

Energy-Efficient Spectral Analysis Method Using Autoregressive Model-Based Approach for Internet of Things

Seiya Yoshida, Shintaro Izumi, *Member, IEEE*, Koichi Kajihara, Yuji Yano,
Hiroshi Kawaguchi, *Member, IEEE*, and Masahiko Yoshimoto, *Member, IEEE*

Abstract—This paper presents an energy-efficient spectral analysis method for the Internet of things. The objective of this work is to reduce the energy consumption of edge devices. The proposed method uses an autoregressive (AR) model for spectral analysis instead of the discrete Fourier transform, and its calculation process is distributed to the edge device and a base station by considering the energy consumption tradeoff of the data processing and the data communication. In this work, the Yule–Walker method is employed for the AR coefficient calculation. The calculation process of Yule–Walker method can be divided into two parts: an autocorrelation calculation and an AR coefficient calculation. The autocorrelation calculation is implemented in the edge devices, and its dedicated hardware is designed using Verilog HDL. Meanwhile, the AR coefficient is calculated in the base station and is used for the spectral analysis. According to this distributed processing approach, the energy consumption of the edge device can be reduced compared with conventional DFT approaches using the fast Fourier transform (FFT). The system level energy consumption is evaluated assuming the IoT edge device, which has a wireless transceiver using Bluetooth low energy. The evaluation results show that the proposed method can reduce 79% of the edge device energy consumption for spectral analysis in a practical application.

Index Terms—Autoregressive (AR) model, data compression, edge computing, energy efficient devices, low power devices and circuits, spectral analysis, Yule–Walker method

I. INTRODUCTION

THE energy consumption of edge devices is an important issue of the Internet of things (IoT). In many applications, the edge device is powered by a battery, and the battery capacity is limited by constraints such as device size, weight, and life time. When many edge devices exist in the system, it is impractical to frequently replace or recharge the battery. Although energy harvesting has also attracted attention, the supply energy is limited in many cases. Therefore, an energy-efficient system design is important.

Recent edge devices contain wireless transceivers such as

Bluetooth Low Energy (BLE), ZigBee, and LoRaWAN. However, the wireless communication circuit consumes a significant amount of energy, and it occupies a large ratio in the total energy consumption of the edge device [1]. Data compression is a straightforward method to reduce communication energy dissipation. In particular, feature extraction is effective because it can reduce the transmission data size when only the necessary feature is transmitted from the edge device to a base station. Data compression also contributes to the efficiency of the wireless communication channel and the memory capacity reduction in edge devices. However, feature extraction requires processing energy. Therefore, we should consider the energy tradeoff between data processing and data communication to reduce the energy consumption of edge devices.

This paper focuses on the spectral analysis as the feature extraction method. The spectral analysis is widely used in many applications of IoT, such as structural health monitoring [2], failure prediction of motors [3], speech analysis [4], and human heart beat variability analysis [5]. The discrete Fourier transform (DFT) is generally used for spectral analysis because the fast Fourier transform (FFT) algorithm can obtain an accurate spectrum with minimal calculation. Furthermore, as most calculations are composed of multiplication and accumulation, it can be easily implemented in a dedicated hardware.

In this work, we employed an autoregressive (AR)-model-based spectral analysis approach instead of the DFT to improve the energy dissipation of the edge devices. The AR model is a linear predictive modeling technique, which assumes that the current value of a signal can be described by a finite linear aggregate of the previous values. The AR model can be used for spectral analysis as an alternative to the Fourier transform [6]. The details of the AR model and spectral estimation are described in Section II. The performance evaluation and hardware implementation results of this method are compared with the FFT in Section III.

This paper is based on the results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

S. Yoshida, K. Kajihara, Y. Yano, H. Kawaguchi, M. Yoshimoto are with the Graduate School of System Informatics, Kobe University, 1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 6578501, Japan.

S. Izumi is with the Institute of Scientific and Industrial Research, Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka 5670047, Japan (e-mail: shin@sanken.osaka-u.ac.jp).

When the model order of the AR model is high, the total computational amount of the AR-model-based approach is larger than that of the FFT. To overcome this drawback, we divide the calculation process into two parts: autocorrelation calculation and AR coefficient calculation, and only the autocorrelation calculation part is implemented in the edge device. When using the Yule–Walker method [6], only the autocorrelation output is required for the AR coefficient calculation and spectral analysis [7]. Furthermore, the data size of the autocorrelation output is much smaller than that of the power spectrum, which is the output of the FFT. Therefore, the communication energy can be reduced and the AR coefficient calculation can be implemented in the base station.

Fig. 1 shows a system-level design of the proposed method. The edge device has a dedicated hardware for the autocorrelation calculation to reduce energy dissipation. When the total energy consumption of the autocorrelation calculation and transmission is smaller than the energy consumption of the FFT calculation and power spectrum transmission, the proposed method can improve the energy efficiency of the spectral analysis. This tradeoff is discussed in Section IV.

II. AUTOREGRESSIVE MODEL AND SPECTRAL ANALYSIS

DFT, which is classified as a nonparametric method, is generally used for spectral analysis. Nevertheless, this paper focuses on the AR model, which is a parametric method, to improve the energy efficiency of the spectral analysis for the IoT. The AR model is a linear predictive modeling technique, which assumes that the current value of a signal can be described by a finite linear aggregate of the previous values. The AR model can be used for spectral analysis as an alternative to the Fourier transform [6].

A. Autoregressive (AR) Model

The AR model expresses time-series data x at time t using its past time-series data and white noise w as the following equation:

$$x_t = \sum_{k=1}^m a_k x_{t-k} + w_t \quad (1)$$

where a_k is the AR coefficient; it is a weight that indicates how much data of a certain time in the past influences the current data. m is the AR order, which is a parameter that determines the past search period in the AR model. w_t indicates the prediction error between the linear sum of the past data and the actual current data.

Subsequently, the power spectral density (PSD) of the time-series data x can be obtained using AR coefficients a_k as the following equation:

$$PSD(f) = \frac{1}{F_s[Hz]} \frac{e_m}{\left| 1 + \sum_{k=1}^m a_k e^{-\frac{2\pi i k f}{F_s}} \right|^2} \quad (2)$$

Here, F_s is the sampling frequency, and e_m is the variance of the prediction error. f represents the frequency component of the PSD.

The model order m is the hyperparameter, which is determined by the application requirement. For example, the

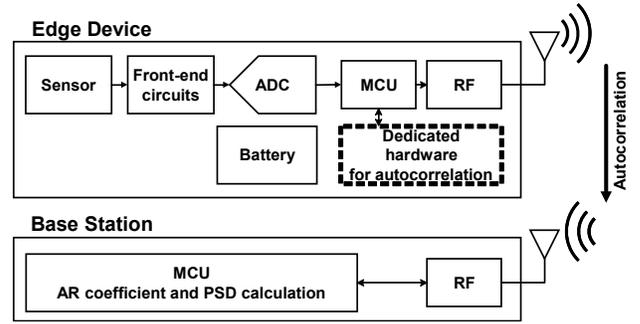


Fig. 1. Proposed system design for AR-model-based spectral analysis.

