Embedded Self Organizing Systems (Vol 11. No 10. 2024) (pp.20-24)

Embedded Selforganizing Systems

Design and Implementation of Biosignal Simulator

Yalalt Otgonsaikhan
Department of Electric Technique
Power Engineering School
Mongolian University of Science and Technology
Ulaanbaatar Mongolia
o.yalalt@must.edu.mn

Bat-Erdene Byambasuren
Department of Electric Technique
Power Engineering School
Mongolian University of Science and Technology
Ulaanbaatar Mongolia
baterdene@must.edu.mn

***Abstract*— Bio-signal simulators are vital in biomedical engineering for replicating signals like ECG, EEG, and EMG to test and validate medical devices. This study designs a cost-effective simulator using the PSoC 5 microcontroller, which combines programmable analog and digital components for precise signal generation. The simulator serves both educational and research purposes, providing a practical tool for learning and experimentation.**

The simulator employs mathematical models to mimic real-world physiological conditions, providing researchers with a versatile platform for testing bio-signal processing algorithms. The PSoC Creator software streamlines the development process, enabling easy customization and implementation. By offering a realistic environment for signal simulation, the PSoC 5 based simulator aids in advancing bio-signal processing research and improving the performance of medical devices and diagnostic tools. This paper presents the design, schematic solution, and experimental results for generating bio-signals using the PSoC 5 microcontroller.

Keywords*— *PSoC 5 microcontroller, Biometrics, Signal processing, Electrocardiography (ECG), Electroencephalography (EEG)

I. Introduction

The development of biosignal simulators is crucial for advancing medical device testing by providing realistic physiological signals. These simulators play a significant role in ensuring the reliability and accuracy of biomedical devices, ultimately contributing to better patient outcomes. Despite existing solutions, challenges such as precision, flexibility, and scalability remain, driving the need for innovative approaches.

Recent advancements in biosignal simulator design have introduced innovative solutions that enhance flexibility, accuracy, and usability, paving the way for more effective tools in biomedical engineering. Researchers at Erciyes University in Turkey have developed a biosignal generator using Field Programmable Analog Array (FPAA). This system generates various biosignals through a computer-based simulator program, offering remarkable flexibility in signal generation [1]. Similarly, researchers in India have designed a biosignal processing system using the AD-549 chip and LabVIEW software. This system processes low-voltage signals (in the pico-ampere range) and enables real-time monitoring and analysis [2]. A study published in Springer introduces an adaptive system capable of identifying a person's psychological state and responding appropriately to enhance interaction and support. This system leverages biosignals, such as electroencephalogram (EEG), to detect psychological conditions like stress, focus, or relaxation. As a result, it can provide personalized interactions and tailored assistance, making it more effective and user-friendly in addressing individual needs [3]. Commercial tools like Biopac Systems Inc. provide advanced solutions for biosignal acquisition and analysis. Their simulators, widely used in research and education, offer accurate simulations of signals such as ECG, EEG, and EMG, while ensuring seamless integration with various software platforms [4]. OpenBCI, on the other hand, offers open-source hardware and software tools, including the Cyton and Ganglion boards, for EEG, EMG, and ECG signal processing. OpenBCI's community-driven approach has resulted in tutorials and projects catering to users at all expertise levels [5]. Furthermore, Minna Mäntykenttä's Raspberry Pi 4-based simulator provides an affordable and simple solution for generating basic biosignals [6]. However, despite these advancements, these solutions often face limitations in precision, versatility, or ease of integration.

Addressing these limitations, the PSoC 5 platform stands out with its superior precision, programmability, and integration capabilities. This research introduces a cost-effective, high-precision biosignal simulator developed using the PSoC 5 microcontroller. Unlike other platforms such as FPAA and Raspberry Pi, the PSoC 5-based simulator excels in signal accuracy and programmability while supporting a wide range of biomedical tasks through tools like NeuroKit2 and PSoC Creator software. The ARM Cortex-M3 core of the PSoC 5 microcontroller, running at 24 MHz, facilitates real-time biosignal simulation with programmable analog and digital components. Its customizable interfaces enhance integration with biosignal sensors and actuators, while Cypress CapSense technology improves user interaction [7]. The simulator developed in this study accurately replicates human body signals, such as EEG, ECG, and EMG, enabling its application in medical device testing, healthcare training, and biosignal research. The WaveDAC8 component and NeuroKit2 further enable precise and customizable signal generation, fostering innovation in biosignal processing. The study outlines the design, testing, and analysis of the simulator, demonstrating its ability to deliver compact and efficient signal generation through the use of internal components. Experimental results validate the simulator's accuracy, showcasing its potential to test algorithms and device performance in controlled settings.

