International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I2P121

Volume 72 Issue 2, 197-202, February 2024

Original Article

Strong Implementation of Huffman Coding on Energy Savings in Electrocardiogram (ECG) Wireless Sensor Device

Guillermo Wenceslao Zarate Segura[1*], Luis Mateo Huapaya[1], Julio cesar Arenas[1]

[1]*Department of Engineering and Systems, Technological University of Peru. Lima Peru*

[*]*Corresponding Author : E15040@utp.edu.pe*

Received: 30 September 2023 Revised: 14 December 2023 Accepted: 25 December 2023 Published: 03 February 2024

Abstract *- The present research aims to enhance energy efficiency saved by implementing Huffman coding in the data obtained through the transmission of an ECG device and a receiver. The data is first compressed and later transmitted by a wireless module. The heartbeat signal is acquired through an ADC (ESP 32 module). The compressed data is scaled into 8 bits, followed by a package method using the Huffman coding. Posterior, the data is sent wirelessly from the transmitter module to the receiver. The beats per minute are calculated by using the received data. In the present method, compressed 400-byte packets and a continuous uncompressed 1-byte packet were tested under energy efficiency scope. One of the outcomes is that the saved energy was roughly 20%. Furthermore, the current consumption is analyzed under multiple package size transmission.*

Keywords *- Energy savings, Heart rate, Wireless transmission, Electrocardiogram (ECG).*

1. Introduction

The current research concentrates on Huffman coding as a data compression method to achieve energy savings in a wireless sensor network that monitors the heartbeat since this algorithm might save energy for extended use [1-2]. Other relevant aspects are the data compression process and the probability of data loss due to transmission. Therefore, it is necessary to balance the compression ratio and signal distortion [3].The literature emphasizes methods such as (Discrete Wavelet Transform, Fast Fourier Transform, Discrete Cosine Transform, and Adaptive Linear Prediction) for signal compression, which combined with Huffman coding to have optimal data transmission, and that's is the main focus of this research [4,5].The signal is directly captured from the human body. The raw signal presents noises, so firstly, It requires a proper treatment of the ECG signal [6, 7]. The literature review offers a broader view of all these integrated stages, from a device that stores cardiac events in an SD-type flash memory to later sending them to another device through a GSM-GPRS modem [8]. Electronic devices send cardiac data through Wi-Fi or Bluetooth technology that allows the wireless connection of multiple sensor devices [9,10,11]. The first one is the acquisition of biomedical signals, specifically the ECG signal, from which the heart pulse or heart rate is obtained. [12]; sensors to detect both the ECG and the conductive skin response (SCR) [7]; and simpler sensors to analyze only the ECG [8]–[11]. The captured signal must be treated to eliminate the inherent noise. The authors, Varela et al. [12] and Mora et al. [6], use different methodologies, such as digital filters. The former author uses a 4th-order low-pass Butterworth filter to obtain a cut-off frequency of 40 Hz. The second one implements different convolution windows to get a low-pass type geometric profile using Gaussian, quadratic, triangular, and trigonometric functions. The data compression process was made using the Huffman coding, which was developed using other authors' research. For instance, Ballesteros et al. [3] pointed out a register of 100 for a Direct Wavelet Transform (DWT) and later their version of Huffman code. The author started analyzing the energy's area between the P and T waves and the QRS segment. Other authors used a different approach, for example. Hamza et al. [4] used registers 100, 109, 115, 119, and 200 to apply a DWT, run-length encoding, and Huffman encoding compression algorithm for their project. After this process had ended, the quantization step started by transforming extensive input values into fewer output values. Bustamante et al. [8] propose an automatic monitor-event device called CARE II. This device obtains bioelectric signals, treatment, analysis, calculation, and storage of the signal data. It uses mobile communication GSM-GPRS to allow access to the internet. It analyses the data collected through a specific application and prepares reports by the corresponding specialist. Another issue is energy efficiency, which Smith et al. [9] and Jacome and Baquerizo [10] propose a system composed of a single server module that processes the signal and transmits it wirelessly to a mobile device (smartphone).

