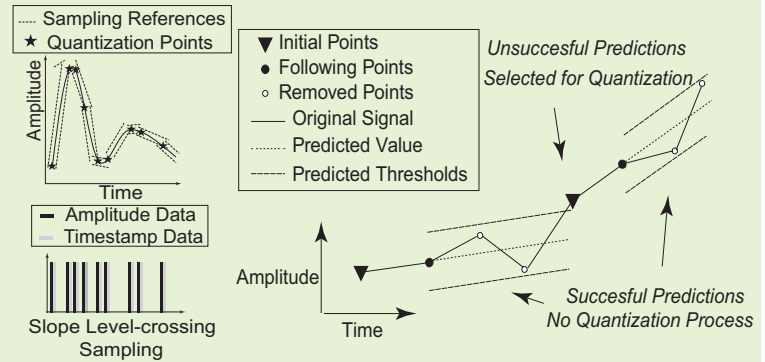


Real-time In-Sensor Slope Level-crossing Sampling for Key Sampling Points Selection for Wearable and IoT Devices

Mario Renteria-Pinon, *Student Member, IEEE*, Xiaochen Tang, *Member, IEEE*, and Wei Tang, *Member, IEEE*

Abstract—This paper presents a slope level-crossing sampling analog-to-digital converter that selects key sampling points for quantization in real time during sensing. It only performs quantization for the turning points in the input analog waveform and provides quantization results of the selected sampling points and timestamps between the selected sampling points. When the input analog signal is sparse, the proposed method reduces digital output data throughput. The processing unit generates a dynamic prediction of the input signal as well as an upper threshold and a lower threshold to form a tracking window. A comparator compares the input signal with the upper and lower threshold to determine if the prediction is successful. Quantization is performed only on unsuccessful predicted sampling points, which are considered key sampling points. A counter records timestamps between the unsuccessful predictions which are the selected key sampling points. The processing unit also includes a neighbor amplitude filter and a slope filter to further reduce the number of sampling points and data throughput when the input signal is associated with high-frequency low-amplitude noise and high-amplitude low-frequency baseline wandering. Reconstruction of the analog signal can be achieved using linear interpolation or polynomial interpolation. The system has been implemented and tested using off-the-shelf components. The simulation and experimental results show that the proposed system can reduce the data throughput and achieve a data compression ratio of 7.1 compared to a conventional successive approximation register analog-to-digital converter with a 10-bit resolution when sampling an ECG signal.

Index Terms—Analog to Digital Converter, Slope Level-crossing Sampling, Prediction-based Sampling, Low Power Circuits.



I. INTRODUCTION

THE development of advanced data acquisition systems and internet-of-things (IoT) technologies have been greatly expected for the next generation wearable biomedical devices and sensing systems, especially in modern human health condition monitoring applications [1]–[3]. For example, the diagnostics of cardiovascular disease (CVD) and cerebrovascular diseases (CeVD) require monitoring Electrocardiogram (ECG) and Electroencephalogram (EEG). However,

current hospitalized monitoring of ECG and EEG costs time and already limited medical resources. Moreover, short-time monitoring may not catch the symptom essential to diagnosis. Thus, long-term real-time ECG and EEG home-monitoring devices play increasingly important roles, which rely on low-power wearable sensors to record and process the analog ECG waveform. Similar applications can be found in other biomedical and internet-of-things applications, such as monitoring brain activities, electrical power consumption, and vibration of buildings and bridges, etc. Therefore long-term real-time data acquisition systems are expected in biomedical and IoT applications with the abilities of digitization, processing, and communication while saving data amount and computing overhead for extended battery lifetime.

Conventional data acquisition systems such as wearable biomedical sensors consist of several main building blocks, including an analog front-end that senses the input analog signals and suppresses the noise, a mid-resolution analog-to-digital converter (ADC), usually realized in Delta-sigma

Manuscript received November 25, 2022; accepted XXX, XX, XXXX. Date of publication XXX, XX, XXXX; date of current version XXX, XX, XXXX. This letter was recommended for publication by Associate Editor XXX XXX. This work was partially supported by United States National Science Foundation grant ECCS-1652944 and ECCS-2015573. The support granted by CONACYT through a graduate fellowship for Mario Renteria-Pinon is acknowledged. (Corresponding author: Wei Tang)

Authors are with the Klipsch School of Electrical and Computer Engineering, New Mexico State University, USA. (email: mario3@nmsu.edu; hypnus@nmsu.edu; wtang@nmsu.edu). Xiaochen Tang and Mario Renteria-Pinon contributed equally to this work

or successive approximation register (SAR) form to convert the analog signal into digital format, and a radio-frequency (RF) transmitter that sends out simultaneous data. Such a system reports the recorded raw data to the central processing station, which is usually at a server on the cloud. The data is then processed by algorithms or human experts, to reconstruct the original analog signal and perform related processing and decision-making. For general applications, a conventional data acquisition system records all raw data of the input signal. However, when the signal is sparse, which means it contains a large part of nonessential data, the general sensor or sensing systems waste efforts on sensing, processing, and communication. For example, in ECG monitoring, since only the fiducial points are important for arrhythmia classification, recording and transmitting all raw data does not help improve the classification performance while increasing the system processing and communication cost.

Moreover, data storage, processing, and communication are also challenging design issues for wearable sensors. Since wearable sensors usually have a small physical size and are powered by batteries, the on-sensor resources are often limited. RF communication is usually one of the most power-hungry units in the system, whose power is proportional to the data throughput of the system. Raw data communication can greatly shorten the battery life of the sensor. The small physical size also restricts the memory size and computation overhead of the on-sensor processing algorithms. In order to solve these problems, efficient sensing methods are proposed to reduce data throughput from the sensor. For example, the event-based level-crossing sampling method was applied for sensing sparse signals [4]–[8]. However, level-crossing sampling cannot detect the turning point of the system, and usually brings insertion-deletion errors [9]. These also bring challenges to wearable sensing and data acquisition systems. More details comparing different nonuniform sampling methods in wearable sensing applications are presented in the next section.

To address the aforementioned problems, this paper proposes a non-uniform sampling method for sensing and data acquisition systems. The proposed method uses active predictions to select the turning points, which are also considered the key sampling points for quantization, instead of using conventional passive sampling and quantization processes. The primary advantage is that by doing so, only critical sampling points are digitized, which reduces digital data throughput. This is important to alleviate the workload of the following digital signal processing circuits. In conventional data acquisition systems, sampling is directly linked to quantization, which is unnecessary in sensing sparse signals. Since quantization consumes much more power than sampling, in the proposed method, power and data throughput can be saved by skipping unnecessary quantization processes. Moreover, the proposed method can be implemented as a fully digital circuit that could be benefited from scaling down using advanced technologies. The circuits can be integrated into the analog-to-digital converter or applied after an ADC, which brings design flexibility. Furthermore, the processing can be achieved in real-time during sensing.

The primary contribution of this paper includes (1) introducing a slope level-crossing sampling mechanism for selecting key sampling points for quantization in order to save output data throughput of the sensor; (2) implementing the proposed slope level-crossing sampling method using discrete components and presenting the measurement results; (3) designing two additional digital filters including neighbor amplitude filter and slope filter to solve the high-amplitude low-frequency baseline wandering problem and low-amplitude high-frequency noise problem. The rest of this paper is organized as follows. Section II presents related work of non-uniform sampling. System Design is introduced in Section III followed by the analysis of the system performance in Section IV. Hardware experimental setup and results are presented in Section V. VI discusses the advantages and compares the results with other recent methods. VII concludes the paper.