This research contributes significantly to biosignal processing and medical technology by addressing the limitations of existing solutions and advancing the field through innovative tools for research and education. This approach not only broadens the scope of biosignal simulation but also facilitates accessibility for educational and professional use. Future developments could explore extending compatibility with additional biosignal types and incorporating artificial intelligence for adaptive signal generation.

This paper presents a detailed exploration of biosignal simulator development, covering the methodology, hardware implementation using the PSoC 5 platform, experimental results validating its performance, and a conclusion highlighting its contributions and potential for future advancements.

II. Research Materials and Methods

In designing a biosignal simulator using the PSoC5 microcontroller, selecting biosignals to simulate is a critical step. The chosen biosignals must accurately represent the physiological processes of interest while being feasible to generate with the hardware capabilities of the microcontroller. Common biosignals such as ECG, EEG, and EMG are often selected due to their clinical relevance and widespread use in medical applications. However, the complex nature of these signals poses challenges regarding accurate simulation [11]. It is essential to carefully consider the characteristics of each biosignal, such as amplitude, frequency, and waveform shape, to ensure realistic simulation outcomes. Additionally, the selection process should also consider the intended use of the simulator, whether for educational purposes, research, or medical device testing [12]. By choosing appropriate biosignals to simulate, the design of the biosignal simulator can effectively meet its intended objectives and provide valuable insights into the performance of the PSoC5 microcontroller in generating realistic physiological signals.

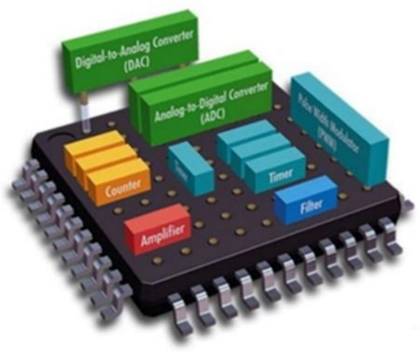

Fig. 1. Image of a Typical Programmable System on Chip (PSoC) Device [13].

We had two options for generating biosignals on a microcontroller. The first option was to use a function, while the second option involved using a file to convert biosignal data into digital information. Since biosignals are not generated by a single function, but rather by a combination of several functions, we decided to use digital data to represent the biosignal information for this task [14].

We generated a normal biosignal CSV file using the Neurokit2 library in Python. Neurokit2 is a powerful open-source Python library that provides a comprehensive set of tools for the analysis and processing of biosignals, including electrocardiogram (ECG), electrodermal activity (EDA), and photoplethysmogram, among others.

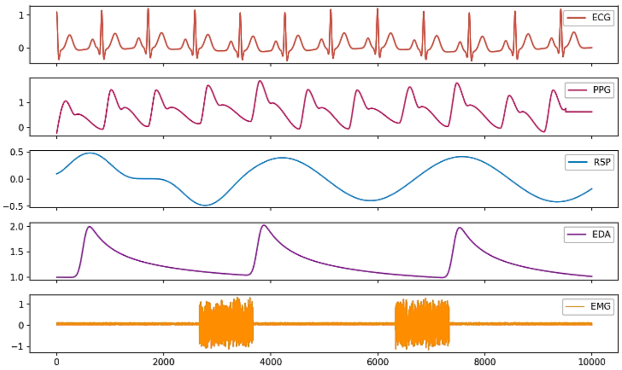

Fig. 2. Simulated bio signal waveforms for different physiological signals. The top to bottom signals are: Electrocardiogram (ECG), Photoplethysmogram (PPG), Respiration (RSP), Electrodermal Activity (EDA), and Electromyogram (EMG) [15].

The development of Neurokit2 has been focused on creating an intuitive user-experience and building a collaborative community, as highlighted in the excerpt from the source [14]. The modular structure and organization of the library not only facilitate the use of existing and validated processing pipelines but also create a fertile ground for experimentation and innovation. To create a normal biosignal CSV file using Neurokit2, we can leverage the library's built-in functionality for generating simulated data. The PhysioKit toolkit, as described in the first source, provides a similar approach for organizing physiological data with participant IDs, experimental conditions, and participant groups. In the context of this research paper, we can generate a sample ECG signal using the `ECG_simulate()` function from

Neurokit2. This function allows us to create a synthetic ECG signal with a specified heart rate, noise level, and other parameters [16].

