# AN ULTRA-LOW-POWER VAD HARDWARE IMPLEMENTATION FOR INTELLIGENT UBIQUITOUS SENSOR NETWORKS

*Hiroki Noguchi, Tomoya Takagi, Masahiko Yoshimoto and Hiroshi Kawaguchi*

Kobe University, Kobe, 657-8501 Japan
h-nog@cs28.cs.kobe-u.ac.jp

## ABSTRACT

We propose a power management method using a digital voice activity detection (VAD) module for intelligent ubiquitous sensor systems. When this VAD module detects a speech signal, a main signal processing circuit is connected to a power source. When no speech signal is detected, most circuits except VAD are blocked off, thereby reducing stand-by power for the specialized sensor nodes used for speech signal processing. We implemented the VAD algorithm, using zero crossing of input signals to an FPGA, thereby achieving 2.10 mW operation. We synthesized this VAD module using CMOS 0.18-μm process, achieving 3.49 μW power consumption for operation at 1.8 V and 100 kHz.

***Index Terms*—** VAD, low power, zero crossing, sensor node, ubiquitous

## 1. INTRODUCTION

In recent years, digital human interfaces have been developed for living spaces, medical centers, robotics, and automobiles. Future applications will enable one person to control thousands of microprocessors without consciousness of their existence. Some speech and face recognition systems are in practical use, but most systems operate only in constrained environments according to installation conditions, angle, or distance to a device. For most people, these constraints are not convenient for everyday life.

Various intelligent ubiquitous sensor systems have been developed as new human interfaces [1]. In the near future, numerous cameras and microphones will be located on walls and roofs of living spaces. They will obtain speech data and visual information automatically and support absolutely hands-free systems. As described herein, we specifically examine speech signal processing with such ubiquitous sensor systems because speech interfaces are the fundamental mode of human communication; moreover, speech interfaces have a much broader range of application. One such specific application is a meeting system with a 128-channel square microphone array [2], which captures speech data from every microphone: each sensor node with some microphones must not only process signal recording but also noise reduction, sound-source separation, speech recognition, speaker identification, and other tasks [2–6].

As described herein, for the intelligent ubiquitous sensor system described above, we implement a voice activity detector (VAD) to reduce the power consumption of each sensor node. The rest of this paper is organized as follows. The next section presents a description of the intelligent ubiquitous sensor system. Section 3 introduces VAD algorithms. Section 4 describes the system implementation and experiment results. Section 5 explains subjects of future work. Finally, section 6 summarizes the paper.

## 2. INTELLIGENT UBIQUITOUS SENSOR NETWORK

Fig. 1 shows a brief description of the intelligent ubiquitous sensor network and a magnified block diagram of one ubiquitous sensor. We assume that each ubiquitous sensor has a microphone array and a microprocessor.

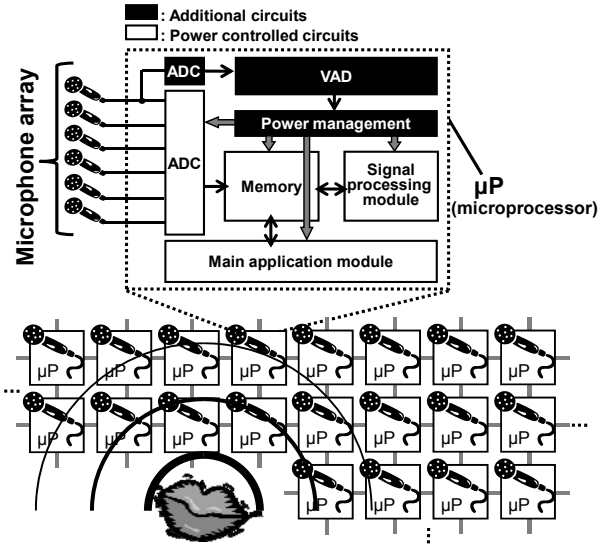


**Fig. 1. Intelligent ubiquitous sensor network and a magnified block diagram showing a microprocessor and microphone array.**

A general button-type battery supplies approximately 60–200 mAh as the total energy budget. The estimated

energy consumption of each sensor is 14.0 mA for a wireless transceiver [7] and 0.1 mA for a microphone. When each sensor samples the signals, a microprocessor runs with estimated power consumption of 10 mA: if all node modules operate continuously, this sensor node can run for about 7 hr with the 150 mAh battery. The sensor node must run for 24 hr to sense speech signals all day long. Therefore, it is necessary to reduce the power consumption during operating. To achieve 24 hr continuous operation, 6.25 mA is the limit of the average energy consumption.

