Design of Low Power VLSI Architectures for Machine Learning Based Wearable Healthcare Devices

A

Thesis submitted

for the award of the degree of

DOCTOR OF PHILOSOPHY

By

Meenali Janveja

DEPARTMENT OF ELECTRONICS AND ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

GUWAHATI - 781 039, ASSAM, INDIA

December 2022

Certificate

This is to certify that the thesis entitled "**Design of Low Power VLSI Architectures for Machine Learning Based Wearable Healthcare Devices**", submitted by **Meenali Janveja** (176102001), a research scholar in the *Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by her under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated: December 26, 2022
Guwahati.

Dr. Gaurav Trivedi
Associate Professor
Dept. of Electronics and Electrical Engg.
Indian Institute of Technology Guwahati
Guwahati - 781 039, Assam, India.

Dedicated to the **Almighty**

for immense blessings,

My guide **Dr. Gaurav Trivedi**

for his guidance and inspiration

&

My **Family**

for their unconditional love and support

Acknowledgements

I am immensely grateful to many people for supporting me in various ways throughout this rather protracted process of completing a PhD thesis. I owe a debt of gratitude to my incredible supervisor, Dr. Gaurav Trivedi, for allowing me to work under his guidance. It is very hard to describe in words to acknowledge my supervisor for his extolling suggestions, continuous guidance, valuable advice, careful scrutiny and explicit encouragement throughout my doctoral studies.

I am also grateful to my doctoral committee members, Prof. H.B Nemade, Prof. P.K Bora and Dr. Aryabratta Sahu, for their encouragement and invaluable suggestions on my work. I want to thank faculty members and the office staff of the Department of Electronics and Electrical Engineering, IIT Guwahati, for their help in carrying out this research work. I also wish to sincerely thank Dr. Jan Pidanic for his valuable suggestions and support for my research work.

My warm and sincere thanks go to all my friends and colleagues from HPC and FPGA Laboratory for their constant encouragement and ungrudging support during my learning process.

Most importantly, I attribute this achievement to my parents, brothers and sisters-in-law for their constant blessings, support, and silent prayers for my success and for making me stand in this position. I would also like to extend my love to my two nieces and nephew, Reya, Anaisha and Vishwank. Their smile gave me all the strength to overcome hurdles along this beautiful journey.

Meenali Janveja

Abstract

According to the World Health Organization (WHO), cardiovascular diseases cause approximately 17.9 million fatalities yearly, which is estimated to be 31% of the global mortality rate. An electrocardiogram (ECG) is a biosignal that provides information on the patient's heart's electrical activity. ECG enables the diagnosis of various cardiac abnormalities, from acute coronary syndrome to cardiac arrhythmias. Therefore, ECG monitoring in daily life is necessary for early diagnosis of heart disease. However, conventional long-time ECG monitoring methods are invasive and expensive, hindering the patients' daily activity. Furthermore, heart activity is usually recorded only for $24-48$ Hrs. in hospitals, which may be effective if the heart anomaly manifests within this period. Further, with the increasing population and inadequate healthcare infrastructure, medical support has become paramount in today's scenario. Due to fewer medical professionals available than required, it has become challenging to administer expert help to people living in urban and rural areas. Even for people living in cities and under constant medical observation, providing them with an efficient mechanism to continuously collect ECG data for analysis is not easy.

Hardware and software developments have led to the development of machine learning-enabled wearable healthcare devices, such as smartwatches and chest patches, which can continuously monitor cardiac functioning easily. The wearable devices provide critical alerts for events that require prompt medical attention or hospitalization, making them highly efficient and practical. Further, these devices can continuously track vital health parameters and send this information to healthcare providers. This reduces the number of yearly visits to a medical professional. A conventional wearable device has three primary modules. The first module is the sensors and analog front, responsible for acquiring the ECG signals and converting them to digital samples. The second module consists of an ECG co-processor, incorporating a feature extraction block and a machine learning-based

classifier responsible for ECG signal analysis and classification of cardiovascular diseases. The final module comprises data compression and transmitter blocks, which transmit ECG data and the classifier output to the cloud servers.

In wearable devices, battery life is critical because most devices monitor ECG continuously. Further, these devices should be small and easy to use. Therefore, area and power-optimized algorithms and their VLSI architectures are required for continuous monitoring of ECG on wearable devices. Thus, we present optimized ECG signal processing algorithms and their low-power and resource-efficient VLSI architectures for cardiovascular disease detection, such as cardiac arrhythmia and myocardial infarction, for wearable devices.

Initially, an optimized ECG feature extraction algorithm and its low-power hardware implementation, which can extract all the critical features of ECG is proposed. Further, this design is integrated with different classifiers, employing neural networks and if-else-based methods to detect different types of cardiac arrhythmia and severity stages of Myocardial Infarction (MI). In this thesis, we propose three different VLSI architectures for classifing five types of arrhythmia beats, predicting ventricular arrhythmia and detecting severity stages of MI. Further, a low-power ECG data compression architecture, enabling the wearable healthcare device to compress a large amount of ECG data, is also developed. The proposed algorithms and their VLSI architectures for ECG signal processing operate at low power and require minimal hardware resources, making them suitable candidates for wearable healthcare devices.

Keywords: ECG Analysis, Cardiac Arrhythmia, Myocardial Infarction, Data Compression, Neural Network.

Contents

List of Figures

List of Tables

1

Introduction

Contents

1.1 Wearable Healthcare Devices

The electronic devices that an individual can wear to monitor important biological parameters as accurately as possible are commonly known as wearable devices. These devices have made the Internet-of-Things (IoT) industry a trillion-dollar giant in the present era. Some classic examples of wearable devices are Apple Watch, Samsung Galaxy Watch, and Fitbit. Additionally, wide varieties of other gadgets lead us towards a better-connected lifestyle. The primary job of these devices is to collect millions of data points ranging from the number of steps an individual takes to one's heart rate, making the wearable industry a booming industry.

Further, it is expected that the wearables industry will reach to a whopping 70 billion industry by 2025 [5]. The significant advancements in wearables have been a welcoming tool for the insurance, healthcare, and sports industries for several reasons. In order to promote healthier lifestyles, these sectors encourage customers to use wearables to track their health data. The data collected from these devices accurately portrays the customer's overall health. Along with the insurance, real-time health data is also helping the sports and fitness industries push the training boundaries of professional athletes worldwide. From marathon runners to football champions, elite athletes use smart compression shirts to maximize every athlete's move. These shirts use a combination of GPS, accelerometers, and biomedical sensors to constantly measure each athlete's performance. Further, wearables influence the business world as well. Arming the employees with wearable devices enhances communication, tracks employee activity, boosts job site safety measures, and can improve the workforce's health and quality of life. A healthier and happier workforce leads to higher employee retention rates, saving companies thousands, if not millions, of dollars annually.

Moreover, the healthcare sector sees the most significant benefits from wearable technology. Smartwatches, like a Samsung Galaxy, Fitbit, or Apple Watch, have pioneered how we keep track of everything from heart rates to our daily steps. Constantly checking our body parameters with them has become a part of a daily routine for millions of users globally, encouraging healthy lifestyles. Knowing how far one can run encourages oneself to push limits to beat old records. Meeting every 10,000-step goal encourages one to walk. Even tracking one's current sleep habits could lead to changes that help one to sleep more soundly at night. Wearables are one of the best ways to gamify the health and

wellness journey. Further, patients who wear these smart devices can measure information ranging from body temperature to heart rate, which relays to their medical team in real-time. This data allows medical professionals better understand the patient's health and enables prompt treatment. The wearables expedite the entire diagnosis and treatment process using machine learning enabled wearable devices. Doctors no longer have to run various tests to determine an illness or disease. They can quickly reference the data collected by a wearable to determine the cause of the medical mishap. Currently available wearables can monitor heart rate, calories burned, seizures, number of steps, blood pressure, and oxygen level. These health parameters integrate into a single unit, like an activity tracker or a smartwatch, and are used for physical training, monitoring overall physical health, and alerting to severe medical conditions. Other healthcare applications can also be explored, as stated below [6].

- Mood changes, stress, and health
- Measurement of blood alcohol content
- Long-term monitoring of cardiac patients with electrocardiogram recording and processing
- Health Risk Assessment applications of Geriatric patients
- Respiratory monitoring for patients with chronic obstructive pulmonary disease
- Monitoring of blood glucose for patients with diabetes type 1 and type 2
- Fetal and neonatal monitoring
- Monitoring of COVID-19 patients, e.g., using body temperature and blood oxygenation level monitoring devices
- Patient monitoring during cancer treatment, e.g., blood pressure, heart rate
- Measurement and recording of dyskinetic symptoms and tremors by patients with neurological disorders

While these wearables can collect data, most of them have limited ability to process it and make any diagnosis based on this data. Therefore, most wearable devices are used primarily for general health information. Wearables consider individual differences; however, most collect data and apply one-size-fits-all algorithms. Modern wearable technology falls under a broad spectrum of usability and works differently based on its categories, such as health, fitness, or entertainment. Predominantly, wearables incorporate microprocessors, memories, batteries, and connectivity modules to the internet to sync data collected with other electronic devices like cellphones or computers. Wearables are also embedded

with built-in sensors that keep track of bodily movements, provide biometric identification or assist with location tracking. Most wearables are worn on the body or attached to clothing. Some wearables use remote smart sensors and accelerometers to track movements, and some use optical sensors to measure heart rate and glucose levels and monitor these data in real time. Smartwatches and fitness trackers are the most prominent kind of wearable devices. However, with the recent developments in IoT-enabled wearables, machine learning is also incorporated into all wearable devices. With machine learning, the wearables can process the data on-chip and aid medical professionals in diagnosis.

1.1.1 Benefits of Wearable Devices in the Healthcare Industry

With wearable technology, medical professionals can monitor and evaluate patients' vitals, which can help doctors better manage their health and safety.It also aids clinicians in monitoring the compliance of medicines and treatments. Healthcare has become more efficient, convenient, and advanced due to the advancements in wearable technology. It also enables doctors to connect with their patients via telehealth solutions and allows them to engage 24×7. Thus, wearable devices in healthcare offer numerous benefits to patients and healthcare providers. Some advantages of wearable technology are as follows.

- **Earlier Disease Detection:** With continuous use, wearables build unique personalized health models for every user by constantly scanning round-the-clock abnormalities. This monitored data can point to the detection of different diseases. It can further assess the acceleration of particular preexisting conditions.
- **Improved Diagnosis:** Wearable Technologies greatly benefit healthcare providers by tracking personalized data. Monitoring certain biometrics continuously over an extended period allows medical professionals to understand patients' conditions better and use this data to create a more accurate and improved diagnosis.
- **Proactive approach of Healthcare:** Due to the constant tracking of biometrics, users now have the potential to take a more proactive approach to their overall well-being. Users benefit from the data collected to detect health issues before any fatal consequences. Wearables also aid in preventing diseases, developing healthy habits, and maintaining conscious and healthier routines.
- **Patient's Engamenet:** The continuous monitoring of the vitals by the users makes them more engaged with their health status. Having all their health information in their hand allows

users to monitor themselves.

- **Instant Alerts:** In chronic conditions, wearables can be fundamental emergency tools that notify users and professionals in real time when users' health indicators are deteriorating and reaching a risk zone. They are also powerful in the detection of anomalies. Detecting variations in the compiled health data aids in the early detection and diagnosis of different diseases.
- **Avoiding Office Visits:** Wearables that continuously track our biometrics and send this information to our healthcare providers, reducing the number of office visits yearly, and making visiting a medical professional more accessible and personalized.
- **Easy sharing of health data:** Wearables enable the democratization and decentralization of patients' information. They generate a new flow of data outside healthcare providers' control, which users can freely administer and access. Another significant benefit is that the Bluetooth and WiFi-enabled devices can easily transmit information to mobile devices, eliminating the data synchronization burden.
- **Cost Reduction**: According to Forbes Magazine, wearables can potentially lower hospital costs by as much as 16% in the next five years. Wearables integrated with artificial intelligence (AI) can predict many hidden diseases before they manifest clinically. Due to the predictive capabilities, these devices can cut down the medical cost incurred by a patient that may arise in fatal situations. These predictions and data also let doctors give more effective treatments and allow hospitals to use their resources better.
- **Alleviating Staff Workload:** The increasingly more significant wearables' usage, functionality, and ability to capture large amounts of patients' data are already benefiting doctors and patients. With the increasing population and fewer medical professionals, especially in developing nations, wearable devices help minimize in-person consultations and alleviate staff workload. Nevertheless, there are still particular challenges to consider in wearable technology.

1.1.2 Challenges Faced by Wearable Devices

Many multinational companies are working on the development of wearable devices and parts of patient monitoring networks. Wearable medical devices can also act as personal emergency-response systems. Wearables also let patients leave hospitals sooner, eliminate many clinic visits, and give doctors accurate data. However, there are several hurdles to making wearable devices from a concept to a profitable product. There are five typical design challenges to monitoring patients and making

wearable devices: power consumption (or battery life), portability (or size), patient safety, secure data delivery, and integration.

- **Battery Life:** Portable and wearable patient monitors are typically battery-powered, and battery life is an important consideration for consumers. Battery life is critical because most patient monitors measure continuously. Battery-operated systems need careful partitioning for optimal use of little space. The users require more features, better-powered deliverance from smaller packages, and longer times between recharging. Some primary techniques, such as standby mode, sleep mode, power save mode, hibernate, and shut down, let the designers reduce power requirements and extend battery life. The features of wake-up and standby modes also play vital roles in minimizing power consumption in wireless connectivity. Despite using low-power microcontrollers (MCUs) and analog integrated circuits, designers would only be able to leverage the latest technologies in architectures if they optimize power management. Choosing an architecture that is as efficient as possible and can extend battery runtime is essential.
- **Portability** Wearable healthcare devices include heart-rate monitoring, blood glucose monitoring, pulse oximeters, and fitness and activity monitors. Many of these devices have disposable batteries that are not rechargeable making the battery's overall form-factor requirements tight. All the sensors and processing units need to be integrated into a space as small as possible. The smaller packaging can also improve integration, save space, and make the wearable more comfortable.
- **Patient Safety:** Multiparameter portable patient monitors estimate biometrics levels and employ power and data isolation to keep users safe. Digital isolators and power isolators are used to keep data and power isolated. Several critical design challenges are associated with isolating power and data, such as output regulation, feedback mechanisms, input voltage, output power and size, and power architectures.
- **Secure Data:** As wearable devices are connected over the internet, security becomes a significant challenge. Healthcare devices handle important private healthcare information and require strict security measures to avoid unauthorized access. Personal health data must be obtained securely to prevent disclosure to a third party. Therefore, there is an inevitable need to secure any attack on the immune system for the wearable healthcare system. Several security measures are already designed to protect Intellectual Property (IP) and data sent between a patient and

a doctor. These measures should protect against attacks and secure data transmissions while processing and converting to vital-sign parameters for display and during transfer.

- **Integration:** The time to develop patient monitors is critical as many standard lab tests and approvals are required before market production at a global level. Speeding the process with little integration effort is possible by relying on connectivity to the cloud. Saving patients' data directly to the cloud saves on-chip memory usage. It must be compatible with Bluetooth, Bluetooth Low Energy, and Wi-Fi. Integration of multiple cores, receivers-transmitters, interface standards, and general-purpose I/O can support various system-level requirements.

As VLSI and biomedical designers resolve these challenges, better devices can be manufactured and made available at affordable prices, smaller sizes, and better connectivity. From hospitals to remote healthcare centres in developed and developing countries, it is anticipated that the healthcare world will quickly adopt newer patches for remote diagnosis providing better care.

1.2 Motivation

Health monitoring has become indispensable in the present-day world. The increasing modernization and unhealthy lifestyle have aggravated diseases, like diabetes, stress, and obesity. With the increasing population and inadequate healthcare infrastructure, medical support has become paramount in today's scenario. Due to fewer medical professionals available than required, it has become challenging to administer expert help to people. Even for people living under constant medical supervision, providing them with an efficient mechanism to continuously collect essential physical parameters to prevent mishaps is not easy. This scarcity of medical facilities has led to a hike in the cost of healthcare facilities [7]. The advent of high-speed internet, sensor technology, and the internet of things (IoT) makes it feasible to bestow medical support comfortably. This advanced technology leads to the development of an efficient medical service ecosystem.

As we know, wearable devices are connected electronic devices worn on the body as accessories, which can glean several health insights. However, the integration of wearable technology in the medical workplace is still in its initial stage. Moreover, the advent of COVID-19 has accelerated the growth of telehealth measures. Among several health disorders, the mortality rate due to cardiac problems is the highest compared to other pathologies [8]. Therefore, a high demand arises for affordable, reliable, low-cost, and compact wearable biomedical devices that monitor patients continuously and connects

them to physicians preventing them from reaching critical conditions.

Electrocardiogram (ECG) is a bio-signal that depicts the electrical activity of heart muscles. Cardiologists use it worldwide to monitor the proper functioning of the heart due to its non-invasive capability of detecting cardiac diseases. ECG is recorded in clinical centres requiring a patient to stay in the hospital for several hours or even days. Portable ECG devices have enabled the monitoring, recording, and transferring of ECG data to health experts, where it can be analyzed for abnormalities. Several wearables available in the market today can monitor ECG signals efficiently. A summary of various wearable devices is presented in Table 1.1, depicting their clinical studies [9].

Comapny	Device Name	Vitals Monitored	FDA Status
Apple	Apple Watch	HR, PA, falls, sleep and ECG	Cleared
SmartCardia	INYU	HR, PA and ECG	Not cleared
Withings	Steel HR, Move,Move ECG, Pulse HR	HR, PA, sleep, ECG and SPO2	Not cleared
iRhythm	Zio Patch	HR and ECG	Cleared
Preventice Solutions	BodyGuardian	HR and ECG	Cleared
Corventis Inc	Nuvant MCT	HR and ECG	Cleared
Bardy Dx	BardyDx CAM	HR and ECG	Cleared
BioTelemetry	BioTel Heart	HR and ECG	Cleared
MediBioSense	MediBioSense MBS HealthStream	HR and ECG	Cleared
Huinno	MEMO Patch	HR and ECG	Not cleared
Samsung	S-Patch Cardio	HR and ECG	Not cleared
AliveCor	KardiaBand (commercially discontinued)	HR and ECG	Cleared
AliveCor	KardiaMobile	HR, single-lead and 6-lead ECG	Cleared
Omron + AliveCor	Complete	HR, BP and ECG	Cleared
SonoHealth	EKGraph	HR and ECG	Not cleared
Komodo Technologies	AIO Smart Sleeve	HR, PA, ECG and sleep	Not cleared
Zephyr	BioHarness 3 clothing	ECG, HR, PA, respiratory rate	Cleared

Table 1.1: Summary of Wearables for ECG Monitoring

These wearable devices are generally designed for monitoring patients remotely. Specific devices can monitor ECG and send the data to the cloud servers for further analysis. However, it can be observed that AI-enabled ECG devices can process ECG data on edge itself to detect different cardiovascular diseases. However, there are still many life-threatening cardiac ailments such as arrhythmia, myocardial infarction, congestive heart faliure that the available devices cannot diagnose. Apple and Samsung Galaxy watches can monitor single lead ECG and blood oxygen. Apple watch can also detect fibrillation using ECG. Similarly, Fitbit sense can record 30 sec ECG recording whenever a user wants, along with stress monitoring. KardiaMobile 6L and QardioCore Chest are personal medical-grade ECG devices that can continuously monitor ECG and measure heart rate variability. Note that these devices efficiently monitor ECG signals and other vital body parameters; however, they are quite expensive and still need to be equipped with on-chip detection of different cardiac diseases, such as

Arrhythmia, Myocardial Infarction.

Moreover, significant developments in IoT have also made it possible to process and transmit data on edge itself. These advancements pose multiple design challenges and tradeoffs. Among many challenges, wearable devices' major hurdle is low energy dissipation. Thus, it is imperative to develop energy-efficient low-cost performance-optimal wearable devices with acceptable accuracy in detecting and predicting cardiovascular diseases in a non-invasive manner.

In this thesis, we aim to develop low-power machine learning enabled VLSI architectures for wearable devices to detect cardiovascular diseases. The proposed architectures perform ECG signal feature extraction and later utilize machine learning models to classify various cardiovascular diseases. Further, the thesis also targets the problem of ECG data compression. As described in previous sections, the wearables continuously monitor the ECG signals and create a large amount of data that needs to be transferred to medical facilities. Therefore, the data can be compressed and transmitted to clinicians to save power. This thesis also proposes a low-power VLSI architecture for ECG data compression.

1.3 Research Objectives

The main research objective of the proposed work is to develop optimized algorithms for ECG signal analysis using machine learning models and implement them on ASIC. The proposed design can detect and predict several cardiovascular diseases and is suitable to be employed in wearable devices.

A conventional wearable device has three significant blocks, as shown in Figure 1.1. The first block is the sensor and analog front, responsible for acquiring the ECG signals and converting them to digital samples. The second block is an ECG co-processor, including feature extraction and a machine learning classifier, responsible for ECG signal analysis and classification of cardiovascular diseases. The third block is the data compression and transmitter block. Based on the survey on wearable devices and identified research gaps, the objectives of the proposed work are given below.

(i) Design low-power VLSI architecture for ECG delineation, which can detect all the features of ECG signal. These features can further be used for different cardiovascular disease detection.

(ii) Develop an optimized algorithm and then implement it on ASIC to detect different types of arrhythmia

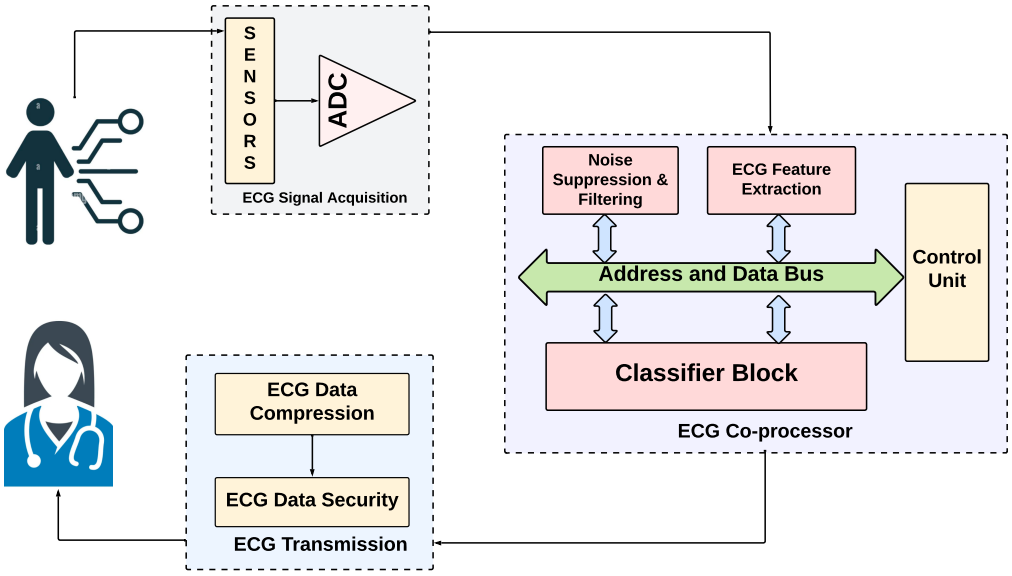


Figure 1.1: Generic Block Diagram of conventional Wearable device ECG SOC

(iii) Propose a low-power design for detecting Ventricular Arrhythmia before its occurrence

(iv) Develop a low-power architecture that can classify different severity stages of Myocardial Infarction

(v) Target to design an optimized architecture for ECG data compression and decompression

1.4 Organization of the Thesis

The work presented in this thesis is organized into six chapters. The content of each chapter is summarized as follows.

- Chapter 2 proposes optimal ECG feature extraction algorithms, which can extract the primary features, namely P-QRS-T waves. The first algorithm utilizes wavelet transform, whereas the other algorithm employs single and double differentiation methods to delineate the ECG features. Further, the proposed methods are implemented on hardware to obtain a low area and low power VLSI architecture, which can be utilized for wearable healthcare devices.
- In Chapter 3, a low power and area-efficient VLSI architecture is designed to classify different arrhythmia beats using ECG signals. A deep neural network is employed for classification. Further, the chapter proposes a design that can predict cardiac arrhythmia before its occurrence. The designs are implemented on hardware using SCL 180nm technology PDKs.

- Chapter 4 proposes a low power VLSI architecture, which can classify different severity stages of Myocardial Infarction(MI). The proposed design utilizes a simple if-else-based classifier to classify various stages of MI. Later, the design is implemented in Verilog HDL and its ASIC realization is performed using SCL 180nm PDKs.
- Chapter 5 presents a low power VLSI architecture for ECG data compression and decompression. The proposed design utilizes an optimal encoding scheme and various hardware optimizations, leading to a low area and low power design of the compression and decompression algorithms.
- Chapter 6 concludes the proposed work in the thesis highlighting the primary contributions. Further, a roadmap for future research work is presented.

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी
Indian Institute of Technology Guwahati

2

ECG Feature Extraction

Contents

Objective

An accurate representation of the ECG signal has a vital role in the diagnosis of various cardiovascular diseases. The fiducial points extracted from ECG are used individually or in groups for different analysis and classification of heart abnormalities. Traditionally, the analysis of ECG takes place on bulky bedside devices or offline using a Holter device. The engineering requirements for the present-day mobile and wearable healthcare system is to develop an automated ECG analysis with low complexity and minimal power demand. The algorithms designed should be immune to the errors that may arise from undesirable noise sources and should ensure the accurate delineation of the heartbeats as someone's life might be at stake. Moreover, to make algorithm hardware efficient, resource utilization has to be optimized, and low power techniques should be incorporated to cater to the battery needs. Therefore, the objective of this chapter is to develop a computationally efficient algorithms and to design memory and power-efficient architectures for extracting ECG features. This chapter presents two low power and computationally simple algorithms which are based on wavelet transform and derivative based techniques

2.1 Introduction

One of the prime application of IoT healthcare is the wearable devices that enable individuals to monitor their health statistics round the clock. The number of patients having heart abnormalities have been increasing due to stressful and unhealthy lifestyle since past few decades. According to the World Health Organization, cardiovascular diseases (CVD) takes a life toll of about 17 million each year worldwide [12]. By the year 2020, heart diseases have become the primary cause of mortality and disability globally. WHO anticipates this figure to increase about 24 million each year by the next decade. The rise in mortality rate is due to scarce medical facilities and healthcare centres that leads to delay in diagnosis of pathologies. Therefore, the need of the hour is to develop personalised cardiovascular disease monitoring wearable devices that are compact and reliable, and can operate in low power. These devices should monitor patients' continuously and connect them to the doctors preventing them from reaching any critical conditions. These devices should have strict power budget requirements so that they can run on low power to improve battery life. It necessitates the development of area and power efficient VLSI architectures that can perform ECG analysis in power and resources constrained environment.

2.2 Preliminaries

Bio-signals refer to the electrical signals generated due to potential differences in tissue, organ, or cell systems. These signals can be measured and monitored in living beings. Typical electrical bio-signals are electrocardiogram (ECG), electroencephalogram (EEG), and electromyogram (EMG). The ECG is a representation of the electrical activity of the heart. The heart is the pumping mechanism of the human body responsible for blood flow to the whole body. It has four chambers, two upper chambers known as atria and the lower ones known as ventricles. The sinoatrial (SA) node, also known as the pacemaker of cells, leads to the contraction of the atria. From the SA node, this signal travels to atrioventricular (AV) node and the bundle of his [10]. This electrical signal leads to the contraction of the ventricles. The electrical impulses follow this set pathway under normal working conditions of the heart. The complete electrophysiology of the human heart is shown in Figure 2.1.