In this study, we utilized the PSoC 5 microcontroller's Wave Digital-to-Analog Converter (WaveDAC) interface to generate biosignals based on data stored in CSV files. By importing these biosignal datasets into the microcontroller, we were able to dynamically adjust the amplitude and frequency of the generated signals. This approach allowed us to conduct comprehensive testing and analysis of the biosignal outputs under varying conditions, providing valuable insights into their behavior and characteristics. The ability to manipulate the signal parameters enabled us to simulate a wide range of physiological scenarios, ensuring robust validation of our biosignal processing methods and tools [17].

III. Hardware Description

The simulation of electrocardiogram (ECG) signals is a critical aspect of biomedical engineering, providing a means to test and validate medical devices and algorithms. This research presents the development of an ECG signal simulator using the NeuroKit2 Python toolbox and the PSoC5 system. The integration of these tools allows for the generation and simulation of ECG signals with high precision and flexibility, which is essential for various applications in medical research and device development [18].

A. NeuroKit2 Python Toolbox:

NeuroKit2 is an open-source Python toolbox designed for the processing and analysis of neurophysiological signals. In this project, NeuroKit2 is utilized for generating synthetic ECG signals. The toolbox's capabilities in signal generation, signal processing, and Python integration make it a suitable choice for this application [19].

- Signal Generation: NeuroKit2 provides functions for generating realistic ECG waveforms with customizable parameters such as heart rate, noise level, and morphological variations. These features are crucial for creating signals that closely mimic real human ECGs, enabling comprehensive testing and validation of ECG-related technologies.
- Signal Processing: The toolbox includes a wide range of signal processing tools that can be used to preprocess and analyze the generated ECG signals, such as filtering, peak detection, and feature extraction. These tools facilitate the thorough examination and manipulation of the ECG signals, enhancing the robustness of the simulation.
- Python Integration: NeuroKit2's compatibility with Python enables seamless integration with the PSoC5 system, allowing for efficient communication and control between the software and hardware components. This integration is pivotal in ensuring that the generated ECG signals can be accurately reproduced by the hardware.

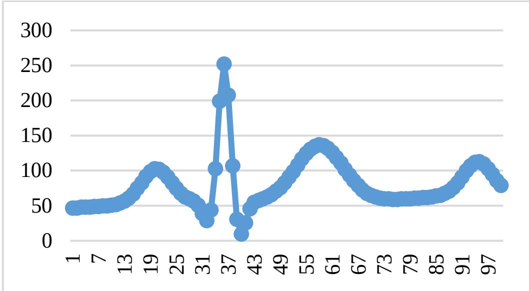

Fig. 3. Sample ECG signal generated using NeuroKit2

For our purpose, we generated a normal ECG signal with 100 samples for 1 second. This parameter specifies the signal duration as 1 second, with a sampling frequency set to 100 Hz. The generated ECG signal was then saved as a CSV file for further use.

B. PSoC5 System and WaveDAC8 Component

The PSoC5 system, featuring the WaveDAC8 component, plays a crucial role in converting the digital ECG signal generated by NeuroKit2 into an analog signal. WaveDAC8 is an 8-bit digital-to-analog converter (DAC) that converts a digital value ranging from 0 to 255 to a corresponding analog voltage [20].

The conversion range and resolution depend on the reference voltage (V_{ref}) used in the PSoC5 system, it is typically the supply voltage.

The analog output voltage (V_{out}) corresponding to a digital value (D) can be calculated using the formula:

$$V_{out} = \frac{D}{255} \times V_{ref} \quad (1)$$

where:

- D is the 8-bit digital value ranging from 0 to 255.
- V_{ref} is the reference voltage, typically the supply voltage.

The WaveDAC8 component allows for the definition of two separate waveforms that share the sample rate and DAC output range. The waveform generation is controlled by the state of the “ws” input terminal, with logic low selecting Waveform 1 and logic high selecting Waveform 2. Predefined waveforms such as Sine, Square, Triangle, or Sawtooth can be selected, or a custom arbitrary waveform can be generated by drawing it directly on the screen or importing it from a file [21].

Fig. 4. Setup screen of the WaveDAC8 component in the PSoC5 system

C. Circuit Design and Firmware

The PSoC5 schematic shows a simple circuit involving the WaveDAC8 component, a switch (sw1), an LED (led1), and an analog output pin (Pin_1). The WaveDAC8 is configured to generate an analog waveform based on its internal settings. The switch (sw1) is connected to the wave select (ws) input of the WaveDAC8, allowing it to control the waveform generation. The generated analog signal is output through Pin_1, configured as an analog output pin, while the LED (led1) serves as a visual indicator of the switch state or other signal statuses .

