Power Optimization for Wearable Devices

Haik Kalantarian, Nabil Alshurafa, Mohammad Pourhomayoun, Majid Sarrafzadeh
University of California, Los Angeles
{kalantarian, nabil, mpourhoma, majid}@cs.ucla.edu

***Abstract*—In recent years, several wearable devices have been proposed for monitoring nutrition intake, tracking energy expenditure, and performing activity recognition. Long device lifetimes are critical because frequent battery replacements increase user burden and yield poor long-term compliance rates. Though countless wearable devices have been proposed in recent years with varying sensors and applications, most system flows are generalizable in terms of their major components: sampling, buffering, processing, and transmission. In this paper, we discuss and evaluate energy-efficiency optimizations for wearable devices, using the NIMON nutrition-monitoring necklace as a case study.**

***Index Terms*—Power Evaluation; Wireless Health; Nutrition; Necklace; Wearable Body Sensors;**

I. Background

Evaluating and monitoring of eating habits has a broad range of applications and uses. These range from addressing the needs of ill and recovering patients in clinical environments, epidemiological research for evaluating the relationship between diet and disease, and as an accessory for individuals to identify and address deficiencies in their diet. Non-invasive wearable devices for nutrition monitoring have the capacity to improving dietary outcomes by providing guidance and feedback to users with relative transparency; their chief benefit is their capability of increasing compliance by minimizing impact on an individual's daily routine. These noninvasive devices typically use audio, video, accelerometry, gesture recognition, capacitive sensing, or a combination of these techniques to record information pertaining to food intake while providing feedback to users with respect to the composition of the meal, eating style, or more long term patterns of consumption.

To realize the intended goal of minimizing burden, it is desirable for wearable devices to remain powered for weeks, or months, without interruption. Required nightly charging or frequent coin-cell battery replacement can be considered a burden to the user, which is undesirable for the aforementioned reasons. Therefore, modern wearable devices are carefully designed to minimize power usage, and run for days or even months. For example, the Misfit Shine activity monitor claims a battery life of four months [1]. Another important drawback of wearable devices with high power requirements are their reliance on large batteries, which may impact the comfort, convenience, and aesthetic appearance of the device. Therefore, it is necessary for wearable devices to carefully factor energy efficiency in their design.

The algorithms, sensor selection, system architecture, sample rate, desired accuracy, and intended system outcomes have broad implications on the power usage of a device. The NIMON necklace presented by Kalantarian et al. in [2] describes one such approach to nutrition monitoring, with a particular emphasis on real-time user feedback, comfort, and ergonomics. The focus of this paper is to analyze the effects of different engineering design decisions on the battery life of the NIMON necklace. Though the focus of this paper is largely on the architecture of the NIMON necklace, the discussed issues and tradeoffs are applicable to a host of other wearable devices targeted towards other applications.

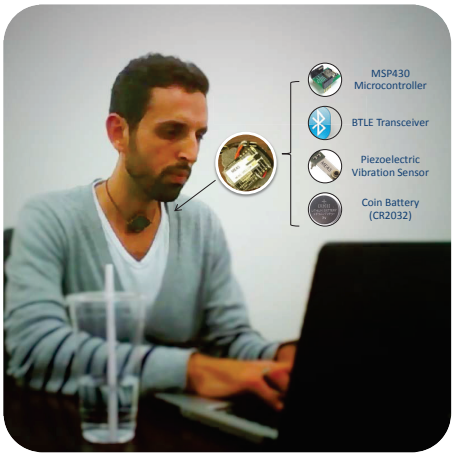

Fig. 1. This figure shows the design and primary components of the NIMON necklace

II. Related Work

The study of nutrition monitoring pertains primarily to the evaluation of different aspects of an individual's dietary habits. These could include estimating the volume of food consumed, the frequency of meals, the number of skipped meals, liquid consumption, food recognition, eating pace, and the amount of chewing. Though countless methods for evaluating food intake have been proposed, two high-level sensor technologies are most promising for real-world use: acoustic monitoring using throat microphones, and piezo-based inertial sensing. In this section, we provide a brief overview of these techniques.