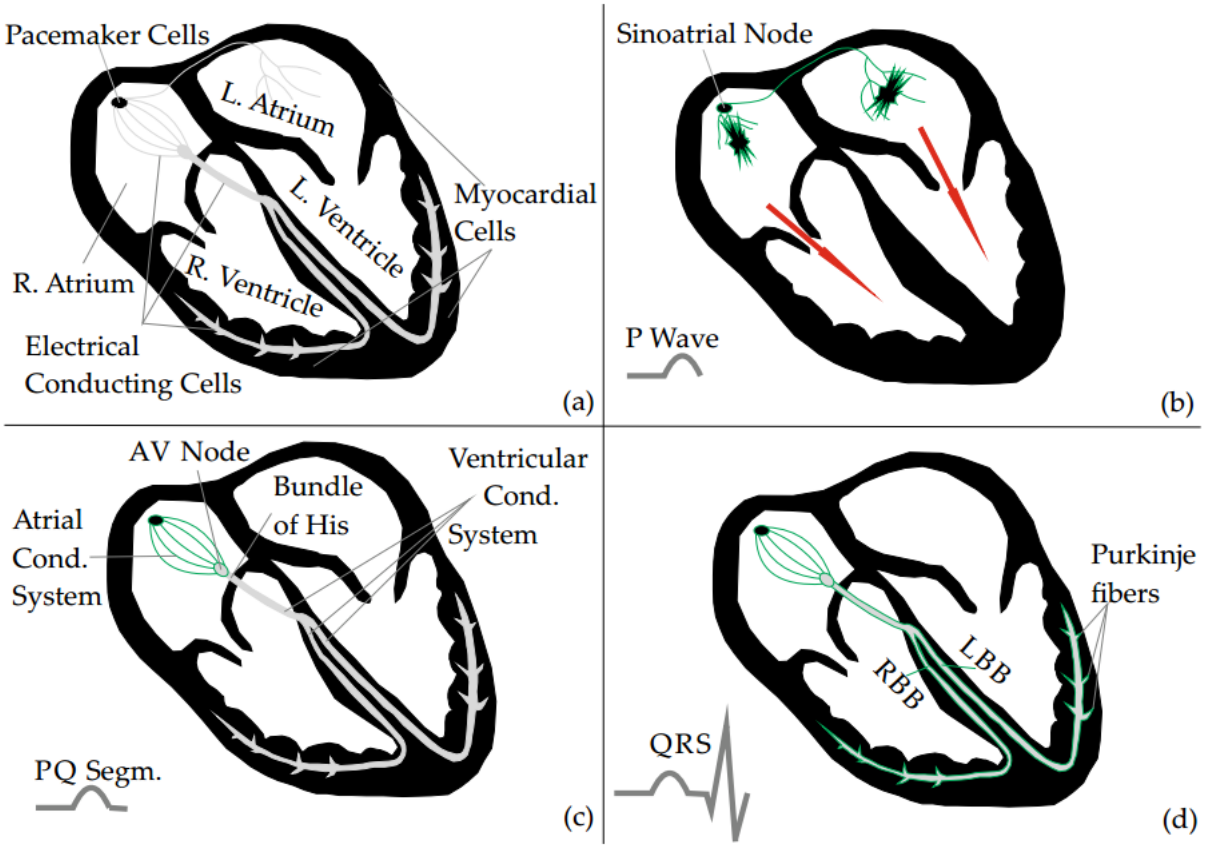


Figure 2.1: Electrophysiology of Human Heart (a) Chambers of Heart (b) Atrial Depolarization (c) Delay of AV Node (d) Ventricular Depolarization [1]

ECG tracks the potential changes throughout the heart's conduction system and the subsequent

contraction of the working myocardium [11]. Figure 2.2 depicts a typical ECG waveform. To understand an ECG, first, one has to understand the grid markings of the ECG paper, as shown in Figure 2.2. The paper has thin and thick lines every 1 mm and 5 mm, forming small and large squares. Various intervals and the cardiac rate is measured on the horizontal lines. The standard paper speed is 25 mm per second, making the thin lines appear at 0.04-second intervals, and thick lines occur at 0.20-second intervals. On the other axis of the scale, the vertical lines enable the measurements of the waveform amplitudes. At the standard calibration of 10 mm per mV, the thin lines are at 0.1 mV increments, and the thick lines are at 0.5 mV increments. Therefore, the dimension of each small square is 0.04 second × 0.1 mV, and that of each large square is 0.20 second ×0.5 mV. Every ECG has a set of seven features that requires systematic examination. These features are mentioned below [10].

- Rate, regularity and rhythm
- P-wave
- PR interval
- QRS complex
- ST-segment
- T-wave
- QTc interval

Most cardiac health information is contained in three dominant waves, the P wave, the QRS complex, and the T wave. These waveforms are systematically approached by analyzing their contours, durations, and amplitudes.

2.2.1 ECG Fiducial Points and Intervals

As described earlier, an ECG has seven main features. These features represent different phases of the cardiac cycle and are widely used to assess heart abnormalities. Table 2.1 gives the range of typical parameters of the normal ECG [2, 10].

- **Regularity and Rhythm:** The cardiac rate and rhythm are rarely regular. Even if the sinoatrial node (SA) initiates electrical impulses normally, the heart rate gets affected by the nervous system and phases of the respiratory cycle. Therefore, the sequence of the cardiac cycles is analyzed to determine the proper functioning of the heart. Generally, a normal ECG has an equal number of QRS complex and P waves. QRS complexes are used to determine the heart rate. It is easily determined by counting the number of the large square between two

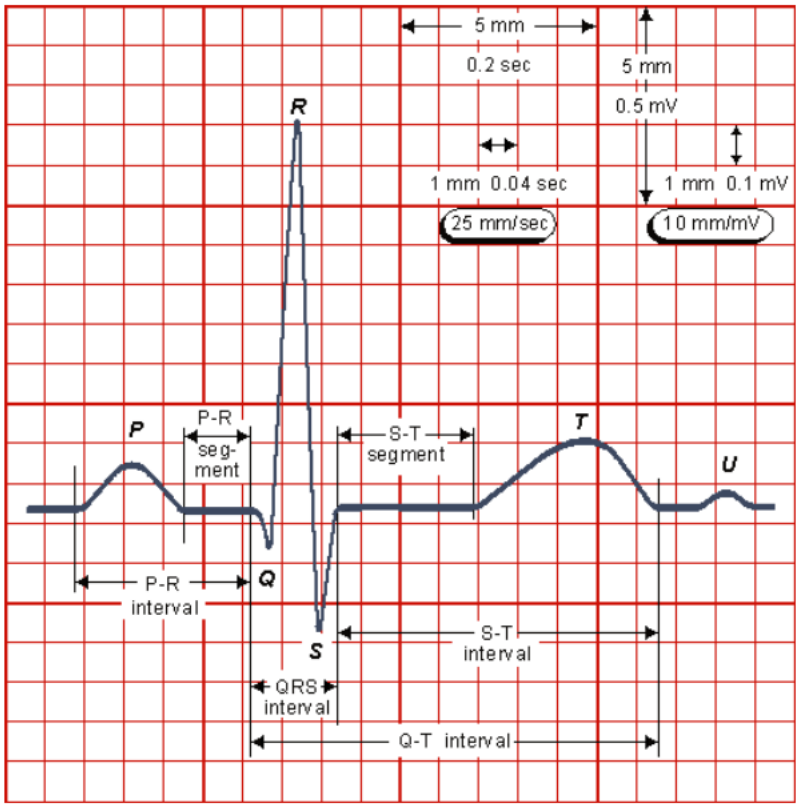


Figure 2.2: Normal ECG Signal [2]

cycles. Since each square is one-fifth of a second, and there are 300 such squares in a minute, determining the large squares and dividing them by 300 gives the heart rate.

- **P Wave:** The P wave is found before the QRS complex in normal or slow heart rates. P waves are hard to detect for fast heart rates as they may merge with the previous T waves. The contour of the P wave is smooth, and is entirely positive or negative. P wave originates due to the contraction of the atria with an amplitude between 0.1 mV and 0.2 mV.
- **PR Interval:** PR interval is measured between the beginning of the P wave to the QRS complex. It represents the time required for an electrical pulse to reach from the SA node to the ventricular myocardium, which represents the atrioventricular delay. The delay signifies the activation of the AV node that maintains the heartbeat. Note that irregularities in the PR interval point to the presence of arrhythmias. The normal range of the PR interval is between 0.12 to 0.20 seconds.
- **QRS Complex:** The QRS complex represents the time taken by ventricles to depolarize. As the ventricles are more muscular than the atria, the conduction velocity of an electrical impulse

Feature	Description	Duration(ms)	Volatage(mV)
RR Interval	Interval between two consecutive R waves	60-120	NIL
P Wave	First short wave of cardiac cycle	90-110	0.25
PR Interval	Measured from the beginning of the P wave to the beginning of the QRS complex	120-220	NIL
QRS Complex	Normally begins with a downward deflection Q, a larger upwards deflection R and ends with a downward S wave	80-120	1.60
ST Segment	End of QRS complex to the beginning T wave	50-150	0
T Wave	Wave after QRS complex	120-160	0.1-0.5
QT Interval	Measured from the beginning of the QRS complex to the end of the T wave	350-440	NIL

Table 2.1: ECG Fiducial Points and Intervals

is more, resulting in the high peaked R wave. Usually, the heart rate is detected using these R peaks. Typically QRS complex range between $60 - 100$ ms.

- **ST Segment:** ST-segment depicts the complete depolarization of the ventricles. This segment has a crucial role in diagnosing ischemia and myocardial infarction. In these conditions, the ST segment can become either depressed or elevated. The total duration of the ST segment is about 0.43 seconds.
- **QTc Interval:** QT interval is the time between the start of the QRS complex and the end of the T wave. It represents the total time a heart needs to depolarize and repolarize. Generally, the QTc interval is considered for evaluation instead of the QT interval. QTc can be described as mentioned below.

$$QT_c = \frac{QT}{RR_{Interval}}$$

- **T Wave:** T wave is found by ventricular repolarization. T wave is sharp or blunt round and has a typical value of 0.10 to 0.25 seconds.

As explained , ECG has three primary features namely P wave, QRS complex and T wave. Any variation in the standard values of ECG features and derived intervals aids cardiologists to diagnose cardiac abnormalities [10]. Hence, accurate and efficient algorithms are required that can process the acquired ECG signal to extract fiducial points. The features extracted from an ECG are used individually or in a group for different analysis and classification of heart abnormalities.

2.3 Prior Works

In this digital and transformational world, monitoring and maintaining one's health may prove to be a great challenge. Therefore, in the late 20^{th} century, when the manual entry of cardiac activi-

ties turned to automation, the use of heart rate monitors mushroomed in the health industry [13]. Gradually heat monitoring commercialized into wearable devices that aids in accessing and recording a person's fitness activities continually. With more advancements, wearable ECG sensors were introduced that can detect and report the heart abnormalities in real-time. If these devices can produce an accurate diagnosis of various heart issues, the number of lives can be saved. But, significant advancements come great challenges. The ECG wearable devices have to be immune to the noise interferences that might corrupt the primary ECG signal. Also, to ensure accurate real-time monitoring, the device and the algorithms running on it should consider the spectrum of heart rates exhibited during various physical activities and heart abnormalities to ensure minimum false detection [14–16].

Accurate analysis and delineation of ECG signals play a crucial role in diagnosing CVD. Therefore, ample research has been done to develop efficient algorithms and hardware for ECG feature extraction. Moreover, to make algorithms hardware efficient, resource utilization has to be optimized, and low power techniques should be incorporated. Pan and Tompkins [17] developed a QRS detection algorithm that has remarkable accuracy of 99.3%. This algorithm includes several complicated operations, such as differentiation, squaring and moving average. Despite a remarkable accuracy, it is inefficient for the area optimal and low power hardware implementation due to computational complexity. Several algorithms are developed that utilise discrete wavelet transform for ECG feature extraction [18–20]. These algorithms employ modulus maxima analysis and time-domain refinement to extract QRS complex and, P and T wave. These proposed algorithms use multiscale resolution coefficients increasing the memory footprint. Tekeste T. et al. introduce curve-length transform for QRS detection and discrete wavelet transform for P and T wave extraction [21]. The architecture developed for this method was realised using $65nm$ technology consuming $642nW$ power while operating at a frequency of $7.5kHz$. However, the use of both curve-length transform and wavelet transform makes it computationally complex. Sanjeev et el. [22] developed an energy-efficient ECG signal processor to detect cardiovascular diseases such as arrhythmia, myocardial infarction and more. The work is implemented on $130nm$ technology utilizing $96pJ$ of energy. The cited work does not address the delineation of boundaries of the P wave and the onset of the T wave. Bote et el. [23] developed a modular design in which the quality of delineation is adjusted in runtime as per medical requirements. The algorithm is processed using the $TIMSP430$ series microcontroller. These general-purpose microcontrollers could be a central processing unit for wearable healthcare devices. However, the active

power dissipation of these microcontrollers is much higher than the custom ASIC processors [24]. Also, the paper utilizes morphological operations to detect the onset and offset of ECG peaks that add up to hardware resource utilization. Table 2.2 presents a complete comparison of different state-of-the-art methodologies for ECG feature extraction.

S.NO	Title	Year	Features Extracted	Methodology	Computational Load
1	A low complexity ECG feature extraction algorithm for mobile healthcare applications [18]	2013	QRS complex, P wave, Twave	Discrete wavelet transform	Medium
2	An Ultra-low power QRS complex detection algorithm based on down-sampling wavelet transform [25]	2013	QRS complex	Continuous wavelet transform	Low
3	Adaptive ECG interval extraction [26]	2015	QRS complex, P wave, T wave	Curve length transform Discrete wavelet transform	Medium
4	A modular low-complexity ECG delineation algorithm for real-time embedded systems [23]	2018	QRS complex, P wave, T wave	Derivative and filters	Medium
5	A modified algorithm for QRS complex detection for FPGA Implementation [27]	2018	QRS complex, R peaks	Integer Discrete wavelet transform	Medium
6	A Nano-watt ECG feature extraction engine in 65-nm technology [28]	2018	QRS complex, P wave, T wave	Curve length transform Discrete wavelet transform	Medium
7	Ultra-low power QRS Detection and ECG compression architecture for IoT healthcare devices [24]	2019	QRS complex	Absolute-Curve length transform, Discrete wavelet transform	Low
8	Simple real-time QRS detector with the MaMeMi filter [29]	2015	QRS complex	Non-linear filter	High.
9	ECG signal classification and parameter estimation using multiwavelet transform [30]	2017	QRS complex	Wavelet transform	High
10	Design and Implementation of an Ultralow-Energy FFT ASIC for Processing ECG in Cardiac Pacemakers [31]	2018		Fast Fourier transform (FFT)	Medium
11	A novel two-dimensional ECG feature extraction and classification algorithm based on convolution neural network for human authentication [32]	2019	QRS complex, P wave, T wave	Neural Network	High

Table 2.2: Literature on ECG Feature Extraction

This chapter proposes two different optimized algorithms to extract all characteristic features of ECG waves. An area and power-optimized VLSI architecture is then designed for the proposed algorithms. The architecture is designed so that it utilizes no floating-point operations. Further, the architecture designed are modular in nature and can be operated as per the medical requirements. If the complete delineation is not needed, some modules such as P and T wave block can be disabled to reduce computational load and power. These modules can be turned on if any anomaly is detected in ECG. This modular architecture makes it a suitable candidate for wearable healthcare devices to detect cardiovascular diseases further.

2.4 Proposed Architecture for Integer Haar Wavelet Transform Based ECG Feature Extraction

This section presents an integer haar wavelet based algorithm to extract fiducial points of ECG signal. Later the VLSI implementation of the proposed algorithm is presented.

2.4.1 Wavelet Transform Basics

Wavelet transform is a linear transformation employed to process non-stationary signals [33]. It performs an overall frequency analysis of the signal and also provides a signal's time-domain based information. It carries out analysis on different levels with a proper resolution by breaking signals into various frequency levels. Hence, in short time duration, high-frequency signals are evaluated, and similarly, long time intervals are considered to analyse low-frequency signals. The advantage of inspecting signals in different frequency levels prove to be efficient for characterising and delineating non-stationary signals, such as ECG. For a signal $f(t)$, wavelet transform in continuous time domain is given as Equation 2.1, where $*$ is the complex conjugate of any function and $\psi_{x,y}(t)$ is the basis function or mother wavelet. x and y are termed as the scaling factor and translation factor, respectively. Therefore, $\psi^*(\frac{t-y}{x})$ represents a shifted and scaled version of basis function.

Discrete wavelet transform (DWT) is potent as compared to continuous wavelet transform (CWT) due to less redundancy and better time complexity for a large range of applications. The DWT of any signal $x[n]$ is calculated by applying a set of low pass and high pass filters with impulse responses g and h, respectively. The output obtained from convolution of signal $x[n]$ with g and h is given by Equation 2.2.

$$\begin{aligned} W(x,y) &= \int_{-\infty}^{\infty} f(t)\psi_{x,y}(t)dt \\ \psi_{x,y}(t) &= \frac{1}{\sqrt{a}}\psi^*(\frac{t-y}{x}) \end{aligned} \tag{2.1}$$

$$\begin{aligned} y[n] &= \Sigma_{k=-\infty}^{\infty} x[k]g[n-k] \\ y[n] &= \Sigma_{k=-\infty}^{\infty} x[k]h[n-k] \end{aligned} \tag{2.2}$$

Detail coefficients and approximate coefficients of the signal are obtained from the output of high pass filter (h) and low pass filter (g), respectively. The two filters are called Quadrature Mirror Filters. In general, the wavelet coefficients are calculated using Mallat's algorithm [18] as shown in Figure 2.3. As we know, the signal $x[n]$ is applied to LPF and HPF to realise detailed and approximate

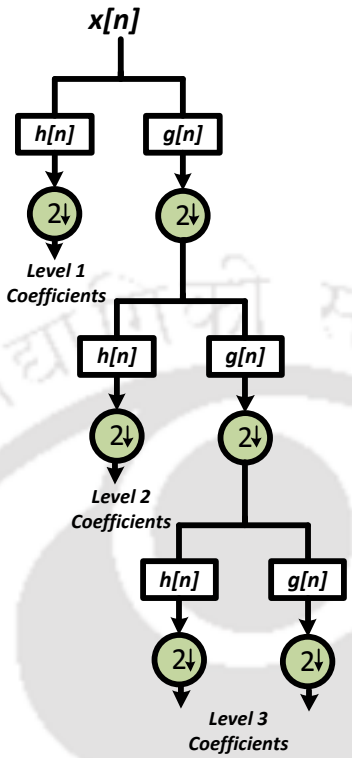


Figure 2.3: Mallat's Algorithm

coefficients. These coefficients are downsampled by 2, thus, generating the coefficients of the level 1 (2^1) in Mallat's algorithm. The approximate coefficients of level 1 are fed to LPF and HPF, and then again downsampled to obtain level 2 coefficient (2^2). This process is continued until the resolution required for a specific application is achieved. There are a wide variety of wavelet functions available [33] which are utilised in various applications.

2.4.2 Integer Haar Transform

The simplest wavelet among all the mother wavelets is the Haar wavelet. The coefficients of the transfer function of LPF and HPF for Haar wavelet are described in Equation 2.3.

$$\begin{aligned} g &= [\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}] \\ h &= [\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}] \end{aligned} \tag{2.3}$$

In [34], the authors elaborated uniqueness and simplicity of the Haar wavelet and compared it with the other wavelets. It is observed that Haar wavelet outperforms other wavelets due to its less memory

requirement and reduced computational complexity. Although Haar wavelet is highly efficient, there are challenges in its hardware implementation due to the involvement of floating-point arithmetic. Therefore, integer Haar wavelet (IHT) is adopted for avoiding these floating point calculations. The approximate and detailed coefficients (CA *and* CD) of IHT are presented in Equation 2.4.

$$\begin{aligned} CA[n] &= \lfloor \frac{1}{2}x[2n] + \frac{1}{2}x[2n+1] \rfloor \\ CD[n] &= x[2n] - x[2n+1] \end{aligned} \tag{2.4}$$

It can be observed that the basic operations, such as addition, subtraction and division by 2 are required to calculate wavelet coefficients. Further, division with 2^n can be realized using a simple right shift operation. The absence of floating point operations reduces complexity of hardware implementation of IHT and makes its realization efficient in the digital logic systems.

2.4.3 Wavelet Based Feature Extraction Algorithm

The prime application of our proposed work is wearable healthcare devices, where the ECG features processing and associated computations take place in power and area constrained environment [35]. Therefore, limiting the complexity and power consumption of the algorithm is essential and critical engineering necessity of the proposed method. Moreover, ECG analysis algorithm needs to fulfil clinical requirements of producing acceptable results by introducing the least error in the delineation of features. These two criteria pose significant constraints on the employment of relevant signal processing methods in terms of their physical implementation in an energy-constrained environment. Therefore, in this section, a modified and efficient algorithm aiming towards optimal power dissipation and clinical accuracy is described in detail.

Algorithm 1 represents proposed algorithm to detect QRS complex. DWT is utilised in this algorithm owing to its effectiveness. The most considerable advantage of DWT lies in its time-scale nature that can inherently separate artefacts like isoelectric line wandering and associated noises from the ECG signal [36] which is processed by employing Integer Haar wavelet (IHT). As explained in the previous subsection, the hardware implementation of IHT is preferred due to its integer nature. IHT avoids any floating-point calculations; therefore, it can be implemented using simple addition, subtraction and shift operations. Although IHT has its limitations, it is an appropriate choice for the implementation of digital logic systems dedicated to low power healthcare [27]. The algorithm is divided into three main stages.

(i) The algorithm begins with the detection of R peak.

(ii) Once the R peaks are found, QRS onsets and offsets can be delineated, if required, before and after the R peak.

(iii) After the extraction of QRS complex, delineation of P and T waves can be performed whenever needed.

QRS Detection Module:

The proposed ECG feature extraction algorithm operates on the ECG signal sampled at $250Hz$. Both frequency analysis and time-domain refinement are employed for delineating ECG fiducial points. Initially, a window of 800 samples (N) of ECG signals is chosen for processing. It is observed that during certain cardiac abnormalities, such as Arrhythmia, the RR interval deviates from its standard value. The minimum two beats are required to calculate RR interval and, P and T waves. A window of 800 samples enables proposed algorithm to capture at least two P-QRS-T beats, when ECG is sampled at $250Hz$ for normal as well as for abnormal patients. This leads to correct delineation of all the primary features in an abnormal ECG as well. As depicted in Algorithm 1, detailed coefficients (CD_3) of third dyadic scale (2^3) are computed for P-QRS-T extraction. These coefficients are calculated using Mallat's algorithm with IHT as mother wavelet, as shown in Figure 2.3. Once these coefficients are computed, the absolute maximum of the (CD_3) coefficients are found (Equation 2.5), and threshold is calculated using Equation 2.6 to determine the position of R peak. The value of the threshold is updated after every 800 samples making it adaptive to the variation in ECG signal.

$$abs\_max = max(CD_3) \tag{2.5}$$

$$th\_R = \frac{abs\_max}{4} \tag{2.6}$$

After calculating the threshold value, an initial estimation of R peak is realised. The values of each CD_3 coefficients are compared with the threshold value. If the magnitude of the coefficient is higher than the threshold value, then its value is stored as initial value of R peak. Later the next 16 samples of (CD_3) are skipped in order to avoid detection of more than one R peak from the same QRS complex. Initial R peak positions in the original ECG signal are found by mapping selected CD_3 coefficients to the signal. This mapping is accomplished by multiplying CD_3 coefficients with 2^3 as it is evident from Figure 2.3.

Algorithm 1.

Input: ECG x[n] of length N
Initial estimation of R peaks
Calculate detailed coefficients CD_3 of **x[n]**
Calculate $abs\_max = max(CD_3)$
Threshold for R peak: $th\_R = abs\_max >> 2$
$j \leftarrow 0$
for $i = 0, \cdots, n_{coef}$ **do**
    if $CD_3 > th\_R$ **then**
        $R\_pos\_temp_j = CD_3 * 2^3$
        $i \leftarrow i + 16$
        $j \leftarrow j + 1$
    else
        $i \leftarrow i + 1$
    end if
end for
R peak Delineation
for $k = 0, \cdots, j$ **do**
    $R\_peak_k = max(x[n])$, $n \in (R\_pos\_temp_k - W1, R\_pos\_temp_k + W1)$
    $R\_peak\_pos = pos(max)$
    $k \leftarrow k + 1$
end for
Q and S peak Delineation
for $l = 0, \cdots, j$ **do**
    $Q_l = min(x[n])$, for $n \in (R\_peak\_pos, R\_peak\_pos - W2)$
    $S_l = min(x[n])$, for $n \in (R\_peak\_pos, R\_peak\_pos + W3)$
    $l \leftarrow l + 1$
end for
RR Estimation
for $m = 1, \cdots, j$ **do**
    $RR_{m-1} = R_m - R_{m-1}$
    $m \leftarrow m + 1$
    if $m > 1$ **then**
        $en\_p = 1$
        $en\_t = 1$
    end if
end for

end

The standard duration of the QRS complex is $120ms$, which may increase and decrease during certain cardiac abnormalities. Therefore, in order to find final values of R peaks, a dynamic window of $2 \times W1$ as per the standard value of QRS complex is chosen to mark R peak. For a positive QRS complex, Q and S peaks are the minimum peaks before and after R peaks. For a standard duration of QRS complex, a window of $60 - 80ms$ ($W1$ *and* $W3$) is selected to delineate Q and S peaks. The minimum of $W1$ and $W2$ are marked as Q and S peaks, respectively. Similarly, the onset and offset of QRS complex are delineated as the maximum value in an adaptive window of $40ms$ after Q and S peaks, respectively. RR intervals are also computed in parallel from the detected R peaks.

P and T Wave Detection:

As we know, P wave is the starting wave of the heartbeat. In normal patients, its duration is less than $120ms$, but it expands for the abnormalities, such as atrial fibrillation up to $170ms$ [37]. The PR interval is of $120ms - 200ms$ [38]. Thus, the P wave is delineated considering a window ($W4$) of $100ms - 200ms$ before the QRS complex. The window should never exceed half of the RR interval. Choosing a window according to the RR interval makes the algorithm adaptive to the varying heart rate of a particular patient; therefore, detection is performed accurately even for abnormal ECGs. Algorithm 2 exhibits the flow of P wave extraction from ECG signal. CD_3 coefficients for this chosen window are examined, and the minimum and the maximum values of the coefficients are obtained. If the minimum value is obtained before the maximum value, then P wave is considered to be positive else it is considered as inverted. The positions of the minimum and the maximum values of CD_3 coefficients are multiplied by a factor of 2^3 to obtain a temporary position of start (P_on) and end (P_off) of P wave in the original ECG signal. For refining the delineation, a window ($W5$) of $40ms$ is chosen on both the sides of initial values of P_on and P_off. The window $W5$ is chosen adaptively considering the sampling frequency of $250Hz$ to get the accurate delineation results on the ECG excerpts. The minimum values of the respective windows are taken as the final boundaries of P waves. The maximum value of the window is considered if P wave is inverted in nature. Value of P peak is extracted by finding the maximum/minimum value in the interval P_on and P_off as per the orientation of P wave.

Algorithm 2.