Akaike information criterion (AIC) [8] and Bayesian information criterion (BIC) [8] are used for the AR order determination. Herein, we evaluate the performance of the spectral analysis with a wide range of model order to accommodate various IoT applications. It is also variable in our hardware implementation, as described in Section III.

As described above, the PSD is estimated using the AR coefficient a_k . In other words, the AR coefficient estimation is the key factor in this approach. Several algorithms can be used to estimate the AR coefficient: Burg's method [7], covariance method [7], and the Yule–Walker method [7]. Herein, we evaluate Burg's method and the Yule–Walker method as the candidates for AR coefficient estimation for IoT applications. The covariance method and modified covariance method are not considered in this work, because they are not in the form of the Toeplitz matrix [9], and it is difficult to reduce the amount of calculation.

B. Burg's Method and Fast Burg's Method

In the original Burg's method [10], the AR coefficient is calculated by the least-squares method for minimizing the power P_m of a forward prediction error vector f_i and a backward prediction error vector b_i as described below:

$$f_i = \sum_{k=0}^m a_k x_{i-k} \quad (3)$$

$$b_i = \sum_{k=0}^m a_k x_{i-m+k}$$

$$P_m = \sum_{i=1}^{N-m} \{f_i^2 + b_i^2\} \quad (4)$$

Here, N denotes the signal length. Subsequently, the AR coefficient a_k is restricted by the Levinson–Durbin recursion as below [11]:

$$a_{i+1} = a_i + a_{mm} a_{m-j, m-1} \quad j = 1, 2, \dots, m-1 \quad (5)$$

Under this constraint equation, parameter λ_i that minimizes P_m is derived as below:

$$\lambda_i = -\frac{2b_i f_i}{f_i f_i + b_i b_i} \quad (6)$$

Subsequently, the $(i+1)$ -th order AR coefficient a_{i+1} can be calculated by the i -th order AR coefficient a_i as follows:

$$a_{i+1} = \begin{bmatrix} a_i \\ 0 \end{bmatrix} + \lambda_i \begin{bmatrix} a_i \\ 0 \end{bmatrix} \quad (7)$$

$$J = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \ddots & 1 & 0 \\ 0 & \ddots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix} \quad (8)$$

Here, J represents an $(i+1) \times (i+1)$ size matrix that inverts the same size matrix vertically and horizontally.

To reduce the computational amount of these calculations, a fast Burg's method [10] has been proposed. In this method, without explicitly calculating the forward prediction error energy f_i and the backward prediction error energy b_i , the reflection coefficient k_i is derived as follows:

$$k_i = -\frac{[a_i^T \ 0]g_i}{[a_i^T \ 0]g_i} \quad (9)$$

where

$$g_i = \begin{bmatrix} g_i + k_{i-1}Jg_{i-1} + \Delta R_{i+1}a_i \\ r_{i+1}^T a_i \end{bmatrix} \quad (10)$$

$$\Delta R_{i+1} = -\begin{bmatrix} x_i \\ \vdots \\ x_0 \end{bmatrix} [x_i \ \dots \ x_0] \quad (11)$$

$$r_{i+1} = \begin{bmatrix} 2c_{i+1} \\ r_i - \begin{bmatrix} x_0 \\ \vdots \\ x_{i-1} \end{bmatrix} x_i - \begin{bmatrix} x_{N-1} \\ \vdots \\ x_{N-i} \end{bmatrix} x_{N-i-1} \end{bmatrix} \quad (12)$$

$$c_i = [x_0 \ \dots \ x_{N-i-1}] \begin{bmatrix} x_i \\ \vdots \\ x_{N-1} \end{bmatrix} \quad (13)$$

Here, c_i represents the autocorrelation.

C. Yule–Walker Method

The Yule–Walker method [12] can estimate the AR coefficient using the autocorrelation of the input data. The AR coefficient a_k ($k = 0, \dots, m$) is calculated by solving the following Yule–Walker equation:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_{m-1} \\ R_1 & R_0 & \cdots & R_{m-2} \\ \vdots & \vdots & \ddots & \vdots \\ R_{m-1} & R_{m-2} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{bmatrix} \quad (14)$$

R_k ($k = 0, \dots, m$) is the autocorrelation of the time-series data and it can be calculated by the following equation:

$$R_k = \sum_{j=k}^{N-1} x_j x_{j-k} \quad (k = 0, \dots, m) \quad (15)$$

In the straightforward manner, the computational amount to solve (14) including the inverse matrix is $O(m^4)$. However, the determinant of (14) is in the form of the Toeplitz matrix, and it can be solved easily using the Levinson–Durbin algorithm (LDA) [12] as follows:

$$a_1 = -\frac{R_1}{R_0}, e_1 = R_0 + R_1 a_1 \quad (16)$$

$$a_{i+1} = \begin{bmatrix} a_i \\ 0 \end{bmatrix} + \lambda_i J \begin{bmatrix} a_i \\ 0 \end{bmatrix}, (i > 1) \quad (17)$$

$$\lambda_i = -\frac{\sum_{j=0}^i a_j R_{i+1-j}}{e_i} \quad (18)$$

$$e_{i+1} = (1 - \lambda_i^2) e_i \quad (19)$$

III. COMPARISON OF SPECTRAL ANALYSIS METHODS

The performance evaluation and hardware implementation results of the spectral analysis methods are described in this Section.

A. Comparison of Computational Amount

First, we evaluate the computational amount of the edge device for the FFT, fast Burg's method, and Yule–Walker method. Here, only real values are assumed as the input data, and the number of FFT calculations is computed based on this assumption [13]. It is noteworthy that the base station can calculate the PSD from the AR coefficients when using the fast Burg's method. Furthermore, when using the Yule–Walker method, the base station only requires the autocorrelation results for the PSD calculation, because the AR coefficients can be calculated only using the autocorrelation results as shown in (16)–(19). In contrast, the FFT cannot realize these distributed computing because the intermediate data size of the FFT is not smaller than the input data size. Therefore, the power spectrum calculation of the FFT, the autocorrelation, the AR coefficient calculation of the fast Burg's method, and the autocorrelation calculation of the Yule–Walker method are compared as the required computational amount in the edge device.

Fig. 2 shows the comparison of the number of multiplications in the edge device. Subsequently, the number of multiplications of the real-value FFT is $N \log_2 N$, when the input data length is N [13]. The number of multiplications of the fast Burg's method and Yule–Walker method are $Nm + 5m^2$ [10] and $Nm - m^2/2$, respectively, when the AR order is m . As shown in Fig. 2, the FFT is advantageous in the number of multiplications when the AR order is high, although the breakeven point is shifted with the long data length. However, the fast Burg's method and Yule–Walker method can reduce the transmitting data size as described above. Because the wireless data transmission energy is often larger than the calculation energy, this advantage is efficient for system-level energy reduction. This tradeoff is discussed in Section IV.