We propose a power management method using a digital voice activity detection (VAD) module and a power management module (marked as black in Fig. 1). The VAD circuit outputs whether an input signal includes speech data or not. When the VAD module detects a speech signal, a main application module and signal-processing module are connected to a power source. When a speech signal is not detected, these circuits are blocked off. According to the speech signal emergence ratio, the power management described above can save energy. To increase this saving factor, it is important to reduce the VAD circuit power consumption.

## 3. VOICE ACTIVITY DETECTION

### 3.1. VAD Algorithms

The VAD algorithms determine the difference between noise wave patterns and speech signals, and find the beginning and end of speech. The VAD algorithms have been used progressively in speech recognition and voice over internet protocol (VoIP) applications [8]. For use in real-time applications, such as internet telephony, the complexity of the VAD algorithm must be low, but for almost all VAD algorithms, the power consumption is merely a secondary concern. Consequently, advanced VAD algorithms attract attention for their use of complicated algorithms such as Fourier Transforms, acoustic, and language model bases [9]

To minimize VAD circuit power consumption, the time-domain algorithm is the most suitable. Although the time-domain VAD algorithms' speech detection performance is poor, they are computationally less complex than frequency-domain algorithms. Frequency-domain algorithms have better immunity to low S/N than the time-domain algorithms, but they have higher computational complexity [8]. The zero-crossing VAD, which is a time-domain algorithm, is used to recover some low-energy phonemes that are rejected by an energy-based detector [8]. In subsections 3.2 and 3.3, we describe the zero-crossing VAD mechanism and algorithm in detail.

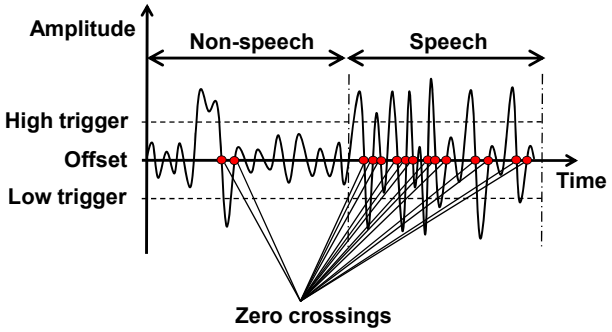


**Fig. 2. Zero-crossing point example. The offset shows the direct current (DC) component.**

### 3.2. Zero-Crossing VAD Algorithm

Fig. 2 portrays the zero-crossing algorithm mechanism. The zero crossing is the first intersection point between the input signal and the offset line, after the signal amplitude crosses the trigger lines: the high trigger and the low trigger. Between a speech signal and non-speech signal, the appearance ratios of this zero crossing differ. The zero-crossing VAD detects this difference and outputs the beginning point and the end point of a speech segment.

### 3.3. Modification the Zero-Crossing Algorithm

For the zero-crossing VAD to detect speech, all requirements are to catch the crossing over the trigger line and the offset line. A detailed speech signal is unnecessary. For that reason, the sampling frequency and the number of bits can be reduced. Once the VAD module detects a speech signal, the main signal processor begins to run and the sampling frequency and the number of bits are increased to sufficient values. These parameters, which decide the analog digital converter (ADC) specifications, are changeable depending on the specific applications that are integrated on the system. As described herein, we adopt standard parameters: the quality of 16 kHz sampling frequency and 16 bits per sample, for which most speech-recognition systems require continuous sensing [10]. Furthermore, only for the VAD algorithm, the sampling frequency is set to 2 kHz and the number of bits per sample is set to 10 bits, which are sufficient to detect human speech.