A. Food Ingestion

Piezoelectric sensors are capable of converting mechanical stress to a measurable voltage. Recently, several works have proposed using piezoelectric sensors for detecting swallow events. Typically, a vibration sensor is placed in the lower or mid throat area (to detect swallows) or on the upper jaw (to

978-1-4799-8425-1/15/$31.00 ©2015 IEEE

detect chewing). A recent work by Fontana et al. [3] developed a sensor fusion model using an accelerometer, a jaw motion sensor, and a hand gesture sensor. Furthermore, the work by Kalantarian et. al [2] placed a vibration sensor in the lower trachea and was able to classify solid vs. liquid foods.

Throat-microphones placed in the lower trachea are capable of recording sounds associated with swallowing and chewing, which are identified using various signal processing techniques. Efficient design of these systems can successfully discriminate between chewing, swallowing, other vocalizations, and ambient noises. The work presented by Kalantarian et al. [4] presents one approach to acoustic monitoring of dietary intake using spectrograms. Several similar approaches have been proposed and validated in the last several years [5][6][7].

B. Energy Optimizations in Wearable Systems

Many other works have covered related techniques to increase battery lifetime of wearable devices. For example, in [8], the authors adjust the duty cycle of the accelerometer in real-time, to reduce energy overhead. The SociableSense system described by Rachuri et al. in [9] not only adjusts the duty cycle of various sensors, but also distributes computation between the local device and the cloud.

The work by Krause at al. in [10] evaluates the battery lifetime of a device called the eWatch. In [11], the authors describe power optimizations in a sound-based activity monitor, also using a MSP430-based platform. The sample rate used in their system was 4.8 KHz, which as we later describe is able to capture most of the spectral information of chewing sounds. Another work by Yan et al. in [12] addresses the problem of performing activity recognition with maximum energy efficiency by dynamically adjusting various parameters such as sampling frequency and classifier features to detect the activity with minimum power overhead.

III. An Overview of the Nimon Necklace

The NIMON necklace, presented by Kalantarian et al. in [2], describes a non-invasive, wearable device capable of detecting swallows by placement of a vibration sensor near the lower trachea. An architectural diagram is shown in Figure 1, which provides an overview of system components. Because the necklace is architecturally similar to many other wearables in terms of sensor sampling, buffering, processing, and transmission, it is used as a case study in this paper. During a swallow event, muscular contractions cause skin motion, which pushes the vibration sensor away from the body and towards the fabric of the necklace, generating a unique output voltage pattern. The various steps in data processing are shown in Figure 2, with the raw value visible at the top. The noticeable dips in the waveform generally correspond with swallows. After initial data acquisition, the data is smoothed using a moving-average low-pass filter with a span of 5, to reduce the impact of noise. Subsequently, a sliding window of length 9, corresponding with .45 seconds of data, is applied with a maximum overlap (shifted one point at a time) to compute standard deviation. These numbers were experimentally determined to be optimal for preserving the critical features of the waveform based on simulations.

TABLE I
A List of System Components used in Evaluation

Hardware Components (NIMON Necklace)	Description
Nordic nRF8002	Bluetooth 4.0 Module
Sandisk 4GB MicroSD Card	SD-Card for local logging
MSP430G2744	Microcontroller
CR2032 Coin-Cell	Battery for powering device

TABLE II
Definition of Terms

Term	Description
adc	Operation to convert an analog value to a digital representation
tx	Operation to transmit a reading to a mobile phone for processing
proc	Operation to process the acquired data before transmitting
f	Sample rate of the microcontroller for acquiring sensor data
P	Payload size (in bytes)
MCLK	Main system clock used by the CPU and MSP430 system
SMCLK	Sub-main clock used by other peripherals such as timers
T_{conn}	Connection interval (in ms)
$I_{read\text{-}SD}$	Average current (1s) to read to the microSD card @ 25MHz
$I_{write\text{-}SD}$	Average current (1s) to read to the microSD card @ 25MHz
$I_{sleep\text{-}SD}$	Average sleep current (1s) of the microSD card @ 25MHz