Both authors aim for the lowest energy consumption. However, the first author implements Wi-Fi technology for wireless communication, using the 802.11n standard for lower transmission and reception consumption. Nonetheless, the second author uses Bluetooth BLE technology for data communication. The main objective was to consume the lowest energy possible because they aimed to use the device for 30 days without charge. Thus, the described methods and Huffman coding have provided promising results [4]. Our objective is to reduce the energy consumption due to data transmission by enhancing the data package transmission using the Huffman coding, allowing increased battery autonomy in the long run [9]. Moreover, this research aims to determine the percentage of energy saved caused by using Huffman compression in wireless data transmission and compare the energy efficiency when a whole data transfer is completed under Huffman coding and in continuous transmission.

This paper flow is stated by naming all the sensors, electronic devices, and software. Then, the current block diagram shows how the data is obtained and transmitted. Next, we test the transmitted raw and filtered data with different size packages using Huffman coding to determine the optimal energy efficiency and less time consumption. The main issue is to gather further knowledge on whether Huffman coding could strongly save energy and how much it does compared to other methods. It presents the outcomes of this research and compares the data rate and the energy saved by measuring the current after the battery is discharged.

1.1. Data Acquisition and Heartbeat Signal

The plethysmography technique was implemented to determine the blood change rate in the body. Plethysmography measures the changes in blood volume, usually in the arms and legs. It is a non-invasive technique that monitors heartbeat and oxygen saturation. The PPG signal is obtained by emitting light on the surface of the tissue. Then, blood flow reflects it, and the photodetector device records the data for further analysis [13]. The cardiac pulse measures heart pulses in a period, usually in 60 seconds. The shape of the signal shows the health state of the patient. The electric wave of the heart is named ECG (Electrocardiogram). The ECG signal is one of medicine's most crucial human signals since it can show different diseases. As a result, it is imperative to look for new ways to compress the signal without losing any signal data [14]. The present study uses a photoelectric sensor SoC (System on chip)ESP32-DOWDQ6 in the board ESP32 DEVKIT, a voltage and current sensor (INA219), and a battery of 5V. The design of the emitter device is shown in Figures 1 and 2.

2. Compression and Uncompression using Huffman Coding

Huffman coding is a method without lost data that uses the probability of how data appear within the sample. It generates a train of bits with an inverse proportion of the frequency data. The probabilities or weights are unknown in Huffman's coding since they are generated in the in-situ experiment. Thus, the Huffman tree is corrected throughout the whole process. There are two algorithms for converting the tree: the Faller-Gallager-Knuth algorithm and the Vitter algorithm [15].

2.1. Signal Conditioning

In Figure 2, the sensor works with photosensor APDS-9008l, changing the analog output according to the reflected light. A filter and amplification stage is also implemented by using an OP-AMP. The analog signal passes in channel 3 of ADC1 (pin39), which sets a 10-bit resolution and a sampling frequency of 200Hz. In Figure 4, pulse peaks are easily identified pulse peaks and the amplitude of the filtered signal (red line) is reduced.