II. RELATED WORK

A conventional data acquisition system sensing general signals usually applies for a Nyquist rate ADC. The ADC samples at a fixed sampling rate and converts each sampling into binary digital bits, as shown in Fig. 1 (a). However, many biomedical signals are sparse in the time domain while most valuable information comes from turning points of the analog signal, which are also called fiducial points in the ECG signal. To catch these key sampling points, a Nyquist rate ADC needs a fast sampling clock. Since the sampling rate is fixed, it results in wasted efforts when sampling the non-essential episode of the signal. Especially, the quantization process of the sampling points during the inactive period of the signal costs much more power if the signal is very sparse. Moreover, such wasted quantization generates many unnecessary bits, which overloads the following digital signal processing and communication circuits. Data savings in ADC is more important than power savings since ADC power is usually a small portion of a whole data acquisition system. The primary challenge of the system is to reduce the data throughput while extracting critical information about the waveform during the analog-to-digital conversion using a low-complexity and low-power fashion.

One popular method to relieve this issue is applying the event-based level-crossing sampling that samples using amplitude thresholds instead of a constant clock [6], [10]–[14], as shown in Fig. 1 (b). A recent comprehensive review paper presents the history, advantages, and challenges of the level-crossing method [15]. Using this method, the input signal is always compared with a group of threshold levels. Sampling is performed only when the input signal crosses a threshold, which is called an “event”. A “positive” event means the input signal is increasing while a “negative” event means the input signal is decreasing. Therefore, sampling and quantization are performed simultaneously. Such single-bit binary information (“positive” or “negative”) and the multi-bit timestamp measuring the duration between “events” forms the digital output of the sensor. If the input analog signal’s amplitude variation is below a certain threshold, no sampling and digital quantization are performed. This is an efficient method to save power and sampling data when the input signal is sparse

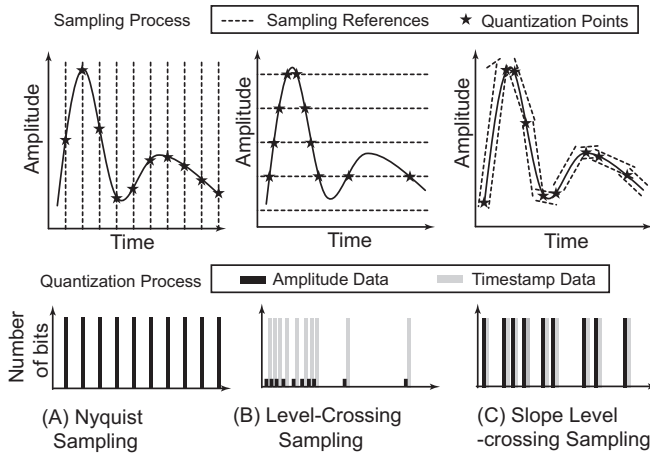


Fig. 1. Differences in Sampling and Quantization Methods: (A) Nyquist Sampling uses a fixed sampling clock and digitizes every sampling result; (B) Level-crossing sampling records the one-bit polarity of each sampling using reference voltages with the associated timestamp in digital format; (C) The proposed slope Level-crossing sampling tracks the input signal and digitizes the turning points and the associated timestamp in digital format.

in the time domain. However, the level-crossing sampling is susceptible to high-amplitude low-frequency baseline wandering, and low-amplitude high-frequency noise. In these two cases, the level-crossing sampling may also generate many unnecessary samplings. Moreover, the level-crossing sampling is not able to identify the turning points (fiducial points) of the input signal since it is only sensitive to the slope but not the slope variation. Another challenge using this method is to record the timing between the level-crossing events. As shown in Fig. 1 (b), the output of level-crossing sampling usually consists of both one-bit binary event data and multi-bit timestamp data. Although some systems use asynchronous processing, most digital processing algorithms still require digitized timing information for each event. Therefore, a level-crossing sampling system usually requires a fast sampling clock to record precise timing data.

Another alternative method to extract key information during the sensing process is using Delta Modulator as the quantizer in low-power sensors [16]. This method converts the input analog signal into a digital bit stream, in which the bit density is proportional to the slope of the input analog signal. Using this method, slope information can be obtained using a counter to identify the morphology of the input signal. So that the total digital data throughput can be reduced. Signal processing algorithms were proposed for such signals [17]. However, the Delta Modulator can not identify the fiducial point since it is only sensitive to the slope but not the slope variation of the input signal. Therefore, a second-order Delta Modulator was invented to localize the turning point of the input signal [18] with the related signal processing algorithm [19]. The second-order Delta modulation converts the input analog signal into a digital pulse stream, whose pulse density is proportional to the input slope variation. Although it could achieve a low-power circuit design, this method is also susceptible to high-frequency noise. The first and the second-

order Delta Modulators can also be combined to compensate for each other to improve the real-time sensing capability [20]. Nevertheless, the primary challenge of the Delta-Modulator-based sensors is that it is difficult to reconstruct the original input waveform due to insertion and deletion errors [21], [22].

A third candidate is a slope-tracking method that only samples the signal when the slope prediction is incorrect [23]–[25]. This method relies on computing the slope difference between samplings and ignores a sampling if its segment has small slope variation compared to the slope of the previous segment. However, if the input signal's slope is varying slightly in one direction for a long time, this method may introduce accumulated errors due to unnecessarily discarding sampling points. So it needs an additional correction circuit to prevent accidental sampling drop. Besides, this method involves complicated analog circuit structures to calculate divisions for obtaining the slope data, so the tolerance of analog noise in the waveform is also a problem in the slope tracking method.

Another related work is piece-wise linear transformation [26] which seeks to use a minimum number of piece-wise linear waveforms to represent the input. The primary goal is to achieve data compression. However, the traditional piece-wise linear transform methods still require the collection of all the input data to form the error channel. The search method has a high computing overhead. These made it not suitable to be applied for low-power sensors running for real-time applications. Another way to save power in reducing transmitting biomedical data volume is to apply the data compression technique. For example, compressed sensing encoders based on sparse measurement matrices [27] and optimal Boolean sampling matrix [28] have been proposed. However, they rely on more data compression techniques and do not take the most advantage of the biomedical signal feature mentioned above.

III. SYSTEM DESIGN

To avoid or alleviate the problems of the methods presented in the previous section, the primary design goal of slope level-crossing sampling is to select only the turning points in the input waveform for quantization during sensing. This is achieved using active predictions of the sampling value. This section presents the algorithm, the circuit implementation, and additional filters to further reduce the effects of high-amplitude low-frequency baseline wandering and low-amplitude, high-frequency noise.

A. Slope Level-crossing Sampling Algorithm

The proposed sensing system records the timing data and amplitude data of the key sampling points of the input analog signal. This process is done in real-time and both the recorded timing and amplitude values are in digital format. The key idea is that the system separates sampling and quantization processes. Although the input signal is always sampled at a fixed sampling rate, only the selected key sampling points are converted into digital data.

