Real-time Biosignal Recording and Machine-Learning Analysis System

Hanrui Li*, Junzhe Wang*, Shiqi Zhao, Fengshi Tian, Jie Yang, and Mohamad Sawan
School of Engineering, Westlake University
Hangzhou, China
{yangjie,sawan}@westlake.edu.cn

***Abstract*—Biosignal recording and processing systems (BRPSs) are in high demand for numerous applications such as brain-machine interfaces, healthcare, and other clinical applications. However, conventional BRPS can only perform simple operations, such as filtering and denoising, but cannot perform robust machine learning-based analyses in real time. This paper proposes an intelligent BRPS that consists of a signal recording front-end for biosignal acquisition, control and visualization hub, and FPGA board for machine learning acceleration. High-speed Ethernet and PCIe interfaces were used to increase the data transmission rate of the system. Moreover, the integrated accelerator in the FPGA is designed in a single-instruction-multiple-data (SIMD) mode to perform complex machine learning operations in parallel to speed up data-processing tasks. The proposed system is validated for various applications, including EEG-based seizure prediction with a convolutional neural network (CNN), EMG-based gesture recognition with a spiking neural network (SNN), and ECG-based arrhythmia detection with a binary neural network (BNN). Experimental results reveal that this system takes 13 ms to process one-second electrophysiological signals at 512 Hz and 32 channels, thus achieving real-time performance. The proposed BRPS is an open-source and expandable system, and different machine-learning approaches can be configured for diverse applications.**

***Index Terms*—Signal Processing System, Biosignal, Graphical User Interface, AI accelerator, FPGA, Neural Networks**

I. Introduction

Biosignals play a significant role in health monitoring and disease diagnosis because they provide critical information about a person's physiological, pathophysiological, and emotional states. With the emergence of machine learning algorithms, intelligent biosignal processing has become available, resulting in various sought-after biomedical applications. For example, Attia et al. proposed a rapid, inexpensive method based on a convolutional neural network (CNN) to detect the signature of atrial fibrillation using an electrocardiogram (ECG) [1]. Borhani et al. adopted a deep-learning-based electroencephalogram (EEG) method for seizure detection and obtained excellent results [2]. Ghassemi et al. built an attention-based hybrid CNN-RNN network to fully utilize the sequential nature of electromyogram (EMG) signals for hand gesture recognition [3]. However, the studies mentioned above cannot be applied to the healthcare market or clinic without a real-time biosignal processing system.

Much effort has been devoted to developing real-time electrophysiological signal recording and processing systems. In 2017, Patel et al proposed an open-source software platform to achieve real-time biosignal acquisition and control in biological experiments [4]. Although the open-source software provides good flexibility and allows users to implement customized protocols and functions, the system performance deteriorates if all processing relies on a general-purpose operating system and CPU. In 2018, Pirog et al. designed "Multimed", a configurable and multi-channel hardware system using FPGA acceleration to lower signal processing latency for real-time analysis [5]. However, with hardware description languages, the flexibility and expandability of the system are greatly limited. More recently, Erickson et al. proposed a multi-channel biosystem on gastrointestinal (GI) field for wireless real-time signal visualization together with an integrated microcontroller that is well suited for portable, ambulatory applications [6]. However, its application may be limited in the neural and cardiac fields, where high sampling frequencies($\geqslant 500Hz$) and high bandwidth streaming are required.

This study proposes an intelligent biosignal recording and processing system (BRPS), which contains a recording front-end to acquire various biosignal, a control and visualization hub based on Python Qt for system control and signal visualization, and a real-time artificial intelligence (AI) processor on an FPGA development board. The FPGA is flexible and can be configured for different neural-network algorithms that can be freely utilized with EEG, EMG, and ECG signals. When connected to biosignal acquisition devices through TCP/IP, recording and processing functions allow users to monitor live-stream biosignals and apply real-time analyses.

The remainder of this paper is organized as follows. Section II presents an integrated system overview, together with individual descriptions of separate parts of the system. In Section III, we present our implementation of the system and algorithm details. This section also presents various applications of this system to EEG, ECG, and EMG. Finally, the conclusions are presented in Section IV.

II. System Description

A. System Overview

Fig. 1 depicts the architecture of the proposed system. It consists mainly of a multichannel recording device front-end, control and visualization hub for system control and

*Equal contributor.

This work is funded by Zhengjiang Key R&D Program Project No.2021C03002.