Calculation of P peaks
When $en\_P == 1$
$flagP \leftarrow 0$
for $c = 0, \cdots, j$ **do**
  $P1_c = min(CD_3[n])$, for $n \in (W4)$
  $P2_c = max(CD_3[n])$, for $n \in (W4)$
  if $P1_c < P2_c$ **then**
    $Pt\_on_c = P1_c * 2^3$
    $Pt\_off_c = P2_c * 2^3$
    $flagP \leftarrow 1$
    if $flagP == 1$ **then**
      $P\_on_c = min(x[n])$, for $n \in (Pt\_on_c - W5, Pt\_on_c)$
      $P\_off_c = min(x[n])$,for $n \in (Pt\_off_c, Pt\_off_c + W5)$
      $P\_peak_c = max(x[n])$,for $n \in (P\_on_c, P\_off_c)$
      $flagP \leftarrow 0$
    end if
  else
    $Pt\_on_c = P2_c * 2^3$
    $Pt\_off_c = P1_c * 2^3$
    $flagP \leftarrow 1$

TH-3060_176102001

```
        if flagP == 1 then
            P_on_c = max(x[n]), for n ∈ (Pt_on_c − W5, Pt_on_c)
            P_off_c = max(x[n]), for n ∈ (Pt_off_c, Pt_off_c + W5)
            P_peak_c = min(x[n]), for n ∈ (P_on_c, P_off_c)
            flagP ← 0
        end if
    end if
    c ← c + 1
end for
T wave extraction
if en_t == 1 then
    Repeat P block with n ∈ W6
end if
```

end

As we know, last characteristic of a heartbeat is T wave. Its duration is of $125ms - 200ms$ in normal ECG, whereas, the QT interval is less than $425ms$ [39]. Considering these circumstances, a window $W6$ of $200ms - 400ms$ after the QRS complex marked to extract the T wave. However, this window should never exceed half of the RR interval. Rest of the procedure to find T wave is similar to P wave delineation mentioned above.

2.4.4 Architecture Details of Proposed Algorithm

The proposed architecture contains three main modules; QRS module and an optional P wave module and T wave module. The P and T wave modules are subdivided into boundary detection and peak detection modules. The modules are enabled and disabled according to the medical requirements or when they are nonoperational.

The hardware implementation of the proposed algorithm is realised in Verilog using Xilinx 2016.4 development environment Vivado and Xilinx Zynq™−7000 FPGA platform [40]. The modules are validated for the desired functionality through simulation using Vivado. Later these modules are ported on FPGA for further verification of the proposed algorithm at the hardware level. The basic architecture of ECG delineation algorithm is illustrated in Figure 2.4. The architecture blocks are controlled by a finite state machine. Different ECG signal samples have been utilised for the analysis of our proposed algorithm. ECG signal is fed serially as input into the system and is stored in a memory. The position of ECG features is obtained as output. Initially, the ECG signal is decomposed into its DWT coefficients utilising a cascaded filter bank structure, as shown in Figure 2.3. The IHT block is designed to decompose a signal into detailed and approximate coefficients using integer Haar transform as shown in Figure 2.5. Three such IHT blocks are pipelined to obtain CD_3 coefficients

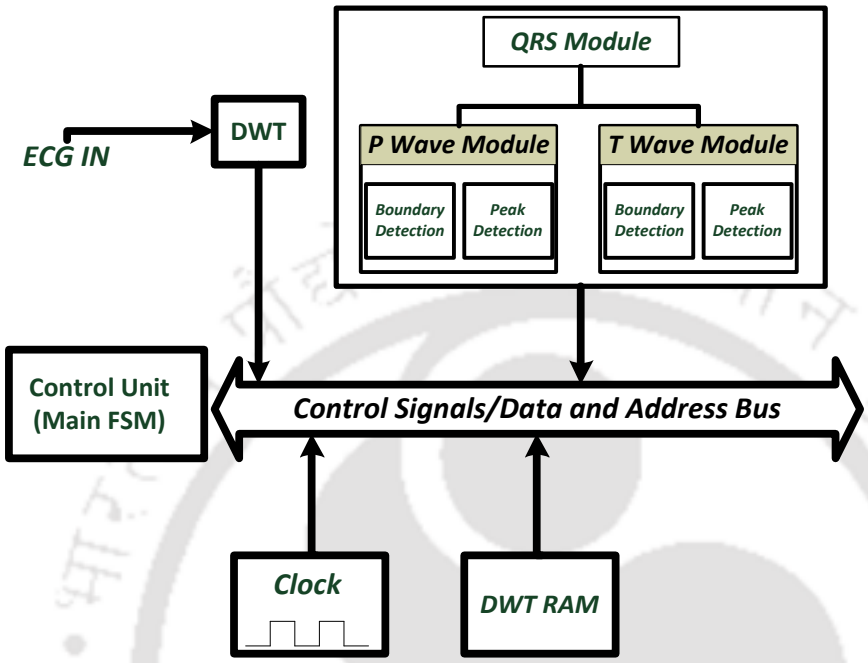


Figure 2.4: Block Diagram of ECG Feature Extraction Algorithm

making a complete DWT block. The coefficients are stored in memory for further processing. The DWT and comparator blocks are pipelined to obtain third scale wavelet coefficients along with their absolute maximum, which reduce delay in the calculation of R peak threshold. Figure 2.6 represents the architecture to compute threshold of R peak.

The present and previously calculated CD_3 coefficients, CD_{3_i} and $CD_{3_{i-1}}$ respectively, are compared to find the maximum and the process is repeated for every new CD_3 coefficient being calculated using the previous step. The threshold for R peak detection is then realised using Equation 2.6. This threshold is updated after every 800 input samples making the proposed architecture adaptive to the variations in ECG. Once the threshold is found, R peaks are detected in CD_3 coefficients, as explained in the algorithm and are mapped to the primary ECG signal subsequently by multiplying with 2^3. This multiplication by 2^3 is implemented as left shift operation in hardware by shifting CD_3 coefficients values by three bits. Once temporary R peaks are calculated, the final R peaks are retrieved by searching a maximum 16 samples before and after temporary R peaks using a comparator. Once the QRS complex is extracted, P and T waves extraction can be performed. In the proposed architecture,

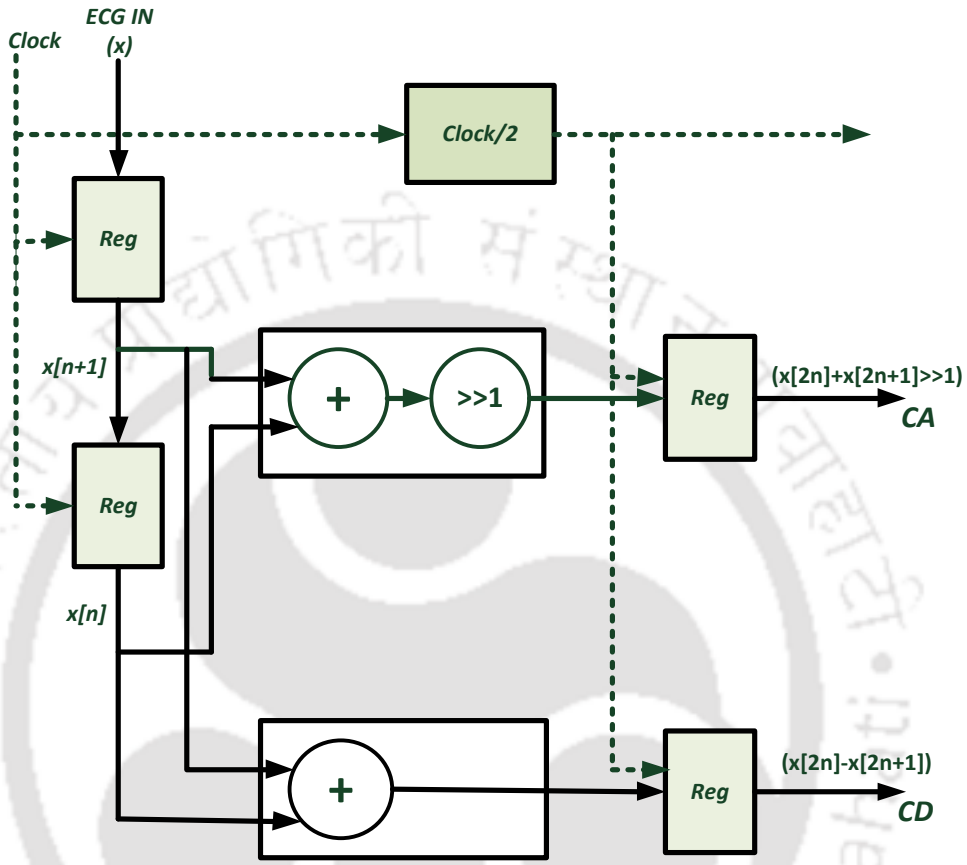


Figure 2.5: Architecture of IHT Module

clock gating technique is utilised to minimise the power consumption of the proposed design [28]. Clock gating controls the activation of different blocks as per the requirement. Since all the processes do not run simultaneously, blocks can be turned off, when not in use, to save power. Clock gating is exhibited in Figure 2.7. As shown in Figure 2.7, P and T modules are controlled by clock and an enable signal. The module operates when enable signal (en_P) and (en_T) becomes high. P and T waves are then delineated as described in the algorithm. Both the modules can operate in parallel, if required, and utilise a simple comparator circuit to find the maximum and minimum values. When QRS module is active, then P wave and T wave module can be disabled after completing their ongoing tasks, which reduces power consumption. The architecture designed utilises fixed point computations and simple hardware blocks making it computationally simple.

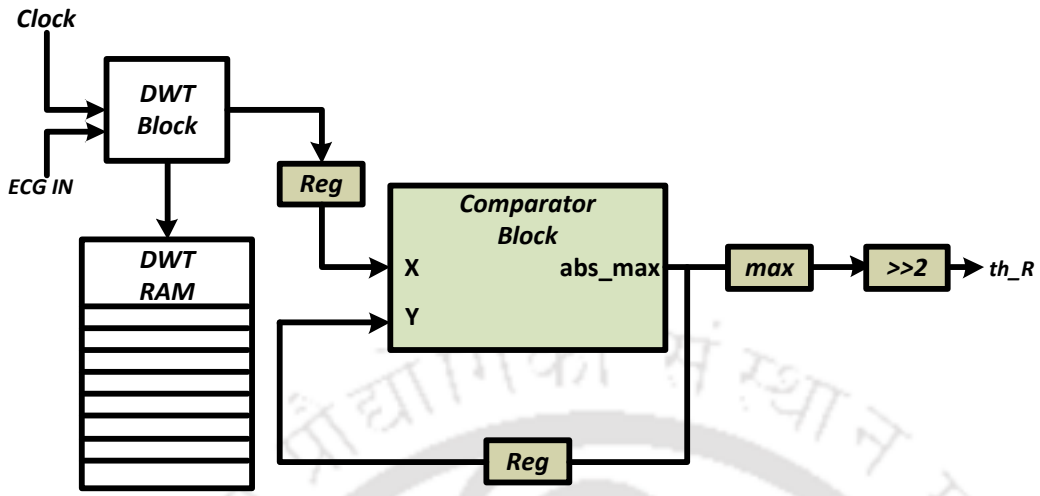


Figure 2.6: Comparator Block

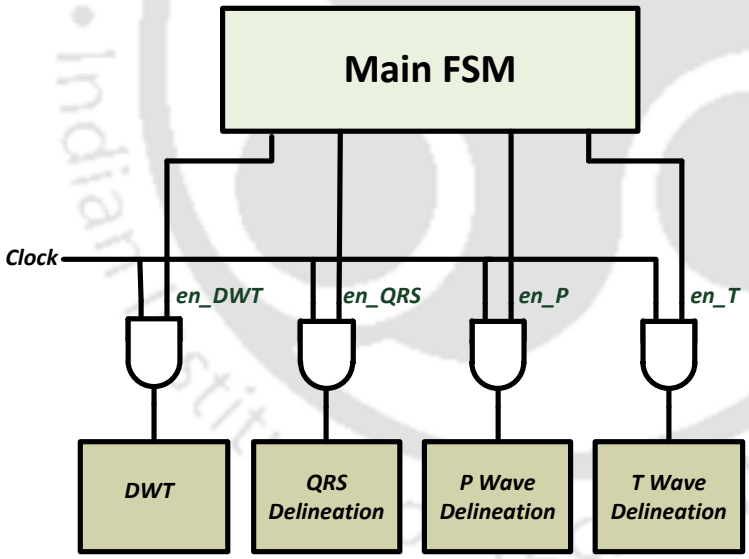


Figure 2.7: Clock Gating of Different Blocks

2.5 Experimental Results of Proposed Design

In this section, experimentation of the proposed methodology and its outcome is discussed in detail.

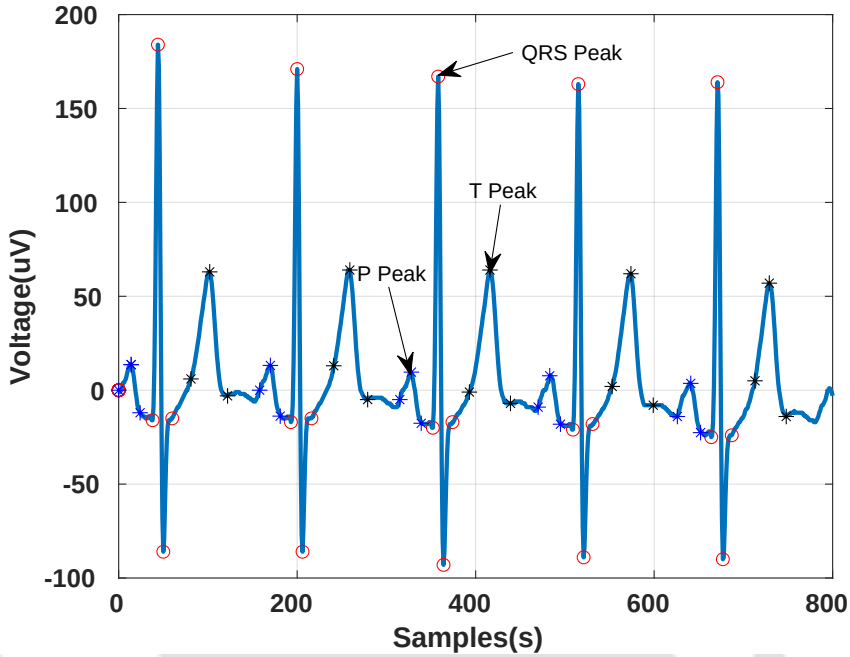


Figure 2.8: ECG Delineation

2.5.1 Validation of the Proposed Methodology

The proposed algorithm is first implemented in MATLAB to validate its correctness. It is tested on excerpts taken from the AHA database and on QT database of the Physionet [41]. It is observed that many of the traditional ECG processing algorithms, such as Pan-Tompkins [17] are implemented in software on a personal computer, which leads to use of floating-point representation of ECG data. As we know, FPGA based implementation of floating-point unit is complex, integer representation is preferred for simplicity as well as to reduce power consumption. Considering the associated constraints, the proposed algorithm is validated, taking integer representation of ECG signal data. Usually, ECG data is recorded in double-precision format. This double precision format is converted into integer format by multiplying it with 10^N, $N = 1, 2, ..$ and is rounded off to its integer value. As explained in [27], we consider ECG data to be multiplied by a factor of 100 and then rounded it off to convert in the integer format. It is to mention that variations in the interbeat intervals are negligible; therefore, ECG is rounded off by considering a factor of 100. Figure 2.8 represents delineation of the boundary and peak points for QT database ($sele0122m$) recording sampled at 250 Hz.

Table 2.3 presents test result for QT database [41] which consists of 105 15min excerpts of 2-lead ECG recordings.

Algorithm	Implementation Platform	Metric	P Wave	QRS Wave	T Wave
Kalyakulina et al. [20]	Software	**Se%**	97.49	98.42	97.2
		PPV%	97.89	98.24	96.55
Sanjeev et al. [22]	ASIC	**Se%**	98.91	100	99.97
		PPV%	91.07	100	97.76
Bote et al. [23]	TI MSP430 series microcontroller	**Se%**	98.23	99.88	98.18
		PPV%	94.38	99.41	96.39
Rincon et al. [42]	Shimmer TM embedded platform	**Se%**	99.88	99.97	99.97
		PPV%	92.04	98.66	98.70
Di Marco et al. [43]	Software	**Se%**	98.15	100	99.72
		PPV%	91.00	-	97.76
Proposed Work	FPGA	**Se%**	97.93	99.46	98.12
		PPV%	94.23	99.06	96.54

Table 2.3: ECG Feature Extraction Results From QT Database

$$Se\% = \frac{TP}{TP + FN}$$
$$PPV\% = \frac{TP}{TP + FP} \tag{2.7}$$

These recordings are sampled at 250Hz. For verification, these excerpts are manually annotated under the guidance of a medical expert. The proposed algorithm works on a single lead channel. The comparison of manually annotated and algorithmically annotated features is performed by selecting a ECG signal. It is found that the algorithmic annotation delineates the same points in (150ms) proximity of the manually annotated signal. The error metric considered is described by the Association for the Advancement of Medical Instrumentation (AAMI) [44]. As shown in Eqution 2.7 *True Positive* (TP) is counted when a feature is correctly detected, then the respective error is calculated between manually and algorithmically annotated features. If no position is detected, then it is considered *False Negative*(FN). Any other point not related to any of the annotated features is regarded as *False Positive*(FP). On this basis, we consider *Sensitivity*($Se\%$) and *Positive Predicted Value*($PPV\%$) metrics for performance evaluation of the proposed algorithm. For validating the algorithm's architecture, different testbenches are created to validate the outcome of the proposed algorithm with the manual annotations. It is seen from Table 2.3 that the proposed architecture gives a better or comparable

Resources	[45]		[46]		[47]		[48]		Proposed Design	
	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised
LUT	40960	29389	343680	119256	239616	131072	69120	24784	53200	6894
Slice Reg	20480	15459	687360	130598	18752	3532	69120	14670	106400	1080
MUX	-	-	-	-	-	-	-	-	26600	126
FF	40960	4845	-	-	-	-	-	-	-	-
IOB/IO	565	30	-	-	152	128	640	58	200	66
BUFG	-	-	-	-	-	-	-	-	32	1
MULT	40	3	-	-	52	20	-	-	-	-
GCLKs	8	3	-	-	-	-	32	6	-	-
BRAM	-	-	2528	786	-	-	148	24	-	-
DSP Blocks	-	-	864	458	-	-	64	7	-	-
Total Resources	103013	49729	1034432	251098	258572	134752	139124	39549	186432	8167
Comparative Resource Utilisation	6.08X		30.74X		16.49X		4.84X		X	

Table 2.4: Comparision of FPGA Results of ECG Feature Extraction Architecture

sensitivity as compared to state-of-the-art methods for delineating an ECG signal verifying the efficacy of the proposed research work.

2.5.2 Hardware Implementation Results

As stated earlier, the proposed method is implemented using MATLAB and Verilog both. Later, execution of the algorithm proposed in this section is validated through FPGA. It has been found that the MATLAB based implementation of the proposed algorithm is in close correlation with its FPGA based implementation. It is observed that the position of R, P and T peaks obtained through MATLAB and Verilog HDL implementation match without any variations. However, a difference of just one sample is noticed in delineation of onset and offset of P and T waves in a few ECG beats. It is to mention that variation in the delineation of P and T waves is still in the error range reported in [44]. In literature, various methods have been reported focusing on the detection of R peaks because many heart diseases can be diagnosed by evaluating variability in the R peaks only. Therefore, R peak module is implemented first and is compared with already existing methods to showcase its effectiveness in terms of efficient hardware utilisation. Comparision details are presented in Tables 2.5 for completeness.

It can be observed from Table 2.5 that the proposed architecture for R peak detection utilizes only 2584 resources out of total available FPGA resources, which is the least among all the previous works

reported. It can also be noticed in Table 2.5 that the design proposed in this section does not include any compute and power intensive elements, such as multiplication units. This makes our proposed design to be more efficient in term of area and power as compared to previously reported designs. It can be observed in Table 2.5 that our proposed design is 28.86% more efficient in terms of resource utilization as compared to the previous best known method reported in [49]. The hardware resource utilisation of complete ECG feature extraction algorithm is presented in Table 2.4. It can be seen that the proposed architecture utilises only 4.38% of the total available resources, which is the least among all other methods reported. It can also be observed that our proposed implementation of complete ECG feature extraction algorithm is 79.35% better than the previous best known method reported in [48]. This is primarily due to designing our proposed architecture using multiplexers, which not only reduces design complexity and makes it simpler but also optimizes power consumption.

Resources	[50]		[51]		[52]		[53]		[49]		[54]		Proposed Design	
	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised	Available	Utilised
LUT	9312	3443	9312	3061	124467	3734	9312	2328	1248012480	14081408	303600	88456	53200	2004
Slice Reg	4656	2901	4656	1809	34240	1712	4656	1489	1248012480	10861086	607200	5728	106400	503
MUX	-	-	-	-	-	-	-	-	-	-	-	-	26600	42
FF	-	-	9312	544	216750	4335	9312	651	-	-	-	-	-	-
LUT-FF Pair	9312	2283	-	-	-	-	-	-	1440	1054	93996	188	-	-
IOB/IO	232	18	67	67	-	-	232	88	172172	8282	700	114	200	34
BUFG	-	-	-	-	-	-	-	-	32	2	32	1	32	1
MULT	20	8	-	-	-	-	-	-	-	-	-	-	-	-
GCLKs	24	1	-	-	-	-	24	1	-	-	-	-	-	-
Total Resources	23556	8654	23347	5481	375457	9781	23536	4557	26604	3632	1005528	94487	186432	2584
Comparative Resource Utilisation	**3.34X**		**2.12X**		**3.78X**		**1.76X**		**1.40X**		**36.56X**		**X**	

Table 2.5: Comparision of FPGA Results of R Peak Extraction

Parameter	[21]	[22]	[55]	[56]	[57]	[58]	Proposed Work
Technology	65nm	130nm	180nm	180nm	180nm	180nm	180nm
Operating Frequency	7.5kHz	4kHz	1MHz	1MHz	NA	0.12kHz	1MHz
Supply Voltage	0.6v	0.9v	1.8v	1.2v	1.0v	1.2v	1.62v
ECG Features	P-QRS-T	P-QRS-T	P-QRS-T	P-QRS-T	QRS	QRS	P-QRS-T
Power	0.642uW	0.384uW	9.47uW	32uW	0.410uW	5.97uW	0.88uW
Energy	85.6pJ	96pJ	9.47pJ	32pJ	NA	49.75nJ	0.88pJ

Table 2.6: Comparision of Synthesis Results of ECG Feature Extraction Architecture

In Table 2.5 and Table 2.4, "-" indicates information not available for comparison in the reported works. The entire design is synthesised using 180nm SCL PDK using Synopsys DC and IC Compiler tools, and its power consumption is compared with some of the most relevant previous known designs, as per our knowledge, and is shown in Table 2.6. It is observed that the power consumption of the architecture proposed in this section is 0.88μW while operating at 1.62V at a maximum operating frequency of 1MHz. It can be observed that the architecture proposed in [21] consumes 0.642uW of power when operated at 7.5kHz, which is less than the power consumption of the proposed design. But, it is reported in [21] that the power consumption of their design is 1.33uW, if it operates at 100kHz. Moreover, our proposed work consumes 0.66X less power while operating at 10X (1MHz) frequency with respect to the design proposed in [21] with the same delineation accuracy. This indicates that the proposed architecture is efficient in terms of power consumption among all the reported state-of-the-art architectures for ECG feature extraction.

Further, for a fair comparison, the proposed research work is compared with different architectures in terms of energy dissipation. It is concluded that the energy utilization of the proposed design is 0.88pJ, which is the least among all other methods as depicted in Table 2.6. The primary challenge of IoT healtcare wearable devices is to operate a device with minimal power resources providing longer battery life [35], [59] and should be compact in nature utilising minimal resources. Therefore, based on the experimental analysis, it is observed that our design utilises minimal resources and can operate at a very low power with high accuracy making it suitable for the applications, which mandatorily require low power and area optimality. It can also be inferred that at smaller technology nodes, the area and power consumption of the proposed design would be lesser, thus, making it a promising

candidate for wearable healthcare applications.

2.6 Proposed Architecture for Derivative Based ECG Feature Extraction

This section presents a derivative based algorithm to extract P-QRS-T features of an ECG signal. Later the VLSI implementation of the proposed algorithm is presented.

2.6.1 ECG Feature Extraction Algorithm

This section describes the second proposed ECG feature extraction algorithm to detect P wave, QRS complex and T wave and mark their fiducial points (onset, offset and peak). The algorithm starts with the extraction of the QRS peak, which is the most prominent feature of an ECG wave and is used to detect a person's heart rate. Any irregularity in the occurrence of the R peak signifies cardiac arrhythmia. Later, QRS boundaries and P, T waves are delineated if any abnormality is detected in $R-R$ interval or as per the medical requirements.

The raw ECG has motion artefacts, isoelectric line wandering and high-frequency noises with the signal components. The signal is first processed to remove these noises to ensure accuracy in extracting fiducial points. The digitized ECG signal is filtered using a band-pass filter of bandwidth $0.5-40Hz$ [22]. Figure 2.9 depicts the flow chart of the proposed algorithm.

Peaks Detection Modules:

The function of the peak detection module is to mark the R peaks and P and T peaks if required. A single and double derivative of the filtered ECG signal is taken as input to this module for performing peak detections. The shape of the QRS complex depends on the selected lead. In some leads such as $V6$, R peak is most significant whereas, in the lead $V1$, R peak is small, followed by a prominent negative S peak. The proposed methodology considers the former as positive QRS complex and later as negative [23]. As shown in the flow chart 2.9, initially, a window of $3.5s$ is considered for processing the ECG signal. Later, the filtered ECG signal's single and double derivative is computed to extract the QRS peak. Once the double derivative is obtained, absolute maximum (abs_max) value of the double derivative signal (ECG'') is calculated for the considered $3.5s$ window. The threshold is then

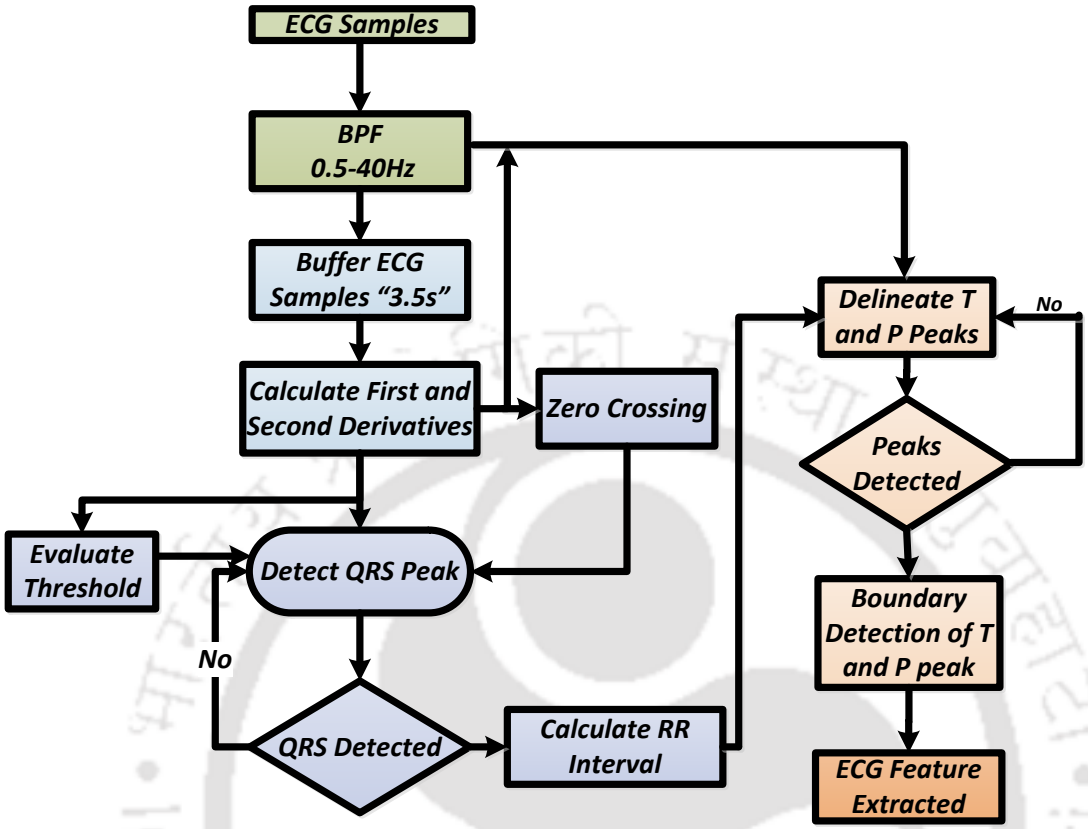


Figure 2.9: Flow Chart of Proposed Algorithm

computed to detect the R peaks according to equation 2.8.

$$th_R = abs\_max/4 \tag{2.8}$$

R peak is marked if a zero crossing is encountered and a magnitude of ECG'' is greater than the described threshold. If the value of ECG'' is positive, then a positive QRS peak is considered; else, it is considered as negative QRS complex. The threshold for finding R peaks is updated with every $3.5s$ window making the detection adaptive to the uneven heartbeat. $R-R$ intervals (RR) are computed along with detecting the R peaks. The duration of the QRS complex is $120ms$. Therefore, if found R peak is positive, then Q and S peak is marked as the minimum value in the window of $60ms$ before and after the R peak, respectively. When the detected QRS complex is negative, the R peak is marked as the maximum value in a $60ms$ window before the obtained negative peak. The negative peak will then be marked as S peak. Once the R peaks are estimated, the RR intervals are calculated as the difference between consecutive R peaks.