When the sensing process begins, the first two analog sampling points are digitized using a standard analog-to-digital conversion process. Then the digital prediction of

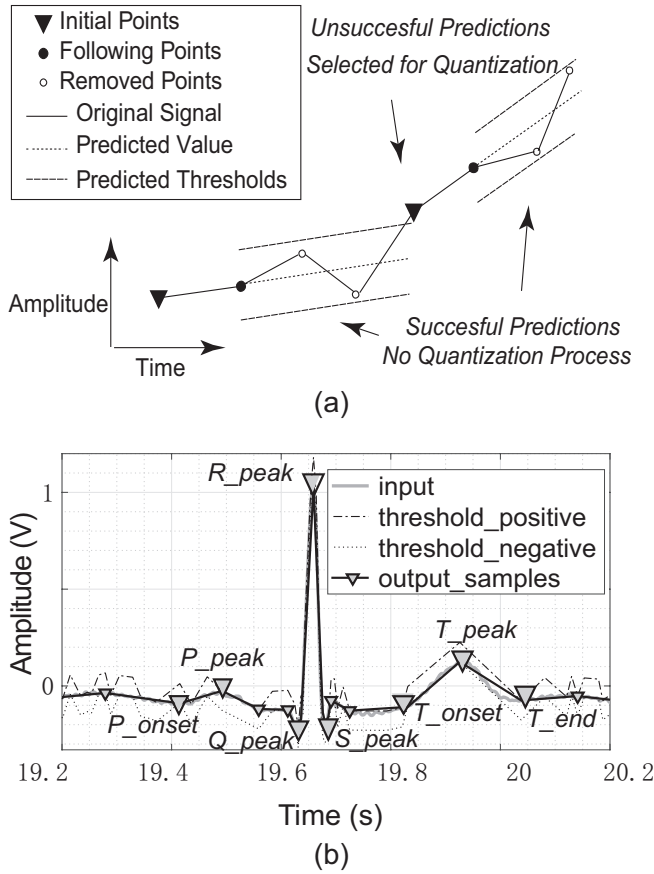


Fig. 2. Slope Level-crossing Sampling Algorithm and Simulation Result. (a) Using prediction and thresholds to select sampling points for quantization only when the prediction is unsuccessful. (b) An example of the selected sampling points using an ECG signal as the input.

the next input analog value is computed using the digital values from the prior two samples with linear extrapolation. Specifically, the *Predicted Digital Value* is calculated using twice the *Last Sampled Digital Value* minus the *Second Last Sampled Digital Value*. As shown in Equation (1).

$$P_D = 2 \times L1_D - L2_D \quad (1)$$

Here P_D is the *Predicted Digital Value*; $L1_D$ is the *Last Sampled Digital Value*; $L2_D$ is the *Second Last Sampled Digital Value*. Equation (1) does not involve an actual multiplication operation since in binary format, multiplying by two can be performed by shifting digital bits to the left. The predicted digital value is then applied to calculate the upper and lower threshold digital values using Equations (2).

$$\begin{cases} UT_D = P_D + \Delta_D \\ LT_D = P_D - \Delta_D \end{cases} \quad (2)$$

Here UT_D is the *Upper Threshold Digital Value*, LT_D is the *Lower Threshold Digital Value*, and Δ_D is the predefined *Delta Step Digital Value*.

The upper and lower threshold values are then converted into analog values using digital-to-analog conversion (DAC). The analog values of these thresholds are compared with the next sampled analog input value. Analog comparisons are

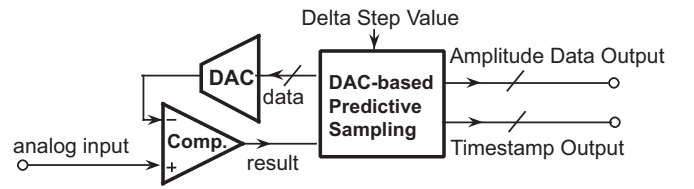
made between the analog input signal, the upper threshold value, and the lower threshold value using one comparator or two comparators. The comparison results decide if the analog input signal is between the *Upper Threshold Analog Value* and the *Lower Threshold Analog Value*, which is represented in Equation (3).

$$LT_A < Input_A < UT_A \quad (3)$$

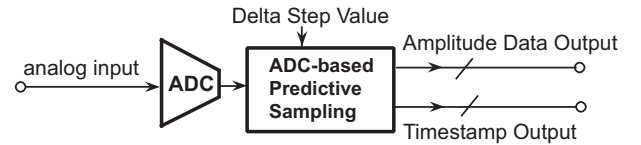
Here LT_A is the *Lower Threshold Analog Value*, $Input_A$ is the *Input Analog Value*, and UT_A is the *Upper Threshold Analog Value*. LT_A and UT_A are obtained by the DAC based on digital values LT_D and UT_D , respectively.

If the input analog value is between the two thresholds, i.e., Equation (3) is valid, the prediction is correct and no analog-to-digital conversion is performed for the input analog signal. The next $L1_D$ is then replaced by the current P_D while the $L2_D$ is replaced by the current $L1_D$. Then the next predicted digital value is calculated using Equation (1). No data is recorded and sent as an output of the system. If the $Input_A$ is not between the two thresholds, i.e., Equation (3) is not invalid, which means the prediction is incorrect, or the $Input_A$ is higher than the UT_A or lower than the LT_A . Then two full analog-to-digital conversions are performed for the input analog samplings to obtain the $L1_D$ and the $L2_D$ to generate the new P_D . Fig. 2 (a) presents a typical slope level-crossing sampling process with both successful and unsuccessful predictions. Using this method, the turning points of the input analog signal can be selected, an example ECG waveform is illustrated in Fig. 2 (b) with the selected turning points using this algorithm.

B. Hardware Implementation



(a) DAC-based Slope Level-crossing Sampling System



(b) ADC-based Slope Level-crossing Sampling System

Fig. 3. Block diagrams of slope level-crossing sampling system implemented using mixed-signal circuits (top) DAC-based implementation (bottom) ADC-based implementation.

The slope level-crossing sampling method can be implemented using either an Analog-to-Digital Converter (ADC) or a Digital-to-Analog Converter (DAC) with a comparator, as shown in Fig. 3. The comparator does not necessarily have to be continuous time, it can be controlled by the ADC clock

and the prediction result. If the prediction is successful, the comparator is not turned on during the quantization sessions so power can be saved. The hardware system selects key sampling points for quantization based on the input analog signal and the digital Delta step value. The output of the system contains both data output and timestamp output. Data output is a multi-bit digital value of the selected sampling point when the prediction is unsuccessful, while the timestamp output records the timestamp between two selected sampling points. This is achieved using a digital timer counting the number of cycles of the sampling clock. Once a prediction is unsuccessful, the digital timer is reset to zero. At the next unsuccessful prediction, the timer sends the recorded timing to the timestamp output.