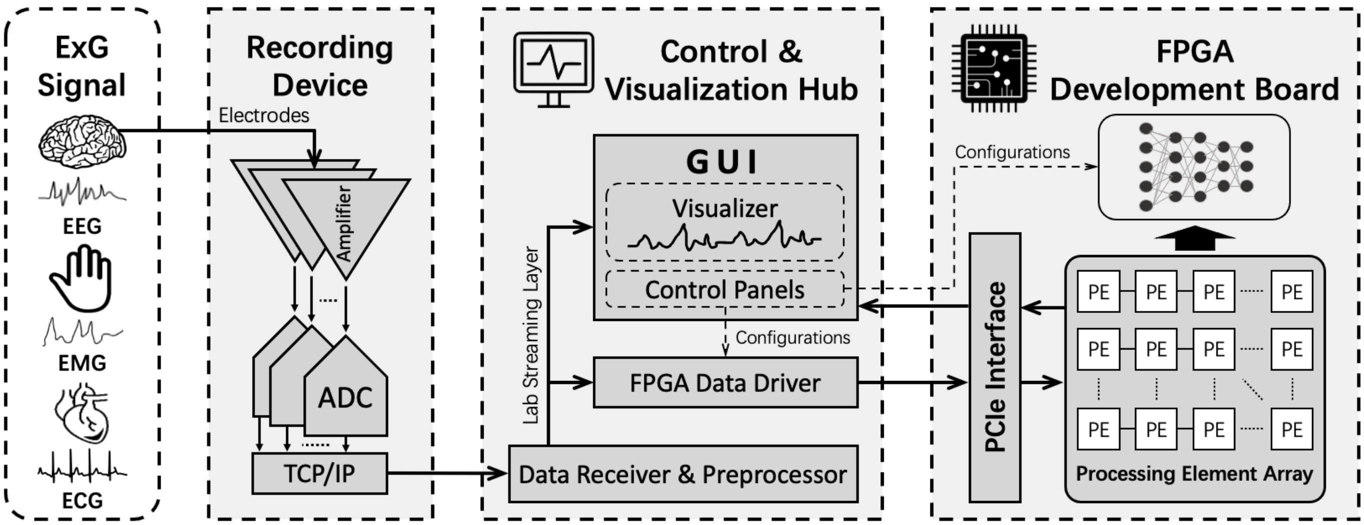

Fig. 1. The proposed ERPS contains three main components: the Recording Device, the Control and Visualization Hub, and the FPGA AI Accelerator. All standard system features, such as the visualizer, are implemented as plug-ins. The plug-in is embedded in the main window through which users can control the plug-in and send instructions to configure FPGA Development board.

monitoring, and AI accelerator implemented on an FPGA board. The biosignals acquired by the recording device are transferred to the control and visualization hub via the TCP/IP protocol. The hub receives real-time biosignals and controls the communications and settings of the AI accelerator by uploading different neural-network parameters to the FPGA board. A high-throughput PCIe interface facilitates communication between the FPGA board and hub, reducing data transmission latency. A programmable AI accelerator is implemented in the FPGA device, which can perform the operations of different types of neural networks. The AI accelerator consists of a two-dimensional (2D) processing element (PE) array that performs the inner product and pooling operations, matrix-multiplication, pooling, and other necessary operations required by neural networks. The 2D PE array performs operations in a single instruction multiple data (SIMD) mode to accelerate data processing. The results delivered by the neural network are fetched and later displayed by the GUI.

B. Recording Device

A BrainVision Recorder (Brain Products GmbH, Germany) is used as the signal-recording device for the proposed system. The maximum number of recording channels is 160, with a recording bandwidth between DC and 7500 Hz.Each recording channel consists of a high-end amplifier and an analog-to-digital converter (ADC). The amplifier's input noise is $2\mu V_{pp}$ and its common-mode rejection rate is 100 dB. The amplified signals are digitized by a 24-bit ADC, after which the signals are transferred to a data processing computer that features a control and visualization hub.

C. Control and Visualization Hub

Fig. 2 illustrates the structure of the control and visualization hub in the proposed system. It mainly consists of

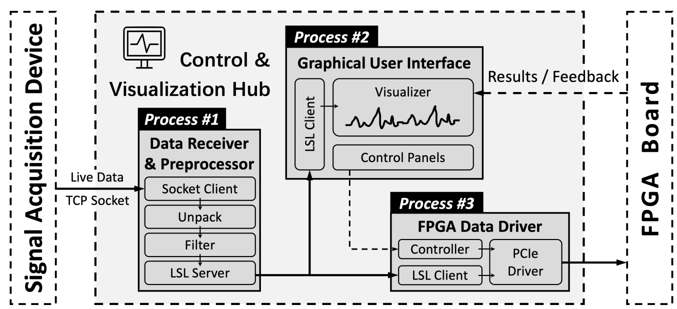

Fig. 2. Control and Visualization Hub

3 functional blocks, data receiver and preprocessor, GUI, and FPGA data driver. To achieve an optimized real-time performance, multiprocessing and multithread programming methods are utilized. The three functional blocks operate through independent processes to handle the acquired biosignal data in a pipeline fashion.