Further, P and T peaks are calculated if required after delineating the respective R peaks. The P wave is the first feature of the ECG wave whose average duration is shorter than $120ms$. However, in patients with atrial fibrillation, its duration can be up to $170ms$ [10]. Therefore, a window less than $170ms$ before the R peak is considered for delineating the P peak. Maximum and minimum of the second derivative of ECG signal are found in a chosen window. If the minimum magnitude is greater than the maximum in a taken window, then a P wave is considered positive; otherwise, it is negative. The value of the ECG sample corresponding to the minimum of ECG'' is marked as P peak for positive P wave and vice versa for the negative P wave. If the absolute value of the minimum or the maximum is less than 1% of abs_max as described above, then P peak is not annotated. The final feature of an ECG wave is T wave. In a normal individual, duration of T wave varies from $125 - 200ms$ [10]. However, it varies when a person is suffering from cardiac issues. Therefore, a similar approach as P wave is used to extract T peak after chosing a window of $200ms$ after the delineated R peaks. The heartbeat (HB) can also be found using equation 2.9. f_S is the sampling rate of the ECG signal.

$$HB = 60 \times (\frac{f_s}{RR\ interval}) \tag{2.9}$$

Boundary Detection Module:

Once the R peak is detected, the onset and offset of ECG peaks can be described. The boundaries of $P - QRS - T$ waves are detected using the zero slope criteria. A suitable window is considered following the standard duration of the respective waves before and after the previously found peak. Once the window is defined, sample values with zero slopes before a peak are marked as onset and after as an offset. However, there may be cases when no samples with zero slope are obtained. To delineate the boundaries in such conditions sample value smaller than 5% of the respective peak value is marked as the boundary of a considered peak in taken windows. Both the procedures are considered for reliable feature extraction of ECG waves.

2.6.2 Architecture Details of Proposed Algorithm

As shown in figure 2.9, the proposed algorithm has two main blocks, peak detection and boundary detection. Among them, the R peak block is mandatory, whereas other sub-blocks are optional and operational per medical requirements.

The proposed algorithm is implemented in hardware as a finite state machine in Verilog HDL using Xilinx Inc's Vivado 2019 development environment. Figure 2.10 represents the state machine that controls the operation of different blocks.

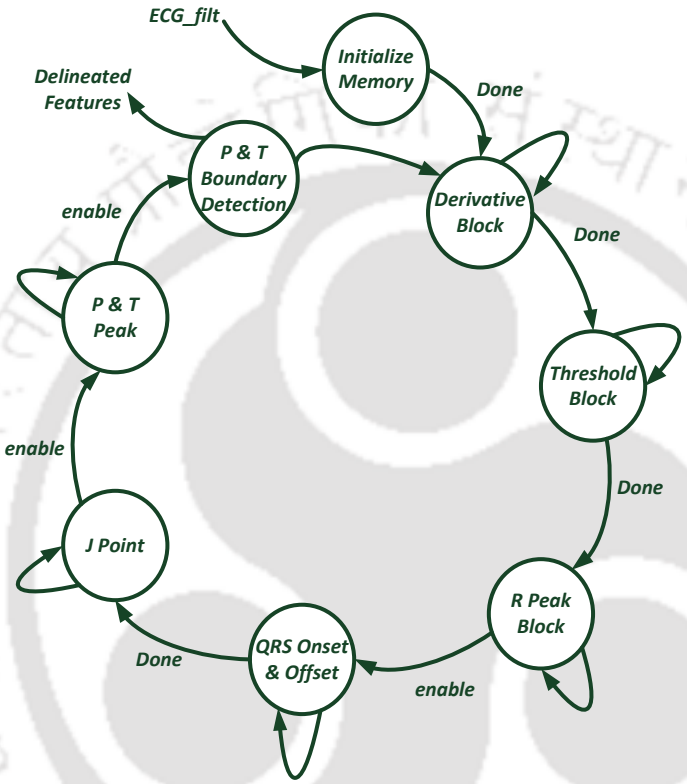


Figure 2.10: Finite State Machine of Proposed Architecture

Initially, ECG signals sampled at $250Hz$ and filtered at a specified frequency of $0.5-40Hz$ are fed serially to the architecture as explained above. Each ECG sample is considered to be of 11 bits. As it is known that floating-point operations are complex to perform in hardware and utilize additional resources, integer representations of ECG signals are considered. ECG samples are available in the double-precision format; they are first converted into integer format by multiplying them by a factor of 100. It is then rounded to its integer value, avoiding the floating-point arithmetic. It is observed that multiplying each sample by a factor of 100 give negligible variation in the interbeat interval of the ECG signal. A memory is first initialized that stores the filtered ECG samples, which will be utilized later to mark the position of the ECG features. The filtered ECG samples are then fed to the differentiator block serially that produces the single derivative (ECG') and double derivative (ECG'') output of

the ECG signal. The double derivative output is fed as input to the threshold block. As shown in equation 2.8, the divide by 4 operation in threshold for R peak delineation (th_R) can be calculated as a shift by 2 operation ($>> 2$). Further, the R peak position can be marked using a simple comparator circuit. To delineate remaining features such as P and T wave, all the window sizes and threshold are considered to avoid any floating-point arithmetic, thus optimizing the hardware resources and reducing computational complexity making the overall architecture simple. As it is evident from the state diagram that the functionality of architecture is divided into several blocks, each controlled by the clock and enable signal. The R peak is the first feature delineated once the signal's double derivative is obtained. Once the R peak is obtained, the modules to examine other features are clock gated to optimize power. Therefore, the proposed architecture is suitable for deploying in resource-constrained wearable healthcare ECG analysis applications. Their key requirement is real-time analysis of ECG signals for any anomalies in the power and resource-constrained environment.

2.7 Experimental Results of Proposed Design

This section elaborates the experimentations performed and output obtained from the proposed methodology.

2.7.1 Validation of the Proposed Methodology

The proposed algorithm is first implemented in Python to validate its functionality. It is then verified on ECG experts taken from $MIT-BIH$ and QT database of Physionet [41]. The $MIT-BIH$ database consists of 48 half-hour excerpts of two-channel ambulatory ECG recordings sampled at $360Hz$. The experts are downsampled at $250Hz$ for use in the proposed algorithm. The QT dataset contains 105 $15min$ excerpts of 2-lead ECG recordings sampled at $250Hz$. The experts are first annotated manually with guidance of an expert for verification. For every ECG signal, a comparison of manual and algorithmic annotated features is performed. It is observed that the proposed algorithm delineates in proximity of $150ms$ of the manually annotated signal. This error metric of ECG feature extraction is described by the Association for the Advancement of Medical Instrumentation (AAMI) [44] for verification of the ECG feature extarction algorithm. Table 2.7 presents the test result of the proposed algorithm on ECG excerpts from QT dataset [41].

It is observed from table 2.7 that the proposed algorithm is much less complex than the algorithms proposed in [22], [23], [43], [20] and [42]. The proposed algorithm displays an equivalent or better

Algorithm	Implementation Platform	Metric	P Wave	QRS Wave	T Wave
Bote et al. [23]	TI MSP430 series microcontroller	**Se%**	98.23	99.88	98.18
		PPV%	94.38	99.41	96.39
Rincon et al. [42]	Shimmer TM embedded platform	**Se%**	99.88	99.97	99.97
		PPV%	92.04	98.66	98.70
Di Marco et al. [43]	Software	**Se%**	98.15	100	99.72
		PPV%	91.00	-	97.76
Sanjeev et al. [22]	ASIC	**Se%**	98.91	100	99.97
		PPV%	91.07	100	97.76
Kalyakulina et al. [20]	Software	**Se%**	97.49	98.42	97.2
		PPV%	97.89	98.24	96.55
Proposed Work	FPGA	**Se%**	98.84	99.86	99.12
		PPV%	96.33	99.56	97.86

Table 2.7: Comparison of ECG Feature Extraction Results on QT Database

sensitivity in extracting the features of ECG compared to the literature, verifying the algorithm's efficacy. Moreover, the computational simplicity of the proposed algorithm makes it a suitable candidate for low power real-time ECG analysis applications due to the area and power-optimized hardware implementation. Robust QRS extraction is achieved using the double derivative of ECG signal with added adaptive threshold and detection windows. The sensitivity of 99.86% and positive predictivity of 99.56% is attained to detect the QRS complex that comprises the peak and boundary detection. The QRS peak marks as a reference point to delineate other features such as P and T wave and intervals such as $R-R$ interval. Moreover, proposed algorithm demonstrates a comparable sensitivity for P and T waves extraction as evident from table 2.7.

2.7.2 Hardware Implementation Results

The proposed work is implemented in hardware using Verilog, and the modules are simulated for functionality in Xilinx 2019 Vivado development environment. Later, the modules are verified using the Xilinx Virtex-7 FPGA board. For validating the proposed architecture, multiple test-bench of different ECG signal experts [41] is created to verify the outcome of the proposed algorithm with the Python implementation and manual annotations. It is observed that the outcome of the extracted features from the architecture is in close correlation with the software annotations and the manual readings.

QRS complex is the most critical feature of an ECG wave, and hence, various methods have been

Architecture	Available LUT	LUT Utilized	% Utilization	Available FF	FF Utilized	% Utilization	Total % Utilization
[50]	9312	3443	24%	4656	2901	62%	45.4%
[51]	9312	3061	32%	4656	1809	38%	34.86%
[52]	124467	3734	3%	34240	1712	5%	3.43%
[53]	9312	2328	20%	4656	1489	32%	27.3%
[49]	12480	1408	11%	12480	1086	8%	9.99%
[54]	303600	88456	29%	607200	5728	0.94%	10.34%
Proposed	303600	422	0.13%	607200	293	0.05%	0.08%

Table 2.8: Comparison of FPGA Implementation of R peak Detection

reported in the literature for delineation of the R peak only. Therefore, for completeness, the R peak module of the proposed architecture is implemented first, and its hardware resources are evaluated and compared with some previous works in table 2.8. It is observed that the proposed architecture utilized only 0.1% of available resources on Virtex-7 FPGA to find the QRS peaks, which is the least among the literature.

Later, architecture for complete ECG delineation is implemented on FPGA and hardware resource utilization is realized. Table 2.9 presents the resources utilized by the proposed architecture for complete ECG feature extraction. It is seen that architecture uses only 0.42% of total available resources available on Virtex-7 FPGA. On comparing this work with the literature in table 2.10, it is observed that the proposed architecture utilized only 2409 lookup tables and 1230 flip-flops. The resource utilization is approximately 11 times better than the most efficient method reported in [48], making the proposed architecture efficient and well-optimized as it employs the least available resources.

Resource	Available	Utilization	Percentage Utilization(%)
LUT	303600	2409	0.79
FF	607200	1230	0.20
DSP	2800	11	0.39
IO's	700	233	38.83
BRAM	1030	1.50	0.15

Table 2.9: FPGA Implementation of Complete ECG Feature Extraction Architecture (Xilinx Virtex-7 FPGA)

Architecture	LUT			Flip Flop		
	Available	Utilized	Percentage Utilization(%)	Available	Utilized	Percentage Utilization(%)
[45]	40960	29389	71	20480	15459	75
[46]	343680	119256	34.7	687360	130598	19
[47]	239616	131072	55	18752	3532	19
[48]	69120	24784	35	69120	14670	21
Proposed	303600	2409	0.79	607200	1230	0.20

Table 2.10: Comparison of FPGA Implementation of Complete ECG Feature Extraction

Further, the entire design is synthesized in $180nm$ CMOS technology to reckon the power consumption of the architecture. The proposed design is compared with other state-of-the-art methods, which are also designed using $180nm$ technology node for fair comparison in table 2.11.

Parameter	[55]	[56]	[57]	[58]	Proposed Work
Technology	180nm	180nm	180nm	180nm	180nm
Operating Frequency	1 MHz	1 MHz	NA	0.12 KHz	1 MHz
Supply Voltage	1.8V	1.2V	1.0V	1.2V	1.98V
ECG Features	P-QRS-T	P-QRS-T	QRS	QRS	P-QRS-T
Power	$9.47\mu W$	$32\mu W$	$0.410\mu W$	$5.97\mu W$	$7.38\mu W$
Energy	9.47pJ	32pJ	NA	49.75nJ	7.38pJ

Table 2.11: Power Comparison of Synthesis Results

The proposed algorithm consumes a mere $7.38\mu W$ of power when operated at $1MHz$ at an operating voltage of $1.98V$, which is less as compared to the literature in [55], [56]. However, the architectures in [57], and [58] utilized $0.410\mu W$ and $5.97\mu W$ of power, respectively, which is less as compared to the proposed design. However, it should be noted that architecture of [57] and [58] reports only QRS detection and thus have fewer architectural requirements than our proposed architecture which is designed to perform complete $P-QRS-T$ feature extraction. Also, they are operated at lower frequency and voltage, leading to less power consumption. Moreover, the proposed architecture is compared in terms of energy with the mentioned state-of-the-art works for a better comparison. It can be concluded from table 2.11 that the proposed design has the least energy requirements ($7.38pJ$) compared to other architectures. Therefore, based on the hardware implementation results, it is summarised that the proposed design consumes low power and hardware resources, making it suitable for

low power wearable healthcare applications.

2.8 Summary and Scope of Present Work

This chapter presents two low complexity algorithms and their area and power efficient VLSI architectures for ECG feature ($P-QRS-T$) extraction. An average sensitivity of 99% is achieved on the QT dataset by the proposed algorithms, comparable or better than the literature. The architectures have low power and resource requirements and could be considered viable for employment as ECG feature extraction block in low power and resource-constrained application of wearable devices. The extracted ECG features can further be used to analyse various ECG intervals for detecting cardiovascular diseases.

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी
Indian Institute of Technology Guwahati

3

Detection and Prediction of Cardiac Arrhythmia

Contents

Objective

Spontaneous detection and prediction of different types of cardiac arrhythmia is crucial to avoid fatalities caused due to the harmful nature of arrhythmia beats. In the past, several hardware designs, reported in the literature, have been proposed to detect and predict cardiac arrhythmias using various features. These features are derived from electrocardiogram signal (ECG) and processed using machine learning classifiers. However, these designs are either more complex or have less detection and prediction accuracy. Therefore, this chapter proposes three different VLSI architetcures utilizing deep neural network to detect different arrhythmia beats and predict ventricular arrhythmia beats. The proposed design has low power requirement with high acuracy making them suitable for wearable healthcare devices.

3.1 Introduction

Cardiovascular Diseases (CVD) are among the primary contributors to the mortality rate worldwide, especially in developing and under developed countries that lack proper health infrastructure. According to the WHO, approximately 17.9 million people lose their lives yearly due to CVDs. Cardiac arrhythmia (CA) is a disorder related to cardiac rhythm of the muscles of the heart, which indicates susceptibility to severe cardiac disease, stroke or cardiac tissue death. It accounts for 80% deaths due to CVDs worldwide [12].

Arrhythmia is a form of heart condition that is characterized by the rate or the rhythm of the heartbeat.The heart has an electrical conduction system which makes your heart pump blood around your body. Arrhythmias are caused by an abnormality in that electrical conduction system. The heartbeat can be faster than normal, or too slow, or have an irregular pattern. Tachycardia occurs when the heartbeat is too fast above 100 beats/min, and bradycardia is the heart disease that is associated with very slow heartbeats below 60 beats/min. The most commonly known cardiovascular diseases include types of arrhythmias such as ventricular fibrillation (VF), premature ventricular contraction (PVC), atrial fibrillation (AF), and premature atrial contraction (PAC), to name just a few.

3.1.1 Types of Arrhythmias

(i) **Sinus Node Arrhythmia:** Sinus node arrhythmia generates from the sino-atrial (SA) node of the heart. The characteristic feature of these types of arrhythmia is the P-wave morphology. These arrhythmias are the following types:

- Sinus Arrhythmia
- Sinus Bradycardia
- Sinus Arrest

(ii) **Premature Atrial Contractions:** Premature atrial contractions (PAC) arrhythmia is an abnormal P-wave morphology followed by a normal QRS complex and a T-wave. This arrhythmia arises when an ectopic pacemaker fires before the SA node. It may occur as a couplet where two PACs are generated successively. The rhythm is called atrial tachycardia when three or more successive PACs occur.

(iii) **Junctional Arrhythmia:** Junctional arrhythmias originate within the atrioventricular (AV) junction in the form of the impulse comprising the A-V node and its bundle. This arrhythmia is caused due to P-wave abnormality. The polarity of P wave would be opposite to that of the normal sinus P-wave.

(iv) **Premature Ventricular Contractions (PVC):** Generally, premature Ventricular Contractions (PVCs) arrhythmia generates from the ventricles. It usually does not depolarize the SA node or the atria. Hence, the morphology of P-waves maintains its underlying rhythm, which occurs at the expected time. PVCs may occur anywhere in the heartbeat cycle. PVCs are described as isolated if they occur singularly.

(v) **Bundle Branch blocks:** There are two bundle branch blocks: The left bundle branch block beat (LBBB) and the right bundle branch block beat (RBBB). When the bundle branch block becomes injured, it may stop conducting electrical impulses appropriately. Due to delayed activation of the left ventricle, the LBBB is generated, which means that the left ventricle contracts later than the right ventricle. A right bundle branch block (RBBB) occurs in the heart's electrical conduction system. An extra deflection in the QRS complex indicates the slow and rapid depolarization of the left ventricle and the right ventricle.

(vi) **Supra-ventricular Arrhythmia:** Supra-ventricular and atrial arrhythmias generate in the areas above the heart's lower chambers or at the heart's upper chamber. It is not a life-

threatening disease like ventricular arrhythmias and may not require any treatment. Atrial arrhythmias can happen from the consumption of tobacco, alcohol, and caffeine and also from cough and cold medicines. Rheumatic heart disease or an overactive thyroid gland (hyperthyroidism) indicates that the patient suffers from supra-ventricular arrhythmia. The shortness of breath, heart palpitations, chest tightness, and a fast pulse indicate supra-ventricular arrhythmia.

(vii) **Atrial Fibrillation:** Atrial fibrillation occurs when single muscle fibers in the heart twitch or contract. It is a fast, irregular rhythm. It can be life-threatening, like stroke, which is generally seen in older people. The blood is pooled in the heart's upper chambers when the patient suffers from atrial fibrillation. The pooled blood leads to the formation of clumps of blood. The clumps of blood are called blood clots. Many patients with atrial fibrillation need immediate anti-platelet therapy, which can prevent blood clots and avoids stroke.

(viii) **Ventricular Escape:** The ventricular escape beat lies between 20 to 40 bpm. These beats have a similar morphology as the LBBB or RBBB. The QRS complexes of ventricular escape beat are 120 ms. When the rate of supra-ventricular impulses arriving at the AV node or ventricle is less than the intrinsic rate of the ectopic pacemaker at that time, junctional and ventricular escape rhythms occur. Severe sinus bradycardia, sinus arrest, sino-atrial exit block, high-grade second-degree AV block, third-degree AV block, and hyperkalemia indicate that the patient suffers from junctional and ventricular escape arrhythmias.

(ix) **Junctional Escape:** The duration of the QRS complex of junction escape arrhythmia is generally narrow, i.e., less than 120 ms. There is no relation between the QRS complexes and any preceding atrial activity (e.g., P-waves, flutter waves, fibrillatory waves). The rhythm rate of the arrhythmia is below the normal heart rhythm i.e., 40-60 bpm.

3.1.2 Cardiac Arrhythmia ECG Database

The MIT/BIH arrhythmia database [60] is used in the study for performance evaluation. The database contains 48 records, each containing two-channel ECG signals for 30 min duration selected from 24-hr recordings of 47 individuals. There are 116,137 numbers of QRS complexes in the database.The subjects were taken from, 25 men aged 32 to 89 years, and 22 women aged 23 to 89 years and the records 201 and 202 came from the same male subject. Each recording includes two leads; the modified limb lead II and one of the modified leads V1, V2, V4 or V5. Continuous ECG

signals are band pass-filtered at 0.1100 Hz and then digitized at 360 Hz. Twenty-three of the recordings (numbered in the range of 100124) are intended to serve as a representative sample of routine clinical recordings and 25 recordings (numbered in the range of 200234) contain complex ventricular, junctional, and supraventricular arrhythmias. The database contains annotation for both timing information and beat class information verified by independent experts.

3.1.2.1 AAMI Standards

MIT-BIH heartbeat types are combined according to Association for the Advancement of Medical Instrumentation (AAMI) recommendation [61]. AAMI standard emphasize the problem of classifying ventricular ectopic beats (VEBs) from the non- ventricular ectopic beats. AAMI also recommends that each ECG beat can be classified into the following five heartbeat types:

(i) N (Normal beat)

(ii) S (supraventricular ectopic beats (SVEBs))

(iii) V (ventricular ectopic beats (VEBs))

(iv) F(fusion beats)

(v) Q (unclassifiable beats)

Each class includes heartbeats of one or more types. Class N contains normal and bundle branch block beat types and escape beat, class S contains supraventricular ectopic beats (SVEBs), class V contain Premature ventricular contraction beats and ventricular escape beat, class F contains beats that result from fusing normal and VEBs, and class Q contains unknown beats including paced beats.

3.2 Prior Works

Several state-of-the-art methods have been reported in the literature to classify arrhythmia using ECG signals. Dutta et al. [62] perform feature extraction using cross-correlation method and, classification utilizing LS-SVM with an accuracy of 95.82%. Haseena et al. [63] employ fuzzy c-mean clustering for feature extraction and probabilistic neural network (PNN) to classify seven different arrhythmias along with regular beats. Tae Joon Jun et al. [3] use a two-dimensional CNN model to classify seven arrhythmia beats from normal beats with an accuracy of 99.05% employing images of ECG beats. Acharya et al. [64] utilize a one-dimensional CNN model to classify MIT-BIH arrhythmia database into five arrhythmia classes with 89.07% accuracy. The methods reported in the literature are implemented only on software platforms and employ complex algorithms, which are not suitable

to be implemented on the hardware.

However, a few state-of-the-art techniques are designed to classify normal and abnormal arrhythmic beats on ASIC and FPGA platforms. Hsu et al. [65] propose a two-stage processing engine to address above mentioned issues. In this first stage, discrete wavelet transform (DWT) is used to extract features like QRS complex, P, T waves. Later, a multivariate autoregressive (MAR) estimator is applied to analyze these characteristics features of ECG. In the second stage, the classification is performed using SVM and a maximum likelihood classifier depending upon the application. Chen et al. [66] introduce a new feature, QRS area ratio for two class classification of CA. Their design comprises of hybrid classifier consisting of a weak linear classifier (WLC) and SVM. Nearly 40% beats can be classified by WLC; whereas, the beats, which are not classifed, are classified using a SVM model. Another ECG classification method is reported using a sparse neural associative memory (SNAM) by Chollet et al. [67, 68], which detect three abnormal heartbeats. It is observed that the reported methods only aim at classifying the abnormal beats from normal beats on the hardware platforms. However, in [69] Sohail et al. propose a multilevel linear support vector machine (ML-SVM) to classify eight types of arrhythmia. The authors utilize eight copies of LSVM classifier with the classification accuracy of 98.5% and 0.91μW of power at 1kHz for one support vector classifier, and 5.36μW of power for all the classifiers. However, the authors have classified arrhythmias, whose occurrence proves to be fatal and it might not give an individual enough time to get medical attention.

A CA classifier proposed in [70] employs SVM to segregate CA beats from the normal heartbeats. However, both the above mentioned works use handcrafted features for the classification, which may affect capability of the classifier to deal with diverse ECGs. Also, classification methods are not as per American National Standard (ANSI/AAMI EC57:1998) prepared by the Association for the Advancement of Medical Instrumentation (AAMI). Authors in [71], [72], [73], [74] and [75] perform CA beat classification as per AAMI standards using different machine/deep learning models. Method reported in [74] has better classification accuracy than the methodology reported in [71], [72], [73]. Despite the appreciable performance, the model utilized in [74] is not suitable for hardware implementation. It employs a complex deep neural network (DNN) with a large number of nodes in the hidden layers, which lead to higher resource utilization making it less suitable to be utilized in wearable device applications. Zhao et al. in [75] uses DNN for CA classification and reports 13.34μW power consumption employing a patient-specific approach for CA classification. Although patient-specific classifiers illustrate bet-

ter performance than the patient-independent classifiers, but they cannot handle ECG diversity and also require expert intervention in the diagnosis [74] making them less suitable for wearable device applications.

The methods reported above focusses on the detection of different types of arrhythmic beats. However, the methods can only detect occurrence of arrhythmia but do not address its prediction. As it is known, ventricular arrhythmia and ventricular fibrillation are fatal arrhythmia beats that may lead to mortality within a few minutes of their occurrence. Therefore, VT/VF beats must be predicted so as to avoid casualties. However, very little literature is available that can predict the occurrence of arrhythmic beats before it happens. Authors in [76] proposed a low power ASIC utilizing Naive Bayes. However, the accuracy for predicting arrhythmia is just 86%. Authors in [77] proposed an algorithm that can predict the VT/VF 15 min prior to its occurrence using decision trees. The two aforementioned methods have exhibited advantages in discriminating between normal and abnormal conditions. However, [76] has low accuracy in predicting VT/VF, and the algorithm of [77] is too complex for hardware implementation. Further, authors in [78] claim to predict the occurrence of arrhythmia. Nevertheless, they have utilized the data from the time of occurrence of the arrhythmia rather than taking ECG beats before occurence of VT in order to make a prediction.