IV. Hardware and Experimental Methodology

Table I presents the hardware components used for power evaluation of the NIMON necklace. The MSP430 has 5 different power modes: Active Mode, and Low Power mode (LP) 1-4. Active mode is used when executing algorithms, while the low power modes supply a clock signal to various peripherals. LP3 disables the CPU, MCLK, and SMCLK. Therefore, the device is configured to alternate between Active Mode and LP3. Note that MCLK and SMCLK are defined in Table II along with other terms used in our work.

The selected Bluetooth transceiver is the Nordic nRF8002, commonly used in several commercial wearables including the Misfit Shine. Power evaluation of the Bluetooth Transceiver was provided by Nordic Semiconductor's nRFGo Studio software (version 1.17), which simulates power demands and battery lifetime based on various parameters. Real-time power debugging of the MSP430 microcontroller was provided by the EnergyTrace++ technology in Texas Instruments Code Composer Studio (CCS) Version 6.0. The SD card used in evaluation is based on datasheets available from the manufacturer. The microSD card is assumed to be operating in default mode at 25MHz, with a capacity of 4GB. The block size is assumed to be limited by the sector size, at 512 bytes.

V. Bluetooth Radio Optimizations

Several parameters can be used to enhance the energy efficiency of the Nordic nRF8002 module. First, the transmit signal strength is adjustable with the following range: output power in connection or advertising can be configured to: -18, -12, -6, 0 dBm. This increases device battery lifetime at the cost of transmit range. A graph showing current consumption for a 100ms connection interval and a 6-byte payload is shown in Figure 3. The graph reveals that modifying transmit power has

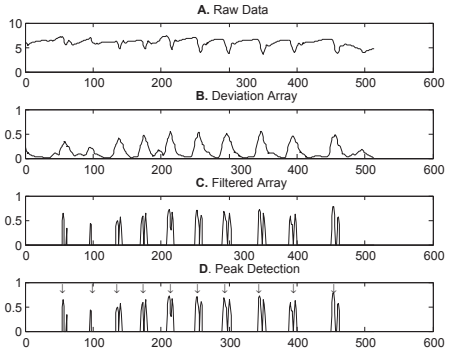

Fig. 2. This graph shows the signal processing flow used by the Nimon necklace to identify swallows.

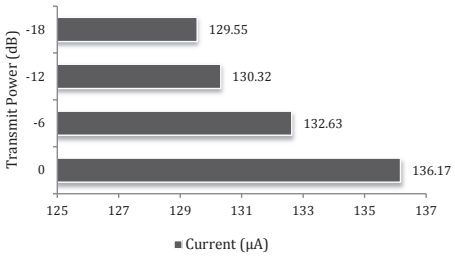

Fig. 3. This chart shows the relationship between transmit power and current consumption for a 100ms connection interval and a 6 byte payload.

a relatively minor effect on decreasing energy consumption. This data suggests that other optimization techniques must be explored to improve battery lifetime.

Secondly, the device can be configured to 'sleep' in low power mode when unused. The average current consuption in sleep mode is 0.50 μA. In 0 dB power mode, the current consumption during the advertisement phase is 23.54 μA and for the connection phase is 16.13 μA, assuming a connection interval and advertising interval set to one second. Therefore, it is critical to set the device into low-power mode whenever data is not being transmitted.Figure 4 shows the relationship between payload size and current consumption with varying connection intervals. This graph reveals that it is significantly more energy efficient to buffer data locally and transmit multiple readings in one payload, due to the overhead of establishing a BTLE transaction. Because the Bluetooth LE protocol packetizes data into 22 byte segments, data beyond 22 bytes will be sent in multiple packets per connection interval, or multiple connection intervals depending on the implementation. Figure 4 suggests that increasing the connection interval is much more critical to power optimizations than decreasing the payload of a packet.