In Process 1, a TCP socket is established between the data receiver and the signal acquisition device. The receiver continuously reads the real-time stream from the socket after the connected device declares the start of transmission. Then, a real-time filter can be optionally added to the raw data before it is sent to any other process via the lab streaming layer (LSL) protocol.

Process 2 mainly entails hosting the graphic user interface (GUI) to help configure the system, visualizing the signals, and displaying processing results from the FPGA board. Fig. 3 shows a screenshot of the graphical user interface (GUI). It is adapted from the App-SigVisualizer with customization to fit our system. Multiple widgets such as algorithm control and operation control are added to help download dedicated network parameters and send commands to the FPGA board. A child thread hosting an LSL client is defined within Process

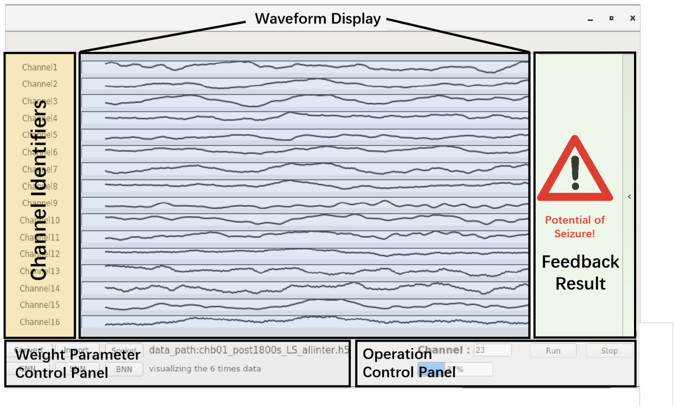

Fig. 3. The layout of the GUI. The Waveform Display visualizes the live stream of the biosignal with channels separated; The Weight Parameter Control Panel provides an option for users to select a desired algorithm out of CNN, SNN and BNN to load the weights; The Operation Control Panel controls the run/stop of the program and defines the number of channels; The Feedback Result presents the results after calculation in the FPGA Board (EEG epilepsy seizure prediction used as an example in the figure)

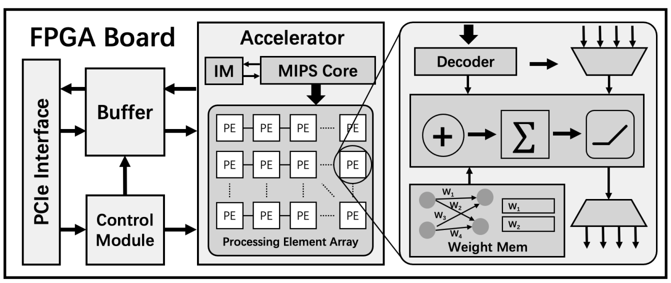

Fig. 4. The FPGA architecture of the proposed system. IM: Instruction Memory; MIPS: Microprocessor without Interlocked Pipelined Stages. According to different algorithms, the control module can give different instructions to the FPGA through RTL or HLS to deploy the intended structure.

2 to receive the signal data from Process 1.

Finally, in Process 3, another LSL client is set up, receiving live biosignal data from Process 1. The FPGA data driver then selects and forwards the data to the FPGA board via a high-throughput PCIe interface.

D. FPGA Platform

The architecture of the FPGA acceleration board is illustrated in Fig. 4. It mainly consists of a PCIe interface, data buffer, control module, and machine learning accelerator. The data from the PCIe interface are automatically stored in the data buffer, which is implemented with dual-clock SRAM memory. The control module receives commands from the PCIe interface to control the data exchange process between the data buffer and the accelerator. The accelerator comprises an instruction memory (IM), MIPS core, and 2D processing element (PE) array. The IM stores instructions that control the operations of both PE and the MIPS core. The instructions are loaded to the IM before the system starts, and can be refreshed based on different neural network types, as dictated by the control hub. The MIPS core fetches the instructions from the IM and broadcasts them to the PE array to carry out neural network operations in a single-instruction multiple-data mode. Inside each PE, the local memory is used to store the weight parameters of the neural network. The control hub preloads the weights before the system starts operating. Arithmetic logic units (ALUs) that perform multiplication, accumulation, and activation functions are also integrated. MUX and DEMUX units are used in the PE to control data communication with neighboring PEs. All PE operations are controlled by signals decoded from the instructions.