When considering the hardware implementation, it is important to adapt to the ADC circuits. The direct current (DC) offset presented in Fig. 2 is the mean value of the ADC outputs; it changes depending on the temperature, voltage, noise, and other operating parameters. Therefore, the output from ADC is usually normalized, such as to a range of 0 to 1, or -1 to 1, to operate correctly as a continuous system. However, to minimize the total computation of VAD, all calculations must be not floating point but integer arithmetic. To solve this problem, we

adopt a DC offset adjust process that is specialized for the zero-crossing algorithm.

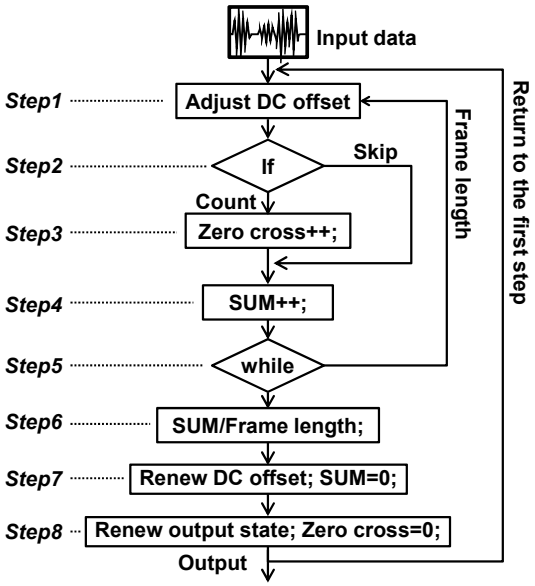


**Fig. 3. Zero-crossing VAD algorithm flow.**

Fig. 3 portrays the VAD flowchart with the DC offset process and details of each step in this flow. The following items describe concrete steps.

***Step 1***: Input data are adjusted to avoid overflowing.
***Step 2***: Input data are judged whether they have a zero cross or not.
***Step 3***: The zero-cross count is incremented if the input data exceed a threshold.
***Step 4***: To calculate the mean value in the present frame, the input data are added to the temporary sum.
***Step 5***: Input data are counted to control the frame length.
***Step 6***: The temporary sum is divided by the frame length only with the shift operation; the mean value in the present frame is obtained.
***Step 7***: The DC offset is adjusted according to the mean value.
***Step 8***: The output state is renewed based on the zero crossing count; the processing returns to the first step.

In step 6, the average of the input amplitudes is obtained using only integer arithmetic. The condition precedent is that the frame length corresponds to a multiple of 2 to obtain the average simply using the adder and the shifter circuits. After obtaining the average of the ADC outputs, VAD can count the zero crossings (steps 2 and 3). The total calculation amount from step 1 to step 8 is approximately 3 kops.

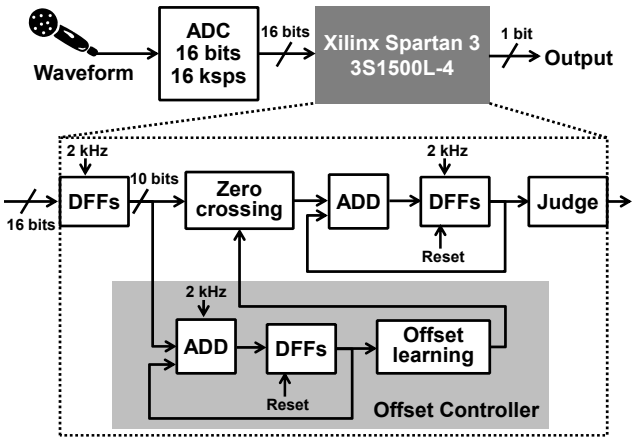


**Fig. 4. Block diagrams of the integrated devices. The D flip-flop (DFF) circuits keep up each input data asynchronously.**

## 4. EXPERIMENTAL MEASUREMENTS

### 4.1. Hardware Implementation

To clarify the proposed VAD performances, we implemented our proposed VAD algorithm using an FPGA (Spartan 3; Xilinx Inc.). We measured the FPGA board power consumption including the ADC, excepting the microphone. Fig. 4 portrays the board block diagram. The supply voltage to the board is 5 V. The ADC that we used takes a sample of 10 bits at a sampling rate within 16 ksps; this sampling rate is controlled by a dedicated circuit configured on FPGA. In Fig. 4, the signals sampled by ADC input to the FPGA chip directly and FPGA chip output the state signal whether the input signal includes speech or not. The calculations executed in this FPGA chip are almost identical to the flow depicted in Fig. 3. The zero crossing, the offset controller and the judgment modules shown in Fig. 4 respectively correspond to steps 1 and 2, steps 4, 6, and 7, and step 8, in Fig. 3. All calculations are integrated using integer arithmetic. Table 1 presents the device utilization summary. The slice flip flops and 4-input LUTs are, respectively, 1,015 and 3,831.