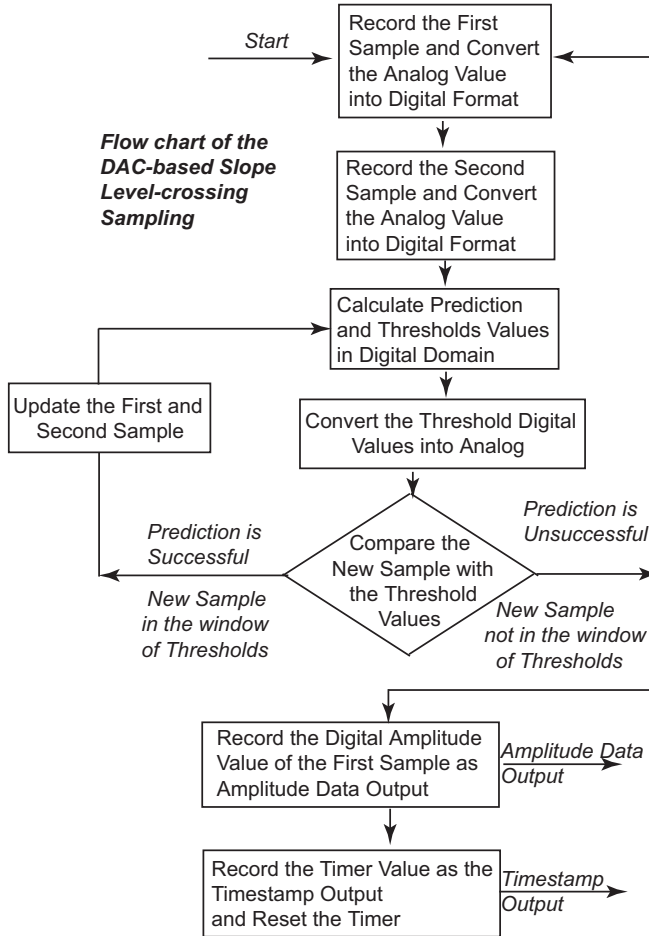


Fig. 4. Flowchart of the DAC-based implementation of slope level-crossing sampling system.

The flow chart of the DAC-based system is shown in Fig. 4. In the DAC-based system, the input analog signal is compared to the analog thresholds that are generated by a digital predicted value and the delta step. The DAC-based prediction and threshold computing circuit are fully digital while the only mixed-signal component is the DAC and the comparator. The system can be implemented using one comparator or two comparators. In a system with two comparators, the digital logic generates both upper and lower thresholds simultaneously. The analog input signal is compared with both thresholds

simultaneously to obtain the result of whether the prediction is successful or not. In a single comparator system, as shown in Fig. 4, the digital logic computes two thresholds consecutively for the DAC and the analog input signal compares with one threshold at a time to decide if the prediction is successful. In the case that the prediction is successful, the digital logic computes the next prediction without sending output data and timestamps. When the prediction is unsuccessful, the digital logic runs a successive approximation logic to obtain the digital value of the analog input while resetting the timer. The digital value and the timer data before resetting are sent to the data and timestamp output.

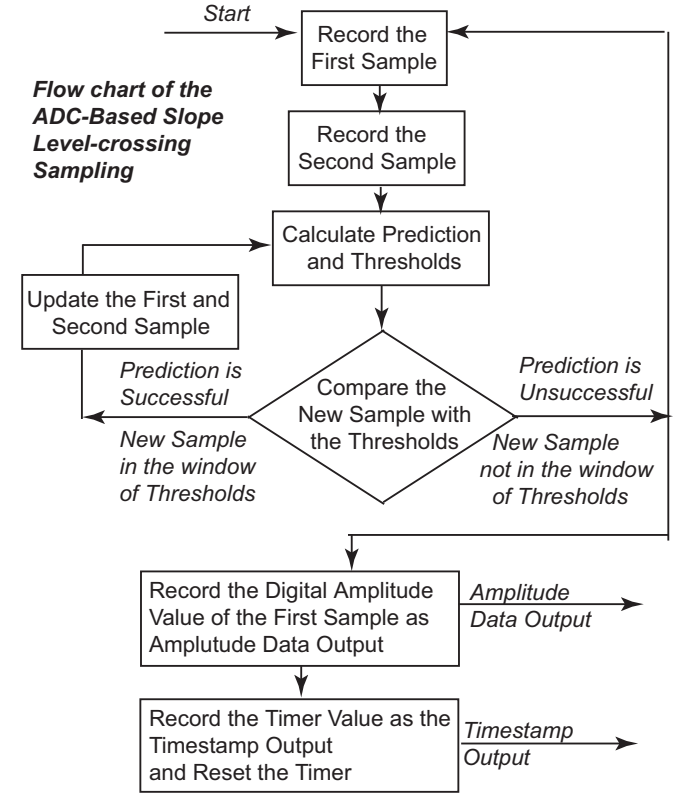


Fig. 5. Flowchart of the ADC-based implementation of slope level-crossing sampling system.

In an ADC-based implementation, the ADC samples at a fixed sampling rate. The digital value of each sampling is processed in the digital domain. When the process starts, the digital values of the first two samplings are used as the $L1_D$ and the $L2_D$. The digital prediction is calculated using Equation (1). The upper and lower thresholds are calculated using Equation (2). The difference between the ADC-based system and the DAC-based system is that in the ADC-based system, the digital values of the upper and lower thresholds are not converted into analog. The comparison is then directly performed in the digital domain using Equation (3). The flow chart of the ADC-based system is shown in Fig. 5. The ADC-based system has a simple digital logic compared to the DAC-based system, which is essentially the same as in [25]. However, since the ADC is running at a fixed sampling rate and performs quantization of every sampling, it costs more

power than the DAC-based implementation. The system can be implemented either with single-ended input or differential input.

C. Additional Filters

The two primary challenges for the proposed slope level-crossing samplings are low-amplitude high-frequency noise and high-amplitude low-frequency baseline wandering. The primary goal is to further reduce the number of sampling points in the original input waveform while keeping a reasonable signal-to-noise ratio in the reconstructed waveform. Two types of filters can be applied to remove the unnecessary samples after the slope level crossing sampling. One filter is a neighbor amplitude filter to remove samples in an episode of low-amplitude high-frequency noise and the other filter is a slope filter to remove samples in a high-amplitude low-frequency baseline wandering. Both filters are implemented digitally after the slope level-crossing sampling system. Although it may require extra memory and digital processing power, these algorithms do not contain complicated arithmetic operations and therefore have low computing overheads.

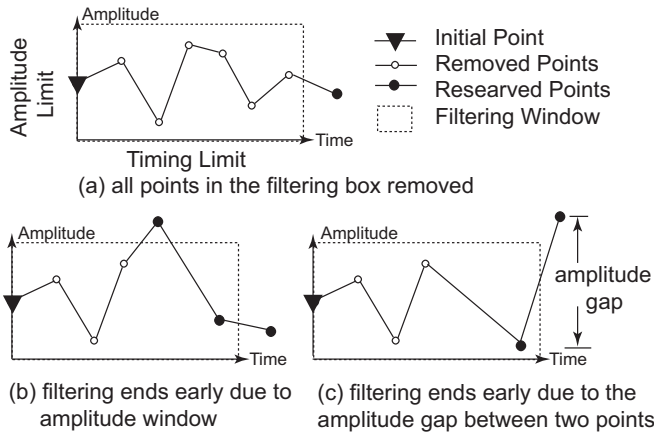


Fig. 6. Neighbor Amplitude Filter can remove more sampling points from the output when there is low-amplitude high-frequency noise in the input signal.