B. Accuracy of Estimated PSD

Fig. 3 shows the comparison of the estimated PSD using FFT, fast Burg's method, and the Yule–Walker method. The input data is a mixed signal of 25, 40, and 55 Hz sine waves. The input signal is resampled by 256-Hz sampling rate, and the data length is set to 256 (1.0 s). The Hamming window is adopted in each method. The AR order of the fast Burg's method and Yule–Walker method is set to 25. The Yule–Walker method is calculated by both the floating point and 16-bit fixed point. As shown in Fig. 3, although the estimated peak power of the PSD is fluctuated, the peak frequencies are detected accurately using

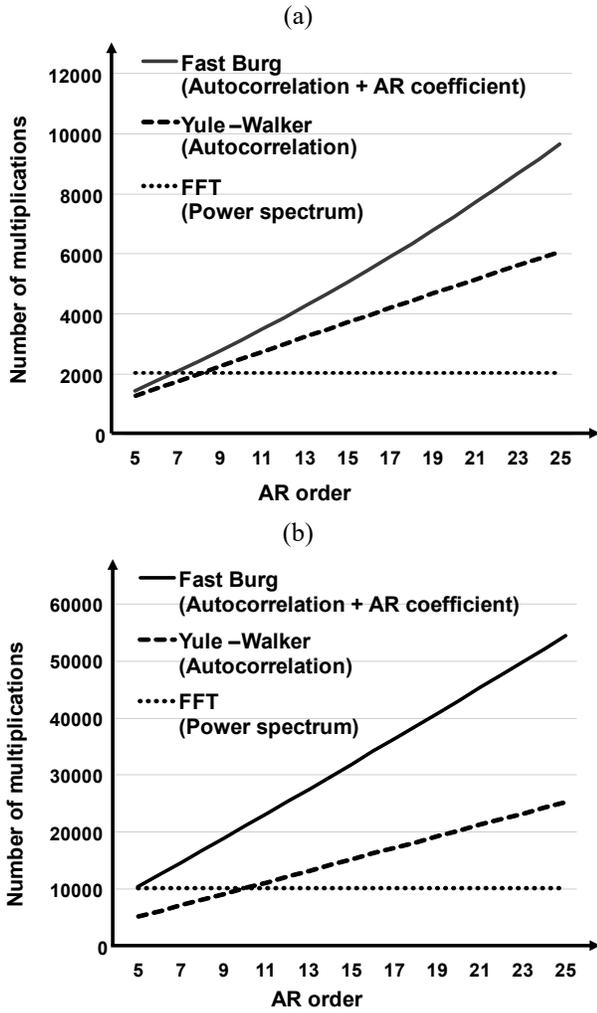


Fig. 2. Relationship between computational amount and AR order: (a) 256 input data length and (b) 1024 input data length.

the AR-model-based methods including the fixed-point Yule-Walker method.

Next, we evaluated the PSD estimation accuracy with a practical application example: heart rate variability analysis. Fig. 4 shows a time series of the human heart beat. The frequency characteristics of the heart beat interval fluctuation indicate the activity of autonomic nerves [5]. Subsequently, the ratio of a low-frequency component (LF) and a high-frequency component (HF) are used as an index. The respective frequency ranges of the LF and HF are 0.05 Hz to 0.15 Hz and 0.15 Hz to 0.40 Hz. The sum of the power spectra in those frequency ranges is calculated. When the LF is larger than the HF, the sympathetic nerve is tensed. Conversely, when the HF is larger than the LF, the parasympathetic nerve is tensed.

Fig. 5 depicts the estimated PSDs from the heart beat intervals shown in Fig. 4. The AR order and data length are respectively set to 25 and 256. As shown in Fig. 5, although the shape of the PSDs appears to be different, the frequency peaks at 0.09 Hz and 0.18 Hz are correctly detected by each method. Furthermore, the error of the ratio of LF/HF is less than 0.07 between the FFT and fixed point Yule-Walker method. This error is sufficiently small for this application.

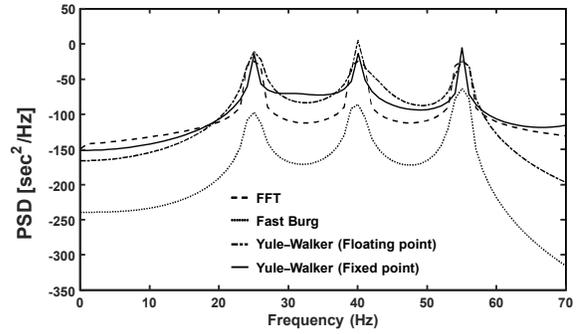


Fig. 3. Comparison of estimated PSD with 25, 40, and 55 Hz sine wave input. Sampling rate, input data length, and AR order are respectively set to 256, 256, and 25.

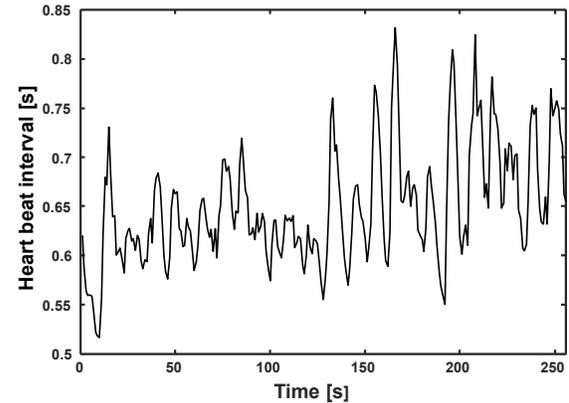


Fig. 4. Measured example of human heart beat for heart rate variability analysis.

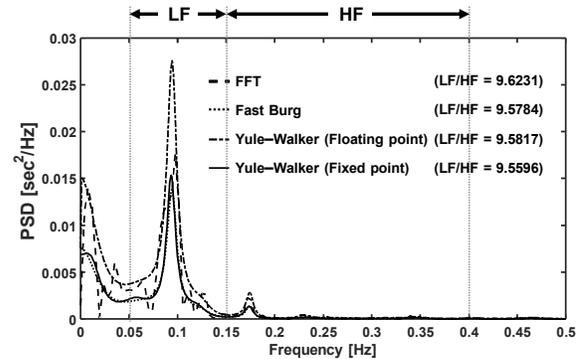


Fig. 5. Comparison of estimated PSD with heart beat intervals. Sampling rate, input data length, and AR order are respectively set to 256, 256, and 25.

C. Hardware Implementation and Comparison of Energy Consumption

As shown in Fig. 2, the computational amount of the edge device for the Yule-Walker method is always smaller than that for the fast Burg's method. Furthermore, it has the smallest transmission data size. Figs. 3 and 5 show that the estimated PSD using the fixed-point Yule-Walker method can be used for practical applications. Thus, we chose the Yule-Walker method with 16-bit fixed point as the spectral analysis method in the IoTs.