A. Window Size vs. Power

The first algorithm for processing acquired data corresponds with steps 1 and 2 in Figure 2. Figure 7 shows the state transitions of the MSP430 microcontroller. The device waits in LMP3 (Low Power Mode 3), in which MCLK, SMCLK, and MODOSC are disabled to save power. When a timer

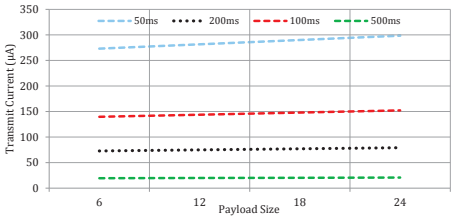

Fig. 4. This chart shows the relationship between payload size and current consumption for varying-sized connection intervals. These values assume a time distribution of 0% advertising, 0% sleeping, and 100% connection.

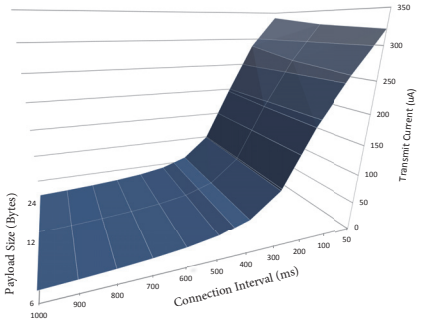

Fig. 5. This chart shows the relationship between payload size, connection interval, and current.

ISR (interrupt service routine) is triggered, the device enters Active Mode, performs the data acquisition and processing, and returns back to LPM3. The software flow on the MSP430 microcontroller is shown in Figure 6. Increasing the window size will increase the amount of computation required per sample, which will require the device to spend a higher percentage of its time processing in Active Mode. In this case, the Window Size parameter can be generalizable to other devices as the amount of local computation required before transmission.

Table III shows the relationship between power usage and window size, at several different sample rates. A key observation is that increasing the window size does not have a substantial effect on power usage, for low sample rates. Also, note that in Table IV, offloading the processing from the necklace to the smartphone decouples the window size parameter from the current usage of the device.

B. Sample Rate vs. Power

Table III shows the relationship between sample rate and power, at several different window sizes. A high sample rate not only incurs processing overhead, but also increases utilization of the on-chip Analog-Digital converter of the MSP430. Table IV shows similar numbers for the case when data processing is offloaded to the mobile phone. In this case, the contributions to the reported power are AD conversion, buffering, and transmission. This table shows that increasing the sample rate between 1 Hz to 32 Hz has a quite limited effect on power usage, since the AD converter is in fact designed for significantly higher sample rates. In comparison

Window Size	Sample Rate				
	1 Hz	**4 Hz**	**8 Hz**	**16 Hz**	**32 Hz**
4	.05 mW	.06 mW	.06 mW	.07 mW	.07 mW
10	.05 mW	.06 mW	.06 mW	.07 mW	.08 mW
20	.05 mW	.06 mW	.06 mW	.08 mW	.10 mW
100	.06 mW	.08 mW	.10 mW	.16 mW	.26 mW

TABLE III
RELATIONSHIP BETWEEN WINDOW SIZE, SAMPLE RATE, AND POWER ON A MSP430 MCU WITH ON-BOARD SWALLOW DETECTION.

Window Size	Sample Rate				
	1 Hz	**4 Hz**	**8 Hz**	**16 Hz**	**32 Hz**
4	.05 mW	.05 mW	.05 mW	.06 mW	.06 mW
10	.05 mW	.05 mW	.05 mW	.06 mW	.06 mW
20	.05 mW	.05 mW	.05 mW	.06 mW	.06 mW
100	.05 mW	.05 mW	.05 mW	.06 mW	.06 mW

TABLE IV
RELATIONSHIP BETWEEN WINDOW SIZE, SAMPLE RATE, AND POWER ON A MSP430 MCU WHEN SWALLOW DETECTON IS OFFLOADED TO THE MOBILE PHONE.

with table III, we conclude that the high sample rate consumes more power largely by increasing the number of data points that must be processed.

