Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Acquiring High-Rate Neural Spike Data with Hardware-Constrained Embedded Sensors

Shahin Farshchi, *Student Member, IEEE,* Aleksey Pesterev, Wan-Lun Ho, and Jack W. Judy, *Senior Member, IEEE*

***Abstract*—In an effort to enable embedded sensors that are hardware and bandwidth constrained to acquire high-frequency neural signals, signal-filtering and signal-compression algorithms have been implemented and tested on a commercial-off-the-shelf embedded-system platform. The sensor modules have been programmed to acquire, filter, and transmit raw biological signals at a rate of 32 kbps. Furthermore, on-board signal processing enables one channel sampled at a rate of 4 kS/s at 12-bit resolution to be compressed via ADPCM and transmitted in real time. In addition, the sensors can be configured to only transmit individual time-referenced "spike" waveforms, or only the spike parameters for alleviating network traffic and increasing battery life.**

I. Introduction

A major challenge to realizing a remote biological monitoring system is creating the miniature wireless biological sensors that serve as the interface between the test subject and the network infrastructure. These wireless biological sensors must be capable of sensing, amplifying, and transmitting biological signals which range from the order of tens of microvolts to several milivolts, while being non-obtrusive (hence compact) and low-power (for sufficient battery life). The vast majority of the power (often as much as 90%) used by the wireless sensor is dedicated to the radio transmitter for signal transmission. Therefore, the addition of local data-processing capabilities can prolong battery life significantly due to the elimination of the requirement for constant high-throughput wireless data transmission. In addition, on-board signal processing capabilities, with the presence of a receiver, can facilitate user-defined multi-mode operation, thus allowing the researcher to switch between low-power event detection and variable rates of real-time biological signal transmission. Therefore, a system with bi-directional communications in addition to on-board computational abilities would be superior to a simple transmitter/receiver combination. Possible approaches for implementing a wireless biological sensor range from assembling commercial-off-the-shelf PC (COTS-PC) components [1] to custom fabricating integrated circuits [2]. COTS-PC components yield large, power intensive units with powerful communications and signal processing capabilities, while custom integrated circuits yield very specialized, compact, and power-efficient solutions. Unfortunately, investing in the development of custom integrated signal acquisition circuits, multi-channel digital signal processors, transmitters, and receivers for non-standardized applications such as biological signal recording (unlike cellular phones, which operate on strict, national standards) may not be economically feasible.

Manuscript received April 3, 2006. This material is based upon work supported by the National Science Foundation under Grant No. 0456125.

S. Farshchi is with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA (phone: 925-323-2784, fax: 509-463-5485, e-mail: shahin@ee.ucla.edu).

A. Pesterev, and J. W. Judy are with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA (e-mail: {alexp, jjudy}@ee.ucla.edu).

II. Definition of a New Approach

A. Existing Miniature-Scale Neural Recording Systems

A thorough review of existing approaches towards developing wireless biological sensors has been covered in [3]. Although the custom-integrated amplifiers and transceivers demonstrated in [6]-[10] feature very small size (~ 5 to 100 mm^2) and low power consumption (~ 2 to 14 mWs), they do not provide digital signal processing or bi-directional communications (except for the system described in [9], which can modulate the inductive power link as a carrier signal for communication from the base station to the sensor). In addition, large re-integration efforts are required for even minor upgrades and improvements (e.g., channel count, signal bandwidth, etc.). Other miniature biological sensors, which are composed of COTS ICs for the amplifier and the transmitter [11]-[13], have performance characteristics similar to those that use custom ICs, but have a much shorter development time, greater size, more mass, and increased power consumption. Although a wireless neural-recording system based on COTS-PC components with ample signal-processing and bi-directional communications abilities has been demonstrated [1], its size, weight, and power consumption is significantly greater than the aforementioned integrated systems.

B. TinyOS and the Mica-Based Sensor Network

TinyOS [16] is a miniature-footprint operating system that has been designed to operate on hardware-constrained embedded computers (frequently referred to as "motes") [4].

W. Ho is with the Computer Science Department, University of Southern California, Los Angeles, CA 90089 USA (e-mail: wanlunho@usc.edu).