3.3 Detection of Eight Types of Cardiac Arhythmia Beats

Electrocardiogram (ECG) is a physiological signal used worldwide by medical professionals to analyze cardiac health and the underlying cardiac ailments. Although a single arrhythmic heartbeat might not have any severe implications on the life, but continuous abnormal arrhythmia beats can result in the fatal circumstances. As shown in Figure 3.1, there are broadly seven types of arrhythmic heartbeats, viz. premature ventricular contractions (PVC), paced beats (PAB), right bundle branch block (RBB), left bundle branch block (LBB), atrial premature complex (APC), ventricular flutter wave (VFW) and ventricular escape beats (VEB), which can cause cardiac criticalities, if they persist for a long time.

For instance, prolonged PVC beats can manifest into a ventricular fibrillation (VF) or ventricular tachycardia (VT) event, which immediately leads to the heart failure. Similarly, VF can lead to stroke-causing mortality. Therefore, in this section we propose a method to classify critical arrhythmia beats using a low power Deep Neural Network based ECG classifier along with the beats, which are not very

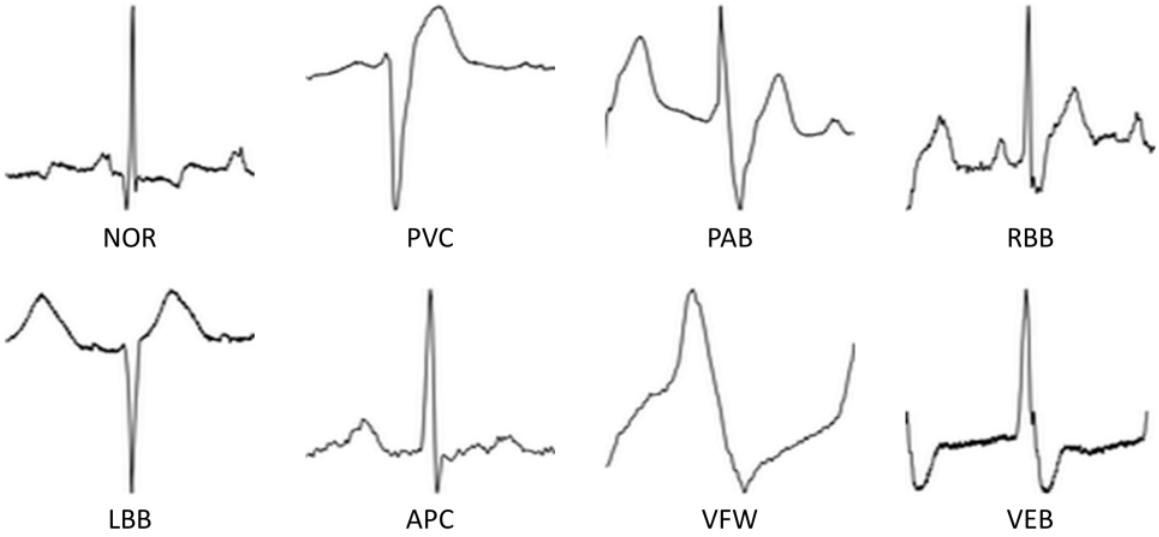


Figure 3.1: Different Arrhythmia Beats of ECG [3]

critical in the initial stages, but can lead to fatal cardiac issues, if they persist for a long time. It is to mention that classifying the non-critical beats in the initial stages of their occurrence would help to alarm an individual having underlying cardiac issues. The low power design of complete classifier architecture makes it suitable for the wearable device applications.

3.3.1 Proposed Arrhythmia Classifier Architecture

For a typical ECG wearable device, an ECG signal is obtained from ECG sensors, and the analog data recorded is converted into the digital format using an ADC. The digitized ECG beats are then processed to classify arrhythmia beats. The contribution of this section is primarily the design and ASIC implementation of CA classification processor block. The classification architecture is based on Deep Neural Network (DNN) as described in Figure 3.2.

In the previously reported works, features are extracted manually from an ECG using feature extraction algorithms and then machine learning methods are applied for the classification of arrhythmia. However, accuracy of the classification of such methods depends on the number of feature vectors chosen and the precision of feature extraction algorithm employed. This motivates us to propose an on-chip DNN-based CA classification architecture having ultra-low power consumption, so that it can be utilized in wearable devices. The proposed design works directly on the filtered digitized ECG beats, therefore, feature extraction is not employed in our design. Using ECG beats for the training

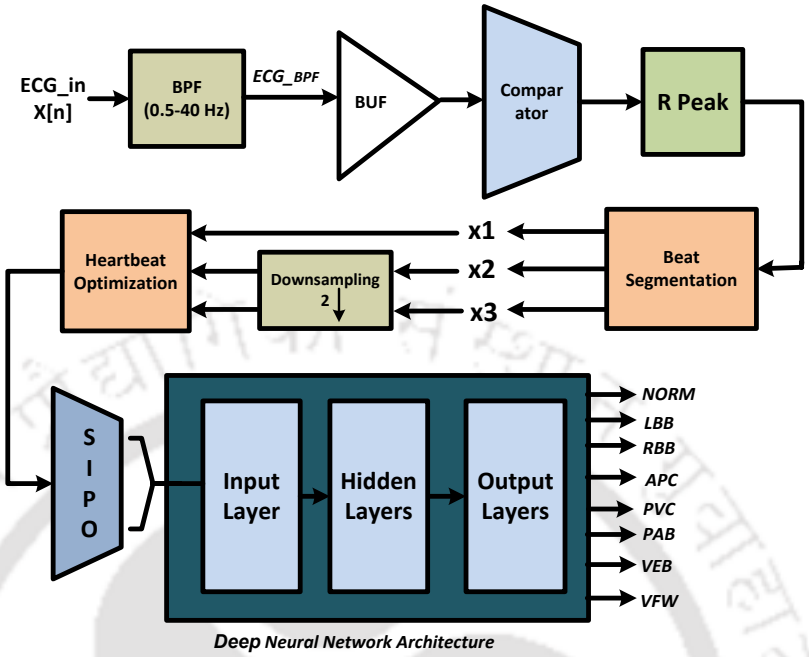


Figure 3.2: Proposed Arrhythmia DNN Classifier

makes our proposed classifier robust with respect to the variation in the morphology of a beat. Therefore, a complete arrhythmia classification system is proposed in this section, which receives ECG data and outputs beat-by-beat eight class classification.

As shown in Figure 3.2, the ECG is filtered by a bandpass filter of the frequency band $0.5 - 40$ Hz to remove the artefacts [79]. Usually, a $0.05 - 40$Hz bandpass filter along with a high pass filter having 0.5Hz cutoff frequency is utilized in the cardiac monitors by medical professionals to have an accurate feature estimation. However, it is observed that a $0.5-40$Hz bandpass filter does not introduce significant error in the classification of beats, when it is analysed using Physionet database [41] without affecting accuracy of the DNN model. The filtered data is fed serially to an ECG segmentation block. It is found that a heartbeat is represented using a window of 400 samples, (W_R), when an ECG is sampled at 360Hz. Further, it is observed that almost 99% of the heartbeats in a dataset can be illustrated using above mentioned window of 400 samples. Therefore, in the heartbeat segmentation process, a window of 400 samples, i.e. size of the beat, is considered. It is to mention that in this section heartbeat and beat are used interchangeably. Later, a sample with the maximum value in (W_R) is marked as *R peak* of the beat.

As the utilization of hardware resources depends on the size of layers in a DNN, size of a beat is further reduced to 250 samples by dividing a single beat of 400 samples into three parts viz.

$x1$, $x2$, $x3$. For $x1$, 100 samples is considered around delineated R *peak*, whereas, $x2$ and $x3$ are obtained by downsampling 150 samples at both the boundries of $x1$ by 2. Since it is known that QRS complex is crucial for the classification of arrhythmia, the maximum information regarding ECG is found in $x1$. Further, it is observed that downsampling $x2$ and $x3$ does not affect the accuracy of classification, but leads to a considerable reduction in hardware utilization. This optimization process finally results in the fixing of size of the beat to 250 samples. Once the heartbeat segmentation is completed, this fixed size ECG beat is fed to a Deep Neural Network classifier. For performing N-class classification, a DNN is designed with $k-1$ hidden layers and an output layer with N nodes having Softmax activation function [3]. For low power wearable applications, the primary objetive of our hardware implementation is to achieve design simplicity. Hence, ReLU activation function (or ReLU) [80] is employed in the hidden layers. It is evident from Equation 3.1, ReLU can be realized in hardware using a simple multiplexer, so that power consumption can be minimized.

$$a_i = ReLU(y_i) = \begin{cases} y_i & \text{if } y_i \geq 0 \\ 0 & \text{if } y_i < 0 \end{cases} \tag{3.1}$$

If $a_n^{(k)}$ is the activation function of n^{th} neuron of the output softmax layer, then the output can be described using Equation 3.2 as mentioned below.

$$z_n = \frac{exp({a_n}^{(k)})}{\sum_{i=1}^{N} exp({a_i}^{(k)})}, n \in 1...N \tag{3.2}$$

The softmax function computes posterior probabilities of an individual class given an input vector y as shown in Equation 3.3. The final classification output is the arrhythmia beat with the maximum probability.

$$z_n = P(class = n|y) \tag{3.3}$$

The hardware implementation of arrhythmia classifier is described in the next section.

3.3.2 Design Implementation

This section explains about dataset and the implementation details of classification architecture.

3.3.2.1 Data Acquisition and Evaluation Scheme

The ECG excerpts used in this work are taken from MIT-BIH arrhythmia database [60]. This database contains 48 recordings. Each recording is of 30 mins duration and is sampled at 360Hz. It includes 47 subjects comprising of 25 men, who are aged between 32 and 89, and 422 women aged between 23 and 89. The dataset has approximate 110,000 heartbeats having different types of arrhythmia and normal beats. Table 3.1 displays ECG excerpts utilized for this study from MIT-BIH database. Our work incorporates seven beats, namely PVC, PAB, RBB, LBB, APC, VFW, VEB, and normal beats, which are broadly used for the arrhythmia classification studies. We have excluded other arrhythmia beats, such as, the fusion of paced and normal beats, and unclassifiable beats etc. during our study, because these are generally ignored in the study of cardiac arrhythmia [3]. The complete dataset is divided into two sets $S1$ and $S2$ in the ratio of 75:25. In the proposed work, DNN model is tranined using $S1$, while $S2$ is employed to test efficacy of our proposed design.

Beat Type	Records ID
NOR	100,101,103,105,108,112,113,114,115,117,121,122,123, 202,205,219,230,234
PVC	106,116,119,200,201,203,208,210,213,215,221,228,233
PAB	102,104,107,217
RBB	118,124,212,231
LBB	109,111,207,213
APC	209,220,222,223,232
VFW	207
VEB	207

Table 3.1: Description of ECG Data

In most of the recordings of the dataset, there are two lead ECG signals. The upper signal is MLII (modified limb lead II), and the lower signal is modified lead V1. Due to the prominence of QRS complexes, we have considered only upper signal to perform the experiments. Also, there are two schemes reported in the literature for the development and performance evaluation of the classifier, namely "class-oriented" and "patient-oriented". The classifier's performance may be overestimated using a patient-oriented method as it employs training and test data that belongs to a single patient. A model trained using above mentioned approach might fail to classify ECG signals from unseen patients due to variations in the characteristics among different individuals. A class-oriented scheme enables a classifier to get trained on the number of ECG beats from multiple subjects, and can

thus perform more realistically. Therefore, in this work, we have utilized a class-oriented scheme for improving the performance of the proposed design for beat-by-beat classification of ECG signals.

3.3.2.2 Architecture of Classifier

The classifier architecture, as shown in Figure 3.2, is first implemented using Python. For making a power-efficient architecture, DNN model is pre-trained on a software platform to obtain optimized weights for the layers. As it is known that the performance of a trained model may change by varying training and test dataset, the DNN model in our proposed design is validated using K-fold cross-validation for stabilizing its performance. In this work, K is fixed to 10, and an even distribution of each data type is used for the dataset. Later, the weights of every layer are obtained and arrhythmia classifier is then realized on hardware using Vivado HLS tool.

For the hardware implementation, filtered ECG data with the resolution of 16 $Bits$ are taken serially as an input to the classifier using fixed-point arithmetic and is fed to the segmentation block. Initially, 400 samples are stored, and the maximum sample value is attained using a simple comparator block. The sample with maximum value is delineated as R $peak$ and, heartbeat is further segmented and downsampled as described in the previous section. Later, the downsampled beat is fed to the classifier using a serial in parallel out (SIPO) register. It is to mention that the pre-trained weights of DNN layers are stored in a register-based memory. Finally, probabilities of the eight classes are obtained as a result of softmax output layer of the classifier, and a class with maximum likelihood is taken as an output. Furhter, RTL of the proposed design is synthesized using Synopsys Design Compiler with SCL Chandigarh CMOS 180nm PDKs. The placement and routing of our proposed design are then performed using Synopsys IC compiler tool. Finally, all the vital statistics of the proposed design are tabulated for the comparison with other state-of-the-art designs available in the literature, which are depicted in the next section.

3.3.3 Experimental Results of Proposed Design

Among many performance indicators used in medical diagnosis, specificity (Sp) and positive predictivity (PPV) are the most important measures to determine diagnostic efficiency of a test. A high PPV indicates that there is high chance of a disease to get detected correctly, whereas, a high specificity aids in ruling out patients, who do not have a specific conditions. As per the medical guidelines [86], it is recommended to choose a test with high specificity to rule out a disease.

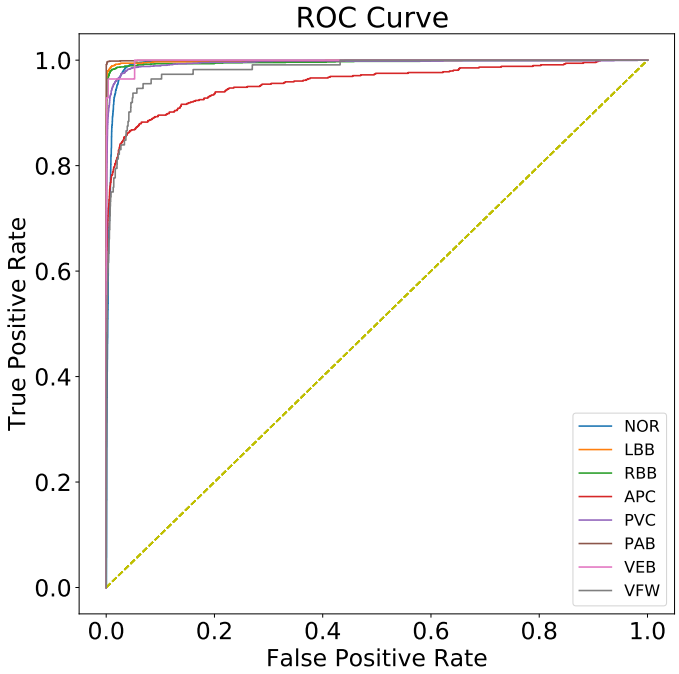


Figure 3.3: ROC Curves for Different Classes

NOR		***Predicted***	
		0	*1*
Actual	*0*	94.83%	5.16%
	1	0.90%	99.09%

LBB		***Predicted***	
		0	*1*
Actual	*0*	99.75%	0.24%
	1	2.69%	97.3%

RBB		***Predicted***	
		0	*1*
Actual	*0*	99.72%	0.27%
	1	6.19%	93.8%

APC		***Predicted***	
		0	*1*
Actual	*0*	99.63%	0.36%
	1	30.23%	69.76%

PVC		***Predicted***	
		0	*1*
Actual	*0*	99.44%	0.55%
	1	9.93%	90.06%

PAB		***Predicted***	
		0	*1*
Actual	*0*	99.95%	0.04%
	1	0.82%	99.17%

VEB		***Predicted***	
		0	*1*
Actual	*0*	100%	0%
	1	32.14%	67.85%

VFW		***Predicted***	
		0	*1*
Actual	*0*	99.92%	0.07%
	1	51.78%	48.2%

Table 3.2: Confusion Matrix of Cardiac Arrhythmia Classifier

A confusion matrix shown in Table 3.2 is derived to estimate the performance measure of a proposed architecture. It is observed from Table 3.2 that the proposed classifier provides high number of true

Class	AUC	Class	AUC
NOR	0.9938	***PVC***	0.9949
LBB	0.9983	***PAB***	0.9996
RBB	0.9963	***VEB***	0.9978
APC	0.9579	***VFW***	0.9823

Table 3.3: Area Under Receiver Operating Curve (AUC)

Paper		[69]	[80]	[81]	[82]	[83]	[84]	[3]	[85]	**Proposed Work**
Implementation Platform		180 nm	180 nm	65 nm	Software	40 nm CMOS LL HVT	40 nm SMIC CMOS	Software	Software	*180 nm*
No of Classification #Arrhythmia		8	5	2	2	2	2	8	2	*8*
Classifier		Multi-level SVM	ANN	CNN	CNN	WLC+SVM	ANN	CNN	ANN	*DNN*
Evaluation Matrix	***Sp***	NA	NA	NA	95	NA	98.9	NA	97.4	*99.09*
	Ac	98.5	98		93.19	98.2	NA	99.05	98.3	*97.01*
	PPV	NA	94	99	NA	NA	NA	NA	NA	*97.63*
Power		5.36 uW	13.34 uW	138.28uW	NA	3.76uW	0.127-0.671uW	NA	NA	*0.785uW*
Database		MIT-BIH Creighton University	MIT-BIH	NA	MIT-BIH	MIT-BIH	MIT-BIH	MIT-BIH	MIT-BIH	*MIT-BIH*
Frequency		1 kHz	10 kHz-25 MHz	128 kHz	NA	10 kHz	360 Hz	NA	NA	*1kHz*
Supply Voltage		1V	1.8V	0.9V	NA	1.1V	0.5-0.9V	NA	NA	*1.98V*
Classification Metric		Class Based	Patient Based	NA	Class Based	Class Based	Class Based	Class Based	Class Based	*Class Based*

Table 3.4: Comparision of the Proposed Work with Other State-of-the-art methods

positives and negatives for the maximum classes. However, due to unavailability of the large number of beats for the classes VEB, VFW and APC, our proposed classifier has less percentage of true positives for these above mentioned classes. It is to mention that the method reported in [3] reports higher accuracy for these three classes. This is because of utilizing complex CNN model for the classification and employing images of the beats as an input instead of raw ECG data in [3]. Hence, their model becomes very complex to get implemented in the hardware. However, our model is very simple for the hardware implementation and can perform better for the above mentioned three classes as well, if more number of beats are available for the training.

In order to utilize more number of beats in the training, the beats of VEB, VFW and APC classes are duplicated, which aids in the increase of number of true positives approximately by 10%. The Accuracy (Ac), Specificity (Sp) and Positive Predictivity (PPV) are calucalated as per the Equations 3.4, 3.5, 3.6, respectively. Here, True Positive (TP) are the beats correctly detected as arrhythmia.

True Negatives (TN) are the beats correctly detected as normal. False Positive (FP) are the beats incorrectly classified as arrhythmia and False Negatives (FN) are the beats incorrectly detected as normal. The proposed design attains an overall accuracy of 97.01%, specificity of 99.09% and positive predictivity of 97.63% on MIT-BIH database.

$$Sp = \frac{TN}{TN + FP} \tag{3.4}$$

$$PPV = \frac{TP}{TP + FP} \tag{3.5}$$

$$Ac = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.6}$$

Figure 3.3 exhibits reciever operating characteristic (ROC) curves of the classifier for different classes. The area under the curve is represented in Table 3.3 for all the eight classes. On macroaveraging the AUC for all the classes it can be seen that our design has AUC of 0.99 which is similar to the one reported in [3] (AUC : 0.989). This states that our proposed model showcase good performance for arrhythmia classification.

It is also evident from Table 3.4 that the proposed work has more or comparable accuracy, specificity and positive predictivity as compared to state-of-the-art works reported in the literature. It is to mention that the methods in [69], [80] and [3] present multiclass classification only for arrhythmia beats. The accuracy reported in [69] and [80] is 98.5% and 98%, respectively, on the ASIC platform, whereas, the accuracy reported in [3] is 99.05% on the software platform. In comparison with the above mentioned designs, our proposed design has an accuracy of 97.01%, which is just 2% less in classifying eight types of arrhythmia beats. However, the design proposed in this section exhibits significant reduction in the power as compared to the previously reported designs in [69] and [80]. Also, in [80], authors have utilised a patient-specific approach for classifying arrhythmia. As it is mentioned earlier that this kind of approach provides high classification accuracy for a particular patient only and cannot be generalized easily. However, this can be overcome when a wide range of data is utilized during realization. As we know, variation in ECG features not only improves accuracy of the implementation but also makes it feasible to get the methodology used widely. Therefore, the class-dependent approach provides more realistic outcome than the patient dependent classifiers.

The ASIC implementation of the proposed design is performed using 180nm CMOS technology

available with SCL Chandigarh. It can attain maximum operating frequency up to 10Mhz without any timing violations. However, for an ECG sampled at 360Hz, our proposed design can operate in real-time at a minimum frequency of 1kHz only. Post placement and routing, the proposed architecture consumes 0.785μW of power at 1.98V supply voltage and 1kHz operating frequency for the ECG signals sampled at 360Hz. Further, this ECG arrhythmia classification architecture utilises an active core area of 0.624 mm^2 post routing at the above mentioned technology node.

It can be observed in Table 3.4 that the power consumed by [84] is lower than the power consumption of our propose design. This is due to the more advanced technology node, lower frequency and supply voltages, but it can also be seen that the design proposed in [84] provides only two classifications, whereas our proposed design outputs eight classifications of cardiac arrhythmia beats. It is to mention that the design proposed in this section, [69] and [80] are implemented using 180nm CMOS Technology, and classify eight, eight and five types of arrhythmia beats, respectively. It can be observed that our proposed design is 6.82× and 17× more power efficient as compared to [69] and [80], respectively. Thus, it is evident from Table 3.4 that our design has the lowest on-chip power with comparable accuracy than the perviously reported methods making our design suitable for low power wearable healthcare applications.

3.3.4 Shortcomings of the Design

Though the proposed design can classify eight kinds of arrhythmia beats primarily used in diagnosing cardiac arrhythmia with an accuracy of 97.01% and specificity of 99.09%. The high multi-class classification accuracy, low area, and power requirements make the proposed architecture suitable for low-power wearable device applications. However, the proposed design has shortcomings which are stated as follows.

(i) Although the design classifies eight types of beats, the classes are different from AAMI/ANSI standards which are considered by medical professionals as well.

(ii) The design utilizes a class-based classification approach in which some beats of the same patient are present in both train and test sets. A better approach of classification (Subject Oriented Approach) should be considered in which the train and test data have ECG of completely different patients.

(iii) ECG beat segmentation and R peak extraction needs to be improved for continuous ECG monitoring. In this work, we do not include ECG pre-processing. ECG beats of fixed length

were input to the classifier directly.

(iv) Further, the RTL of DNN needs to be further optimized for better hardware implementation.

3.4 Cardiac Arrhythmia Classifier to Detect Five Types of Arrhythmia

To overcome the shortcomings of our previous work, a low power VLSI architecture for DNN based patient-independent and subject-oriented cardiac arrhythmia classifier co -processor is proposed in this section. The proposed design can classify five types of arrhythmia with high accuracy making it suitable to be used in wearable healthcare devices. Figure 3.4 presents the architecture of the proposed five class classifier.

3.4.1 ECG Feature Extraction and Beat Alignment

The primary step in classifying CA is to extract fixed-length vectors from the raw ECG signals, which are input to the classification block. The steps to extract feature vector are described in Algorithm 3. The raw ECG signal is first filtered using a bandpass filter of bandwidth $0.5-40$Hz [79] to remove the noise. This filtered signal is then passed through a feature delineation block, which extracts R peaks. Several methods have been reported previously for the detection of R peaks using ECG signals. However, the methods reported in the literature employ complex mathematical operations, such as, squaring operations [22, 79] or wavelet transforms [20] etc. These methods utilize more resources and memory as they require storing both ECG samples and intermediate output arrays. Therefore, in the proposed work, a simplified R peak extraction algorithm is employed, which utilizes simple difference and comparing operations to delineate R peaks, and stores only initial ECG samples.

Initially, 850 samples of ECG are fed serially to the feature delineation block and the maximum value of these samples is found using a comparator. The ECG samples are stored for further processing and a threshold value is updated as $\frac{ECG_{max}}{2}$ for finding R peaks. Once the threshold is updated, a counter is initiated to traverse stored ECG samples. R peaks are delineated, if the following two conditions are satisfied. First, $ECG_i > threshold$, and second, a zero-crossing is obtained i.e $MSB(ECG_{i+1} - ECG_i) \ \wedge \ MSB(ECG_i - ECG_{i-1}) = 1$. Thus, the proposed architecture needs no intermediate sample values to be stored, and utilize a comparator and XOR operation to find the R peak making our implementation resource and power efficient as explained in line 1-14 of Algorithm 3. Once the R peaks are marked, respective RR intervals are reckoned as the difference between two

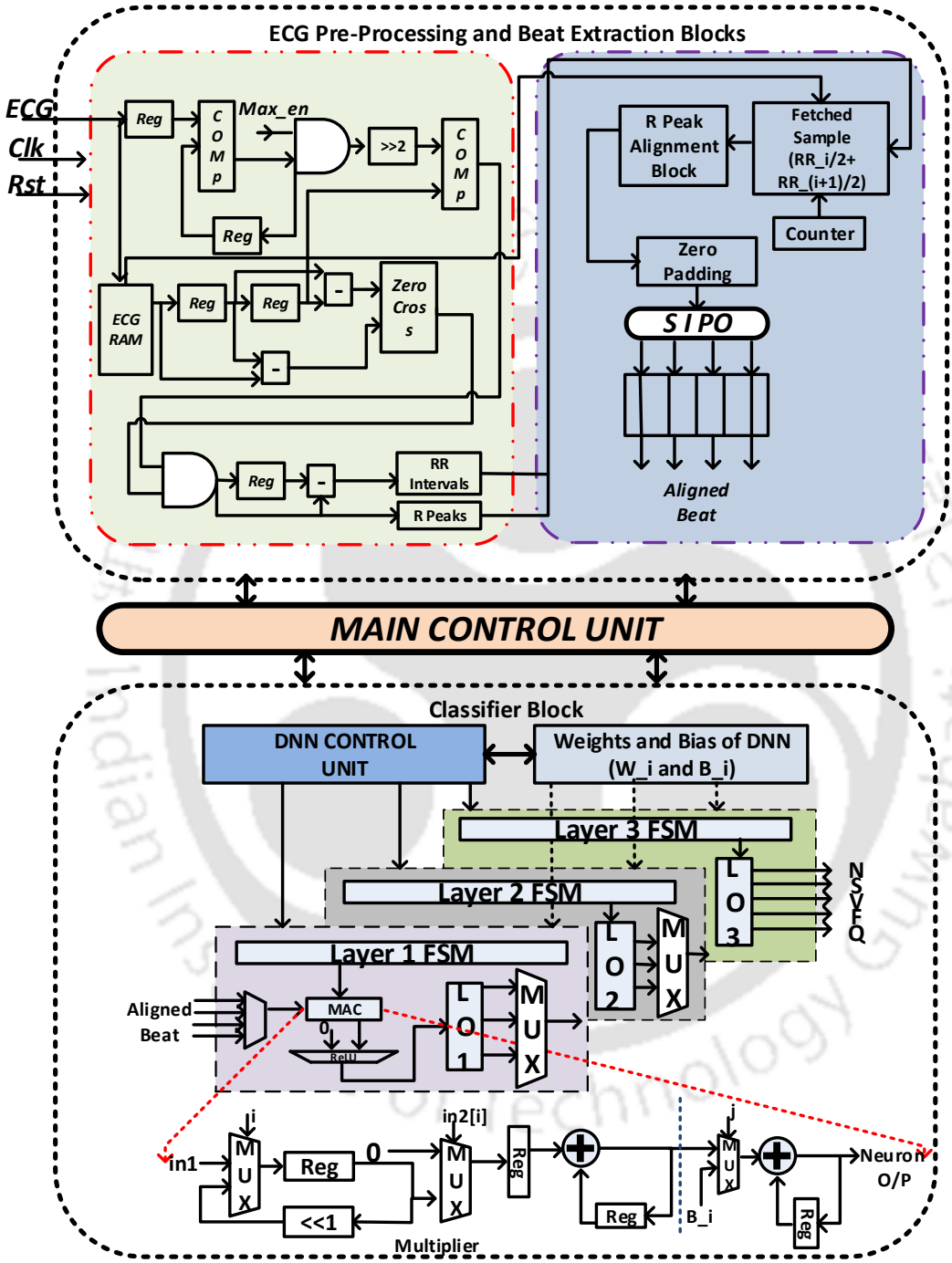


Figure 3.4: Architecture for ECG Processing and DNN Implementation

consecutive R peaks. Later, the beat segmentation block is enabled, as shown in Figure 1.