The neighbor amplitude filter removes sampling points that are too close to their neighboring sampling points in both time and amplitude. As shown in Fig. 6, a protection window for a starting sampling point is set with a certain time and amplitude to identify other sampling points after the starting point within the window. All the other sampling points after the starting point that are located within the window are removed. An “exception” is if a sampling point has a high amplitude difference compared to its consecutive sampling point. All sampling points after such a “different point” are preserved and the neighbor amplitude filter stops for the starting sampling point. The neighbor amplitude filter then searches for the next starting sampling point.

The slope filter calculates the slopes of certain sampling points with their prior and later sampling points. As shown in Fig. 7, if the relative difference between the prior slope and the later slope is lower than a slope threshold, this sampling point

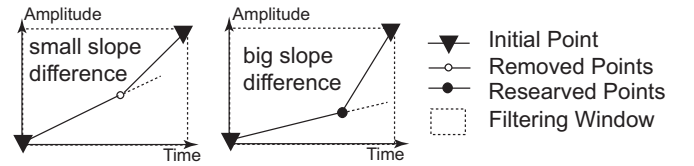


Fig. 7. Slope Filter can remove more sampling points from the output when there is high-amplitude low-frequency baseline wandering in the input signal.

can be removed. Since both the neighbor amplitude filter and slope filter can be cascaded in the system, each time a new filter is applied, the time stamp would be re-calculated using the sampling points that were dropped from the filter. Both timestamp data and the amplitude data of the selected sampling points from the original input waveform are applied for further processing or reconstruction. Since the filters only remove high-amplitude low-frequency baseline wandering and low-amplitude high-frequency noise, it will not affect the timing intervals, such as PR, ST, and QRS, since all fiducial points are recorded.

In reconstruction, the selected sampling points are connected using a first-order approximation to form a piece-wise linear waveform. In many data acquisition applications that involve time-series signals, such as biomedical, speech, and audio signal processing, the original analog waveform could be simplified into a piece-wise-linear waveform without losing key information. Advanced reconstruction methods can also be applied to further reduce the errors introduced by removing certain sampling points, at a cost of increased computing overhead.

IV. PERFORMANCE EVALUATION

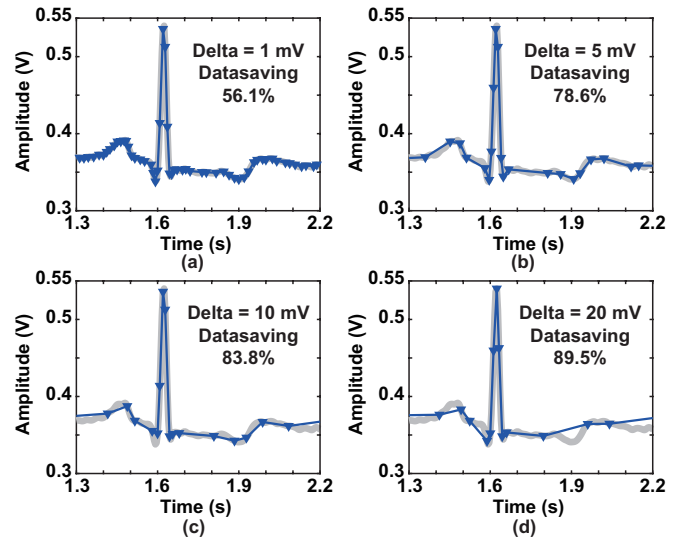


Fig. 8. Trade-off between compression factor and distortion of the reconstructed signal by adjusting Delta level. The input signal is ECG data from the MIT-BIH database. A higher Delta value results in more data saving at a cost of a worse distortion.

The primary advantage of the proposed slope level crossing sampling comes from the reduced digital data throughput. This

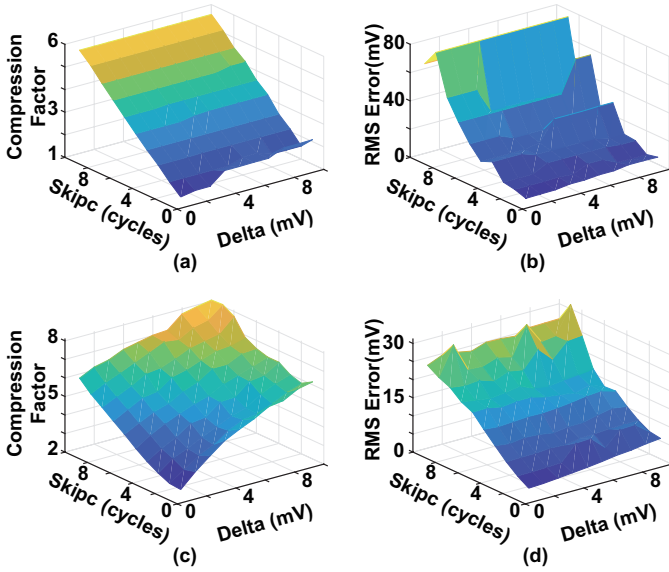


Fig. 9. Trade-off between data compression ratio, root-mean-square error of the reconstructed signal, with the Δ value and the $skipc$ number for different inputs (a) and (b) sinusoidal signal input; (c) and (d) ECG signal input.

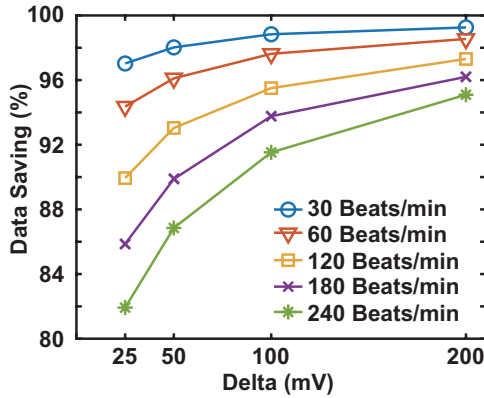


Fig. 10. Data saving performance in ECG data acquisition with different signal sparsity levels.

is achieved by actively predicting the analog input value of the future samplings and the error range of the prediction defined by the predefined Δ_D . The successful prediction results in the reduction of the efforts to perform quantization. In particular, in each sampling, the prediction costs two additional comparisons between the input sampling and the two thresholds, but it has the potential to waive ten comparisons which are required for each sampling in a 10-bit SAR ADC. This property is signal-dependent, which brings more data and power savings when the signal is sparse or has more linear portions than turning portions. Therefore, the performance evaluation depends not only on the system but also on the input signal. We note that although this system could save the power of the ADC since ADC power is usually a small portion of a data acquisition system, the primary advantage is data saving and key sampling points selection during the sensing process.