Fig. 5 presents the equipment—the FPGA board, microphone, and tester—used for the experiments described herein. Measurement results show that the board, except the microphone, requires 0.42 mA electric current and 2.10 mW power consumption. Results show that the stand-alone VAD module can run about 70 hr with a 150 mAh battery.

**Table 1. Device utilization summary.**

| Logic utilization | Used | Available | Utilization |
|---|---|---|---|
| # of slice flip flops | 1,015 | 26,624 | 3% |
| # of 4-input LUTs | 3,831 | 26,624 | 14% |

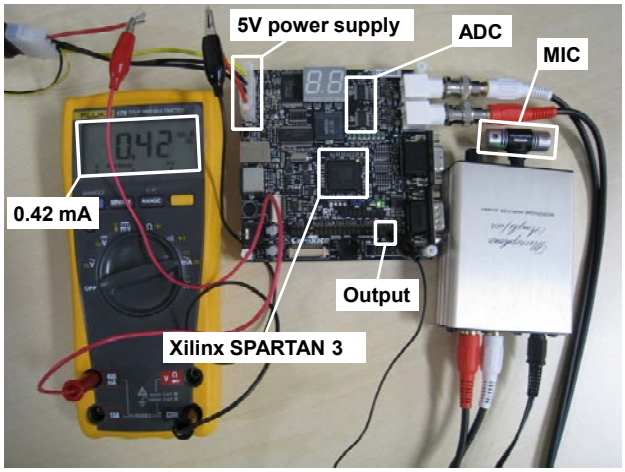


**Fig. 5. FPGA board with a microphone and a current tester.**

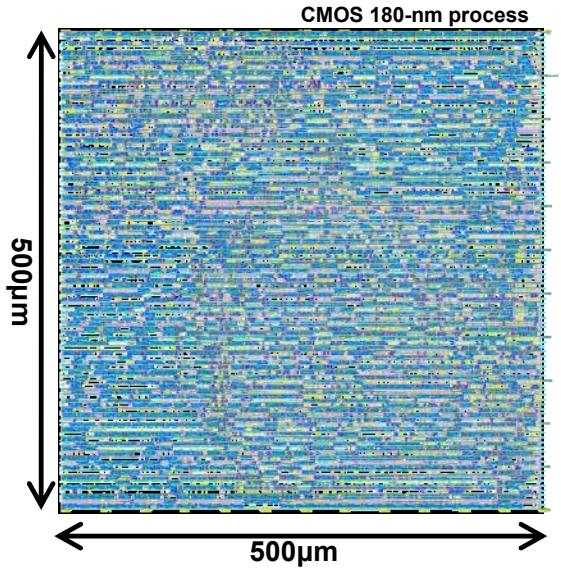


**Fig. 6. Layout plot of the zero-crossing algorithm integrated using CMOS 0.18-μm process technology.**

All blocks of the zero-crossing VAD module are implemented using CMOS 0.18-μm process technology. Fig. 6 depicts the layout plot. The power consumption is 3.49 μW at the 1.8 V supply voltage and 100 kHz frequency, which implies 1,700-day operation with the 150 mAh battery.

### 4.2. Experimental Results

The S/N easily affects the zero-crossing VAD algorithms because they are based only on changes in amplitude. For S/N dependencies of the VAD performance, we experiment using various S/N environments of -20 – 20 dB. In every S/N condition, we use identical 15-min speech data comprising 24 ATR phoneme balanced sentences [11]. The frame length of the VAD algorithm, depicted in Fig. 3, is 256. In each S/N condition experiment, the number of VAD results is 7,030. For this experiment, we counted the quantities of ***correct***, ***surplus***, and ***deficit*** VAD results. Each condition is defined as follows.