C. Connection Interval and Payload

Connection interval, which is how often the BTLE slave device is queried for data, is a function of payload size. This is expected, since less frequent transmissions will necessitate larger payloads, to maintain the same bandwidth. Figure 4 shows the relationship between transmit current and payload size, for four different connection intervals.

Using curve fitting, we find the following equation, which relates transmit current to connection interval and payload size. Figure 5 shows a surface plot, illustrating the relationships between these parameters.

$$I = 70.9 * \frac{P}{T_{conn}} + \frac{7732.7}{T_{conn}^{0.875}} \quad (1)$$

An important observation is that connection interval appears to be a much greater contributor to power use than payload size. Therefore, it is critical to buffer data locally and transmit as infrequently as possible.

D. Benefits of Local Swallow Detection

The NIMON necklace has a sample rate of 10 to 20 Hz; each sample acquired a signal from the vibration sensor and transmitted the raw values (with some metadata) to a mobile phone for processing. The algorithms for determining if a series of samples corresponded to a swallow were originally implemented entirely on a mobile phone, for design simplicity. However, it should be noted that analyzing the data on the necklace could dramatically reduce the number of Bluetooth transmissions, albeit with a higher computational overhead. Because the algorithms run on the mobile device, all data acquired from the vibration sensor must be transmitted. If this algorithm is executed on the embedded device, rather than the smartphone, it will not be necessary to transmit all acquired data from the necklace to the mobile phone. Instead, it will be

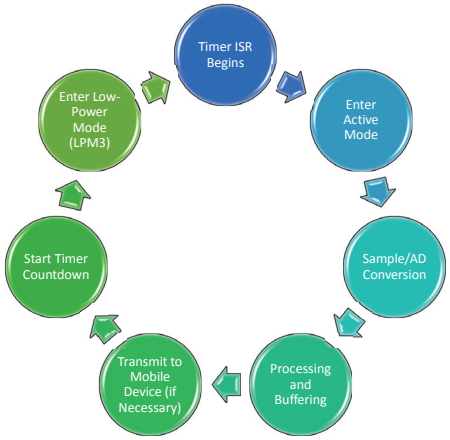

Fig. 6. This chart shows the system flow of the NIMON necklace, on the MSP430 microcontroller.

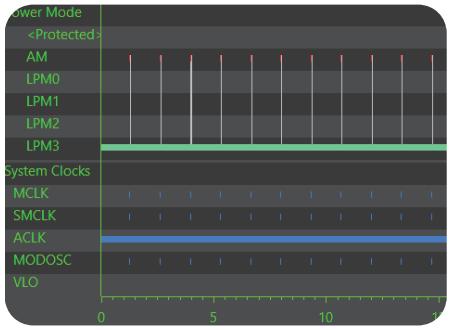

Fig. 7. This figure shows the state transitions of the MSP430 microcontroller, which occur on an interrupt callback. in LPM3, all clocks besides ACLK (necessary for the timer) are disabled.

possible to reduce transmission by locally counting swallows, and periodically transmitting a swallow count. Therefore, an important tradeoff is between computational workload on the necklace vs. minimizing wireless data transfer to the mobile phone. Using simulation and power-debugging software, an evaluation was conducted to determine the optimal tradeoff.