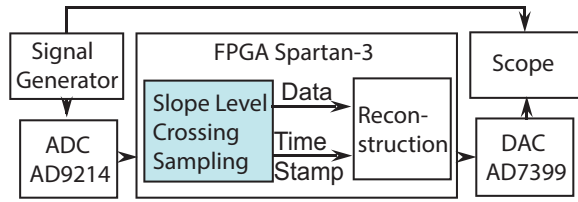
Data saving can be evaluated using the compression factor, which is defined by the ratio of the data amount between the conventional Nyquist sampling and the proposed slope Level-

crossing sampling. We should note that for each sampling point, a 10-bit conventional Nyquist sampling records ten bits of data, while the slope Level-crossing sampling needs to record the 10-bit amplitude data and the additional timestamp data. Here we assume the timestamp data has the same size as the amplitude data. Therefore, for each sampling point, the data amount from the Level-crossing sampling is doubled. Data saving in the slope Level-crossing sampling only comes from the reduced number of sampling points for quantization. In the case that a higher-resolution ADC is used, the proposed system can have even better performance since it can save more bits. This is because for each sampling, if the prediction is successful, the quantization process is skipped and the system does not generate output digital bits. For example, when a 24-bit resolution ADC is applied instead of a 10-bit resolution ADC, each successful prediction can save 24 bits instead of 10 bits.

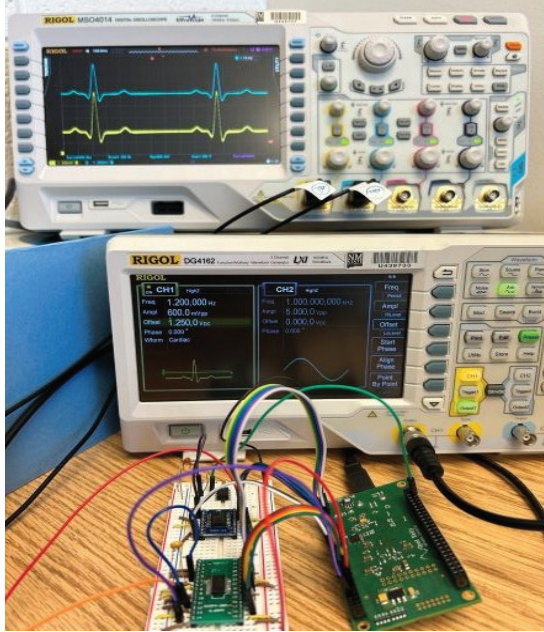
The cost of achieving data savings comes from the additional error when reconstructing the input signal using the selected sampling points. The error performance is evaluated by comparing the reconstructed signals between the proposed slope level crossing sampling and conventional Nyquist rate ADC. The root-mean-square (RMS) error is calculated using the difference between the two reconstructed signals. Compared to a conventional Nyquist rate ADC, since some of the sampling points are skipped for quantization, the reconstructed signal-to-error ratio also depends on the reconstruction method. In this study, we use the simple piece-wise-linear method to reconstruct the signal, which represents the worst-case scenario. When advanced reconstruction methods are applied, we expect the error performance can be improved.

Both the data saving and RMS error depend not only on the input signal but also on the predefined Δ_D . A larger Δ_D results in fewer selected sampling points for quantization, which means a higher data compression factor. However, a larger Δ_D may skip some potentially important sampling points, resulting in a higher RMS error in the reconstructed signal. On the other hand, a smaller Δ_D that causes more sampling points and data throughput reduces the compression ratio but reduces RMS error. Fig. 8 shows simulated ECG waveforms using slope level-crossing sampling with different Delta steps, which shows that a higher compression ratio may result in a distorted reconstructed waveform. Therefore, balancing the trade-off between RMS error and data compression ratio by choosing the Δ_D value is critical in specific applications.

Another parameter applied in evaluating the data saving and RMS error performance is the number of steps between two quantized sampling points for prediction. In the case when the input signal is smooth, the system can use two consecutive sampling points for prediction. However, when the input signal contains high-frequency low amplitude noise, using consecutive sampling points for prediction may result in wrong predictions. In such a case, the system can use a predefined number “skip count” $skipc$ to skip a few sampling points and use the first and the last sampling points between the episode of $skipc$ to calculate the prediction. Such a design brings the benefit of a more accurate prediction, results in fewer sampling points and a higher compression ratio, with



(a) Measurement Setup Diagram



(b) Measurement Setup

Fig. 11. Experiment Setup (a) Measurement Setup diagram. (b) The prototype of the slope level-crossing sampling system was implemented using discrete components and tested using a signal generator and an oscilloscope, which shows both the input and reconstructed waveform.

the cost of a higher RMS error and additional computing overhead. Both the Δ_D and the *skipc* are used for evaluating the system performance of data compression ratio and RMS error for specific input signals.

We use both sinusoidal signal and ECG signal for evaluating the system performance. Fig. 9 (a) and (b) present the simulation result of using a sinusoidal input while Fig. 9 (c) and (d) illustrate the simulation result of ECG input. In both simulations, we study the effects of parameters including the Delta Step value Δ_D and “skip count” *skipc* in terms of their effects in data compression ratio and RMS error. From the simulation result shown in Fig. 9(a), the output data volume can be reduced by more than 50%. Signal difference increases with a more significant value of parameter setup, due to the loss of more data samples from ADC.

A sinusoidal signal is not a good representation of sparse signals in biomedical applications. To evaluate data-saving performance on sparse signals, ECG data from the MIT-BIH Arrhythmia database is also applied as the input signal, and the results are shown in Fig. 9 (c) and (d). The simulation result shows that even with the smallest *skipc* and *delta*, i.e., with the least information loss, we can save more than 56.5% of data. *skipc* and *delta* have similar influences on data saving performance, while *skipc* has a more significant

impact on signal quality than *delta*. A higher data saving can be achieved if the signal has a higher sparse level as shown in Fig. 10 simulated using ECG signals with different beat rates.

V. EXPERIMENTAL SETUP AND RESULTS

The system was implemented and tested using off-the-shelf components. For the processing unit used to generate the slope level-crossing sampling logic, an Opal Kelly XEM 6001 integration module was used. The XEM6001 uses a Xilinx Spartan-6 FPGA. The DAC-based system was implemented using an AD7399 Quad, serial-input 10-bit DAC, along with an LM393 Dual Comparator. The ADC-based system used an AD9214 10-bit 65-MSPS ADC. The testing equipment used included a RIGOL DG4162 waveform generator and a RIGOL MSO4014 Digital Oscilloscope. Fig. 11 shows the components and equipment used. In the oscilloscope a reconstructed waveform, using a Δ_D of 25mV, could be seen in yellow.

For testing a 1.2-Hz 600 mVpp ECG signal was used. Reconstruction of the analog input is done by using an additional DAC, if a prediction is successful the predicted value is sent as a data output. If the prediction fails the actual digital value is sent as the data output. Fig. 12 shows the results from the ADC-based system. The reconstructed signal shows that with a 25mV and 50mV Delta, the system is able to keep the morphology of the waveform for arrhythmia classification. Higher Delta discards important sampling points, as seen on both the 100mV and 200mV waveforms, with the last one completely missing the critical P and T waves.

VI. DISCUSSION

The main difference between slope level-crossing sampling and regular level-crossing sampling is that in slope level-crossing sampling, the thresholds are updated after every sample using two prior sampling values to form virtual slopes. Even if the prediction is successful, the threshold value changes over time and tracks the input analog slope. While in regular level-crossing sampling, the thresholds are calculated using only one sampling value while the thresholds are constants over time if the prediction is successful. Using slope level-crossing sampling, the system achieves a low computing overhead by removing multiplication, since multiplying by 2 can be realized using shift registers. Also, the computing of predictive value and thresholds is performed in the digital domain, which reduces the noise effects of the analog circuitry. Moreover, the proposed method provides a guaranteed maximum error that avoids the accumulated errors introduced by small slope variations using the signal-depending sampling method. The Delta Step in the second-order level-crossing sampling can also be adjusted digitally.