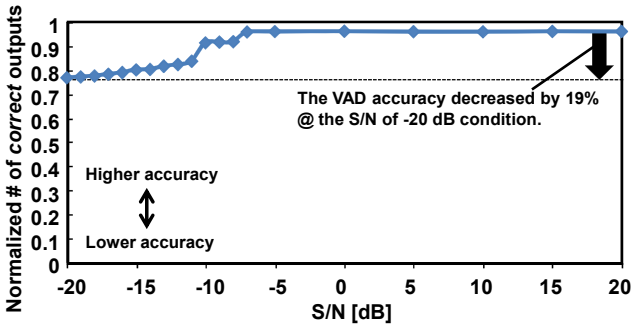


**Fig. 7. The number of *correct* VAD outputs using the number of outputs from VAD as normalized criteria.**

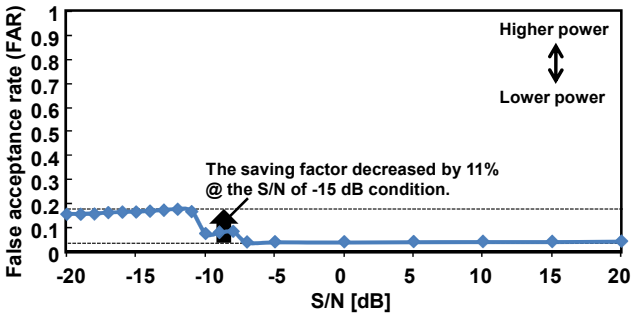


**Fig. 8. The false acceptance rate (FAR) in VAD outputs using the number of non-speech frames of the recorded condition as normalized criteria.**

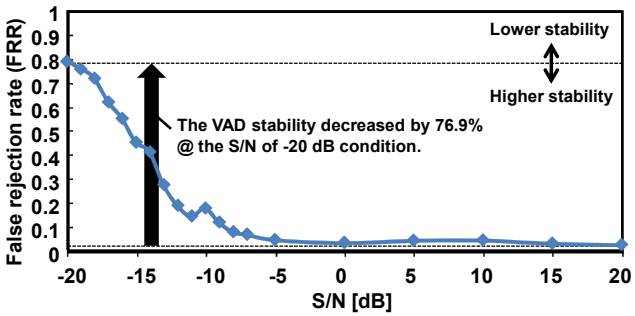


**Fig. 9. The false rejection rate (FRR) in VAD outputs using the speech frames of the recorded condition as normalized criteria.**

***Correct***: A case in which the VAD output is correct.
***False acceptance (FA)***: A case in which the VAD output is speech, although the input frame is non-speech.
***False rejection (FR)***: A case in which the VAD output is non-speech, although the input frame is speech.

Figs. 7, 8 and 9 respectively depict results of ***correct***, ***FA***, and ***FR*** VAD output quantities. Fig. 7 shows that zero-

crossing VAD retains greater than 80% accuracy, even for S/N of -20 dB, compared to the S/N of the 20 dB condition. Figs. 8 and 9 show that the power saving factor and stability of the zero-crossing VAD decreases according to deterioration of the S/N. Fig. 10 shows input waveforms and the VAD results for S/N of -20 dB, 0 dB, and 20 dB.