To evaluate the accurate energy consumption of the Yule-Walker method, we designed a dedicated hardware for the

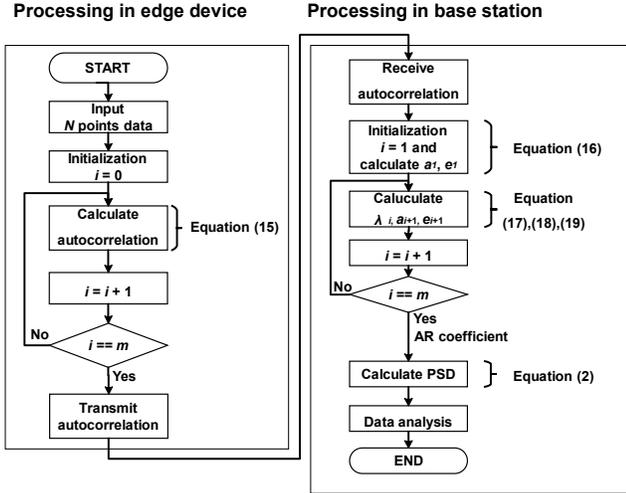


Fig. 6. Processing flow of PSD estimation using AR-model-based approach with Yule-Walker method.

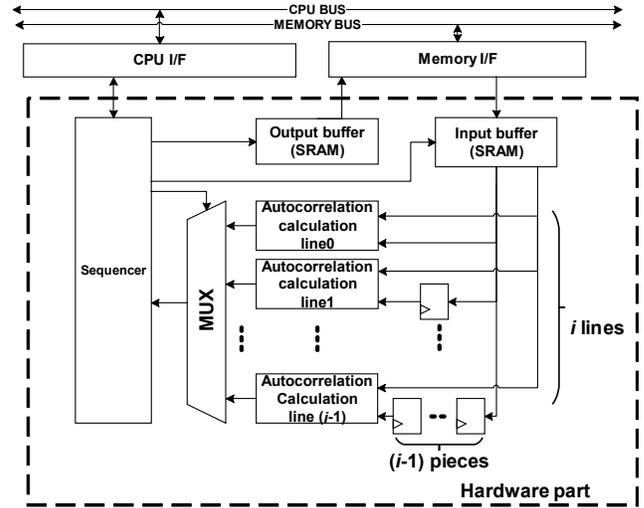


Fig. 8. Block diagram of autocorrelation processor with parallelism

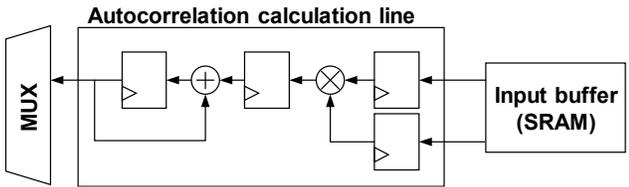
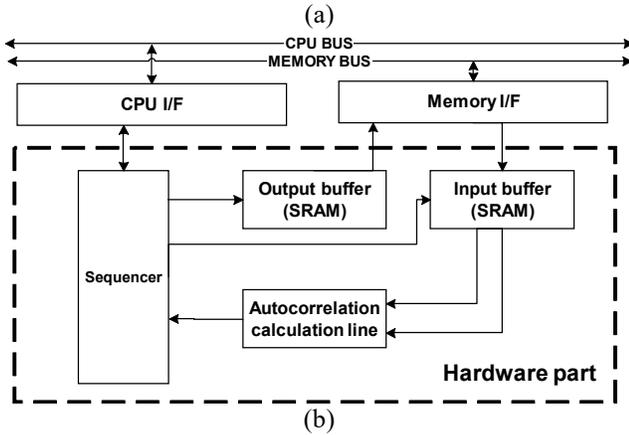


Fig. 7. (a) Block diagram of autocorrelation processor without parallelism, and (b) implementation of autocorrelation calculation line.

autocorrelation calculation. Because most operations of the autocorrelation are multiply-accumulate, it can be easily implemented as the dedicated hardware.

Fig. 6 shows the processing flow of the PSD estimation using the Yule-Walker method. As shown in Fig. 6, only the autocorrelation process is implemented in the edge device.

Fig. 7 illustrates the block diagram of the hardware-implemented autocorrelation without parallelization. This hardware can easily extend to a parallel architecture because the autocorrelation is the appropriate calculation for spatial parallelism. To reduce the memory access and latency, a parallel architecture is designed as shown in Fig. 8. When the degree of parallelism is set to i , the required hardware resource is decided as shown in Table 1. Subsequently, the latency of each autocorrelation calculation line is illustrated in Fig. 9. As shown in Table 1 and Fig. 9, a tradeoff exists between the degree of parallelism and latency. If a mismatch exists between

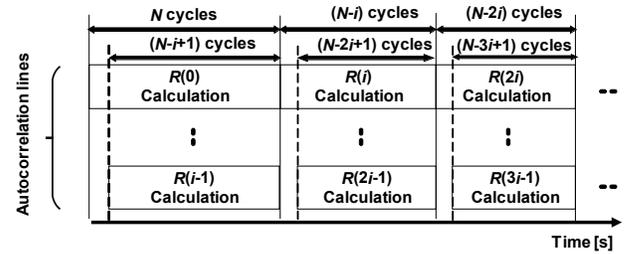


Fig. 9. Timing chart of parallel autocorrelation processing.

TABLE I
REQUIRED HARDWARE RESOURCE FOR AUTOCORRELATION

Adder	Multiplier	Registers	Latency
$i + 3$	i	$64i + 33$	$\frac{(Nm - m^2/2)}{i}$

AR order and parallelism, the higher parallelism architecture causes redundant circuit and calculation energy consumption. For example, when the AR order is 20 and parallelism is 16, 12 autocorrelation lines consume unnecessary energy.

Next, the autocorrelation hardware is implemented using Verilog HDL to evaluate the accurate energy consumption. Logic synthesis and power estimation are performed using Synopsys Design Compiler Version 2017.09 with standard cell and SRAM libraries designed by the 65-nm CMOS process. Table 2 shows the summary of the implementation result.

Fig. 10 shows the energy consumption of the proposed hardware for autocorrelation when the input data length is 256 and the AR order is 25. The energy consumption of memory access (active) and memory retention (static) is dominant in the total energy consumption without parallelism. When the degree of parallelism is increased, both the active and static power of the memory can be reduced, because both the number of memory access and the calculation cycles are reduced.

Fig. 11 shows the effect of the AR order on the normalized energy consumption. The conventional FFT hardware implementations [14]-[16] are also indicated in Fig. 11. It is noteworthy that these energy consumptions are normalized by the following equation [15]:

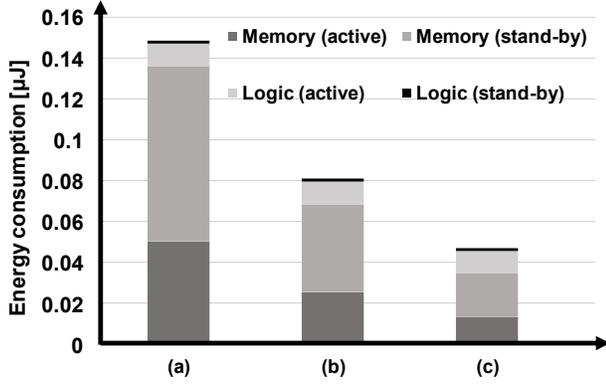


Fig. 10. Energy consumption of autocorrelation processing: (a) without parallelism, (b) 2-parallel, and (c) 4-parallel architecture.