1-4244-0033-3/06/$20.00 ©2006 IEEE.

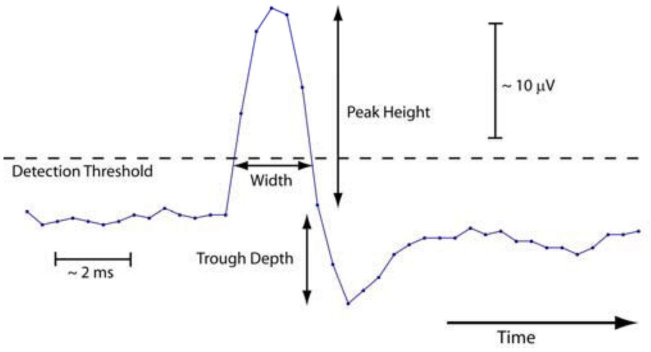

Figure 1: Neural spike parameters of interest (adapted from [25]).

The operating system gives the developer high-level control over low-level hardware. The motes have been used in a variety of low-frequency, distributed sensing applications. The type of mote used in this work is the TelosB mote produced by Crossbow Technology Inc. (San Jose, CA, USA) [14] and Moteiv (El Cerrito, CA, USA) [15], which operates on TinyOS. Data is processed by a microprocessor (MSP430, Texas Instruments, Dallas, TX, USA) with 1 MB of flash memory. The TI MSP430 has 8 analog input channels that are time-multiplexed onto a single analog-to-digital converter (ADC). Data transmission is handled by a ZigBee-compliant (IEEE 802.15.4) 2.4-GHz transceiver (Chipcon CC2420, Oslo, Norway). An antenna embedded on the printed-circuit board is used for wireless communication. When two 1.5-V batteries (Panasonic Industrial AA, Secaucus, NJ, USA) are installed, the TelosB mote becomes approximately the size of a matchbox ($65 \times 31 \times 6$ mm^3). Users have the option of using more compact 3-V batteries that may be more suitable for their application (e.g., coin cells for experiments involving rodents). One TelosB mote, running a signal-acquisition, filtering, and transmission framework [17], has been interfaced with the test subject via a biological interface. A second TelosB mote interfaced with a gateway module, which in this experiment is a laptop (Thinkpad X21, IBM, Armonk, NY, USA) running a modified version of Emstar [18] to emulate a Stargate Gateway (Crossbow Technology Inc., San Jose, CA, USA), wirelessly receives and forwards sensor readings over the network (Ethernet). The gateway module also provides sensors with configuration data for remote adjustment of filter properties.

A basic wireless neural recording, archiving, and hosting system based on embedded sensors has been demonstrated; however, continuous signal transmission limits battery life and spike recording to a single channel [5]. A similar vital sign monitoring system that uses TelosB motes has also been demonstrated in [19]; however, the fastest signal it acquires is a single channel of ECG. The work in this paper has been directed toward investigating computationally-efficient software filters and compression algorithms to enable chronic, multi-channel wireless biosignal recording with embedded sensors that are hardware and bandwidth constrained.

III. Software Designs

A. Neural Signal Filters

To best leverage the limited processing capabilities of the TelosB for the purpose of improving battery life, computationally-efficient filters were designed for each biological signal of interest to minimize the amount of raw data being transmitted from the radio while still providing useful biological data. An excellent example of where sensor-level signal processing can yield great bandwidth and hence power savings, is detecting and classifying single-neuron firings when investigating single-unit activity. Raw neural recording and transmission normally requires a bandwidth in excess of 40 kbps per channel [24]. Numerous methods have been investigated for detecting a neural spike (or discharge of a single neuron) to ease network traffic and bandwidth requirements.

The memory and computational resources required by each spike-detection algorithm vary from requiring powerful desktop PCs [23] to simple analog circuits [22]. Obeid et al. [21] have performed an evaluation of neural spike-detection algorithms, and concluded that for systems with limited computational resources, taking the absolute value of the neural signal before applying a threshold (in combination with a refractory period) is nearly as effective as applying more elaborate energy-based detectors. In addition, basic spike sorting can be achieved by measuring the width and height of each individual spike waveform [24]. The spike features that neuroscientists use to categorize the spikes are illustrated in Figure 1.