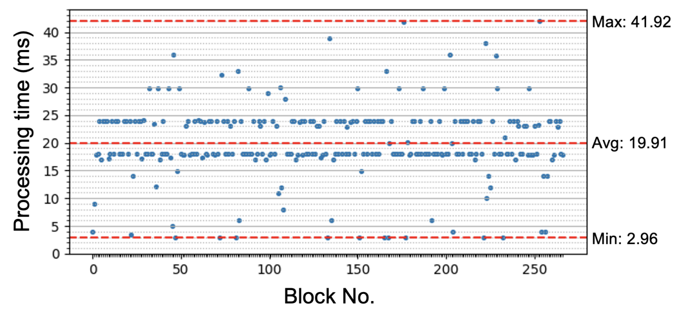

Fig. 5. Real-time Latency Analysis for 268 Iterations

III. Implementation and Measurement

A. Implementation

For the system implementation, a BrainVision product was adopted as the signal acquisition device for our system. This involves 24-bit electrophysiological signal recording on 32 channels, with a sampling rate of up to 100 kHz. The control and visualization hub is implemented using Python on a Windows 10 PC with an Intel Core i5-7500 CPU and 8G memory. The FPGA platform adopted in the system is an Xilinx ALVEO U250 Card with a PCIe Gen 3 interface which provides an 8.0GT/s bandwidth for the communication with the PC.Moreover, the FPGA-featured AI accelerator consumes 161,348 LUTs (9.34%), 95,885 FFs (2.78%), and 773.5 kB BRAMS.

B. Experimental Results

To evaluate the real-time performance of our system, we measured and analyzed the delay between data acquisition and visualization for 268 iterations. Fig. 5 depicts the result, showing an average real-time display latency of approximately 20 ms. Furthermore, to demonstrate the intelligent signal processing performance of the proposed system and benchmark with other works, we evaluated the system with EEG, ECG, and EMG analysis tasks using the CHB-MIT [8], MIT-BIH arrhythmia [9], and Nina Pro DB1 [10] datasets, respectively.

First, we implemented the binary convolutional neural network (BCNN) architecture proposed in [11] to perform EEG-based seizure prediction. The network architecture is illustrated in Fig. 6 (a). The network was first trained with CHB-MIT, and the weight parameters were fully mapped onto the FPGA through the control and visualization hub. The measured average processing delay of the FPGA was 13 ms when processing one-second EEG signals at 512 Hz. An accuracy of 94.26% seizure prediction was achieved. The

TABLE I
PERFORMANCE COMPARISON WITH OTHER OPEN-SOURCE SYSTEMS

	Open Ephys [7]	RTXI [4]	Multimed [5]	This work
Processing Latency	20ms	74.8ms	10-20 ms	13ms
Processor	PC	PC	FPGA	FPGA
Development Environment	JUCE, C++	MATLAB, C++	VHDL	Python, C++, Verilog
Supported Algorithms	-	Preprocessing function	SNN	CNN, BNN, SNN Preprocessing function
Compatible Signal Types	Intracellular and Extracellular electrophysiology, EEG	EMG, Neural Signal	EEG, EMG, ECG Pancreatic Cells	EEG, EMG, ECG
Applications	Closed-loop stimulation of hippocampus	Dynamic-clamp technique Distorted auditory feedback Investigate feedback-driven tACS	Dectect electrical activity of pancreatic islets Long term continuous processing	Epileptic seizure prediction Hand gesture recognition Arrhythmia detection