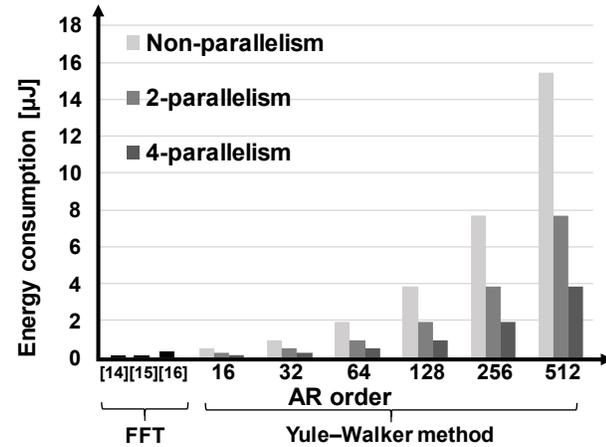
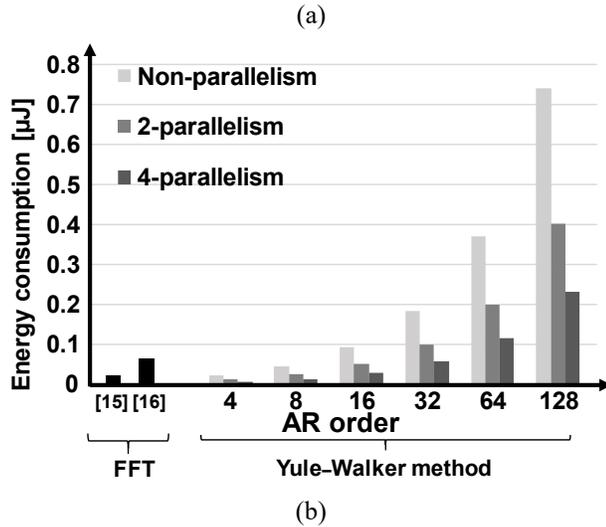


Fig. 11. Comparison of processing energy consumption in edge device; (a) 256 input data length and (b) 1024 input data length.

$$E_{\text{norm}} = \frac{\text{Power [nW]} \times \text{Execution time [sec]}}{\left(\frac{\text{Technology [nm]}}{65 \text{ [nm]}}\right) \left(\frac{V_{\text{dd}} \text{ [V]}}{1.2 \text{ [V]}}\right)^2} \quad (18)$$

The prior work [14] does not explain the FFT hardware power consumption with 256 input data length. Furthermore, the FFT hardware presented in prior works [15, 16] can input complex values. These hardware consumes two times larger power

TABLE II
SUMMARY OF HARDWARE IMPLEMENTATION FOR ENERGY EVALUATION

Method	Yule-Walker
Technology node	65 nm
Supply voltage	1.2 V
Frequency	300 MHz
Bit width (input/output)	16 bits
Input buffer memory size	$(N \times 16)$ bits
Output buffer memory size	$\{(m + 1) \times 16\}$ bits
Input data length N^*	256, 1024
AR order m^*	From 2 to $(N - 1)$

* N and m are programmable in our hardware design using Verilog HDL.

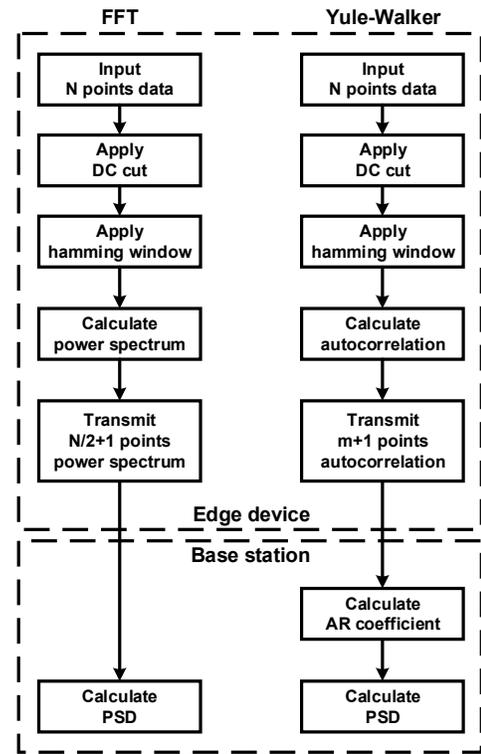


Fig. 12. Comparison of PSD estimation flow in edge device and base station (BS).

compared with real value input FFT hardware, when assuming that four times real value multiplication are required for one complex value multiplication. Therefore, in this work, we compared half the value of normalized energy consumption with the proposed hardware implementation. When the degree of parallelism is doubled, the circuit area is increased by approximately 1.8 times. The required execution cycle is approximately $(Nm - m^2/2)/i$ when the degree of parallelism is i .

IV. SYSTEM LEVEL PERFORMANCE EVALUATION

As shown in Fig. 1, we aim to distribute the processing of the spectrum analysis to the edge device and the base station. As described in Section III, the Yule-Walker method can be

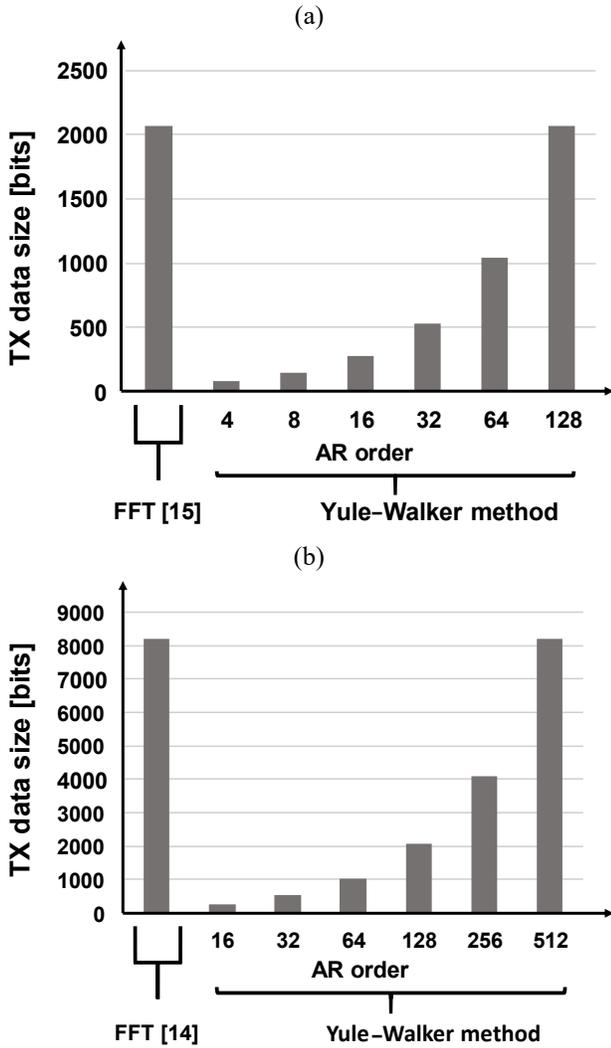


Fig. 13. Comparison of transmitted data size from edge device to base station; (a) 256 input data length and (b) 1024 input data length.