A comparison between typical non-uniform sampling methods for biomedical sensing is summarized in Table I. Compared to regular level-crossing sampling [11], slope level-crossing sampling is able to select the turning points in the waveform in real-time during sensing, which is critical for the following digital signal processing. Delta Modulator [16] can detect the slope of the input analog waveform but cannot precisely identify the turning point. Although the second-order Delta Modulator [20] can detect the turning point, it is difficult

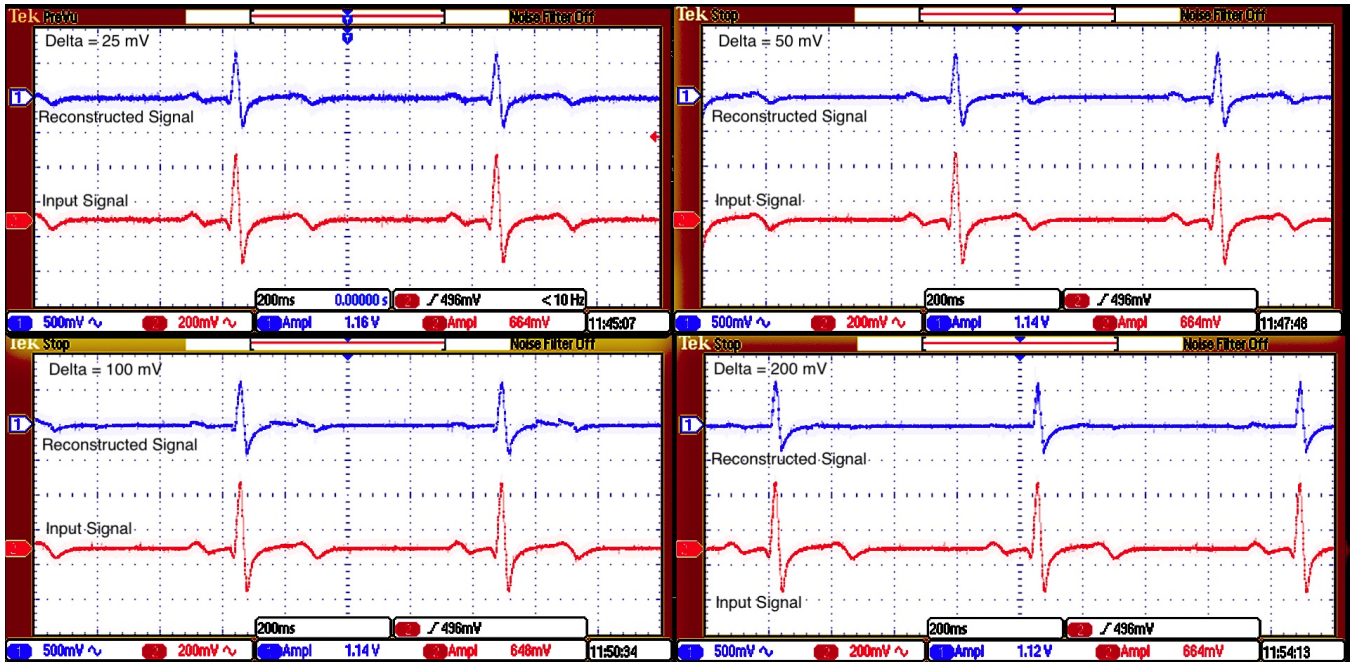


Fig. 12. Experimental Results of the recorded input and reconstructed output signal using different δ values.

TABLE I

COMPARISON BETWEEN THE PROPOSED WORK AND RECENT REFERENCES OF NON-UNIFORM SAMPLING SENSING SYSTEMS.

	This Work	TCASI 2020 [24]	Sensors Journal 2022 [25]	TBCAS 2018 [16]	Sensors Journal 2022 [20]	TCASI 2013 [11]	JBHI 2018 [29]
Method	Slope Level Crossing	Analog Slope-Dependent	ADC + Slope-Dependent	First Order Delta Modulator	Second Order Delta Modulator	Level Crossing Sampling	Digital Second Order Derivative
Fiducial Point	Yes	Yes	Yes	No	No	No	Yes
Sampling Value	Real	Real	Real	No	No	No	Yes
Analog Division	No	Yes	Yes	No	No	No	No
System Implementation	Off-the-Shelf Components	Integrated Circuit	Off-the-Shelf Components	Integrated Circuit	Integrated Circuit	Integrated Circuit	Off-the-Shelf Components
Sampling Rate	1 kHz	1 kHz	1 kHz	1 kHz	1 kHz	20 kHz	1 kHz
Digital Resolution	10 bits	12 bits	12 bits	N/A	N/A	4.8 bit	>12 bit
Compression Factor	6.17	6.1	1-29	N/A	N/A	N/A	N/A

to reconstruct the signal since it uses pulse density modulation, which is not able to precisely measure the amplitude of the turning point. Compared to digital signal processing methods following the ADC such as derivative calculation [29] and slope tracking methods [25], slope level-crossing sampling saves power and computing overhead since the linear portion of the input signal is not quantized. Finally, the proposed level-crossing sampling method is integrated into the digital circuit inside the ADC, which avoids complicated analog division circuitry in [24].

Drifting is a problem of reconstructing non-uniformed sampling systems such as in level-crossing sampling. However, the proposed method solved this problem since it is essentially a constant sampling system. There is a fixed-rate sampling clock that always samples the signal at the sampling rate. The difference between a conventional Nyquist sampling system and the proposed system is that the system automatically

skips the quantization process for certain sampling points for saving data. The recorded signal can be reconstructed without any drift errors. Even if there are errors during sampling or quantization, the reconstructed signal will have an error point, which could also happen in other systems with a constant sampling rate, but it does not affect the following points.

Both IoT systems and wearable systems face the similar challenge of limited power supply while tracking sparse signals. In IoT applications, using the proposed method, the variation of the input signal can be monitored more efficiently since only turning points of the input signal are recorded, this can greatly reduce the power consumption of data communication and signal processing. For example, when tracking the maximum and minimum value of the input, which are both turning points, the system can simply select the maximum and minimum value from the recorded turning points, without paying attention to the slope or linear portion of the input

signal. For low-frequency input signals, the system can apply a slower clock to save power. As long as the signal is sparse (as in most IoT applications), the frequency of the input signal does not change the advantage of the proposed method.

VII. CONCLUSION

This paper presents a slope level-crossing sampling method for reducing data throughput in sensors for IoT devices. The system is able to select turning points in the input waveform for quantization in real-time during sensing. Two additional filters are also presented to further reduce the number of sampling points during baseline wandering and noise. Simulation results report the trade-off between data compression ratio and root-mean-square error as a function of the Delta threshold and the skip count during sensing. Hardware implementation is achieved using off-the-shelf components for a proof-of-concept demonstration. The proposed system provides unique features and reasonable performance compared to other non-uniform sampling methods for sensing and data acquisition systems.