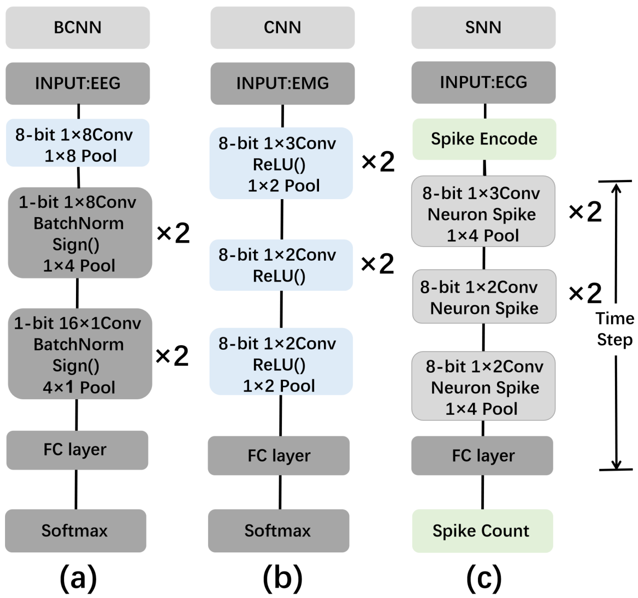

Fig. 6. Neural network structure of the proposed system. (a) BNN-based EEG seizure prediction problem, (b) CNN-based EMG gesture recognition problem, (c) SNN-based ECG arrhythmia detection problem.

CNN architecture shown in Fig. 6 (b) was used for the EMG-based gesture recognition. We used an 8-bit quantized CNN to achieve an accuracy of 86.39% for five gesture classification problems. Spiking CNN architecture shown in Fig. 6 (c) was implemented on an AI accelerator for the ECG-based arrhythmia detection. The MIT-BIH arrhythmia dataset was used for training and evaluation. With the time step set to 8, four types of arrhythmia can be detected with an accuracy of 84.21%.

Table I shows a comparison between the proposed BRPS and other open-source systems. With the help of the FPGA-based acceleration, our experiments indicate that the processing speed is increased compared to other works. In addition, the control and visualization hub was implemented under Python environment, which facilitated the training and parameter conversion process due to its compatibility with existing machine learning frameworks such as TensorFlow or PyTorch. Our system is validated with various machine learning algorithms for different applications without deteriorating the signal analysis accuracy.

IV. CONCLUSION

This paper presents a real-time biosignal recording and processing system for various applications. It employs an open-source and python-based control and visualization hub for biosignal visualization and system configuration and an FPGA for machine learning processing acceleration. Experimental results show that the system achieves low visualization and processing latency. Various neural network-based biosignal analyses can be carried out with the proposed system in high accuracy with real-time performance.

REFERENCES

[1] Z. I. Attia, P. A. Noseworthy, F. Lopez-Jimenez, S. J. Asirvatham, A. J. Deshmukh, *et al.*, "An artificial intelligence-enabled ecg algorithm for the identification of patients with atrial fibrillation during sinus rhythm: a retrospective analysis of outcome prediction," *The Lancet*, vol. 394, no. 10201, pp. 861–867, 2019.

[2] S. Borhani, R. Abiri, Y. Jiang, T. Berger, and X. Zhao, "Brain connectivity evaluation during selective attention using eeg-based brain-computer interface," *Brain-Computer Interfaces*, vol. 6, no. 1-2, pp. 25–35, 2019.

[3] M. Ghassemi, K. Triandafilou, A. Barry, M. E. Stoykov, E. Roth, *et al.*, "Development of an emg-controlled serious game for rehabilitation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 2, pp. 283–292, 2019.

[4] Y. A. Patel, A. George, A. D. Dorval, J. A. White, D. J. Christini, and R. J. Butera, "Hard real-time closed-loop electrophysiology with the real-time experiment interface (rtxi)," *PLoS computational biology*, vol. 13, no. 5, p. e1005430, 2017.

[5] A. Pirog, Y. Bornat, R. Perrier, M. Raoux, M. Jaffredo, *et al.*, "Multimed: an integrated, multi-application platform for the real-time recording and sub-millisecond processing of biosignals," *Sensors*, vol. 18, no. 7, p. 2099, 2018.

[6] J. C. Erickson, B. Reed, J. Wharton, U. Thapa, J. Robey, and R. Shrestha, "Open-source 128-channel bioamplifier module for ambulatory monitoring of gastrointestinal electrical activity," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 4429–4432, IEEE, 2020.

[7] J. H. Siegle, A. C. López, Y. A. Patel, K. Abramov, S. Ohayon, and J. Voigts, "Open ephys: an open-source, plugin-based platform for multichannel electrophysiology," *Journal of neural engineering*, vol. 14, no. 4, p. 045003, 2017.

[8] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, *et al.*, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[9] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[10] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, *et al.*, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Scientific data*, vol. 1, no. 1, pp. 1–13, 2014.

[11] S. Zhao, J. Yang, Y. Xu, and M. Sawan, "Binary single-dimensional convolutional neural network for seizure prediction," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2020.