divided into two parts: the autocorrelation calculation and the AR coefficient calculation. Because the data size of the autocorrelation results is smaller than the power spectrum output of the FFT, the Yule-Walker method is advantageous as it can reduce the transmitting data size from the edge device to the base station. Subsequently, only the autocorrelation calculation process is implemented in the edge device. Fig. 12 summarizes the difference in PSD calculation flow with FFT and the AR model using the Yule-Walker method.

The transmitted data size depends on the input data length N and AR order m . Fig. 13 shows the transmitted data size in each AR order. Subsequently, the bit width of the input data is set to 16. When the AR order is set to N/m , the transmitted data size of the Yule-Walker method is the same as that of the FFT. If the AR order is smaller than $N/2$, the Yule-Walker method is advantageous in terms of the transmitted data size.

However, the Yule-Walker method is disadvantageous in terms of the processing energy, as discussed in Section III. Therefore, a tradeoff exists between the transmitting energy and processing energy.

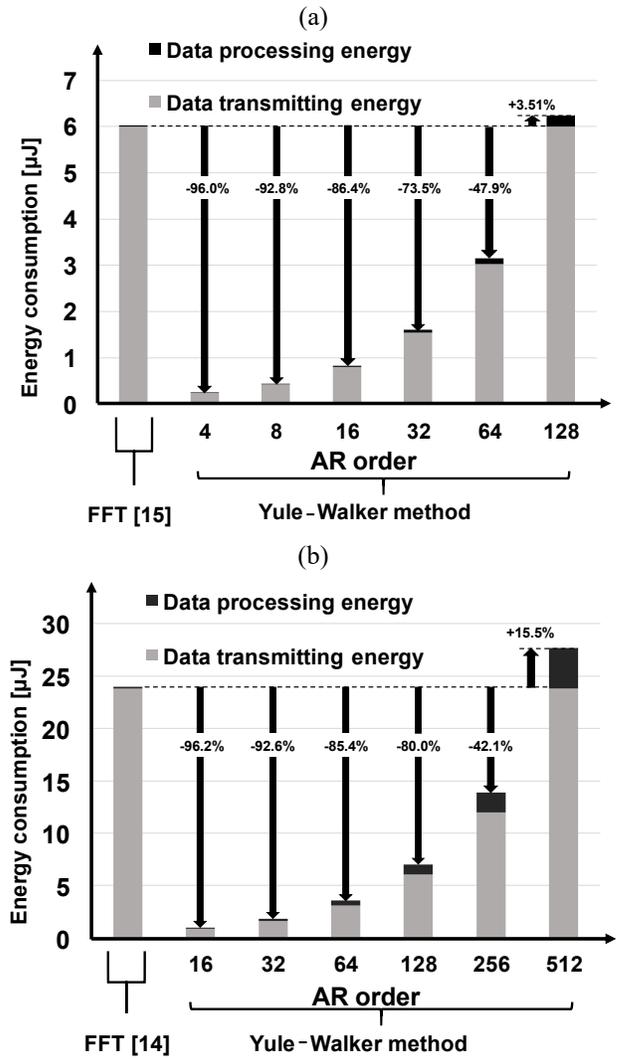


Fig. 14. Comparison of total energy consumption in edge device including processing energy and transmitted energy; (a) 256 input data length and (b) 1024 input data length.

In this work, BLE is assumed as the wireless communication standard for the edge device. In a state-of-the-art BLE transceiver [17], approximately 2.9-mW transmitting energy is achieved with 1Mbps data rate. This transmitting energy is used in this evaluation. The overhead of the link connection and packets is not considered for simplicity.

Fig. 14 shows the total energy consumption of edge device for spectral analysis. The proposed method assumes 4-parallelism. The Direct Current (DC) component elimination and hamming window calculation are negligible because they consume less than 1% of the processing energy shown in Fig. 14.

As shown in Fig. 14, the Yule-Walker method achieves smaller energy consumption, when the AR order is smaller than $N/4$. Although the processing energy consumption of the Yule-Walker method is larger than that of the FFT and it increases in proportion to the AR order, the transmitting energy is still dominant because the proposed dedicated hardware can achieve small energy dissipation comparable with the FFT hardware. Detailed performance comparison result is

summarized in Table 3. Here, the energy for processing in Table 3 shows the normalized energy consumption calculated by (18).

As discussed in Section II.B, the Yule–Walker method with 256 input data length and the 25-th AR order can be used in a practical application: heart rate variability analysis. In this condition, the proposed method can achieve 1.25- μ J energy consumption in the edge device, which is 79% smaller than the conventional method using FFT. Subsequently, as shown in Fig. 5, the error caused by the proposed method is less than 0.07 in the LF/HF ratio, which is sufficiently small for the required accuracy in this application. When raw data is transmitted in this application, the estimated energy consumption is 11.9- μ J. Compared with this result, the proposed method achieves 89.5% lower energy consumption. Furthermore, when the edge node performs all process of PSD calculation using AR model, the computational amount of AR coefficient calculation is $Nm + m^2/2$ according to (15)-(19). When the data length N and AR order m are respectively set to 256 and 25, the number of calculations is approximately 6.7 kilo-operations. The computational amount of PSD calculation for each frequency domain is $2m$ according to (4). We assumed that the frequency resolution of the PSD is 0.002 Hz and the frequency range is between 0 and 0.5 Hz with 1-kHz sampling rate. Then, approximately 13 kilo-operations are required, because 250-frequency domains are calculated to obtain the PSD. On the other hand, only 6.1 kilo-operations are required to calculate the autocorrelation coefficient R , which is the part of AR coefficient calculation, as shown in Fig. 2. It is three times smaller than the total computational amount of the AR coefficient and PSD.

V. DISCUSSION

In this paper, we focused on the AR-model-based spectral analysis approach instead of the Fourier transform to improve the energy dissipation of the edge devices. For example, literature [5] shows that the AR-model based approach can achieve equivalent results to FFT in the heart rate variability analysis. Furthermore, this method has an advantage compared with the FFT that more accurate PSD can be obtained with short window length. The evaluation results show that the AR-model-based spectral analysis can be implemented with lower power consumption compared with the FFT.

