will take us one giant step closer to realizing the original vision of Smart Dust.

8. ACKNOWLEDGMENTS

This material is supported in part by the U.S. ARL and the U.K. MOD under agreement W911NF-06-3-0001, NSF awards CCF-0820061 and CNS-0964120, a University of Michigan Research Incentive Award, and UCLA's NSF CENS. It is also supported in part by a Swiss National Science Foundation Fellowship and a Microsoft Research Graduate Fellowship. Any opinions, findings and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the listed funding agencies. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

9. REFERENCES

[1] M. Dohler, T. Watteyne, T. Winter, and D. Barthel. Routing Requirements for Urban Low-Power and Lossy Networks. RFC 5548 (Informational), 2009.

[2] P. Dutta, D. Culler, and S. Shenker. Procrastination might lead to a longer and more useful life. *HotNets-VI*, 2007.

[3] P. Dutta, J. Taneja, J. Jeong, X. Jiang, and D. Culler. A building block approach to sensornet systems. In *SenSys*, 2008.

[4] J. Eisner. Meraki white paper: Meraki hosted architecture. White paper, Meraki, Inc., 2009.

[5] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *MobiCom*, pages 263–270. ACM New York, NY, USA, 1999.

[6] O. Gnawali, K.-Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, and E. Kohler. The tenet architecture for tiered sensor networks. In *SenSys*, pages 153–166, New York, NY, USA, 2006. ACM.

[7] J. Hill. *System architecture for wireless sensor networks*. PhD thesis, University of California, 2003.

[8] J. W. Hui and D. E. Culler. Ip is dead, long live ip for wireless sensor networks. In *SenSys*, New York, NY, USA, 2008. ACM.

[9] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler. Design and implementation of a high-fidelity ac metering network. In *IPSN*, 2009.

[10] R. Ketcham and J. Frolik. A low-complexity, compact antenna for mitigating frequency-selective fading. In *IPSN*, 2007.

[11] R. Nowak and U. Mitra. Boundary estimation in sensor networks: Theory and methods. In *IPSN*, 2003.

[12] J. Rabaey, M. Ammer, J. da Silva Jr, D. Patel, and S. Roundy. Picoradio supports ad hoc ultra-low power wireless networking. *Computer*, 33(7):42–48, 2000.

[13] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler. An analysis of a large scale habitat monitoring application. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 214–226, 2004.

[14] Texas Instrument. Z-Stack - ZigBee protocol stack. http://focus.ti.com/docs/toolsw/folders/print/z-stack.html, 2009.

[15] G. Werner-Allen, P. Swieskowski, and M. Welsh. Motelab: A wireless sensor network testbed. In *IPSN*, page 68. IEEE Press, 2005.