In the proposed work, an approach similar to [74] is employed for the beat segmentation and alignment. An ECG beat is taken as samples between two consecutive RR intervals, $RR_i/2$ and

$RR_{i+1}/2$. Later, the R peak is aligned in the centre and a fixed-size feature vector is generated by taking 208 samples from both the sides of R peak. Zero paddings are performed, if the number of samples is less than 417 between $RR_i/2$ and $RR_{i+1}/2$. However, it is observed that zero paddings are applied for most of the ECG beats because the difference between $RR_i/2$ and $RR_{i+1}/2$ is less than 417 in the maximum ECG excerpts. These unnecessary zero padding adds to redundant hardware resources as they contain no information of an ECG signal. Thus, to reduce the redundant hardware, we gradually decrease input layer size from 417 to 210 without causing any loss in the information and achieve comparable accuracy for the multiclass classification. It is to mention that further reduction in the beat size decreases the accuracy of DNN drastically. Therefore, our proposed methodology employs a similar approach as [74] as explained in lines 15-20 of Algorithm 3 to generate feature vectors of size 210 instead of 417. It is to mention that reducing beat size optimizes resource utilization by almost half with the acceptable performance.

Algorithm 3.

1: **Input:** Filtered Samples of ECG
2: **R Peak Extraction and Beat Alignment**
3: Initialize $ECG_{max} = 0$
4: Serially input ECG data ECG_i and store it in reg based memory
5: **if** $ECG_i > ECG_{max}$ **then**
6: $\quad ECG_{max} = ECG_i$
7: **end if**
8: Threshold for R peak: $Th = ECG_{max} >> 1$
9: $j, i \leftarrow 0$
10: **if** $ECG_i > Th$ and
11: $(ECG_{i+1} - ECG_i)[15] \wedge (ECG_i - ECG_{i-1})[15] == 1$ **then**
12: $\quad R[j] = i$
13: $\quad j = j + 1$
14: **end if**
15: $boundL = (R_{i-1} + R_i) >> 1$ and $boundR = (R_{i+1} + R_i) >> 1$
16: **if** $R_i - boundL < 105$ **then**
17: $\quad beat = zero\_paddings + ECG_i$, $i \in boundL$ *to* R_i
18: **else**
19: $\quad beat = ECG_i$, $i \in R_i - 105$ *to* R_i %Similar process for right side of beat taking boundR
20: **end if**
21: **Arrhythmia Classifier**
22: $k \leftarrow 0$ %k Controls the layer number
23: Begin
24: **for** $i = 0, \cdots, n\_layer_{k+1}$ **do** %Controlling neuron operations
25: $\quad$ **for** $j = 0, \cdots, n\_layer_k$ **do**
26: $\quad\quad sum_i = sum_i + (input\_sample_j \times W_{ji})$
27: $\quad\quad j \leftarrow j + 1$
28: $\quad$ **end for**

29: $y_i = sum_i + B_i$
30: **if** $k < 3$ *and* $y_i > 0$ **then**
31: $out_i = y_i$
32: **else**
33: $out_i = 0$
34: **end if**
35: $k \leftarrow k + 1$
36: Repeat from 23 for other layers
37: **end for**

end

3.4.2 Design and Implementation of CA classifier

Once the heartbeat segmentation is completed, the fixed-size feature vector is fed to DNN based classifier. Figure 3.4 illustrates proposed architecture of the DNN. As it is known, a DNN has input layers and hidden layers having neuron as a fundamental unit comprising of a multiplier and an accumulator (MAC), whose operation can be expressed as $y = \sum_{i=1}^{N} W_i x_i + B_i$. Here, W_i and B_i are the weight and bias of a neuron, respectively. Therefore, computation complexity and the hardware cost of a DNN are primarily determined by two factors. First is the size of input and hidden layers and, second is the efficiency of multiplier and adder units employed in a MAC. Therefore, we have employed two-stage optimizations to limit the power consumption and resource utilization of the proposed architecture. As already described, the design proposed in [74] has a large number of neurons and consumes large area and power, if implemented on the hardware. Therefore, we propose an optimal neural network, which is suitable for the resource-constrained applications. After reducing input layer size and several epochs of training, the size of DNN is optimized to $210 \times 35 \times 25 \times 5$, which sustains sufficient accuracy and optimal computational resources on the hardware. After optimizing DNN layers, a pre-trained network is implemented in the hardware using $Verilog\ HDL$.

As we know, weights and biases of the neurons are represented using floating-point number system, which is not only complex but also consumes more power as compared to the fixed point and integer arithmetic operations, when implemented in the hardware. Therefore, weights and biases are presented in 16-bit fixed point format, where first MSB 6 bits are fixed for the integer part and 10 bits are reserved for the decimal part. It is to mention that converting weights represented in the floating-point format to the fixed point does not incur any loss in the accuracy, and simplifies the implementation of a MAC unit.

Further, as shown in Figure 3.4, DNN has two dense layers and an output layer controlled by

a DNN control unit implemented as a Finite State Machine (FSM). This control unit controls an intermediate FSM of every layer. As soon as *Enable* signal of DNN becomes high, an extracted beat of 210 samples is input to the DNN. Later, control unit of layer 1 becomes functional and starts processing the inputs (Lines $21-36$ of Algorithm 3). Each input and its respective weight, which are stored in $2's$ complement format, are fed serially to the MAC unit.

As explained above, a MAC unit consists of a multiplier and an adder to accumulate final result. This multiplier requires shifters, half adders and full adders to perform the operation. There are a few previously reported methods, which utilize different kinds of approximate and fast adders and multipliers in the MAC unit for optimizing power [71,87]. However, approximate multipliers and adders affect overall accuracy of the network, whereas, fast adders and multipliers add to area overheads. Therefore, we propose a novel and optimized sequential shift multiplier with reduced registers to minimize resource utilization. In this multiplier a simple shift-by 1 operation and a single $32-$bit adder are employed, whose operations are controlled by a clock signal as shown in Figure 3.4. Implementation of this multiplier costs extra clock cycles, which increases latency, but it significantly optimizes resource utilization. Since pre-trained weights are less than one, 16 intermediate bits of 32 bit multiplier output is considered as its final output without affecting overall accuracy. This $16-$bit output is then sent to the accumulator for addition. The area and power consumption of the proposed MAC unit is reduced by $2.76\times$ and $2.34\times$, respectively, as compared to the conventional MAC unit synthesized at $180nm$ technology without any penalty to the accuracy. After multiplying and accumulating weights and inputs using a MAC unit, bias is added and the final output of a single neuron is obtained. Further, this MAC unit is shared with other DNN layers obtaining three times improvement in the resource optimization. The output obtained from MAC are fed to the "ReLU" activation function implemented using a simple multiplexer. The layers operate sequentially after having output from the previous layers controlled by DNN control unit. This lead to the optimization of area and power consumption of the proposed MAC unit as compared to the conventional MAC unit realized using $180nm$ technology. It is to mention that the proposed co-processor performs beat by beat classification in real-time at 12 kHz, which is suitable for low power.

While software training, softmax function is considered as an activation function of the final output layer for multiclass classification. A softmax function is utilized because it is differentiable at all the points and gives an accurate estimation of output posterior probabilities causing no information loss

making it suitable for multiclass classification. Since we employ a pre-trained network for the hardware implementation, therefore, for optimizing the architecture, a simple "max" function is employed as an activation function in the final output layer instead of a softmax function, and is implemented using a simple comparator. This further optimizes resources and power consumption making it suitable for ultra low power wearable devices.

3.4.3 Experimental Results of Proposed Design

3.4.3.1 Validation of the Proposed Methodology

In this section, a detailed discussion is presented on the outcome of the proposed design, when it is validated with various testcases. The designed architecture is tested and verified on the MIT-BIH arrhythmia database [88]. As per American National Standard (ANSI/AAMI EC57:1998) [89] issued by AAMI, the heartbeats are combined into five classes, namely, normal beat (N), ventricular ectopic beat (V), supraventricular ectopic beat (S), fusion of a normal and a ventricular ectopic beat (F) and unknown beat type (Q). We have utilized two evaluation schemes for estimating the performance of the proposed classifier. First is the subject-oriented approach as described in [72] *(Experiment 1)*, in which training and test sets are completely different from each other as per $ANSI/AAMIEC57$ standards. Therefore, it gives more realistic estimate on the ability of the classifier in practice. Second is the class-oriented approach to test the model on a larger database and to address data imbalance issue in the MIT-BIH data. In this scheme, data is divided in the ratio of 75 : 25 for testing and training, respectively *(Experiment 2)*. It is to mention that a model trained for the subject-oriented approach according to AAMI standard is used for the proposed hardware implementation. Further details about data can be obtained from [72]. Also, as suggested by $ANSI/AAMIEC57$ standard, we have given more importance to the classification of two majority classes of arrhythmia that are *Class S* and *Class V*.

The performance metrics for *Experiment* 1 and *Experiment* 2 are presented in Table 3.5 and Table 3.6, respectively. It is observed from Table 3.5, for class V, the proposed classifier has comparable sensitivity and specificity with respect to previously reported works. It also exhibits an overall accuracy of 91.6%, which is more than [71], [72], and a comparable accuracy to [73]. It has only 3.1% less accuracy than the classifier reported in [74]. However, it is to be noted that the previously reported works have complex classifiers, which are not suitable for the low power hardware implementation, and can only be used in a hospital setup having a software platform. Also, [71], [72] and [73] utilize

features of ECG as an input to the classifier instead of a raw ECG beat that again adds to the hardware cost. On the other hand, our proposed classifier is optimized for the low power applications for performing real-time arrhythmia detection. Also, it is evident from Table 3.5 that our method has comparable AUC values than [74] indicating it to be a promising candidate for the wearable healthcare applications. It can be observed from Table 3.6 that the class-oriented scheme has an overall accuracy

Methodology		**[71]**	**[72]**	**[73]**	**[74]**	**Proposed**
Overall Accuracy(%)		85.9	86.4	89.9	94.7	91.6
Class S (S v/s Non S	***Se%***	75.9	60.8	80.8	77.3	86.45
	Sp%	95.4	97.7	96.7	97.7	82.21
	AUC	NA	NA	NA	0.858	0.886
Class V (V v/s Non V)	***Se%***	77.7	81.5	82.2	93.7	83.37
	Sp%	98.8	96.4	99.0	98.8	94.06
	AUC	NA	NA	NA	0.951	0.925

Table 3.5: Comparision Results for Experiment 1 (Subject-Oriented Scheme)

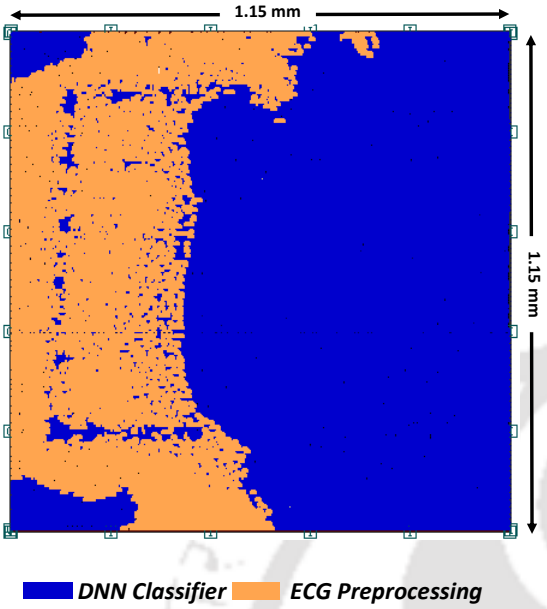
of 97.35% and the sensitivity of 94.83%, which is again better or comparable to the previously reported works employing class-oriented schemes. It is observed that the method reported in [90] cites highest accuracy and sensitivity as compared to our proposed as well as other state-of-the-art methods. It utilizes stacked convolutional and long short-term memory networks to classify ECG signals in only two classes. However, our method can perform multiclass classification using a simpler model with comparable accuracy and sensitivity. It is also to mention that for a class-oriented approach, an average AUC for *Class S* and *Class V* is 0.97, but due to the unavailability of data of other methods, AUC results with other state-of-the-art approaches could not be compared.

Method	**[72]**	**[91]**	**[92]**	**[90]**	**[93]**	**[94]**	**Proposed**
Overall Accuracy (%)	99.3	99.41	94.90	99.85	94.30	98.3	97.35
Se%	NA	96.08	99.13	99.85	93.18	95.51	94.83
Sp%	NA	NA	81.44	NA	NA	NA	93.25

Table 3.6: Comparision Results for Experiment 2 (Class-Oriented Scheme)

3.4.3.2 FPGA Implementation

The proposed CA classifier is first implemented on $Xilinx Virtex-7$ FPGA for initial functional verification of the architecture. Table 3.7 presents resource utilization of the proposed design. It is observed from Table 3.7 that our design utilizes 1.52% of total available resources making it resource-



CMOS Process	SCL 180nm
Area	$1.32mm^2(1.15mm \times 1.15mm)$
Voltage	1.98V
Frequency	12kHz
Dynamic Power	7.5403uW
Static Power	1.2097uW

Figure 3.5: The Layout Photograph of Co-processor and its Specifications

efficient.

Resources	Total Available	Resources Utilized	Percentage Utilization(%)
Slice LUT	303600	11125	3.66
Slice REG	607200	4884	0.80
F7 MUX	151800	1080	0.71
F8 MUX	75900	255	0.33
IOB	600	24	4
BUFGCTRL	32	1	3.125
Total Resources	1139132	17369	1.52

Table 3.7: FPGA Resource Utilization

3.4.3.3 ASIC Implementation

The DNN based CA classifier is implemented as an ASIC employing $180nm$ Bulk CMOS Technology available with SCL Chandigarh as shown in figure 3.5. Post placement and routing using Synopsys IC compiler, the design operates at $12kHz$ consuming power of $8.75\mu W$ having area of $1.32mm^2$.

Table 3.8 presents a comparison of our proposed classifier with some well-known state-of-the-art CA classifiers on the hardware. It is observed from Table 3.8 that our proposed architecture exhibits

Method	[75]	[76]	[95]	[70]	Proposed
Process	180nm	65nm	40nm	40nm	180nm
Frequency (Hz)	10k-25M	10k	1M	10k	12k
VDD(V)	1.8	1	1	1.1	1.98
Power(W)	13.34μ	2.78μ	14.14m*	3.76μ	8.75μ
Area(mm^2)	0.9250	0.112	0.135	0.12	1.32
Model Utilized	ANN	Naive Bayes	SVM	WLC+SVM	DNN
Evaluation Scheme	Patient-Specefic (PS)	Class-Oriented(CO)	Subject-Oriented (SO)	Class-Oriented (CO)	Subject-Oriented(SO)
Overall Accuracy(%)	99.39	86	88.06	98.2	91.6
Figure of Merit(pJ)	3.36×10^3	NR	7.23×10^6	NR	2.08×10^6
Number of Classes	5	2	3	2	5

*Power is calculated from Energy and Frequency reported in paper [95] **FoM is classification time x power ***NR:Not Reported

Table 3.8: Comparision Results of ASIC Implementation

higher accuracy than the methods reported in [76] and [95]. It is also seen that the overall accuracy of our proposed methodology is less than the methods reported in [75] and [70]. It is to mention that [70] showcases more accuracy because it is designed for two-class classification only. In contrast, our method reports five-class classification as per AAMI standards described in section III. On the other hand, [75] reports better accuracy for five- class classification. However, the method in [75] utilizes a patient-specific approach instead of a subject-oriented approach to classify arrhythmia into five different classes. Also, [75] employs a conditional biasing scheme, in which the data from different classes are merged to perform classification that leads to better accuracy due to availability of more data for a particular class at the time of training. In comparison, our method do not employ biasing of data in any form and is implemented as per AAMI standards described in [74]. It is to be noted that in the subject-oriented method, classifier is trained on diverse ECG beats and is tested on completely blind data. Therefore, it performs better in the practical wearable healthcare applications by classifing ECG beats correctly even for unknown individuals as well. However, design reported in [75] exhibits the best performance in the patient-specific classification.

Further, it can also be observed from Table 3.8 that our architecture has higher power consumption

than the design reported in [70] and [76]. It is to mention that these designs perform two-class classification only utilizing less power than our proposed design. Also, their Figure of Merit (FoM) cannot be compared due to the unavailabilty of necessary data. It is further observed from Table 3.8 that our proposed architecture has higher FoM compared to design reported in [75] because of the following reasons. In [75], the design is implemented using a patient-specific approach at a fixed heart rate, which requires retraining for every individual under expert intervention for correct marking of the ECG beats. Further, this design employs conditional grouping and biased training, where data of some classes are merged with other classes. Thus, it requires a small network size to achieve high accuracy in the classification of grouped classes. However, the classification of merged classes is not recommended by AAMI. Also, in [75], classification time and power consumption are presented only for the classifier module. This is because of the implementation of pre-processing of ECG signals in the software.

On the other hand, in our proposed work, complete co-processor including ECG pre-processing and classification is implemented in hardware to classify five classes of arrhythmia without any data grouping. Thus, the delay and power consumption associated with its pre-processing lead to higher FoM. It is to mention that the proposed design can perform classification in real time at 12 kHz only. Therefore, our design exhibits less power consumption as compared to [75], which needs higher operating frequency to sustain its operations. However, the proposed design has better FoM as well as power consumption than [95], in which both ECG pre-processing and classifier are implemented in hardware to perform three-class classification employing subject-oriented approach. Therefore, our proposed design can be the most suitable to be used in the wearable devices.

3.5 Prediction of Cardiac Arrhythmia

Among all the chronic CVDs, malignant Ventricular Arrhythmia (VA), namely Ventricular Tachycardia (VT) and Ventricular Fibrillation (VF), is highly common and responsible for a large number of sudden cardiac deaths. Ventricular arrhythmia (VT/VF) is an abnormally fast rhythm in the heart's bottom chambers. VT is a rhythm of more than three consecutive beats originating from the ventricles at a rate of more than 100 beats/min [96]. VF is another rhythm characterized by the chaotic activation of ventricles. It causes immediate cessation of blood circulation and degenerates into a pulseless or flat ECG signal indicating no cardiac electrical activity. A VT/VF, which lasts longer than 30

seconds, is detrimental to health because hemodynamic compromise may occur and lead to sudden cardiac death [97]. The implantable cardioverter-defibrillator has been considered the best protection against sudden death from ventricular arrhythmias in high-risk individuals. However, most sudden deaths occur in individuals who do not have high-risk profiles. Many methods have been proposed that can detect VA events using ECG signals in the past. However, as patients with VT/VF episodes are at high risk of sudden cardiac death, it is imperative to predict rather than detect the events of VT/VF. Predicting VT/VF would enable individuals to get prompt treatment and avoid fatalities. Therefore, in this section, a novel architecture of a co-processor for arrhythmia prediction (CoAP) is proposed. CoAP can predict the occurrence of VT/VF with high accuracy and consumes low power making it suitable for wearable devices.

3.5.1 System Overview of CoAP

A typical ECG processor has two significant operation blocks. The first is the analog front end responsible for acquiring the ECG signal from the on-body sensors and converting them to digital samples. The second block is the co-processor block responsible for processing the acquired data and further detecting the abnormalities present in the ECG signal. As explained above, the occurrence of ventricular arrhythmia turns out to be fatal if timely treatment is not provided. Therefore, the co-processor block "CoAP" proposed in this section can act as a life-saving system for individuals susceptible to VT by alerting them about their medical condition. As already described in Chapter 1, a conventional ECG systems acquire the ECG signals from the analog front end and transmit them over a channel for further analysis. CoAP can act as an integrated co-processor that acquires ECG signals from the sensors, processes them, and performs on-chip classification without any external interaction. Such design methodology immediately alerts an individual, which is essential in critical conditions. Moreover, processing and classifying the data locally on a chip reduces the amount of data transmitted through a data channel, optimizing power consumption.

CoAP further consists of three primary blocks: the ECG delineation, descriptor processing, and classifier block. First, the ECG delineation block is responsible for extracting the fiducial points of ECG. The raw ECG signal obtained from the ADC is first filtered using a bandpass filter of band $0.5 - 40Hz$. Filtered ECG excerpts of three-second windows are considered for processing. Later, the $P - QRS - T$ waves of ECG signals are extracted using our derivative based ECG feature extraction algorithm, which is described in chapter 2. However, new techniques for optimizing the existing

Descriptor		Details	Original Definition	Optimized Definition
QRS Duration	mean	QRS complex duration represents ventricular depolarization.	$QRS_m = \frac{\sum(QRS_{off} - QRS_{on})}{n}$	$QRS_{m_{opt}} = \frac{1}{16}(\frac{\sum(S_{peak} - Q_{peak})}{n})$
RT Interval	mean	Depicts the time from the beginning of ventricular depolarization to the end of its repolarization. It also includes electrical events that take place in the ventricles	$RT_m = \frac{\sum(T_{off} - QRS_{on})}{n}$	$RT_{m_{opt}} = \frac{1}{16}\frac{\sum(T_{peak} - R_{peak})}{n}$
	variance		$RT_{var} = \frac{\sum(RT - RT_m)^2}{n}$	$RT_{var_{opt}} = \frac{1}{64}(\frac{\sum(RT_{opt} - RT_{m_{opt}})^2}{n})$
PS Interval	mean	It represents both atrial depolarization and ventricular depolarization	$PS_m = \frac{\sum(P_{on} - QRS_{off})}{n}$	$PS_{m_{opt}} = \frac{1}{16}\frac{\sum(P_{peak} - S_{peak})}{n}$
iCEB	mean	It signifies balance and imbalance of depolarization (QRS duration) and repolarization (QT) of the cardiac electrophysiology $iCEB = \frac{QT}{QRS}$	$iCEB_m = \frac{\sum(\frac{(T_{off} - QRS_{on})}{(QRS_{off} - QRS_{on})})}{n}$	$iCEB_{m_{opt}} = \frac{\sum(\frac{(T_{peak} - Q_{peak})}{(Q_{peak} - S_{peak})})}{n}$
	variance		$iCEB_{var} = \frac{\sum(iCEB_m - iCEB)^2}{n}$	$iCEB_{var_{opt}} = \frac{\sum(iCEB_{m_{opt}} - iCEB_{opt})^2}{n}$

Table 3.9: Deatils of Optimized Descriptors

feature extraction algorithm are incorporated in this section, which include employing adaptive search windows and thresholds to increase the robustness of the proposed design. The detailed hardware description of ECG delineation is presented in section 3.5.2.

The second block is the descriptor processor block, which is responsible for extracting the critical information from the variation of primary ECG features. In the proposed work, the features required as an input feature vector for the classifier block are termed "*Descriptors*". Researchers have conducted many studies to discover various ECG features that could reflect underlying VT/VF [98]. Some ECG features have predictive values for arrhythmic events, such as heart rate variability (HRV), T-peak T-end (TpTe), and QT wave intervals. Authors in [99] have utilized features, such as the mean of QRS intervals, to detect VT/VF. They have also used standard and root-mean-square deviations of Q, S and R waves to predict the occurrence of VT/VF by employing decision trees on the software platform. However, the complex square root and division operations required to estimate these features might significantly add to the hardware utilization of the co-processor. Further, in [76], the authors used seven unique sets of interval-based features to predict the VT/VF, which does not require any complex mathematical operations. However, ECG features with onset and offsets of $P - QRS - T$ waves need to be delineated to find these intervals. Also, the accuracy obtained after using these features is only 86%, which is less than [99]. Therefore, we propose an optimal feature set that balances the tradeoff

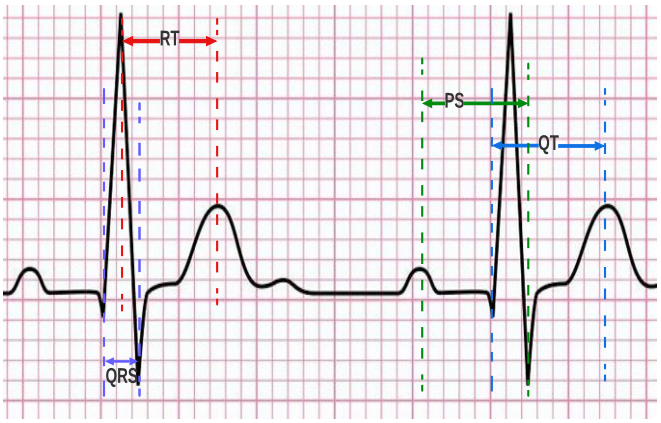


Figure 3.6: Optimized ECG Intervals

between accuracy and hardware resource utilization in this work. This is crucial for enhancing the robustness and acceptability of the proposed method for practical applications.

3.5.1.1 Descriptor Optimization

After a rigorous study of different ECG characteristics associated with VT/VF, we developed a set of six descriptors, which considerably help in the prediction of VT/VF. These descriptors are the mean and variance of RT interval (RT_m), mean of PS intervals(PS_m), mean of the duration of QRS complexes, and mean and variance of the index of cardiac electrophysiological balance ($iCEB_m$ and $iCEB_{var}$). The description of these features is given in Table 3.9. Note that an ECG excerpt of a length of three seconds is employed to find the descriptors at a time. Conventionally, RT interval, PS interval, QRS duration, and QT intervals are the difference between the boundaries of the respective $P-QRS-T$ waves. For instance, RT is estimated as the difference between QRS_{on} and T_{off}. Estimation of these features requires delineating the peaks and boundaries of the $P-QRS-T$ waves. However, estimating the boundaries of $P-QRS-T$ waves is critical due to the varying morphology of the ECG waves and requires separate modules for its implementation. This significantly increases resource utilization. Therefore, in this chapter, features are optimized by taking the RT interval as the difference between the R peak and the T peak instead of the boundaries, as shown in Figure 3.6.

Other features are optimized similarly, as shown in Table 3.9. Further, it was observed that employing the interval features might cause overflow while implementing a deep neural network on hardware. Thus, the number of bits required to represent data needs to be increased to get high

classification accuracy to accommodate the values of estimated features. Therefore, the features are further optimized by dividing them with a specific threshold described in Table 3.9 to avoid increasing the number of data bits without reducing accuracy. These descriptors are finally fed as input feature vectors to the final classifier block of the ECG co-processor. In the proposed work, a deep neural network is utilized as a classifier to predict the excerpts susceptible to ventricular arrhythmia. The novelties of the proposed design are summarised as follows.