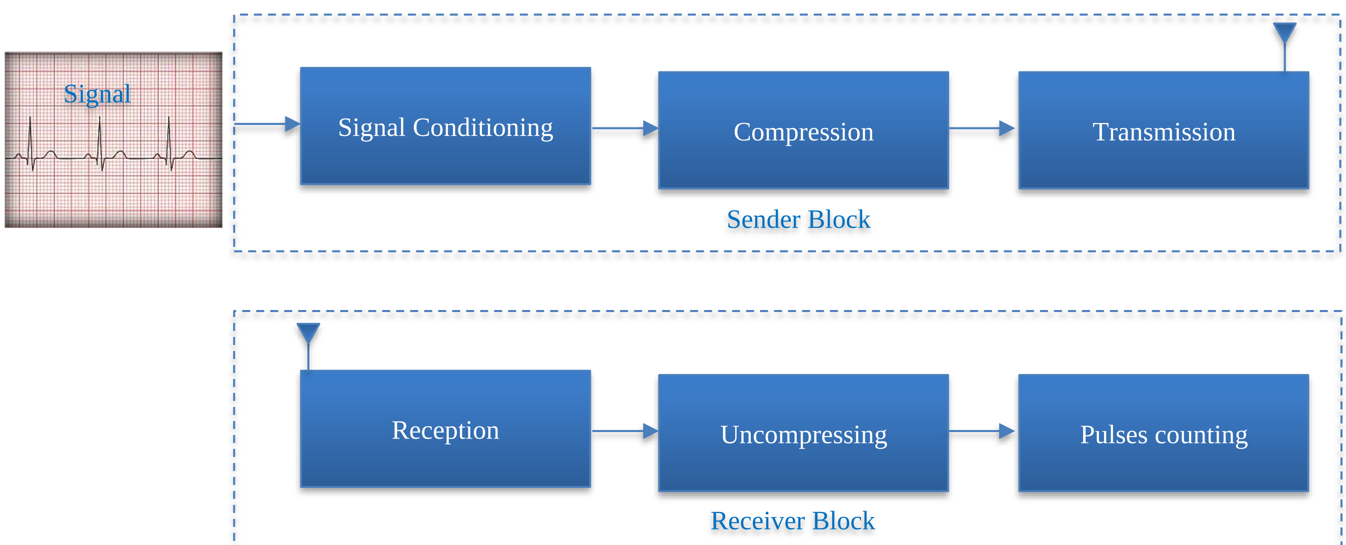


Fig. 1 Block diagram of the project

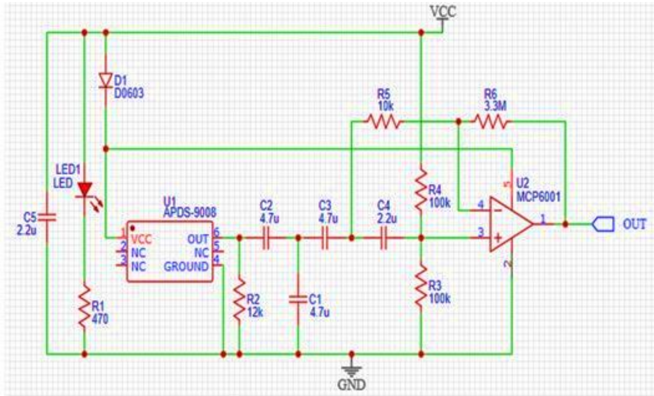


Fig. 2 Transmitter device electronic schematic

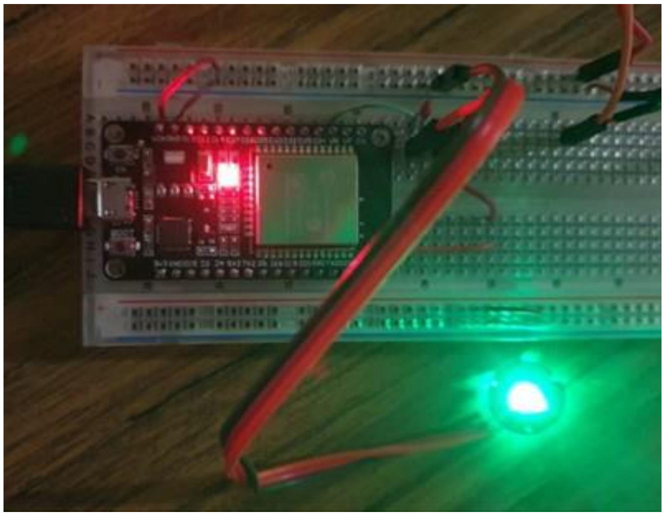


Fig. 3 Real transmitter device

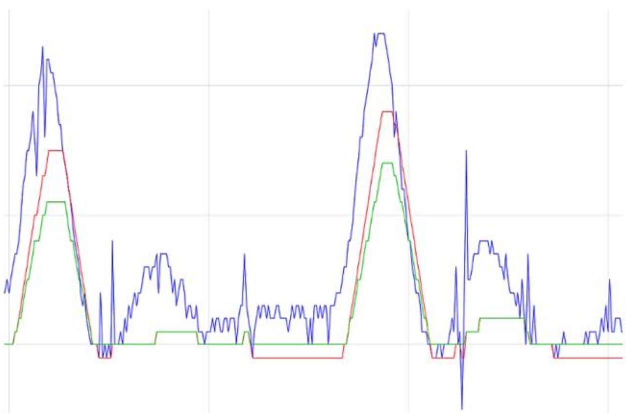


Fig. 4 Filter signal (red), raw signal (blue), and tracing signal (green)

2.2. Signal Compression

This sub-block stores the scaled data in a one-dimensional array of 300 bytes. This matrix is compressed with an algorithm based on Huffman coding. The code used for compression and decompression is based on a work published on GitHub [16]. Table 1 shows the compression percentage for a data sample that presents different values, such as the QRS zone of the cardiac signal.

A compression for different data packets is obtained considering only the code generated by the compression, without considering the Huffman tree's structure. It can be observed for the 50 and 100-byte packets, and adequate reduction is not achieved since the Huffman tree structure increases its size, and the structure is necessary to send it for its later decoding.

2.3. Data Transmission

In this last sub-block of the transmitter stage, the data generated by the compression is stored in a data structure defined both in the transmitter and the receiver. This is established by the EspNow protocol, which sends