For detecting the spikes, a simple spike-filtering algorithm has been designed that continuously buffers the signal until its absolute value exceeds a user-defined number of standard deviations of the baseline noise [22], which is calculated via a sliding-window algorithm, or a user-defined threshold. To avoid noise (such as movement artifacts) from being classified as neural spikes, the height, width, and trough depths of the detected spike (see Figure 1) are also measured against a range of acceptable values pre-determined by the user. If the measured spike parameters fit within these ranges, the spike is accepted by the filter. The data points representing the spike are compressed via adaptive differential-pulse-code modulation (ADPCM) [25], and marked for transmission over the radio. This filtering method provides users with a time-reference record of the individual spike waveforms. A second filter passes the height and width of each spike along with its time of occurrence, which enables the client to statistically categorize the spikes based on its features [24].

Data transmission from the TelosB takes place in packet format. Signal resolution, filter type, and a time reference are included in the packet header. The transmitted data is received by a second TelosB mote which is interfaced with

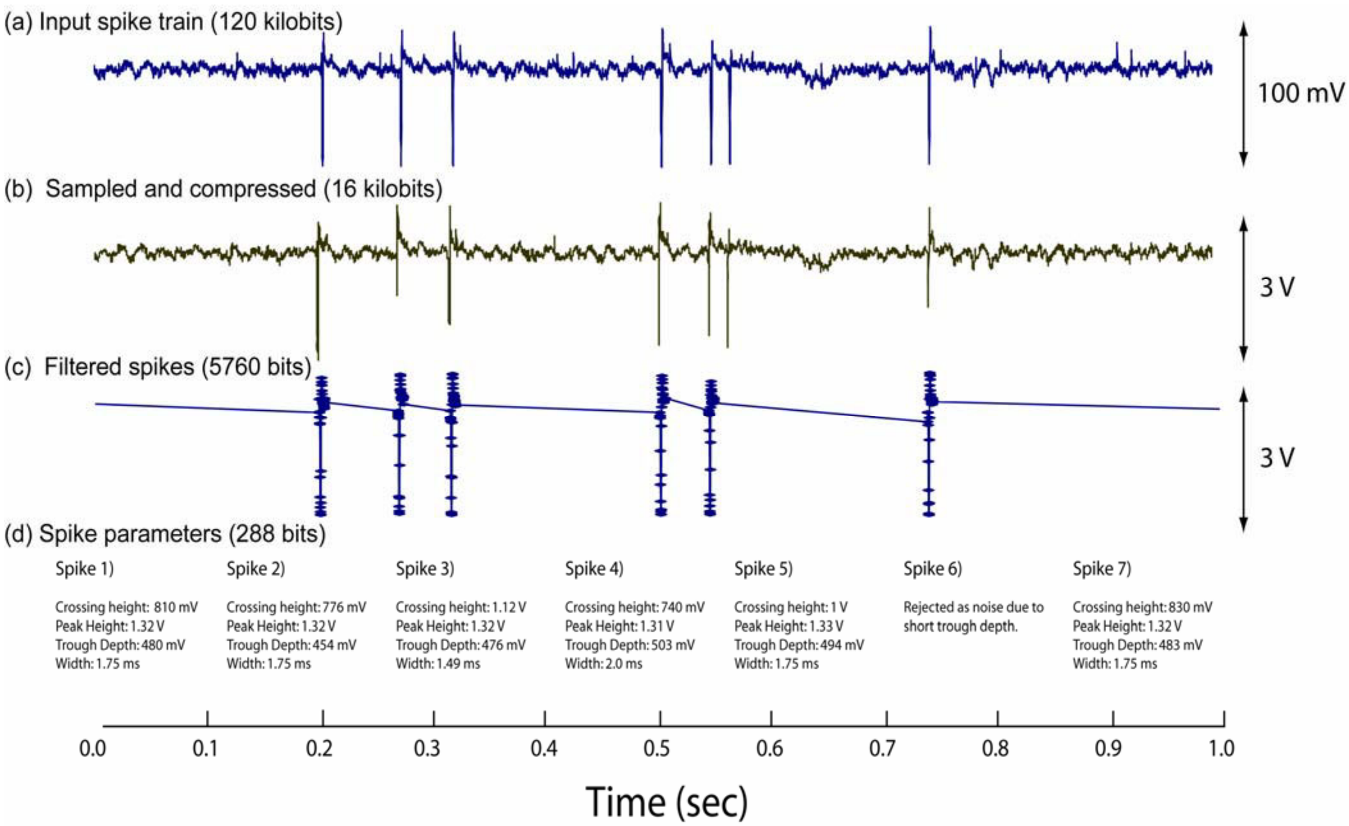

Figure 2: Neural signal (a) applied to the mote, (b) ADPCM-compressed, c) filtered waveforms, (d) spike parameters, and the bandwidth required for transmitting them. To demonstrate the ability of the filter to accept spikes while rejecting unwanted noise (such as motion artifacts, which result in spike-like patterns in the input signal), the filter parameters (i.e., window of acceptable spike heights, widths, and trough depths which are user programmable) were programmed to reject the sixth spike waveform in the dataset—as though it were noise—as it has a very low trough depth.

an Emstar-enabled [18] PC-class device. The gateway module parses and packages the biological signals by type, which are specified in the transmitted packet headers. SQL queries are generated and transmitted via TCP/IP to an archive server.