- The designed co-processor can predict VT/VF 15 minutes before its onset.
- The design utilizes an optimized feature set to attain maximum accuracy with low power and area requirements.
- The design employs an optimized ECG delineation algorithm for low power implementation.
- An optimized multiplier is proposed for low power implementation of pre-trained DNN on hardware while maintaining high accuracy.

3.5.2 Implementation Details of CoAP

This section presents the architecture of CoAP, as shown in figure 3.7. The CoAP architecture is divided into three stages, which are controlled using the main control unit (MCU). The intermediate modules are controlled with a separate finite state machine (FSM), as shown in figure 3.7. An ECG excerpt of three seconds, with each sample represented in $16-$bit $2's$ complement format sampled at a frequency of 250 Hz, is considered at a particular time for processing. These samples are fed serially to the CoAP through a testbench during simulations.

3.5.2.1 ECG Delineation and Feature Extraction

It is known that ECG has three primary features, P wave, QRS complex, and T wave [11], which are considered for manual and automated detection of cardiac abnormalities. Over the decades, several ECG delineation algorithms have been proposed that claim to have high-performance metrics in extracting ECG features, for instance, [22–24]. In this work, we have utilized an optimal ECG delineation algorithm, which is previously reported in [100] which is also described in chapter 2. Moreover, the previous ECG delineation is optimized in this section by making it adaptive to irregular heart rates. As we know, the raw ECG signal has motion artifacts, isoelectric line wandering, and high-frequency noise with the signal components. Therefore, the raw ECG signal is first processed to remove noise to ensure accuracy in extracting fiducial points. The proposed algorithm uses a single

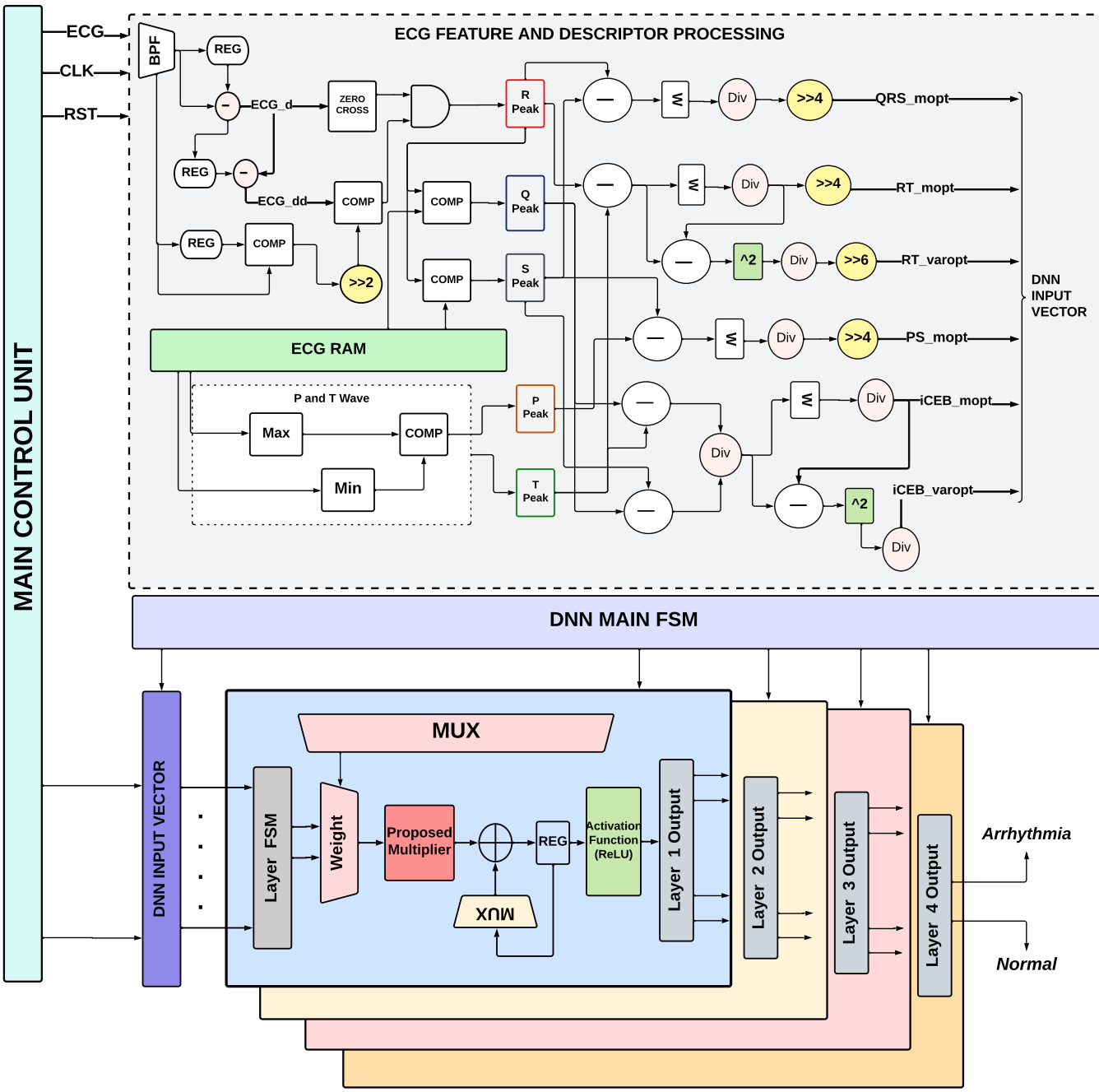


Figure 3.7: Design of Architecture of CoAP

bandpass filter of bandwidth $0.5 - 40Hz$ [10] to obtain the filtered ECG signal (ECG) in a 3-second window. Initially, the QRS peak is extracted with the help of single (ECG_d) and double (ECG_{dd}) derivatives of the filtered ECG signal (ECG). Some previously reported methods, such as [23], also employ derivatives of ECG signals and threshold values to extract the R peaks. However, they utilize thresholding techniques, which involve complex multiplication and averaging operations having higher hardware cost. Therefore, for low power and resource-efficient implementation, a threshold is computed as 3.7 in the proposed design.

$$Th\_R = abs\_max >> 2 \tag{3.7}$$

In 3.7, abs_max is the absolute maximum of ECG samples in a 3-second window. The chosen threshold is estimated using a simple comparator and shift operation, thus optimizing the hardware resources. Later, the R peak is extracted, if a zero crossing is encountered in ECG_d and the magnitude of ECG_{dd} is greater than the threshold, as shown in equation 3.8. ECG_d and ECG_{dd} are estimated in real-time with the help of the filtered signal ECG to avoid the need to store different samples.

$$\begin{aligned} &ECG_{d_i}[15] \wedge ECG_{dd_{i+1}}[15] == 1 \\ &ECG_{dd} >= Th\_R \end{aligned} \tag{3.8}$$

The threshold for finding R peaks is updated with every 3-second window, enabling detection to be adaptive to an irregular heartbeat. Once the R peak is detected, a counter is incremented by a window of 85 ms ($W1$). $W1$ is chosen as per standard QRS complex interval for an ECG signal sampled at 250 Hz. This increment avoids delineating two R peaks in the same ECG beat. Steps $3 - 17$ of the algorithm 4 explain the complete R peak extraction.

The $R - R$ intervals (RR) are computed while detecting R peaks as the difference between two consecutive R peaks. Later, minimum values estimated in a 60 ms window ($W2$) before and after the R peak are marked as Q and S peaks, respectively. Once the QRS complex is delineated, P and T peaks are extracted as presented in steps $20 - 28$ of algorithm 4, which are later used to estimate the descriptors necessary for classification. The P wave is the first feature of an ECG wave with a standard duration shorter than 120 ms. However, during certain CVDs, its duration can reach up to 170 ms [10]. Therefore, a window adaptive to RR intervals is chosen for delineating P peaks. Employing windows as per RR intervals makes the feature extraction adaptive to varying

heart rates. Further, the windows are taken in such a way as to avoid the floating-point operations, keeping hardware requirements simple. The maximum and minimum of the ECG_d are computed in the window ranging from $RR/2$ to $RR/4$. If the minimum of ECG_d is greater than the maximum of ECG_d in a chosen window, then a P wave is considered positive; otherwise it is treated as negative. Later, the ECG sample corresponding to the maximum of ECG_d is marked as the P peak for the positive P wave and vice-versa for the negative P wave. The final feature of an ECG wave is the T wave. In a healthy individual, the duration of the T wave varies from $125 - 200$ ms [10]. However, it varies when a person is suffering from cardiac issues. Therefore, an adaptive window ranging from $3RR/8 - RR/2$ is taken after R peaks. A procedure similar to the delineation of the P wave is then employed to extract T peaks. As explained above, the employment of optimized features avoids the need to extract boundaries of the $P - QRS - T$ waves required in the state-of-the-art methods mentioned before. This optimizes the hardware and computation requirements of our design.

Algorithm 4.

Input: Filtered Samples of ECG
Input: Filtered Samples of ECG(y)
$j, i \leftarrow 0, abs_{max} = 0$
if $ECG[i] > abs_{max}$ **then**
    $abs_{max} = ECG[i]$
end if
$Th_R = abs_{max} >> 2$
R Peak Extraction
$j, i \leftarrow 0$
$ECG_d = ECG[i+1] - ECG[i]$
$ECG_{dd} = ECG_d[i+1] - ECG_d[i]$
if $ECG_{ddi} > Th$ and
$(ECG_{di+1} - ECG_{di})[15] \wedge (ECG_{di} - ECG_{di-1})[15] == 1$ **then**
    $R[j] = i$
    $j = j + W1$
else
    $j = j + 1$
end if
$Q[i] = min(ECG)$, $S[j] = min(ECG)$, $i\ in R[i] - W2\ to\ R[i], j\ in R[i]\ to\ R[i] - W2$
$boundP = (R_i\ to\ R_i - (RR_i/2 - RR_i/4)$
$boundT = (R_i\ to\ R_i + (3RR_i/8 - RR_i/2)$
Delineate T wave in boundT similar to P wave
Find min and max of ECG_d and ECG_{dd} in boundP
if $|min\ ECG_d| > |max\ ECG_d|$ **then**
    $P_{pos} = max\ ECG_{dd}$
else
    $P_{pos} = min\ ECG_{dd}$
end if

end

3.5.2.2 Descriptor Processing

Once the $P-QRS-T$ peaks are delineated for a 3-second window, these peaks are input to the descriptor processing block. The descriptors are calculated using the optimized definitions described in Table 3.9. To estimate the descriptors, initially, the 16-bit values of the features are zero-extended to 32 bits for the intermediate operations. Later, this is truncated to 16 bits as required. As it is evident from Table 3.9, to estimate the descriptors, we mainly need subtraction, squaring, and division operations. However, it is observed that a 3-second ECG signal sampled at 250 Hz has four ECG beats at most. Therefore, the number of peaks, i.e., "n", is less than or equal to four. Thus, the summation of every feature needs to be divided by an integer less than or equal to four to find the mean and variance of an ECG excerpt. Therefore, the descriptor processor is implemented using simple case statements, which become functional as per the number of R peaks. Division by "n" operation is converted to right shift by 1 or 2, if n is 2 or 4, respectively. However, if the value of n is 3, then dividing by three is approximated using equation 3.9.

$$div\_out = (x << 4 + x << 2 + x << 1) >> 5 \tag{3.9}$$

The complete descriptor block is implemented as an FSM. In the first state, the difference operations are executed. Once the difference values of peaks as per the optimized definition of descriptors are obtained, the summation of these values is performed in the second state. In the third state, mean values are estimated to find $QRS_{m_{opt}}$, $RT_{m_{opt}}$, $PS_{m_{opt}}$, and $iCEB_{m_{opt}}$. Once mean values are obtained, these values are divided by the respective thresholds, as shown in Table 3.9. As evident from this table, the thresholds are taken so that they can be implemented using simple right shift operations, which avoids the need for division operations. Similar operations are done to find the variance of RT and $iCEB$, once the mean of respective descriptors is estimated.

3.5.2.3 DNN Classifier

The DNN classifier is enabled once the estimated descriptors are fed as input to the classification block. In this work, we have utilized a low power implementation of a deep neural network trained on software using an Intel(R) Xeon(R) CPU E5-2630 machine. After a number of trainings on the employed dataset, the model's size is fixed as $32 \times 16 \times 8 \times 2$ with an input feature vector size of 6. During the implementation, each descriptor is represented in $2's$ complement format using $16-$bits. It

is worth mentioning that the classification accuracy has no distinct drop after limiting the bit length of the descriptors to 16−bits, where 10 bits are used to represent the integer part, and 6 bits are for the fractional part. Further, the weights and biases are presented in a similar fixed-point format. As the weights obtained after training are less than 1, the neuron output is also truncated to 16 bits without reducing accuracy.

3.5.2.4 Optimized Implementation of Activation Function

For power-restricted wearable applications, the primary consideration for implementation is the simplicity of the hardware. Therefore, ReLU is chosen as an activation function in the proposed design because it requires only a simple multiplexer (MUX) in the implementation.

$$ai = ReLU(xi) = \begin{cases} xi & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{3.10}$$

As we perform two-class classification, a Sigmoid is utilized as an activation function for the final output layer. In general, only a single output node with a sigmoid activation function is employed for a two-class classification. However, in this work, we have utilized two output nodes instead of one, each with a sigmoid activation function. It aids in the hardware implementation of the neural network, which is explained below. A sigmoid activation function has an exponent function, as shown in equation 3.11.

$$Sigmoid(xi) = \frac{1}{1 + e^{x}} \tag{3.11}$$

Hardware implementation of exponential function requires complex algorithms like CORDIC or Taylor series, requiring more area and power resources. Therefore, to optimize the hardware implementation of the trained neural network, we utilize two output nodes during the training. Post-training, the sigmoid function was replaced by a simple comparator in hardware. The comparator compares the output of two final nodes, and the one with the maximum value is chosen as the output. Thus, employing two nodes enabled us to eliminate the sigmoid function during hardware implementation, optimizing the area and power considerably.

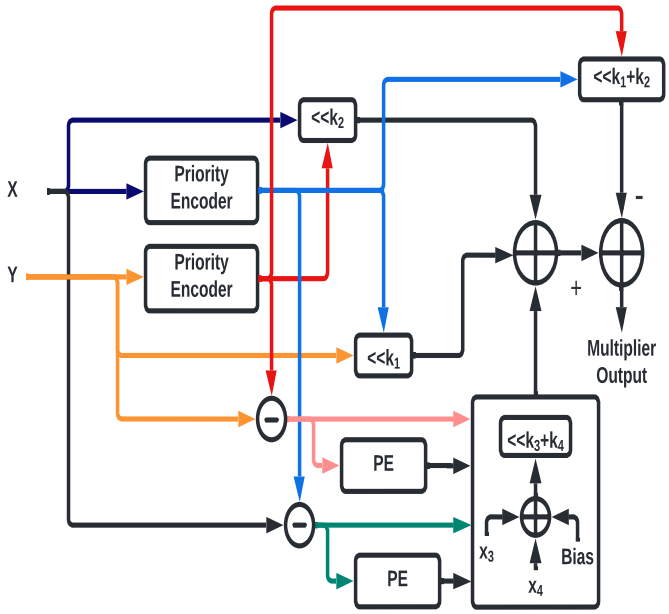


Figure 3.8: Proposed Optimal Multiplier

3.5.2.5 Proposed Optimized Multiplier

For low-cost and compact wearable devices, power and area are the primary considerations over speed. Therefore, a parallel architecture of DNN is unnecessary as it increases leakage power and utilizes a larger silicon area. Therefore, in the proposed work, a single multiplier and accumulated unit (MAC) is employed, which is reused by neurons of every layer. In our previous work we proposed a reduced register sequential multiplier for the efficient implementation of a MAC unit to optimize the power requirements of a DNN. However, the sequential operation of the proposed multiplier requires a higher clock frequency for real-time operation, which leads to higher power dissipation. Therefore, in this section, we propose an area and power optimal approximate multiplier. Despite this minimal error introduced in multiplication due to approximations, the accuracy of DNN is not affected when compared to the software implementation. Moreover, utilizing the approximate multiplier enabled us to maintain low-frequency operation for real-time processing of CoAP. Note that the proposed multiplier is an advancement over our previous work [101] and is described in detail below.

In [101], we employed an approximate multiplier based on the Mitchell algorithm, which approximates the complete product using a multiplier and a multiplicand as a whole. However, to further increase the efficiency of the algorithm, we use multiplication by splitting the multiplier and multipli-

cand into two parts, "I" and "F". As the "F" part is very small, the error approximation is much less than the error, which is introduced if the complete binary product is approximated as in [101]. The proposed approximate multiplier uses shifting operations, which reduces hardware computation costs. The operation and implementation of the proposed multiplier are described below.

As we know, the product of any number with 2^n can be reduced as a left shift operation($<< n$). Therefore, to calculate the approximate product, binary numbers $a = a_{n-1}\ a_{n-2}...a_0$ and $b = b_{n-1}\ b_{n-2}...b_0$ are first represented as equation 3.12.

$$\begin{aligned} a &= 2^{ka}(1 - x_a) = 2^{ka} - 2^{ka}x_a \\ b &= 2^{kb}(1 - x_b) = 2^{kb} - 2^{kb}x_b \end{aligned} \tag{3.12}$$

Let 2^{ka} and 2^{kb} be represented as A_i and B_i. Further, let $2^{ka}x_a$ and, $2^{kb}x_b$ be represented as A_f and B_f. Therefore, the product of two binary numbers a and b is approximated as equation 3.13.

$$\begin{aligned} a \times b &= (A_i - A_f)(B_i - B_f) \\ &= A_i \times B_i - A_f B_i + A_f B_f - A_i B_f \\ &\approxeq A \times 2^{kb} + B \times 2^{ka} - 2^{k_a+k_b} + x_a x_b \times 2^{k_a+k_b} \end{aligned} \tag{3.13}$$

$A \times 2^{kb}$, $B \times 2^{ka}$, $2^{k_a+k_b}$ is calculated using simple shift operations, which would not introduce any errors, whereas $x_a x_b \times 2^{k_a+k_b}$ is estimated employing a modified Mitchell algorithm proposed in [101].

Several accurate and approximate multipliers are proposed in the literature. However, approximate multipliers are widely used for applications where speed and area are prime considerations. Among all the available methods of approximate multiplications, Mitchell's algorithm [102] is the most simplified approach for integer and fixed-point multiplication.

Mitchell has proposed an efficient approximate multiplier (AM) [102], which utilizes the *log* multiplication property. It estimates approximate multiplication by linearly approximating *log* and *antilog* values. First, the approximate characteristic part of *log* is obtained by finding the position of the leading one, i.e., the leftmost one in the binary sequence. Later, the remaining value is used as an approximate fractional part of the *log*. Further, an approximate *antilog* operation is performed by adding the two operands, which generates an approximate product.

Let an N bit integer be Y with bits $Y_{n-1}, Y_{n-2}, \cdots Y_0$, which is represented as $Y = \sum_{i=0}^{N-1} 2^i y_i$. Assuming the leading one occurs at position K, where $(N-1) <= K <= 0$, Y can be presented as

Equation 3.14 without any loss in accuracy.

$$Y = 2^k \left(1 + \sum_{i=0}^{k-1} 2^{i-k} y_i\right) \tag{3.14}$$

Equation 3.14 is further represented as $Y = 2^k(1+x)$, where $x = \sum_{i=0}^{k-1} 2^{i-k} y_i$. Therefore, log_2 of Y can be presented as Equation 3.15, where K being an integer represents characteristic value of log_2 and $log_2(1+x)$ is the fractional part.

$$log_2 Y = k + log_2(1+x) \tag{3.15}$$

As, $0 <= x < 1$, therefore, the linear approximation of $log_2 Y$ can be presented as Equation 3.16, where function $\widetilde{log}$ is representation of approximate log_2.

$$\widetilde{log}_2 Y = k + x \tag{3.16}$$

Further, let us assume that there are two N-bits binary numbers, Y_1 and Y_2, with leading ones at k_1 and k_2. These numbers can be presented as $Y_1 = 2^{k_1}(1+x_1)$ and $Y_2 = 2^{k_2}(1+x_2)$. Therefore, the approximate product $(\widetilde{P})$ is estimated as equation 3.17.

$$\widetilde{log}_2(\widetilde{P}) = k_1 + k_2 + x_1 + x_2 \tag{3.17}$$

Next, to calculate approximate *antilog* of Equation 3.17, 1 is added to the fractional part (x), which is in range $[0,1)$ and is scaled with respect to the characteristic part. Since $0 \leq (x_1, x_2) < 1$, thus, in this case $0 \leq x_1 + x_2 < 2$. Further, the approximate product is decomposed into two cases. The first case is $0 \leq x_1 + x_2 < 1$, in which a carry is not generated from the fractional part to the characteristic part, whereas the second case is $1 \leq x_1 + x_2 < 2$, in which a carry is generated to the characteristic part.(Equation 3.18).

$$\widetilde{P} = \begin{cases} 2^{k_1+k_2}(x_1 + x_2 + 1) & x_1 + x_2 < 1 \\ 2^{k_1+k_2+1}(x_1 + x_2) & x_1 + x_2 \geq 1 \end{cases} \tag{3.18}$$

This approximation reduces the multiplication operation to the basic add and shift operations. However, this decomposition introduces certain errors in the product obtained. Thus, an efficient error reduction technique is utilized for a more accurate multiplication of the weight and bias of a neuron, which considerably reduces the error of the multiplier. This technique is described in our

previous work [101]. A bias is calculated by averaging the error across the entire range of fractional part x, which is added to the approximate product to improve accuracy. Next, an error E in the approximate product is estimated by equation 3.19, where P is the logarithm product of Y_1 and Y_2, and $\widetilde{P}$ is the approximate product output by Mitchell's algorithm.

$$\begin{aligned} E &= \widetilde{P} - P \\ &= \begin{cases} -2^{k_1+k_2}(x_1 x_2) & x_1 + x_2 < 1 \\ -2^{k_1+k_2}(1 + x_1 x_2 - x_1 - x_2) & x_1 + x_2 \geq 1 \end{cases} \end{aligned} \tag{3.19}$$

Further, average error (bias) is calculated using Equation 3.20.

$$\begin{aligned} E_{avg} &= \frac{1}{(1-0)(1-0)} \int_0^1 \int_0^1 E dx_2 dx_1 \\ &= \int_0^1 \int_0^{1-x_1} -2^{k_1+k_2}(x_1 x_2) dx_2 dx_1 + \\ &\quad \int_0^1 \int_{1-x_1}^1 -2^{k_1+k_2}(1 + x_1 x_2 - x_1 - x_2) dx_2 dx_1 \\ &= -2^{k_1+k_2}(0.083333) \end{aligned} \tag{3.20}$$

It is noted that the average error (bias) is always negative and depends on k_1 and k_2. Bias is shifted and added as per the position of the leading ones in Y_1 and Y_2.

Note that the average error (bias) is always negative and depends on k_1 and k_2. Bias is shifted and added as per the position of the leading ones in Y_1 and Y_2. The implementation of the proposed multiplier is depicted in Figure 3.8. The leading one finder is optimally implemented in the hardware as a simple priority encoder to estimate k_1 and k_2. Further, the binary numbers can be shifted and rearranged in a fixed point representation to get respective fractional parts x_1 and x_2. Finally, the approximate product is obtained by adding and shifting x_1, x_2, k_1, and k_2 according to Equation 3.18. It was observed that the proposed multiplier reduces the error in the output by 1.69% as compared to the mitchell multiplier. This reduction in error maintains the accuracy of DNN on a hardware platform as compared to the software.

3.5.2.6 Operation of DNN Classifier

The operational principle of the proposed DNN classifier is presented in Figure 3.7. Once the descriptor processing block computes the descriptors, the top FSM is activated. This further enables

the FSM of the first DNN layer, which reads the input from the descriptor block and weights of the first neuron. Next, these weights are sequentially multiplied by the inputs. Once the multiplication and accumulation (MAC) of weights and input are completed, a bias is added to the MAC output. Further, this output is passed through a multiplexer (MUX) of ReLU to obtain an output of the first neuron. The same procedure is repeated for all the neurons of a given layer. The output of neurons acts as input to the second layer, which is fed through a serial-in parallel-out (SIPO) register. Once the operation of layer-1 is completed, the top FSM passes the control to the FSM of the second layer, which repeats the same process. The MAC unit and MUX are reutilized in successive layers, which minimizes area and power. Additionally, folding the hardware mentioned above also optimizes leakage power. Since we use a pre-trained network for the hardware implementation, a simple comparator is employed as an activation function in the final output layer instead of a sigmoid function, which optimizes resources without causing classification errors.

3.5.3 Experimental Results of Proposed Design

This section presents a detailed discussion of the outcome of the proposed design and its validation using various testcases. The initial algorithm verification and model optimization is performed using Python. Further, the complete design of CoAP is implemented using Verilog-HDL, whose performance was tested using several testbenches created using ECG excerpts. This subsection first presents the performance metrics of CoAP. Further, it depicts the outcome of FPGA and ASIC implementations of CoAP. Next, a detailed description of the comparison of the proposed design with the contemporary state-of-the-art designs is given. Finally, we summarize the findings and novelties of the proposed design in this section.

3.5.3.1 Dataset Description

For detecting VA, the features extracted from the ECG segment before the onset of VT/VF is utilized as training and test data. It is imperative to predict VA rather than detect it to save lives. Therefore, analyzing the feature variations of the ECG segments preceding VA beats is necessary. To incorporate this methodology, we consider ECG segments from the MIT-DB dataset [41]. The MIT-BIH arrhythmia database [88] consists of 48 two-lead ECG recordings, and each is of half-hour duration and is sampled at $360Hz$. The dataset has $108,655$ beats, labeled by expert cardiologists into different types of arrhythmia as per the American National Standard (ANSI/AAMI EC57:1998) [89].

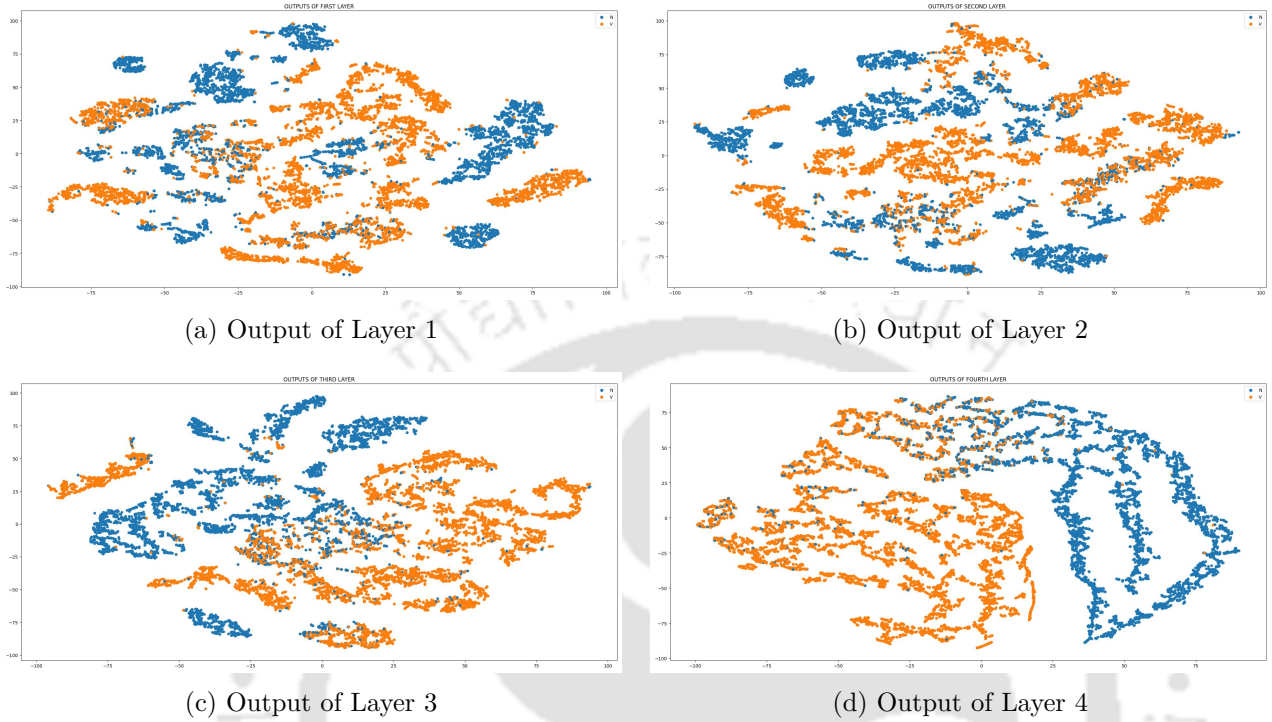
MIT-BIH database has two ECG lead recordings; modified limb lead II (MLII) and modified lead (V1). Only MLII is used for the ECG classification because of the prominent QRS complexes in the proposed work. Later, the ECG data is downsampled to 250 Hz to optimize the number of samples in an ECG segment. Note that downsampling the data reduces the number of samples per ECG segment without affecting the overall performance of the classifier.