information between the modules. The EspNow packet payload supports a maximum of 255 bytes. However, the load generated for a test data array does not exceed this limit after compression. Additionally, the transmitting module must know the MAC address of the receiving module to establish communication.

2.4. Data Reception

In this sub-block, the data received is delayed by the transmitter module through the instruction ("esp_now_register_recv_cb"). Once the data is obtained, it is copied to the reception structure for subsequent decompression.

2.5. Decompression

The received data structure contains the encoded information and the information from the Huffman tree necessary to decipher the code. The data was decompressed.

2.6. Pulses Count

This sub-block is in charge of detecting the peaks of the decoded signal and counting until the next peak is detected. Knowing the value of the counter and data sample taken every 5ms, the heart rate or Beats Per Minute (BPM) can be known using the following equations.

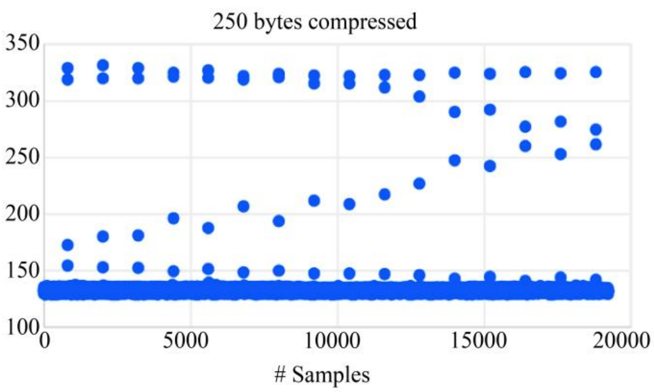
$$BPM = 60 * (Tm * C)^{-1} \, [s] \tag{1}$$

Where Tm is the sampling time (ms), and C is the count from the last peak. Considering a sampling time of 5ms, the formula remains as (2) for this case.

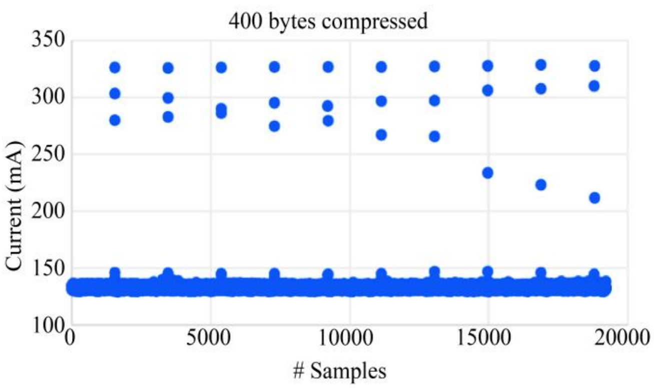
$$BPM = (5ms * C)^{-1} * 60 = 12000 \, \left[\frac{1}{C}\right] \tag{2}$$