B. Data Compression

In an effort to further reduce the amount of data being transmitted from the wireless biosensors, several data-compression algorithms were investigated. These algorithms were implemented in C, and tested on the pre-recorded neural dataset to gauge their compression efficiency and required computational overhead. A major challenge set fourth by the motes is their limited memory and computational resources (i.e., the motes cannot perform most transforms). In spite of this, we implemented the following compression algorithms, and compared them to see which would be a suitable candidate for compressing neural spikes on a mote: delta encoding, adaptive delta encoding (ADPCM), Huffman encoding, LZ77, and LZW [25]-[28]. Huffman encoding and LZW are dictionary-based algorithms, while the other algorithms can perform compression in real time. All of the algorithms, excluding ADPCM, are lossless, which means that the uncompressed signal is not distorted as a result of the compression.

IV. SYSTEM TESTING

For testing the spike-filtering characteristics of the system, an arbitrary-waveform generator (33120A, Agilent Technologies Inc, Palo Alto, CA, USA) was programmed with prerecorded spike datasets. The data was originally acquired in vivo from freely moving rats using 5 four-channel MOSFET-input operational amplifiers mounted in the cable connector to remove movement artifacts. Data were recorded wide band (0.1 Hz to 5 kHz) and sampled at 10 kHz/channel (16 channels) with 12-bit precision. Spikes were obtained by applying a high-pass filter with an f_{3dB} frequency of 300 Hz.

To assess the performance and computational overhead of the compression algorithms, each algorithm was applied to the spike dataset. The amount of clock cycles and memory consumed was predicted by counting each function call and calculating the number of operations and memory required to execute each function in the compression algorithm.

V. RESULTS

A. Neural Signal Filters

Figure 2(a) displays the original dataset acquired from an oscilloscope, which was attached to the output of the signal