REFERENCES

- [1] J. De Roose, H. Xin, M. Andraud, P. J. Harpe, and M. Verhelst, "Flexible and self-adaptive sense-and-compress for sub-microwatt always-on sensory recording," in *ESSCIRC 2018 - IEEE 44th European Solid State Circuits Conference (ESSCIRC)*, 2018, pp. 282–285.
- [2] Y. Liu, P. M. Furth, and W. Tang, "Hardware-efficient delta sigma-based digital signal processing circuits for the internet-of-things," *Journal of Low Power Electronics and Applications*, vol. 5, no. 4, p. 234, 2015. [Online]. Available: <http://www.mdpi.com/2079-9268/5/4/234>
- [3] Y. He, F. Corradi, C. Shi, S. van der Ven, M. Timmermans, J. Stuijt, P. Dettler, P. Harpe, L. Lindeboom, E. Hermeling, G. Langereis, E. Chicca, and Y.-H. Liu, "An implantable neuromorphic sensing system featuring near-sensor computation and send-on-delta transmission for wireless neural sensing of peripheral nerves," *IEEE Journal of Solid-State Circuits*, vol. 57, no. 10, pp. 3058–3070, 2022.
- [4] P. Martínez-Nuevo, S. Patil, and Y. Tsvividis, "Derivative level-crossing sampling," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 1, pp. 11–15, 2015.
- [5] J. Van Assche and G. Gielen, "Power efficiency comparison of event-driven and fixed-rate signal conversion and compression for biomedical applications," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 4, pp. 746–756, 2020.
- [6] C. Weltin-Wu and Y. Tsvividis, "Event-driven clockless level-crossing adc with signal-dependent adaptive resolution," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 9, pp. 2180–2190, 2013.
- [7] H. Wang, F. Schembari, and R. B. Staszewski, "An event-driven quasi-level-crossing delta modulator based on residue quantization," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 2, pp. 298–311, 2020.
- [8] J. Van Assche and G. Gielen, "A 10.4-enob 0.92-5.38 w event-driven level-crossing adc with adaptive clocking for time-sparse edge applications," in *ESSCIRC 2022- IEEE 48th European Solid State Circuits Conference (ESSCIRC)*, 2022, pp. 261–264.
- [9] Q. Hu, C. Yi, J. Kliever, and W. Tang, "Asynchronous communication for wireless sensors using ultra wideband impulse radio," in *2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2015, pp. 1–4.
- [10] B. Schell and Y. Tsvividis, "A Continuous-Time ADC/DSP/DAC System With No Clock and With Activity-Dependent Power Dissipation," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 11, pp. 2472–2481, 2008.
- [11] W. Tang, A. Osman, D. Kim, B. Goldstein, C. Huang, B. Martini, V. A. Pieribone, and E. Culurciello, "Continuous Time Level Crossing Sampling ADC for Bio-Potential Recording Systems," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 6, pp. 1407–1418, 2013.
- [12] Y. Li, D. Zhao, and W. A. Serdijn, "A Sub-Microwatt Asynchronous Level-Crossing ADC for Biomedical Applications," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 7, no. 2, pp. 149–157, 2013.
- [13] N. Ravanshad, H. Rezaee-Dehsorkh, R. Lotfi, and Y. Lian, "A Level-Crossing Based QRS-Detection Algorithm for Wearable ECG Sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 1, pp. 183–192, 2014.
- [14] X. Zhang, Z. Zhang, Y. Li, C. Liu, Y. X. Guo, and Y. Lian, "A 2.89 μ W Dry-Electrode Enabled Clockless Wireless ECG SoC for Wearable Applications," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 10, pp. 2287–2298, 2016.
- [15] Y. Zhao and Y. Lian, "Event-driven circuits and systems: A promising low power technique for intelligent sensors in aiOT era," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 7, pp. 3122–3128, 2022.
- [16] X. Tang, Q. Hu, and W. Tang, "A Real-Time QRS Detection System With PR/RT Interval and ST Segment Measurements for Wearable ECG Sensors Using Parallel Delta Modulators," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 4, pp. 751–761, 2018.
- [17] X. Tang, Z. Ma, Q. Hu, and W. Tang, "A Real-Time Arrhythmia Heartbeats Classification Algorithm Using Parallel Delta Modulations and Rotated Linear-Kernel Support Vector Machines," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 4, pp. 978–986, 2020.
- [18] X. Tang and W. Tang, "A 151nW Second-Order Ternary Delta Modulator for ECG Slope Variation Measurement with Baseline Wandering Resilience," in *2020 IEEE Custom Integrated Circuits Conference (CICC)*, 2020, pp. 1–4.
- [19] —, "An ecg delineation and arrhythmia classification system using slope variation measurement by ternary second-order delta modulators for wearable ecg sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 15, no. 5, pp. 1053–1065, 2021.
- [20] X. Tang, S. Liu, P. Reviriego, F. Lombardi, and W. Tang, "A near-sensor ecg delineation and arrhythmia classification system," *IEEE Sensors Journal*, vol. 22, no. 14, pp. 14 217–14 227, 2022.
- [21] Y. Liu, W. Tang, and D. G. Mitchell, "Efficient Implementation of a Threshold Modified Min-Sum Algorithm for LDPC Decoders," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. Early Access, pp. 1–5, 2020.
- [22] Y. Liu, X. Tang, D. G. M. Mitchell, and W. Tang, "Ternary ldpc error correction for arrhythmia classification in wireless wearable electrocardiogram sensors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 69, no. 1, pp. 389–400, 2022.
- [23] H. Mafi, M. Yargholi, M. Yavari, and S. Mirabbasi, "Digital Calibration of Elements Mismatch in Multirate Predictive SAR ADCs," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 12, pp. 4571–4581, 2019.
- [24] E. H. Hafshejani, M. Elmi, N. TaheriNejad, A. Fotowat-Ahmady, and S. Mirabbasi, "A low-power signal-dependent sampling technique: Analysis, implementation, and applications," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 12, pp. 4334–4347, 2020.
- [25] E. Hadizadeh Hafshejani, N. TaheriNejad, R. Rabbani, Z. Azizi, S. Mohin, A. Fotowat-Ahmady, and S. Mirabbasi, "Self-Aware Data Processing for Power Saving in Resource-Constrained IoT Cyber-Physical Systems," *IEEE Sensors Journal*, vol. 22, no. 4, pp. 3648–3659, 2022.
- [26] K. Konstantinides and B. Natarajan, "An architecture for lossy compression of waveforms using piecewise-linear approximation," *IEEE Transactions on Signal Processing*, vol. 42, no. 9, pp. 2449–2454, 1994.
- [27] W. Zhao, B. Sun, T. Wu, and Z. Yang, "On-chip neural data compression based on compressed sensing with sparse sensing matrices," *IEEE transactions on biomedical circuits and systems*, vol. 12, no. 1, pp. 242–254, 2018.
- [28] Y. Wang, X. Li, K. Xu, F. Ren, and H. Yu, "Data-driven sampling matrix boolean optimization for energy-efficient biomedical signal acquisition by compressive sensing," *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 2, pp. 255–266, 2016.
- [29] J. M. Bote, J. Recas, F. Rincón, D. Atienza, and R. Hermida, "A modular low-complexity ecg delineation algorithm for real-time embedded systems," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 2, pp. 429–441, 2018.