2.7. Energy Consumption

This section shows the energy consumption of the emitter block. A 5v DC battery powers this block. Through the INA219 module, It is possible to know the current consumption in the circuit. This applies both for sending encrypted and unencrypted information. The current consumption has been recorded in the encrypted data sent for 30 seconds. The sampling time for the current is 1.04ms. This process used 250- and 400-byte packets for data compression (Figure 5 a and b).



(a)



(b)

Fig. 5 Current consumption of the Sender block: (a) for a package of 250 bytes, (b) for 400 bytes

Table 1. Percentage of Compression for different data package sizes.

Data package	Compression	
	Generated code	Code and Huffman tree
50	46%	198%
100	43%	119%
150	40%	91%
200	42%	80%
250	40%	71%
400	39%	58%

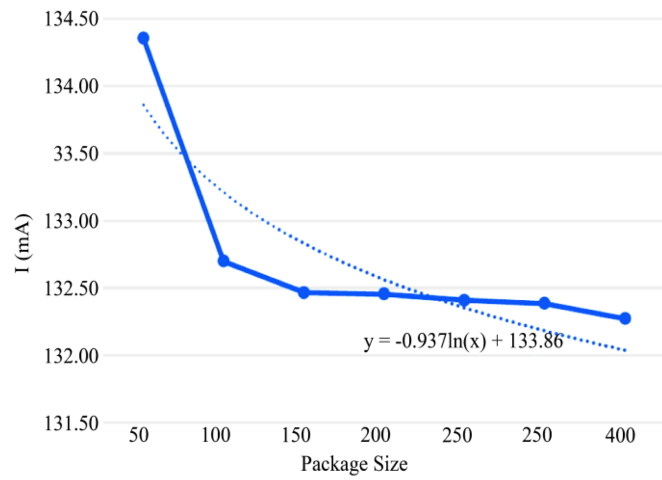


Fig. 6 Current (mA) versus the size of the package of the transmission data

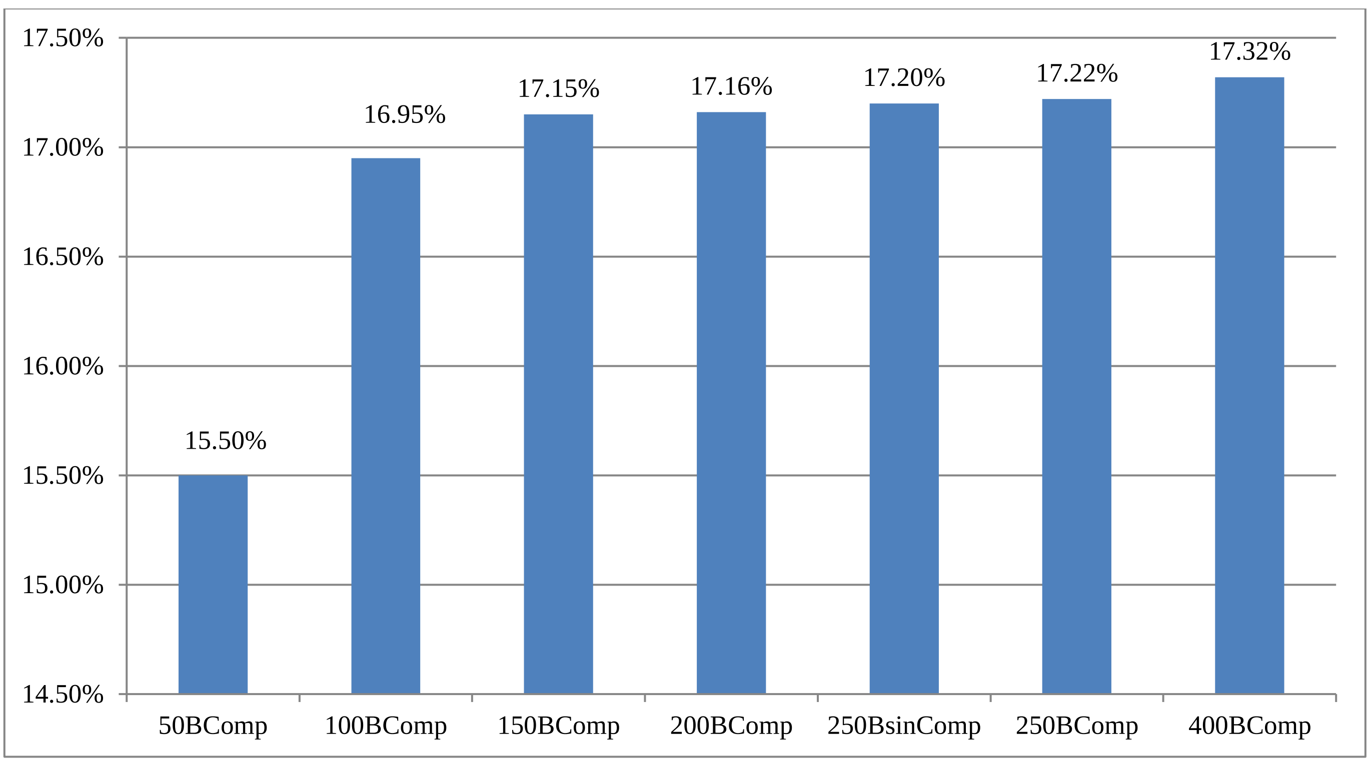


Fig. 7 Battery's lifespan percentual rises based on the transmission of 1 byte

Table 2. Estimation of the lifetime of the battery (5000mAh)

	# Bytes	Lifespan		
		Days	Hours	Minutes
Compressed	50	1	13	12.9
	100	1	13	40.8
	150	1	13	44.7
	200	1	13	44.9
	250	1	13	46.1
	400	1	13	48.1
No-compressed	250	1	13	45.7
	1	1	8	13.2

Table 3. Lifetime of the battery in Deep-sleep mode

		Lifespan		
Reading time (sec)	# readings per hour	Days	Hours	Minutes
10	1	15	6	49
	2	14	22	8
20	1	14	22	8
	2	14	5	57

3. Discussion and Conclusion