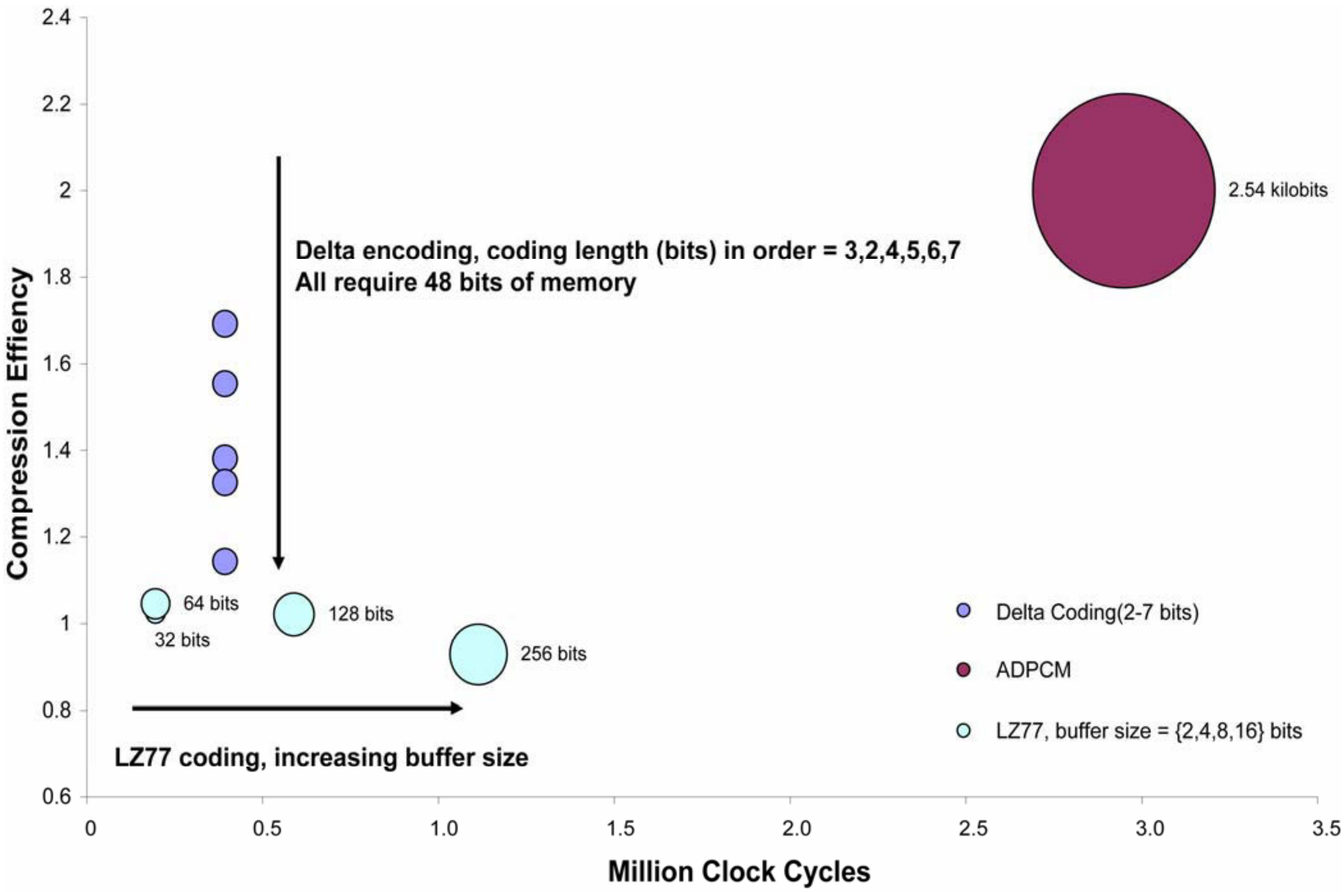

Figure 3: Performance of non-dictionary-based compression algorithms against the spike dataset

generator programmed to output the spike waveforms. Figure 2(b) displays the dataset that was acquired and transmitted by the TelosB mote at 4000 12-bit samples per second followed by ADPCM compression. Figure 2(c) displays the transmitted signal when the mote was programmed to acquire the neural signal at 8000 12-bit samples, and then to detect and transmit time-referenced spikes using the absolute-value-thresholding technique discussed in [21]. To illustrate the ability of the filter to accept spikes in the presence of unwanted noise (such as motion artifacts which could result in signals that resemble spikes), the filter parameters were chosen such that the required signal trough depth exceeded that of the sixth spike; hence, the spike was rejected as though it were unwanted noise. The signal parameters extracted from the spike waveform are listed in Figure 2(d). The amount of data throughput necessary for transmitting each waveform is also labeled in Figure 2. Transmitting the spike parameters only (e.g., spike time, peak height, and trough depth) requires only 48 bits per spike, thus lowering the required bandwidth for transmitting the 1-second signal to only 288 bits. The normalized correlation of the received ADPCM-compressed raw spike signal to the auto-correlated original waveform is over 99%.