Assuming a sample rate of 16 Hz and a window size of 10, which are both typical numbers for this application, Table III shows that .07 mW of power are required for P_{proc} + P_{ADC}, from which at most .01 mW comes from P_{proc}, the remainder coming from the ADC and from powering other on-chip peripherals. It is then necessary to analyze the benefits of not transmitting each sample, based on equation 1. We assume that each sample requires 6 bytes to transmit, considering meta data and other overhead. We also assume a relationship exists between connection interval and payload, necessary to maintain a fixed bandwidth as defined below in equation 2. This relationship ensures a fixed number of samples per second. Note that payload is in bytes and connection interval in milliseconds. Furthermore, payloads greater than 22 bytes are sent in multiple packets in the BTLE protocol.

$$T_{conn} * P = 300 \quad (2)$$

TABLE V
AVERAGE TRANSMIT POWER FOR A FIXED BANDWIDTH (20HZ) AT 3.7V

T_{conn} (ms)	Payload (bytes)	Avg Current (μA)	Avg Power (μW)
50	6	328.37	1214.97
100	12	260.70	964.59
200	24	195.66	723.94
300	36	99.98	369.92
500	60	41.98	155.32
1000	120	19.90	73.62
3000	360	7.18	26.56

The results for various connection intervals are shown in Table V. Average power ranges from 1.2 mW at a connection interval of 50ms, to 0.026mW at a connection interval of 3 seconds. Compared to the extra .01 mW for performing swallow detection locally, it is clear this is more optimal to run detection algorithms locally for anything responsive enough to be considered real time user feedback (10 seconds or less).

E. Exclusive Local Buffering with SD-Card Logging

One possible optimization in a wearable device is to avoid transmitting data in real-time. An alternative is to simply log the data locally in memory (such as Flash storage), allowing the user to sync their data periodically via PC or mobile phone. This design decision can minimize or eliminate Bluetooth transmissions, though significant energy may be expended writing to Flash storage. In this section, we provide a discussion of this local-storage based architecture based on a 4GB microSD card from SanDisk.

Our prior analysis assumed 6 bytes are required per sample. Because the block size of the microSD card is 512 bytes, approximately 85 samples can be written in one block write. In the datasheet, $I_{write\text{-}SD}$, which is the average current required to write a block to the SD card in standard mode (25MHz), is 100mA. The average time required for a block write is specified as 250 ms. Furthermore, the sleep current of the microSD card is rated at 150μA. Therefore, the power dissipation during a write is (.1A * 3.7V), which is 370mW. The microSD card will be in sleep mode otherwise. Based on the assumption that the block size can support 85 samples, the equation for power dissipation of the SD card reader as a function of sample rate can be found in equation 3. This model assumes that all samples are buffered locally to the SD card one block at a time, and no data is transmitted.

$$P_{microSD} = V * [P_{write-SD} + P_{sleep-SD}]$$

$$P_{write-SD} = \frac{I_{write-SD} * T_{write-SD} * f}{85} \quad (3)$$

$$P_{sleep-SD} = \frac{I_{sleep-SD} * T_{write-SD} * (324 - f)}{85}$$

$$0 \leq f \leq 324$$

The results for various sample rates are presented in Table VI. These results suggest that logging to an SD card may be prohibitive from a perspective of energy efficiency. As the data in Table V shows, the quite high connection interval of 50ms consumes 1.2mw of power, while a more reasonable connection interval of one second reduces power to 73.62 μW. Therefore, the savings of buffering locally rather than transmitting are negligible in comparison to the power required to write to the microSD card and maintain it in sleep mode otherwise.

TABLE VI
POWER DISSIPATION OF THE SD CARD AT VARIOUS SAMPLE RATES

Sample Rate (Hz)	$P_{Write\ (mw)}$	$P_{Sleep\ (mW)}$	$P_{Total\ (mW)}$
1	1.088	0.527	1.615
3	3.264	0.524	3.789
5	5.44	0.52	5.96
8	8.705	0.515	9.221
10	10.88	0.51	11.39
15	16.32	0.50	16.82
20	21.76	0.49	22.26
50	54.41	0.44	54.85