The compression percentages are written in Table 1 for a sample of cardiac signal data. It is impossible to properly compress a 50 to 100-byte package since Huffman's tree structure has exceeded the package size. The best option is to send the package of 400 bytes since It presents the best compression on the present table. Then, we plot the current need to transmit the data against the package size. The plot is presented in Figure 6, and the current consumption is reduced from 50 to 100 bytes. The current required to transmit 200-, 300-, and 400-bytes size is approximately the same as 100 bytes. Thus, the aim to transmit only 400-byte packages was successful. Figure 7 plots the battery's lifespan using the current consumption in different-size transmissions. The main outcome is a rise in the battery's lifespan by less than 20%, approximately 17% in cases over 100 bytes in size. Using the current, the estimation of how long the battery of 5000mAh will last until its next recharge; the data is presented in Table 2. Indeed, the Huffman compression for a maximum size of 400 bytes allows an energy saving of 17.32% concerning a continuous sending of one-byte data without compression and lasts one day, 13 hours, and 48 minutes for a 5000mAh battery. With the Deep-sleep mode activated, It can be extended up to 15 days, 6 hours, and 49 minutes, see Table 3. Compression of the package from 100 to 400 bytes does not save energy since the effect of energy saved is less than 1 %.

The recommendation is to use an analog-digital external converter since the module used (ESSP32) presents a distortion in the reading of the ADC signal when the data is sent through the ESPNOW protocol. A more robust sensor is recommended since technical issues were obtaining a stable signal without noise.