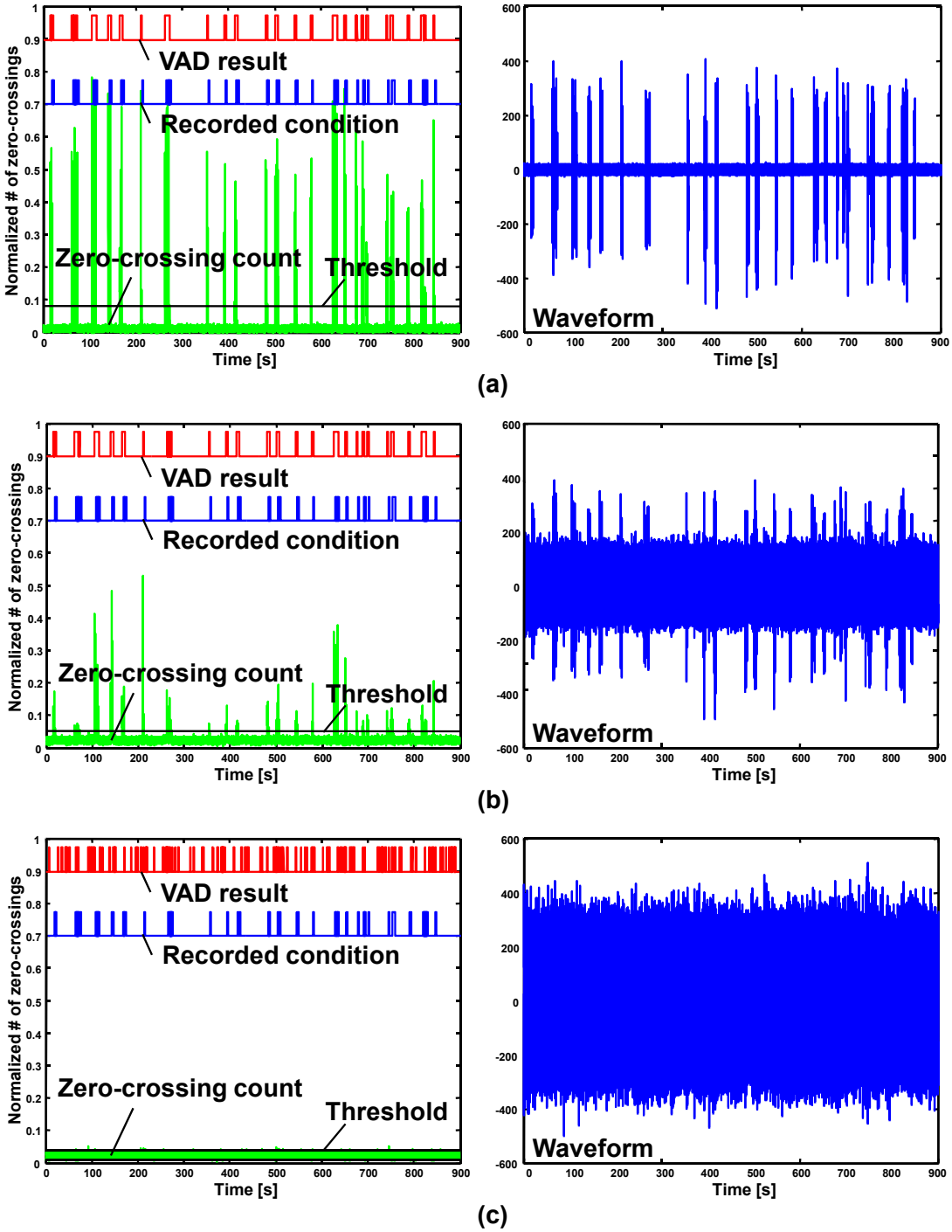


**Fig. 10. Waveforms and zero-crossing results at (a) S/N=20 dB, (b) S/N=0 dB, and (c) S/N=-20 dB.**

## 5. FUTURE WORK

To adapt to various S/N environments, the threshold parameters, the high trigger, and the low trigger (shown in Fig. 2) must be adjusted adaptively according to the zero-crossing counts while a speech signal is not detected. These parameters can be updated using the standard deviation (*SD*). The *SD* is calculated using the zero crossing and averages of amplitude, which are obtained while a speech signal is not detected. Equation (1) is the correction equation of *SD*.

$$modified\ SD_n = \frac{SD_{n-1} + \alpha SD_n}{1+\alpha} \quad (1)$$

The system is adjustable to the circumstances, with various S/N, by changing the threshold using learning coefficients $\alpha$, as shown in Eq. (1). In Eq. (1), modified $SD_n$ is obtained from $SD_{n-1}$ and observed $SD_n$. The learning coefficients $\alpha$ make the temporal changes of *SD*s in input signals smooth. The system becomes unstable for an accidental noise when $\alpha$ is large, because *SD* is renewed sensitively. However, the system becomes stable but the time for convergence of *SD* increases for small $\alpha$, because *SD* is renewed gradually. That is, $\alpha$ is a trade-off parameter for the system robustness. Therefore, we should examine the optimum $\alpha$ to realize a fully autonomous sensor node. Furthermore, we must adjust the learning equation above to low-power sensor nodes.