In prior works, a hybrid data compression algorithm [18] and compressed sensing schemes [19, 20] are used for data compression and sensor active rate reduction. Adaptive sampling schemes have also been proposed to reduce the sensor active rate [21]. These conventional methods can reduce the transmission data by decimating data and lossy compression. In contrast, the proposed method reduces transmission data using the characteristics of frequency analysis based on AR model: the data amount of correlation coefficients, which is intermediate data calculated by the edge node, is smaller than the PSD calculated in the base station. When using FFT, the transmission data cannot be compressed in the same manner, because the amount of intermediate data is larger than the PSD (see Fig. 12). As shown in Fig. 14, compared with conventional

dedicated hardware implementations of FFT, the proposed scheme has an advantage in the system level power consumption even in higher AR order. Then, the compression ratio can be controlled by the AR order. Note that this paper is first hardware implementation of AR-model-based frequency analysis. Furthermore, the conventional sampling rate reduction techniques can be combined with the proposed method.

In our implementation, the FFT, fast Burg's method, and Yule-Walker method respectively requires about 1-kilo operations, 8 kilo-operations, and 6 kilo-operations computational amount, when the data length and AR order are respectively set to 256 and 25. Then, the PSD calculation is executed every seconds. Furthermore, the transmission data required by the FFT, fast Burg's method, and Yule-Walker method are 2,048 bits/s, 400 bits/s, and 400 bits/s, respectively. Because the BLE transceiver with 1-Mbps data rate is assumed in this work, it has enough capacity for this application. Note that the transmission energy is 2.9-nJ/bit and the evaluation results shown in Fig. 14 is calculated based on these values.

The proposed method can realize 2.04- μ W operation with 256 data length and 25-th AR order. Then, we assume the energy dissipation shown in Fig. 14 and the environmental setting shown in Table 2. It is assumed that the clock and power source are gated while stand-by state. This power dissipation is sufficiently small value compared to the state-of-the-art low-power IoT edge devices (e.g. 17.4- μ W in literature [22] and 24-uW in literature [23]).

When the data length N and AR order m are respectively set to 256 and 25, the required energy for the frequency analysis is about 1.3 μ J. The expected average power consumption is about 1.3 nW, when assuming that the frequency analysis is performed every seconds. Then, the estimated operating time is about 32,000-hours with CR1220 Lithium battery, which has 1.0-g weight and 35-mAh capacity. This power consumption can also be covered by an energy harvester like wearable thermoelectric generator (TEG), which can generate 20- μ W power from 14.4-K temperature difference with 0.97-cm² area [24].

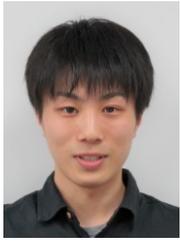
VI. CONCLUSION

We herein proposed an energy-efficient spectral analysis method using the AR-model-based approach to reduce the edge device energy consumption. The network distributed processing and the dedicated hardware for the Yule–Walker method, which calculates the AR coefficient to estimate the PSD, were implemented and evaluated in this work. The evaluation result shows that the proposed method is advantageous in terms of the system-level energy consumption, because the Yule–Walker method can reduce the transmitting data despite the computational amount overhead. The wireless transmitting energy is still dominant in the system level, even if the state-of-the-art transceiver is used in the edge device. Furthermore, the proposed dedicated hardware implementation of the autocorrelation process of the Yule–Walker method achieves small energy dissipation, which is comparable with the FFT hardware. The system-level evaluation result shows that

the proposed method can achieve smaller energy consumption compared with conventional methods using FFT. When the input data length and the AR order are respectively set to 256 and 25, the proposed method can reduce 79% of the energy consumption in practical applications.

REFERENCES

- [1] O. Landsiedel, "Accurate prediction of power consumption in sensor networks," in *Proc. The 2nd IEEE Workshop on Embedded Network Sensors*, Sydney, Australia, May. 2005.
- [2] A. Sabato, C. Niezrecki, G. Fortino, "Wireless MEMS-Based Accelerometer Sensor Boards for Structural Vibration Monitoring: A Review," *IEEE Sensors Journal*, vol.17, no.1, pp. 226–235, 2017.
- [3] B. Li, M. Y. Chow, Y. Tipsuwan, J.C. Hung, "Neural-network-based motor rolling bearing fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol.47, no.16, pp. 1060–1069, October 2000.
- [4] L. Muda, M. Begam, I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques," *Journal of computing*, vol. 2, no. 3, March 2010
- [5] M. V. Kamath, M. A. Watanabe, A. R.M. Upton, "Heart Rate Variability(HRV) Signal Analysis," in *Clinical applications*, CRC Press, 2012, pp.10–22.
- [6] J.M. Spyers-Ashby, P.G. Bain, S.J. Roberts, "A comparison of fast Fourier transform (FFT) and autoregressive (AR) spectral estimation techniques for the analysis of tremor data," *Journal of Neuroscience Methods*, vol.83, no.3, pp. 35–43, October. 1997.
- [7] P. Stoicssa, R. L. Moses, "Spectral analysis of signals," Upper Saddle River, New Jersey, USA: Prentice Hall, pp.90–91, March 2005.
- [8] V. SI, "Model selection and psychological theory: A discussion of the differences between the Akaike information criterion(AIC) and the Bayesian information criterion(BIC)," *Psychological Methods*, vol.17, no.2, pp. 228–243, June. 2012.
- [9] S. M.Kay, and S. L. Marple, JR., "Spectrum Analysis-A Modern Perspective," *Proceedings of The IEEE*, Vol.69, No11, November. 1981.
- [10] K. Vos, *A Fast Implementation of Burg's Method*, August. 2013, [online] Available :www.opuscodec.org/docs/vos_fastburg.pdf.
- [11] Paul Shaman, "Generalized Levinson–Durbin sequences, binomical coefficients and autoregressive estimation," *Journal of Multivariate Analysis*, no.101, pp. 1263–1273, January. 2010.
- [12] J. L. Salinet, N. Masca, P.J.Stafford, G.A. Ng, and F. S.Schindwein, "Three-dimensional dominant frequency mapping using autoregressive spectral analysis of atrial electrograms of patients in persistent atrial fibrillation," *Bio Medical Engineering Online*, vol.15, no.28, pp. 1–15, 2016.
- [13] Texas Instruments. *Implementing Fast Fourier Transform Algorithms of Real-Valued Sequences With the TMS320 DSP Platform*, August. 2001.
- [14] M. Garrido, N.K.Unnikrishnan, K.K.Parhi, *A Serial Commutator Fast Fourier Transform Architecture for Real-Valued Signals* [Online early access] DOI: 10.1109/TSCII.2017.2753941. Published Online: Sep. 18, 2017.
- [15] C. H Yang, T.-H. Yu, D. Markovic, "Power and area minimization of reconfigurable FFT processors: A 3GPP-LTE example," *IEEE J. Solid-State Circuits*, vol.47, no.3, pp.757–768, Mar. 2012.
- [16] A. Wang, A.Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology," *IEEE J. Solid-State Circuits*, vol.41, no.2, pp.483–495, Feb. 2006.
- [17] H. Liu, Z. Sun, D. Tang, H. Huang, T. Kaneko, W. Deng, R. Wu, K. Okada, A. Matsuzawa., "An ADPLL–Centric Bluetooth Low–Energy Transceiver with 2.3mW Interference–Tolerant Hybrid–Loop Receiver and 2.9mW Single–Point Polar Transmitter in 65nm COMS," *IEEE International Solid–State Circuits Conference*, vol.28, no.2, pp. 444–445, 2018.
- [18] C. J. Deepu, C. Heng, Y.Lian, "A Hybrid Data Compression Scheme for Power Reduction in Wireless Sensors for IoT," *IEEE Transactions on Biomedical Circuits and Systems*, vol.11, no.2, pp.245–254, April. 2017.
- [19] M. Tsai, C. Chou, A. A. Wu, "Robust Compressed Analysis Using Subspace–based Dictionary for ECG Telemonitoring Systems," *Proc. IEEE International Workshop on Signal Processing Systems*, October. 2017.
- [20] A. M. Abdulghani, A. J. Casson, E. Rodriguez–Villegas, "Compressive sensing scalpe EEG signals: implementations and practical performance," *Medical Biological Engineering Comput.*, vol.50, pp.1137–1145, 2012.
- [21] Y. Jon, "Adaptive Sampling in Wireless Sensor Networks for Air Monitoring System," *Computing and Control for the Water Industry Leicester Procedia Engineering*, pp.1356–1365, 2015.
- [22] M. Khayatzaedeh, X. Zhang, Y.Lian, "A 0.7–V 17.4–uW 3–lead wireless ECG SoC," *IEEE Transactions on Biomedical Circuits and Systems*, vol.7, no.5, pp.583–592, October. 2012.
- [23] H. Bhamra, Y.–J. Kim, J. Joseph, J. Lynch, O. Z. Gall, H. Mei, C. Meng, J. –W. Tsai, P.Irazoqui, "A 24 uW , batteryless, crystal–free, multimode synchronized SoC bionode for wireless prosthesis control," *IEEE Journal of Solid–State Circuits*, vol.50, no.11, pp.2714–2727, 2015.
- [24] F. Suarez, A. Nozariasbmarz, D. Vashae and M. C. Ozturk, "Designing thermoelectric generators for self-powered wearable electronics", *Energy Environ. Sci*, pp.2099–2113, 2016.