References

[1] Juan José Garza-Saldaña, Arturo Díaz-Pérez, and Arturo Medina Puente, "Energy Saving for Body Sensor Node Networks through Physiological Data Compression," *Magazine National Congress of Engineering and Technologies for Sustainable Development*, vol. 1, no. 1, pp. 35–41, 2015. [Google Scholar]

[2] Christopher M. Sadler, and Margaret Martonosi, "Data Compression Algorithms for Energy-Constrained Devices in Delay Tolerant Networks," *SenSy '06: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, pp. 265–278, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[3] Dora María Ballesteros Larrotta, Camilo Andrés Lemus Bernal, and Carlos Alberto Suárez López, "Compression of ECG Signals using DWT and Huffman Coding," *Science and Technology*, vol. 1, no. 41, pp. 340–345, 2009. [Google Scholar] [Publisher Link]

[4] Khalida Shaaban Rijab, and Mohammed Abdul Redha Hussien, "Efficient Electrocardiogram Signal Compression Algorithm Using Dual Encoding Technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 3, pp. 1529–1538, 2022. [CrossRef] [Publisher Link]

[5] Bharat Garg, Sameer Yadav, and G.K. Sharma, "An Area and Performance Aware ECG Encoder Design for Wireless Healthcare Services," *2016 20th International Symposium on VLSI Design and Test (VDAT)*, Guwahati, India, pp. 1-6, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[6] Miguel Mora González et al., "Digital Noise Reduction in ECG Signals Using Convolution Filtering," *Research and Science*, vol. 16, no. 40, pp. 26–32, 2008. [Google Scholar] [Publisher Link]

[7] F.J. Alvarado Rodríguez et al., "Development of a Computational Tool for the Analysis of Galvanic Skin Conductance, ECG and Respiratory Rate due to Respiratory Sinus Arrhythmia (SCRATER)," *Mexican Journal of Biomedical Engineering*, vol. 38, no. 1, pp. 7–18, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[8] John Bustamante Osorno et al., “Development of the Hardware Component of a Device for Tele-Monitor Wireless of Cardiac Events,” *Research and Science*, no. 39, pp. 445–448, 2008. [Google Scholar] [Publisher Link]

[9] Rubén Cobo Alea, Rafael E. Smith Colás, and Carlos R. Vázquez Seisdedos, “Design of a Wireless Electrocardiography Monitoring System for Android Devices,” *Electronic, Automatic and Communications Engineering*, vol. 41, no. 2, pp. 63–79, 2020. [Google Scholar] [Publisher Link]

[10] Pablo Jacome Ruiz, and Christian Baquerizo Cardenas, “Portable Heart Monitor with a Bluetooth Interface “CARDIO UEES”,” *Unemi Science*, vol. 9, no. 20, pp. 36–49, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[11] Ahmed Aboalseoud et al., “Wireless ECG Monitoring System for Telemedicine Application,” *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, pp. 300–305, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[12] J.L. Varela Benítez et al., “High Sensitivity Capacitive Electrode for the Detection of Electrical Biopotentials,” *Mexican Journal of Biomedical Engineering*, vol. 36, no. 2, pp. 131–142, 2015. [Google Scholar] [Publisher Link]

[13] Assim Boukhayma, Antonino Caizzone, and Christian Enz, “An Ultra-Low Power PPG and mm-Resolution ToF PPD-Based CMOS Chip towards All-in-One Photonic Sensors,” *IEEE Sensors Journal*, vol. 19, no. 24, pp. 11858–11866, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] Erick Alexánderson Rosas, and Gerardo Gamba Ayala, *Cardiovascular, Renal and Respiratory Physiology*, 1st ed., Editorial El Manual Moderno, pp. 1-277, 2014. [Google Scholar] [Publisher Link]

[15] Thomas H. Cormen et al., *Introduction to Algorithms*, 3rd ed., London, England: The MIT Press, 2009. [Google Scholar] [Publisher Link]

[16] SadiManna/Compression, Data Compression Techniques in C and Matlab, 2018. [Online]. Available: https://github.com/sadimanna/compression