VI. ENERGY EVALUATION OF FOURIER-BASED ALGORITHMS

In this section, we briefly cover the device lifetime challenges using acoustic sensors. The work by Amft et al. in [7] recognized chewing with high precision using a sample rate of 44kHz, which is quite high for an embedded application. This is especially the case when compared to the NIMON necklace in which sample rates as low as 10 Hz have been validated. Though a 44 kHz audio signal will have more information for classification than a much smaller sample rate, the implications of battery life will be substantial. In another work by Amft et al. in [13], audio data was also acquired at 44 kHz and processed with a 512-point FFT. However, prior works [14][15] have shown spectral energy from potato chips to be primarily between 0-10kHz, with highest amplitude frequency ranges between 1 and 2 kHz. Based on Nyquist-Shannon sampling theory, this conservative estimate would require a sample rate of 4 kHz. Though this is well within the specifications of the MSP430 ADC unit, it far exceeds the sample rate of 10-20Hz required by the vibration sensor. The overhead of transmitting up to 400x more data, not to mention the computational overhead of the FFT in comparison to the simple windowing algorithm, makes the vibration sensor a better choice.

A 16-point FFT algorithm was implemented on the MSP430 microcontroller. In a real time application, the FFT algorithm must run periodically to analyze frequency domain features of incoming data. Table VII shows results for four different FFT rates (1, 16, 64, and 128). For example, an FFT rate of 16 represents an operating mode in which a 16-point FFT is evaluated 16 times per second (therefore processing 256 samples per second). The system was designed such that the MSP430 microcontroller would immediately enter LPM3 (low power mode 3) to conserve energy between FFT operations. The table shows that average power is quite high for high FFT

TABLE VII
POWER DISSIPATION FOR A 16-POINT FFT

FFT/Sec.	Mean Power (mW)	Min Power (mW)	Max Power (mW)
1	0.02	0.01	0.03
16	0.13	0.12	0.23
64	0.50	0.48	0.52
128	0.64	0.52	0.68

TABLE VIII
MSP430 RUNTIME OF DIFFERENT MODES DURING FFT

FFT/Sec.	Active	LP3	FPADD	MPYD
1	1.7%	98.3%	0.7%	0.1%
16	17.9%	82.1%	9.2%	2.4%
64	77.8%	22.2%	41.1%	11.2%
128	100%	0%	55.0%	13.2%

rates such as 128 (corresponding with a sample rate of 2kHz); mean power approached 0.64 mW, not counting the overhead of sampling, buffering, and transmission. Table VIII shows that as the frequency of FFT operations increases, the CPU spends a higher percentage of time in Active Mode. At the rate of 128 FFTs/second, the device spends all of its time in Active Mode, and 55% of its time performing a floating point add operation. This suggests that the microcontroller is unable to perform FFT operations at the requested rate, since it should enter LPM3 after completion. At this rate, power dissipation is over ten times greater than that of a vibration-sensor based system with a sample rate of 8 Hz and a window size of 20, as shown in Table III.

VII. CONCLUSION

In this paper, we evaluate the power-efficiency of various design decisions in wearable devices, using the NIMON necklace as a case study. Our findings reveal that the largest determinant of power usage in transmission is the connection interval. Increasing the connection interval of the NIMON necklace from 1 second to 2 seconds while maintaining the same bandwidth can reduce the power usage by 52.1%. Furthermore, we show that the performance advantages of offloading processing swallow detection data for small window sizes (less than 20) and sample rates less than 16Hz are not substantial.

We evaluate the energy savings of buffering data to a microSD card to avoid the Bluetooth transmission costs. Our findings show that using SD card logging instead of BTLE transmission can increase power requirements by a factor of x30. Lastly, we simulate the power demands of a 16-point FFT (Fast Fourier Transform) on sensor data, and show that the high sample rates and processing algorithms used to analyze audio data in the frequency domain make audio-based dietary monitoring techniques a poor choice for wearables.