Seiya Yoshida received B.E. degrees in Computer Science and Systems Engineering from Kobe University, Kobe, Japan in 2018. He is currently in the master course at Kobe University. His current research is low-power signal processing for Internet of Things.



Shintaro Izumi respectively received his B.Eng. and M.Eng. degrees in Computer Science and Systems Engineering from Kobe University, Hyogo, Japan, in 2007 and 2008. He received his Ph.D. degree in Engineering from Kobe University in 2011. He was a JSPS research fellow at Kobe University from 2009 to 2011, and an

Assistant Professor in the Organization of Advanced Science and Technology at Kobe University from 2011 to 2018. Since 2018, he has been an Associate Professor in the Institute of Scientific and Industrial Research, Osaka University, Japan. His current research interests include biomedical signal processing, communication protocols, low-power VLSI design, and sensor networks.

He has served as a Chair of IEEE Kansai Section Young Professionals Affinity Group, as a Technical Committee Member for IEEE Biomedical and Life Science Circuits and Systems, as a Student Activity Committee Member for IEEE Kansai Section, and as a Program Committee Member for IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips). He was a recipient of 2010 IEEE SSCS Japan Chapter Young Researchers Award.



Koichi Kajihara received B.E. degrees in Computer Science and Systems Engineering from Kobe University, Kobe, Japan in 2017. He is currently in the master course at Kobe University. His current research is low-power signal processing and device power management for Internet of Things.



Yuji Yano received the M.S. degree in electronic engineering from the Hiroshima University, Hiroshima, Japan, in 2004. In 2004 he joined the SOC Division, Renesas Technology Corp., Hyogo, Japan. Since 2016, he has been a researcher at Graduate School of System Informatics, Kobe University, Hyogo, Japan, where he has

been engaged in the design and development of the ultra-low power (ULP) system LSI solutions.



Hiroshi Kawaguchi received B.Eng. and M.Eng. degrees in electronic engineering from Chiba University, Chiba, Japan, in 1991 and 1993, respectively, and earned a Ph.D. degree in electronic engineering from The University of Tokyo, Tokyo, Japan, in 2006. He joined Konami Corporation, Kobe, Japan, in 1993, where he developed arcade

entertainment systems. He moved to The Institute of Industrial Science, The University of Tokyo, as a Technical Associate in 1996, and was appointed as a Research Associate in 2003. In 2005, he moved to Kobe University, Kobe, Japan. Since 2007, he has been an Associate Professor with The Department of Information Science at that university. He is also a Collaborative Researcher with The Institute of Industrial Science, The University of Tokyo. His current research interests include low-voltage SRAM, RF circuits, and ubiquitous sensor networks. Dr. Kawaguchi was a recipient of the IEEE ISSCC 2004 Takuo Sugano Outstanding Paper Award and the IEEE Kansai Section 2006 Gold Award. He has served as a Design and Implementation of Signal Processing Systems (DISPS) Technical Committee Member for IEEE Signal Processing Society, as a Program Committee Member for IEEE Custom Integrated Circuits Conference (CICC) and IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips), and as an Associate Editor of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences and IPSJ Transactions on System LSI Design Methodology (TSLDM). He is a member of the IEEE, ACM, IEICE, and IPSJ.



Mashahiko Yoshimoto joined the LSI Laboratory, Mitsubishi Electric Corporation, Itami, Japan, in 1977. From 1978 to 1983 he had been engaged in the design of NMOS and CMOS static RAM. Since 1984 he had been involved in the research and development of multimedia ULSI systems. He earned a Ph.D. degree in Electrical Engineering from Nagoya

University, Nagoya, Japan in 1998. Since 2000, he had been a professor of Dept. of Electrical & Electronic System Engineering in Kanazawa University, Japan. Since 2004, he has been a professor of Dept. of Computer and Systems Engineering in Kobe University, Japan. His current activity is focused on the research and development of an ultra low power multimedia and ubiquitous media VLSI systems and a dependable SRAM circuit. He holds on 70 registered patents. He has served on the program committee of the IEEE International Solid State Circuit Conference from 1991 to 1993. Also he served as Guest Editor for special issues on Low-Power System LSI ,IP and Related Technologies of IEICE Transactions in 2004. He was a chair of IEEE SSCS (Solid State Circuits Society) Kansai Chapter from 2009 to 2010. He is also a chair of The IEICE Electronics Society Technical Committee on Integrated Circuits and Devices from 2011-2012. He received the R&D100 awards from the R&D magazine for the development of the DISP and the development of the realtime MPEG2 video encoder chipset in 1990 and 1996, respectively.

TCAS-I-01216-2018

He also received 21th TELECOM System Technology Award in 2006.