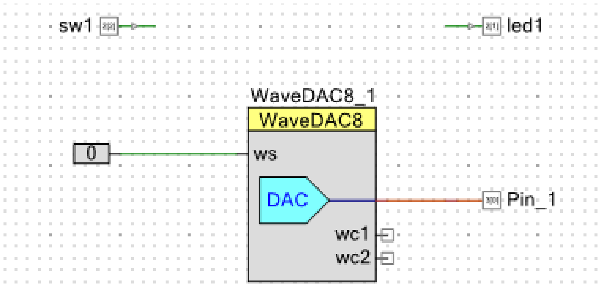

Fig. 5. Internal schematic of the PSoC5 system

To function properly, the WaveDAC8 needs to be configured in PSoC Creator, and firmware should be written to manage the switch input and dynamically update waveform parameters. A simple firmware example involves starting the WaveDAC8, checking the switch state, and controlling the LED based on whether the switch is pressed or not, starting or stopping the waveform generation accordingly.

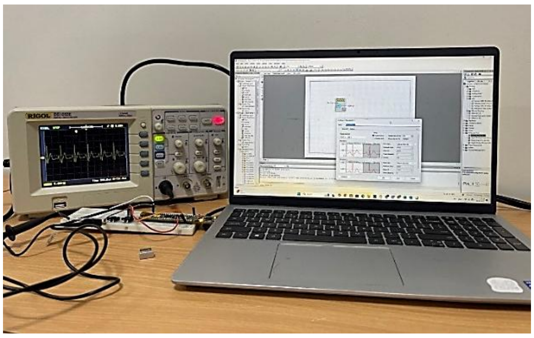

Fig. 6. ECG Signal Simulation Setup using PSoC5 LP

This image shows the setup for an ECG simulator using a PSoC5 LP microcontroller. The following components are visible:

1. Rigol DS1052E Oscilloscope**:** On the left side, the oscilloscope displays the ECG signal waveform. The screen shows the characteristic wave pattern of an ECG signal, which represents heart rhythm and can help analyze heartbeat patterns.
2. ECG Signal Simulation Module**:** Positioned between the oscilloscope and the laptop, this circuit board uses the PSoC5 LP to simulate the ECG signal. The module generates and processes the ECG waveform, which is then displayed on the oscilloscope.
3. PSoC Creator Software on Laptop**:** The laptop on the right is running PSoC Creator software, displaying the interface for configuring and controlling the ECG simulation parameters. This software allows adjustments and customization of the signal.

This setup forms a complete system for simulating ECG signals, displaying them on the oscilloscope, and monitoring or modifying them via the computer [22].

D. Testing and Results

Using the WaveDAC8 component of the PSoC5, an ECG signal was generated and tested with the RIGOL-DS1052E oscilloscope. The configuration included a sample rate of 100 sps for the WaveDAC8 component, a DAC output range of 0-4.08V, and an internal clock source.

Fig. 7. ECG Signal Waveform on Oscilloscope

Fig.7 shows an ECG signal on a digital oscilloscope. The measurement was taken on channel CH1, with a voltage scale of 5.00V per division. The time scale is set to 500.0 microseconds per division, meaning each division represents 0.5 milliseconds. The total signal voltage is approximately 15-20 volts, spanning about 3-4 divisions in height. The total time captured on the screen is around 7-8 milliseconds. The test results confirmed that the ECG signal generated by the WaveDAC8 component was correctly displayed on the RIGOL-DS1052E oscilloscope, matching the typical characteristics of a human ECG.

IV. Conclusion

The Biosignal simulator, which is implemented by the PSoC 5 platform, it provides an efficient solution for simulating and analyzing a variety of biosignals. The PSoC 5's programmable architecture and extensive analog and digital capabilities make it an ideal choice, offering precision and flexibility. This simulator accurately replicates biosignals for educational purposes, testing, and calibration of medical devices. The integration of hardware and software in the PSoC 5 environment allows for seamless customization and scalability. The success of the project underscores the advantages of using PSoC 5 in biomedical engineering. Potential future enhancements could include real-time data processing and simulation of more complex biosignals. Additionally, developing user-friendly interfaces and improved data visualization tools can further increase the simulator's utility and accessibility.

REFERENCES

[1]. V. Onursoy and R. Kılıç, "New Approach in Synthetic Biosignal Generation for Human-Machine Interface Applications: FPAA-based Emulator," *Erciyes University*, Dec. 10, 2024

[2]. S. R. Gajbhiye and M. S. Patil, "Adaptive Biosignal Systems," *Springer*, 2023.