B. Data Compression

Figure 3 illustrates the summary performance of the non-dictionary-based compression algorithms on the spike dataset. The number of clock cycles is an estimated value based on the number of mathematical operations that need to be performed for each compression. The size of each bubble indicates the memory required to execute the compression of the dataset. Compression efficiency is the ratio of the size of the original dataset to that of the compressed dataset. It is immediately apparent that delta-encoding requires a constant amount of memory and clock cycles, whether it is 2-bit or 16-bit, which implies that the differences between successive data points are encoded as 2-bit or 16-bit values, respectively. LZ77 yields relatively poor compression efficiency, while requiring a modest amount of computational overhead. ADPCM, which was originally designed for compressing quantized analog waveforms, exhibits the strongest performance, while requiring the most amount of computational overhead with respect to the other non dictionary-based approaches. The computational overhead required by ADPCM, however, can be provided by the TelosB TI MSP430 microprocessor, which is why it has selected in this work.

The dictionary-based compression algorithms require far more resources without a significant benefit in compression efficiency than ADPCM. For example, the LZW algorithms yielded roughly the same amount of compression efficiency as ADPCM while requiring almost triple the RAM and an order of magnitude more clock cycles (with dictionary sizes and maximum word lengths ranging from 300 and 2 to 1000 and 10, respectively). Huffman encoding consumes considerably less memory (less than a quarter) while yielding less compression efficiency (approximately 1.2) as its memory and processor-intensive LZW counterpart, and the more efficient ADPCM compression algorithm.

VI. CONCLUSION

In this work, we have demonstrated high-frequency biological signal acquisition using embedded sensors that are hardware and bandwidth constrained. The wireless biological sensors leverage the signal-filtering capabilities of the COTS TelosB wireless processor modules on which they are based. By applying efficient filters that are designed based on existing methods used for interpreting biological signals, and the ADPCM compression algorithm, power efficiency has been improved by a factor of up to 400, which results in vastly increased battery life, hence lower system maintenance.

ACKNOWLEDGMENT

The authors would like to thank Dr. Anatole Bragin for providing the pre-recorded neural datasets.