## 6. CONCLUSIONS

For intelligent ubiquitous sensor systems, we proposed a power management method using a voice activity detector (VAD) module. This VAD module reduces the stand-by power. We implemented the VAD algorithm using the zero crossing of input signal to an FPGA. Using the measurement result, the FPGA implementation achieved 2.10 mW operation. We also synthesized the VAD module to an LSI using CMOS 0.18-μm process, and achieved 3.49 μW power consumption for operation at 1.8 V and 100 kHz.

## 7. ACKNOWLEDGMENT

This research has been supported by the Semiconductor Technology Academic Research Center (STARC).

## 8. REFERENCES

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Journal of Computer Networks*, vol. 51, no. 4, pp. 921-960, Mar. 2007.

[2] Y. Tamai, S. Kagami, H. Mizoguchi, K. Sakaya, K. Nagashima, and T. Takano, "Circular microphone array for meeting system," in *Proc. of IEEE Sensors*, vol. 2, pp. 1100-1105, Oct. 2003.

[3] A. Eriksson, P. Stoica, and T. Soderstrom, "On-line subspace algorithms for tracking moving sources," *IEEE Transaction on Signal Processing*, vol. 42, no. 9, pp. 2319-2330, Sep. 1994.

[4] H. Tanaka and T. Kobayashi, "Estimating positions of multiple adjacent speakers based on MUSIC spectra correlation using a microphone array," in *Proc. of IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, pp. 3045-3048, May 2001.

[5] K. Nakadai, H. Nakajima, M. Murase, S. Kaijiri, K. Yamada, T. Nakamura, Y. Hasegawa, H.G. Okuno, and H. Tsujino, "Robust Tracking of Multiple Sound Sources by Spatial Integration of Room And Robot Microphone Arrays," in *Proc. of IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, pp. 929-932, May 2006.

[6] M. Rubsamen and A.B. Gershman, "Root-music based direction-of-arrival estimation methods for arbitrary non-uniform arrays," in *Proc. of IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*, pp. 2317-2320, Mar. 2008.

[7] T. Takeuchi, S. Izumi, T. Matsuda, H. Lee, Y. Otake, T. Konishi, K. Tsuruda, Y. Sakai, H. Fujiwara, C. Ohta, H. Kawaguchi, and M. Yoshimoto, "A 58-μW Single-Chip Sensor Node Processor Using Synchronous MAC Protocol," *2009 Symposium on VLSI Circuits Digest of Technical Papers*, pp. 290-291, Jun. 2009.

[8] R. Venkatesha Prasad, Abhijeet Sangwan, H.S. Jamadagni, Chiranth M.C, Rahul Sah, and Vishal Gaurav, "Comparison of Voice Activity Detection Algorithms for VoIP," in *Proc. of the Seventh International Symposium on Computers and Communications (ISCC)*, p. 530, Jul. 2002.

[9] H. Sakai, T. Cincarek, H. Kawanami, H. Saruwatari, K. Shikano, and A. Lee, "Voice activity detection applied to hands-free spoken dialogue robot based on decoding using acoustic and language model," in *Proc. of ACM International Conference on Robot Communication and Coordination (ROBOCOMM)*, pp. 303-310, Oct. 2007.

[10] T. Fujinaga, K. Miura, H. Noguchi, H. Kawaguchi, and M. Yoshimoto, "Parallelized Viterbi Processor for 5,000-Word Large-Vocabulary Real-Time Continuous Speech Recognition FPGA System," in *Proc. of 10th Annual Conference of the International Speech Communication Association (Interspeech 2009)*, to appear.

[11] T. Kobayashi, S. Itahashi, S. Hayamizu, and T. Takezawa, "ASJ continuous speech corpus for research," *Journal of Acoustical Society of Japan*, vol. 48, no. 12, pp. 888-893, 1992, in Japanese.