REFERENCES

[1] "Misfit wearables faq," http://www.misfitwearables.com/supportl.

[2] H. Kalantarian, N. Alshurafa, and M. Sarrafzadeh, "A wearable nutrition monitoring system," in *IEEE Body Sensor Networks*, 2014.

[3] J. Fontana, M. Farooq, and E. Sazonov, "Automatic ingestion monitor: A novel wearable device for monitoring of ingestive behavior," *Biomedical Engineering, IEEE Transactions on*, vol. 61, no. 6, pp. 1772–1779, June 2014.

[4] H. Kalantarian, N. Alshurafa, M. Pourhomayoun, S. Sarin, T. Le, and M. Sarrafzadeh, "Spectrogram-based audio classification of nutrition intake," in *IEEE EMBS HIPT*, 2014.

[5] E. S. Sazonov, O. Makeyev, S. Schuckers, P. Lopez-Meyer, E. L. Melanson, and M. R. Neuman, "Automatic detection of swallowing events by acoustical means for applications of monitoring of ingestive behavior," *IEEE Trans Biomed Eng*, vol. 57, no. 3, pp. 626–633, Mar 2010.

[6] M. Nagae and K. Suzuki, "A neck mounted interface for sensing the swallowing activity based on swallowing sound," in *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE*, Aug 2011, pp. 5224–5227.

[7] O. Amft, M. Kusserow, and G. Troster, "Bite weight prediction from acoustic recognition of chewing." *IEEE Trans. Biomed. Engineering*, vol. 56, no. 6, pp. 1663–1672, 2009. [Online]. Available: http://dblp.uni-trier.de/db/journals/tbe/tbe56.html

[8] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh, "A framework of energy efficient mobile sensing for automatic user state recognition," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '09. New York, NY, USA: ACM, 2009, pp. 179–192. [Online]. Available: http://doi.acm.org/10.1145/1555816.1555835

[9] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow, "Sociablesense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '11. New York, NY, USA: ACM, 2011, pp. 73–84. [Online]. Available: http://doi.acm.org/10.1145/2030613.2030623

[10] A. Krause, M. Ihmig, E. Rankin, D. Leong, S. Gupta, D. Siewiorek, A. Smailagic, M. Deisher, and U. Sengupta, "Trading off prediction accuracy and power consumption for context-aware wearable computing," in *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, Oct 2005, pp. 20–26.

[11] M. Stager, P. Lukowicz, and G. Troster, "Implementation and evaluation of a low-power sound-based user activity recognition system," in *Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on*, vol. 1, Oct 2004, pp. 138–141.

[12] Z. Yan, V. Subbaraju, D. Chakraborty, A. Misra, and K. Aberer, "Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach," in *Wearable Computers (ISWC), 2012 16th International Symposium on*, June 2012, pp. 17–24.

[13] O. Amft, M. Stger, P. Lukowicz, and G. Trster, "Analysis of chewing sounds for dietary monitoring," in *UbiComp 2005: Ubiquitous Computing*, ser. Lecture Notes in Computer Science, M. Beigl, S. Intille, J. Rekimoto, and H. Tokuda, Eds. Springer Berlin Heidelberg, 2005, vol. 3660, pp. 56–72. [Online]. Available: http://dx.doi.org/10.1007/115512014

[14] D. Brochetti, M. Penfield, and S. Burchfield, "Speech analysis techniques: A potential model for the study of mastication sounds," *Journal of Texture Studies*, vol. 23, no. 2, pp. 111–138, 1992. [Online]. Available: http://dx.doi.org/10.1111/j.1745-4603.1992.tb00515.x

[15] W. E. HI, A. E. Deibel, C. T. Glembin, and E. Munday, "Analysis of food crushing sounds during mastication: Frequency-time studies1," *Journal of Texture Studies*, vol. 19, no. 1, pp. 27–38, 1988. [Online]. Available: http://dx.doi.org/10.1111/j.1745-4603.1988.tb00922.x