REFERENCES

[1] I. Obeid, M. A. L. Nicolelis, and P. D. Wolf, "A multichannel telemetry system for single unit neural recordings," *Journal of Neuroscience Methods*, vol. 133, pp. 123–135, February 2004.
[2] P. Mohseni and K. Najafi, "A battery-powered 8-channel wireless FM IC for biopotential recording applications," *Digest of Technical Papers, 2005 IEEE International Solid-State Circuits Conference*, San Francisco, CA, February 6-10, 2005, pp. 560-561.
[3] S. Farshchi, P. H. Nuyujukian, A. Pesterev, I. Mody, J. W. Judy, "A MICA2-enabled TinyOS-based wireless neural interface," *IEEE Transactions on Biomedical Engineering*, to be published.
[4] J. L. Hill and D. E. Culler, "Mica: a wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, Issue 6, pp.12-24, Nov/Dec 2002.
[5] S. Farshchi, P. H. Nuyujukian, A. Pesterev, I. Mody, J. W. Judy, "A MICA2-enabled TinyOS-based wireless A TinyOS-Based Wireless Neural Sensing, Archiving and Hosting System," *Proc. of the 2nd Int. Conf. of the IEEE EMBS Conference on Neural Engineering*, March 16-19, 2005, Arlington, VA, USA.
[6] K. D. Wise, D. J. Anderson, J. F. Hetke, D. R. Kipke, K. Najafi, "Wireless implantable microsystems: high-density electronic interfaces to the nervous system," *Proc. of the IEEE*, vol. 92, pp. 76-97, Jan. 2004.
[7] H. J. Song, D. R. Allee, and K. T. Speed, "Single chip system for bio-data acquisition, digitization and telemetry," *Proc. of the 1997 IEEE International Symposium on Circuits and Systems*, Hong Kong, June 9-12, 1997, vol.3, pp. 1848-1851.
[8] G. A. DeMichele and P. R. Troyk, "Integrated multichannel wireless biotelemetry system," in *Proc. of the 25th International IEEE-EMBS Conf.*, Cancun, Mexico, September 17-21, 2003, pp. 3372-3375.
[9] J. Parramon, P. Doguet, D. Martin, M. Verleyssen, R. Munoz, L. Leija, and E. Valderrama, "ASIC-based batteryless implantable telemetry microsystem for recording purposes," in *Proc. of the 19th International IEEE-EMBS Conf*, Chicago, IL, Oct. 30 to Nov. 2, 1997, pp. 2225-2228.
[10] P. Irazoqui-Pastor, I. Mody, and J. W. Judy, "Transcutaneous RF-powered neural recording device," in *Proc. of the 24th Annual Conf. and the Annual Fall Meeting of the EMBS/BMES* Conf., Houston, TX, October 23-26, 2002, vol. 3, pp. 2105-2106.
[11] A. Nieder, "Miniature stereo radio transmitter for simultaneous recording of multiple single-neuron signals from behaving owls," *Journal of Neuroscience Methods*, vol. 101, pp. 157-164, Sep. 2000.
[12] S. Takeuchi and I. Shimoyama, "A radio-telemetry system with a shape memory alloy microelectrode for neural recording of freely moving insects," *IEEE Transactions on Biomedical Engineering*, vol. 51, pp. 133-137, Jan. 2004.
[13] M. Modarreszadeh and R. N. Schmidt, "Wireless, 32-channel, EEG and epilepsy monitoring system," in *Proc. of the 19th International Conference of IEEE/EMBS*, Chicago, IL, Oct 30-Nov 2, 1997, pp. 1157-1160.
[14] Crossbow Technology Inc. [Online]. Available: http://www.xbow.com
[15] Moteiv [Online]. Available: http://www.moteiv.com
[16] J. L. Hill, "System architecture for wireless sensor networks," Ph.D. Dissertation, Dept. of Computer Science, Univ. of California, Berkeley, CA, 2003.
[17] Ben Greenstein, Alex Pesterev, Christopher Mar, Eddie Kohler, Jack Judy, Shahin Farshchi, and Deborah Estrin, "Collecting High-Rate Data Over Low-Rate Sensor Network Radios," University of California, Los Angeles Center fo Embedded Networked Sensing Technical Report #55, [Online]. Available: http://research.cens.ucla.edu/pls/portal/url/item/05002645A3A05A96E0406180528D2EF9.
[18] J. Elson, L. Girod, and D. Estrin, "EmStar: Development with High System Visibility," *IEEE Wireless Communications Magazine*, Dec. 2004.
[19] K. Lorincz, D. J. Malan, T. R. F. Fulford-Jones, A. Nowoj, A. Clavel, V. Shnayder, G. Mainland, and M. Welsh, "Sensor Networks for Emergency Response" Challenges and Opportunities," *IEEE Pervasive Computing*, Oct-Dec. 2004.
[20] A. Bragin, J. Engel, C. L. Wilson, I. Fried, G. W. Mathern, "Hippocampal and Entorhinal Cortex High-Frequency Oscillations (100-500 Hz) in Human Epileptic Brain and Kainia Acid-Treated Rates with Chronic Seizures," *Epilepsia*, vol. 40, pp. 127-137, Feb. 1999.
[21] I. Obeid, P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application," *IEEE Transactions on Biomedical Engineering*, vol. 51, pp. 905-912, June 2004.
[22] P. T. Watkins, G. Santhanam, K. V. Shenoy, and R. R. Harrison, "Validation of adaptive threshold spike detector for neural recording," *Proc. or the 26th International Conf. of the IEEE EMBS*, San Francisco, CA, Sept 1-5, 2004, pp. 4079-4083.
[23] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Journal of Comput. Neural Syst.*, vol. 9, pp. R53-R78, 1998.
[24] M. A. L. Nicolelis, "Methods for neural ensemble recordings," CRC Press LLC, 1999, ch. 4.
[25] P. Cummiskev, N. S. Jayant, and J. L. Flanagan, "Adaptive quantization in differential PCM coding of speech." *Bell Syst.Tech. J.*, vol. 52, pp. 1105-1118, Sept. 1973.
[26] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the I.R.E.*, Sept. 1952, pp 1098-1102.
[27] J. Ziv and A. Lempel, "A Universal Algorithm for Sequential Data Compression," *IEEE Transactions on Information Theory*, May 1977.
[28] T. A. Welch, "A technique for high-performance data compression," *Computer*. Vol. 17, pp. 8-19. June 1984.