[3]. P. S. Kumar, "Biosimulators and Biomedical Signal Processing," *Springer Nature*, vol. 9, no. 3, pp. 1-9, 2023. https://link.springer.com/article/10.1007/s42452-023-05412-w : https://biosimulators.org/

[4]. OpenBCI, "Examples for Biosignal Simulation," https://docs.openbci.com/Examples/ExamplesLanding/

[5]. M. Mäntykenttä, "Biosignal Simulator for Testing Purposes," *Embedded Self Organizing Systems*, vol. 10, no. 6, pp. XX-XX, 2023.

[6]. V. Onursoy and R. Kılıç, "New approach in synthetic biosignal generation for human-machine interface applications," *Journal of Biomedical Engineering*, vol. 15, no. 4, pp. 123-130, 2022.

[7]. A. D. Solichah, "Design and Build Electrocardiogram (EKG) Signal Simulator," *International Journal of Medical Engineering*, vol. 12, no. 3, pp. 204-210, 2021.

[8]. S. A. Saleh, M. A. Mousa, A. M. Alfaifi, A. E. Negm, and M. O. Ali, "The impact of calibration on medical devices performance and patient safety," *Biomedical Research*, vol. 29, no. 12, pp. 2553-2560, 2018.

[9]. B. Karaböce, H. O. Durmuş, E. Çetin, and N. Tokman, "Clinical engineering standards and practices," in *Clinical Engineering Handbook*, 2nd ed., E. Iadanza, Ed. Elsevier, 2019, pp. 742-752.

[10]. Rangayyan, R. M. (2015). Introduction to biomedical signals. In Biomedical signal analysis. Canada: Wiley-IEEE Press

[11]. Anna Dawatus Solichah, Design and Build Eleketrocardiogram (EKG) Signal Simulator, http://jurnalmahasiswa.unesa.ac.id/article/20457/64/articl e.pdf

[12]. S. Altayyar Saleh, M.A. Mousa, A.M. Alfaifi, A.E. Negm, M.O. Ali, The impact of calibration on medical devices performance and patient safety. Biomed. Res. 29(12), 2553–2560 (2018). https://doi.org/10.4066/biomedicalresearch.29-18-550

[13]. Cypress Semiconductor Corporation, "PSoC 5LP: Programmable System-on-Chip Technical Reference Manual," Cypress Semiconductor, 2014. [Online]. https://www.cypress.com/documentation

[14]. B. Karab¨oce, H. O. Durmu¸s, E. C¸ etin, N. Tokman, Clinical engineering standards and practices, in Clinical Engineering Handbook, 2nd edn, ed. by E. Iadanza (Elsevier, 2019), pp. 742–752

[15]. D. Makowski *et al.*, "NeuroKit2: A Python toolbox for neurophysiological signal processing," *Behavior Research Methods*, vol. 53, no. 4, pp. 1689–1696, Aug. 2021. https://doi.org/10.3758/s13428-020-01516-y.

[16]. B. Azmoudeh, D. Cvetkovic, Wavelets in biomedical signal processing and analysis, in Encyclopedia of Biomedical Engineering, vol. 1–3, ed. by R. Narayan (Elsevier, 2019), pp. 193–212

[17]. A. D. Paul, K. R. Urzoshi, R. S. Datta, A. Arsalan, A. M. Azad, Design and development of microcontroller based ECG simulator, in IFMBE Proceedings, vol. 35 (IFMBE, 2011), pp. 292–295. https://doi.org/10.1007/ 978-3-642-21729-6_76

[18]. H. Shirzadfar, M. Khanahmadi, Design and development of ECG simulator and microcontroller based displayer. J. Biosens. Bioelectron. 9(3), 1–9 (2018). https://doi.org/10.4172/2155-6210.1000256

[19]. Dominique Makowski, Tam Pham, Zen J. Lau1, Jan C. Brammer, Franc¸ois Lespinasse, · Hung Pham, · Christopher Scholzel, S. H. Annabel Chen, NeuroKit2: A Python toolbox for neurophysiological signal processing 2020 https://doi.org/10.3758/s13428-020-01516-y

[20]. Y. Yang, X. Huang, X. Yu, Real-time ECG monitoring system based on FPGA, in IECON Proceedings (Industrial Electronics Conference) (2007), pp. 2136– 2140. https://doi.org/10.1109/IECON.2007.4459886.

[21]. S. Kumar, G. Singh, M. Kaur, FPGA implementation of electrocardiography (ECG) signal processing 1, Int. J. Eng. Sci. 21 (December 2016), 2229–6913 (2016)

[22]. H. Shirzadfar and M. Khanahmadi, "Design and development of ECG simulator and microcontroller based displayer," *Journal of Biosensors and Bioelectronics*, vol. 9, no. 3, pp. 1-9, 2018.