The ECG records taken from the MIT-DB database are processed in the following manner. First, the VT/VF beats are identified. Next, ECG segments of three seconds are considered before this onset of VT/VF beats. The 3-second windows are extended to different time intervals, "T". The data between this "T" and the onset of VT/VF beat is considered abnormal (Class A). The data before this "T" interval is considered normal (Class N). Later, this data is divided into two groups: DT1 (75% of total data), the training set, and DT2 (25% of total data), the test set. Analyzing the ECG signal in such a way enabled the classifier to predict the VT/VF event 15 minutes (T) before its occurrence.

3.5.3.2 Validation of the Proposed Methodology

Among the performance metrics used for medical diagnosis, sensitivity (Se) and specificity (Sp) are the crucial measures of diagnostic accuracy. High sensitivity enables clinicians to rule out disease, whereas a high specificity of a test helps identify a particular disease in the patients [103]. Therefore, sensitivity (Se) and specificity (Sp) are employed to measure the diagnostic ability of the classifier in Class N and Class V. Further, the accuracy of CoAP is also calculated to estimate overall system performance.

The performance of CoAP is tested on different values of "T", i.e, from five minutes before the onset to 30 minutes before the onset of VA beats. However, we observe that CoAP can predict arrhythmia 15 *min* before its occurrence with 91.6% accuracy, 91.94% sensitivity, and 91.42% specificity. Considering "T" more than 15 minutes leads to very low accuracy. As it is essential for a wearable device to generate minimal false alarms, DNN is implemented on hardware, which is trained on the dataset taken 15 minutes before the onset of VA.



(a) Output of Layer 1

(b) Output of Layer 2

(c) Output of Layer 3

(d) Output of Layer 4

Figure 3.9: TSNE Plots of Different Layers of DNN

TSNE plots are also presented to illustrate the functioning of hidden nodes. The t-distributed stochastic neighbour embedding (t-SNE) is a nonlinear dimension reduction method utilized to visualize high-dimensional data on a lower-dimensional (two-three) space. The output of the first, second, and third hidden layers of the DNN are extracted from the input vectors. We have applied t-SNE on the 6-dimensional input vectors and the hidden layers. Figure 3.9 presents the results for Class N and V. It can be observed from Figure 3.9 that there are no separate clusters for the feature vectors (Fig. 3.9a). As we progressively move forward to the hidden layers (Figs. 3.9b-3.9c), the clusters become visible. However, classes N and V still have multiple small clusters for the first two hidden layers. This shows that nonlinear operations are needed for a complete classification. However, it becomes evident that in the final layer (Fig. 3.9d), large clusters of both classes are clearly visible. This showcases the discriminative nature of DNN as we move from the bottom to the top layer. The hidden layers have disentangled the classes from the considered ECG excerpts, making the final layer representation distinctive.

3.5.3.3 FPGA Implementation

The proposed architecture is first implemented on $XilinxVirtex-7$ FPGA for functional verification. Table 3.10 presents the resource utilization of the proposed design. It is observed from Table 3.10 that our design utilizes 0.69% of total available resources, making it resource-efficient.

Resources	Total Available	Resources Utilized	Percentage Utilization(%)
Slice LUT	303600	5257	1.73
Slice REG	607200	2354	1.55
F7 MUX	151800	206	0.13
F8 MUX	75900	64	0.08
DSP	2800	2	0.07
BRAM	1030	1	0.09
IOB	600	21	0.35
BUFGCTRL	32	1	3.125
Total Resources	1142962	7906	0.69

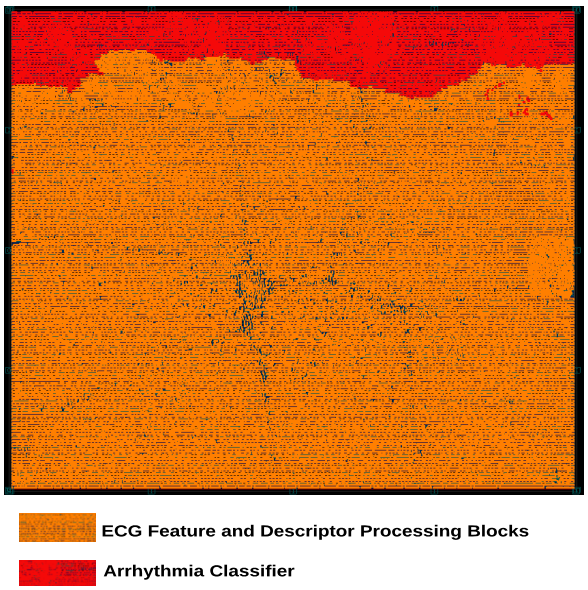
Table 3.10: FPGA Resource Utilization

3.5.3.4 ASIC Implementation

CoAP is implemented as an ASIC using $180nm$ Bulk CMOS Technology available with SCL Chandigarh as shown in figure 3.10. The design is synthesized using Synopsys DC compiler, and placement and routing are performed employing Synopsys IC compiler. It is observed that the proposed design can operate at the minimum frequency of $12.5kHz$ to perform beat-by-beat classification in real-time, consuming $4.69\mu W$ power. Therefore, despite the ability of the proposed architecture to operate at a higher clock frequency till $1MHz$, it is sufficient to choose $12.5kHz$ as an operating frequency to fulfill the low power requirement.

3.5.3.5 Comparison with State-of-the-Art

Table 3.11 presents a comparison of our proposed classifier with well-known state-of-the-art arrhythmia classifiers on the hardware as well as the software platforms. It is observed from Table 3.11 that CoAP exhibits lower accuracy than the methods reported in [70, 75, 78]. However, the accuracy reported in [70, 75, 78] is for the detection of arrhythmia and not prediction, and an alert is sent to an individual only when arrhythmia has occurred. Nevertheless, our proposed classifier can predict the arrhythmia beats 15 *min* before their occurrence with an accuracy of 91.6%. It is evident from Table 3.11 that the architecture of CoAP has better or comparable performance compared to [99] and [76], which also report the prediction of VT/VF over detection. It can be observed that CoAP



CMOS Process	SCL 180nm
Area	1.8 mm^2
Voltage	1.98V
Frequency	12.5kHz
Dynamic Power	1.8323 μW
Static Power	2.8594 μW

Figure 3.10: The Layout Photograph of CoAP Architecture and its Specifications

has higher accuracy than [76]. On the other hand, CoAP has higher specificity than [99], but it is slightly less sensitive than the design reported in [99]. However, [99] is implemented only on software by utilizing complex input feature vectors and decision trees as a classifier. Although the decision tree implementation is not critical in hardware, the complex input features, which lead to a slight increase in the sensitivity considered in [99], require complete ECG feature delineation. These features also need to find the boundaries of $P-QRS-T$ waves. Therefore, the hardware requirements of [99] would be higher if it is implemented as an ASIC. Therefore, it can be stated that the proposed design has the highest performance metrics than the other state-of-the-art methods. On the other hand, the power requirement of the proposed design is less than [75] and [78] but more than [70] and [76]. The designs reported in [70] and [76] are implemented on the lower technology nodes with less operating voltage. Further, the design reported in [70] can only detect the arrhythmia beats, and therefore, it utilizes simple R peak based features for the classification, which have lower hardware requirements.

Parameter	TBIoCAS 2020 [75]	DATE 2018 [78]	PLOS ONE 2020 [99]	TVLSI 2016 [76]	Proposed
Approach	Detection	Detection	Prediction	Prediction	Prediction
Database	MIT-BIH	MIT-BIH	NSRDB, VFDB	MIT-BIH NSRDB	MIT-BIH
Classifier	ANN	Threshold Based	Decision Trees	Bayes Naive	DNN
Accuracy	99.68%	97.02%	NA	86%	91.61%
Sensitivity	NA	94.64%	95%	NA	91.94%
Specificity	NA	99.41%	90%	NA	91.42%
Platform	ASIC	ASIC	Software	ASIC	ASIC
Technology Node	180nm	180nm	NA	65nm	180nm
Voltage (V)	1.8	1.8	NA	1V	1.98V
Frequency (Hz)	25M	1k	NA	10k	12.5kHz
Area (mm^2)	0.9246	NA	NA	0.112	1.8
Power (W)	13.34	5.04	NA	2.78	4.69

Table 3.11: Comparison with State-of-the-Art Methods

On the other hand, the design proposed in [76] employs a naive bayes classifier and interval-based input features. The computation of these features is easier than the statistical features used in the proposed work. However, the accuracy of the design proposed in [76] is less than CoAP, making it less suitable for wearable devices. Therefore, it can be inferred from Table 3.11 that CoAP is most suitable for low power wearable devices to predict arrhythmia as it has higher accuracy and low power consumption.

3.6 Summary and Scope of Present Work

This chapter presents two low power VLSI architectures. First one is a low power, patient-independent, generic and adaptive ECG cardiac arrhythmia co-processor incorporating a novel simplified reduced register shift multiplier processor to classify all five types of ECG beats as per AAMI standards. It employs a simple ECG pre-processing method and uses a complete ECG beat as an input to DNN, which allows classifier to capture all the morphological changes in the arrhythmic beats. The utilization of a simplified MAC unit, resource sharing and optimization of *softmax* function make our co-processor area and power efficient. It consumes 8.75μW at 12 kHz with 91.6% accuracy when realized with 180nm CMOS technology. Secondly, a low power ECG co-processor for predicting cardiac arrhythmia is proposed, which can predict the occurrence of ventricular arrhythmia and ventricular fibrillation 15 min before its occurrence with an accuracy of 91.6%. The proposed design utilizes an optimal feature set and novel DNN classifier architecture, enabling it to operate at a minimum frequency of 12.5kHz, consuming 4.69μW of power. The low power operations of both the designs make them suitable for wearable device applications.

4

Detection of Myocardial Infarction

Contents

Objective

Myocardial Infarction (MI) is a critical heart abnormality causing millions of fatalities worldwide every year. MI progress in three stages based on its severity causing several changes in an Electrocardiogram (ECG) signal. It is very critical to capture these variations, which requires continuous monitoring of the ECG signal of the patient. Therefore, it becomes imperative to develop a low power VLSI architecture to address the prognosis of MI. Therefore, the objective of this chapter is to design an area and power efficient design of a five stage classifier which detects the progression of various stages of MI using ECG beats in real time.

4.1 Introduction

Amidst all the diseases, cardiovascular diseases (CVD) have scourged the world with millions of deaths every year. The leading cause of death due to cardiovascular diseases globally is Myocardial Infarction (MI). It is estimated that MI causes approximately nine million fatalities yearly, which is projected to increase up to 12 million by 2030 [104]. Myocardial infarction occurs due to complete blockage of a coronary artery, which spawns due to rupture of atherosclerotic plaque [105]. The rupture causes deprivation of oxygen to the heart muscles, making them necrotic. MI progresses in three stages, namely early MI (EMI), acute MI (AMI) and chronic MI (CMI). In the early MI stage, if fibrinolytic therapy (FiBT) procedure is performed to reperfuse the blocked artery within 40 minutes, the damage to myocardial tissues can be reversed. In AMI, arteries' obstruction gets prolonged for $12 - 24$ hours; necrosis starts leading to irreversible damage to the tissues. Partial restoration of the myocardium can be done with the help of percutaneous coronary intervention (PCI) and coronary bypass surgery (CABG) [106]. However, in CMI, irreversible damage occurs to the affected myocardium when occlusion remains for more than 24 hours leading to mortality. Figure 4.1 illustrates progression of Myocardial Infarction. These specific clinical procedures can be enabled for a patient if an on-time diagnosis of MI is performed. Therefore, there arises a need for a time-critical response system for MI, which can alert a person before any life-threatening condition.

ECG is widely used by clinicians worldwide to record the electrical conduction of the heart muscles for the diagnosis of MI. ECG has three primary features, P wave, QRS complex, and T wave [11] which are considered for manual and automatic detection of cardiac abnormalities. During the progression of MI, ECG features undergo variations that aids cardiologists to diagnose myocardial infarction

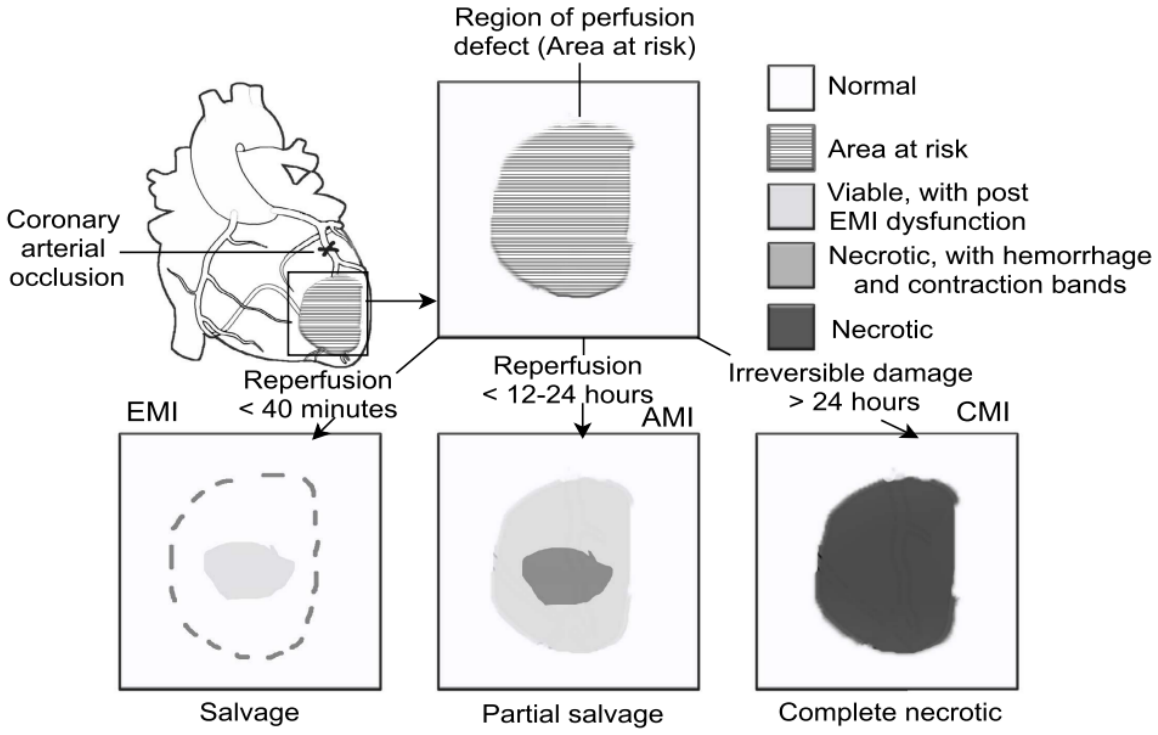


Figure 4.1: A Schematic Illustration of the Progression of Myocardial Infarction [4]

properly. In early MI, abnormal T wave behaviour is noticed, such as peaking and inversion, and slight ST elevation occurs. With progression, severe ST segment elevations and T wave inversions are observed during AMI. In the final stage of necrosis, pathological Q waves start to manifests. These feature variations of ECG signals are widely used by professionals for the prognosis of Myocardial Infarction. However, manual examinations of these variations are very typical and require a high level of expertise. In several cases of MI, a person does not feel any symptom before reaching to a critical condition [107]. This is because of the common symptoms of MI progression, such as, feeling uneasy, dizziness, and light headedness are often ignored by many people. Further, it may also happen that ECG performed in the initial stage of MI might not be diagnostic, as the evolution of ECG variations may vary considerably from person to person. Therefore, it becomes important to take serial electrocardiograms in certain intervals to get an assured diagosis.

Several approaches have been introduced that focus on detecting abnormal heartbeats of MI patients from normal beats of a healthy individual with the help of ECG variations. However, the existing methods neither include non-MI CVD ECG beats in the classification dataset nor considers

classifying the progression stages of MI. Moreover, the methods that address MI stage-wise classification are implemented using software platforms and utilize complex deep learning models, making them unsuitable for hardware implementation. Hence, a VLSI architecture, which can be integrated as a wearable device for continuous monitoring of ECG, would aid in early stage classification of MI avoiding fatalities. These devices can alert an individual in the early MI stages, which can reduce mortality rates due to irreversible damage that occurs in MI. Further, these devices can capture and store progression of morphological changes of an ECG signal, which can assist clinicians later. However, due to their portable nature, these devices have to operate with restricted power and area resources. Considering these requirements, low power and area optimal VLSI sub-system is proposed in this chapter MI stages classification for wearable device application.

4.2 Prior Works

There are a number of state-of-the-art methods available, which can classify ECG beats into normal and ischemic on the software platforms suitable for in-hospital setups [108], [109]. Despite the high performance metrics of these methods, their complex methodologies makes them unsuitable for the low power VLSI implementations. Moreover, very few methods are available, which can detect MI on the hardware platforms. For instance, a feature extraction algorithm proposed in [22] is implemented on ASIC, which transmits data to a smartphone through bluetooth. Thereafter, MI is detected using an application installed on the smartphone. However, transmitting data via bluetooth is expensive in terms of power, as it drains battery of the wearable devices [110]. An ASIC implementation is proposed in [111], which can classify ECG beats as MI or non-MI. This design utilizes only ST segment as an indicator for MI detection. However, in the real world scenario, besides ST segment, T wave morphology and Q wave peakings should also be considered for the accurate diagnosis of MI. Also, despite low power implementation and capability to perform on-chip classification of ECG beats into MI and non-MI, it could not perform classification of various stages of MI progression.

As evident from the literature, none of the reported methods can classify stages of MI progression, which helps in alerting an individual before reaching to a critical condition. Therefore, in this chapter, first time a low power VLSI architecture *MInSC* (Myocardial Infarction Stages Classifier) is proposed, which can classify ECG beats into five distinct classes, viz., Normal, EMI, AMI, CMI and Non-MI. Its on-chip classification reduces power consumption as well as execution time due to less frequent

communication between cloud and the wearable devices.

4.3 Proposed Design of MInSC

The proposed work describes a novel VLSI architecture (MInSC) for classifying the progression of Myocardial Infarction, whose details are explained as follows. The MInSC works in three stages. In the first stage, features of ECG ($P-QRS-T$ waves) are extracted using a novel optimized ECG feature extraction algorithm. In the second stage, these delineated ECG features are used to calculate "markers" for the stages of MI. It is to mention that the characteristics features required for the classification of MI are termed in this chapter as *Markers*. In the third stage, these markers are utilized by classifier to detect various stages of MI.

4.3.1 ECG Delineation

Figure 4.2 illustrates a complete overview of the proposed MI classification architecture. As shown in Figure 4.2, the first step is to process an ECG signal and delineate its features. As we know, ECG has three primary features, P wave, QRS complex, and T wave [11], which are considered for manual and automatic detection of cardiac abnormalities. It is important to mention, in diagnosing MI, correct delineation of R peak and isoelectric line is crucial as all other features necessary for the classification are calculated with respect to them. In this work, we utilize an optimal ECG delineation algorithm, which is previously reported by us in [100] and described in Chapter 2. Moreover, this ECG delineation method is further optimsed in the chapter 3 by making it more adaptive to the varying heartrate. However, in chapter 3, estimation of boundaries of P-QRS-T peaks are not required. Therefore, in this work the boundary detection modules are also utilized for extracting onset and offset of previously detected peaks as described in chapter 2.

4.3.2 Markers for Classifier

During the progression of MI, ECG features undergo variations that aid cardiologists to diagnose myocardial infarction properly. In *EMI*, abnormal T wave behaviour is noticed, such as peaking and inversion etc., and slight ST elevation occurs. With the progression of MI, severe ST segment elevations and T wave inversions are observed during *AMI*. In *CMI*, pathological Q waves start manifesting, which may lead to fatality, if it is not treated on time. Thus, the proposed design utilizes a set of four ECG *markers*, namely, T wave morphology, ST-segment deviation, abnormal Q peaks and an

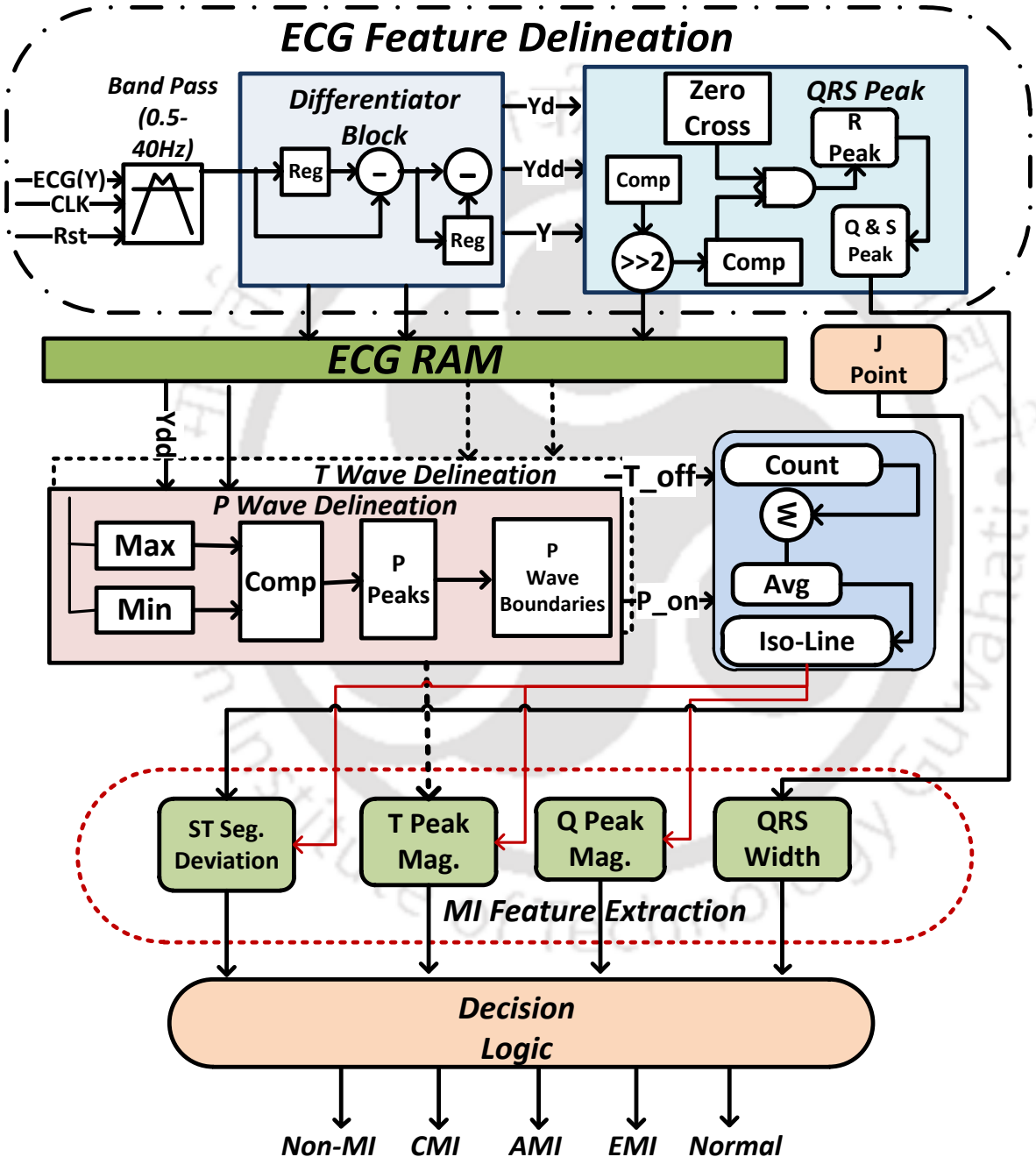


Figure 4.2: Proposed Architecture for ECG Delineation and MI Classificationize

additional feature, QRS width which helps cardiologist to classify various stages of infarction [11]. It is to mention that QRS width is considered to perform differential diagnosis, as explained in [111]. Therefore, first, QRS width is estimated as a difference between QRS onset and offset just after their delineation and, later, isoelectric line is calculated to find rest of the markers.

In ECG, an isoelectric line can either be determined using a window between the offset of P_i and the onset of QRS_i or as an interval between the offset of T_{i-1} and the onset of P_i. However, certain cardiac issues like pericarditis, a condition which can clinically mimic ischemia, may lead to depression in the interval between the offset of P_i and the onset of QRS_i [11]. Thus, it is clinically more accurate to take an interval between the offset of T_i and the onset of P_{i+1} as an isoelectric line, and the same is considered in the proposed algorithm. Isoelectric line is estimated as shown in Equation 4.1.

$$Iso\_line = \frac{\sum_{T\_off_i}^{P\_on\_i+1}(ECG_j)}{(P\_on_{i+1} - T\_off_i)} \tag{4.1}$$

As evident from Equation 4.1, an adder and a divider circuit are required for the correct estimation of isolectric line. However, as we know, implementation of a divider is very expensive in the hardware. Therefore, as shown in Figure 4.3, in this chapter we propose an optimized divider based on the mutliplicative division [112] to effectively estimate isoelectric line in the hardware with minimal resources.The proposed divider is explained as follows.

Proposed Divider: Let "a" and "b" be the dividend and the divisor, respectively, which can be represented in the normal form as shown in Equation 4.2.

$$a = c_1 \times 2^{k_1} \; and \; b = c_2 \times 2^{k_2} \tag{4.2}$$

where, $c_1, c_2 \in (0.5, 1)$. The quotient of $\frac{a}{b}$ can be presented as $\frac{a}{b} = (c_1 \times \frac{1}{c_2})(2^{(k_1-k_2)})$. The first step involves finding of the leading one using a priority encoder and to calculate k_1 and k_2. Next, a shift and compare operation is employed to estimate c_1 and c_2. Further, $\frac{1}{C_2}$ is estimated using pre-calculated values stored in a look-up table (LUT). Finally, c_1 and $\frac{1}{c_2}$ are multiplied and shifted by $k_1 - k_2$ to get the final quotient. Once the isoelectric line is estimated, ST segment block gets enabled and ST segment deviation is computed using the difference of the voltages between ST segment and isoelectric line. As per the RR interval, ST segment is considered as either $J + 60ms$ and $J + 80ms$ [11]. It is observed that T waves start to peak during early stages of MI and later gets inverted. Thus, in parallel to ST segment deviation, T wave magnitude is also quantified with respect to isoelectric line. Further, at the onset of necrosis, pathological Q waves may reach more than 25%

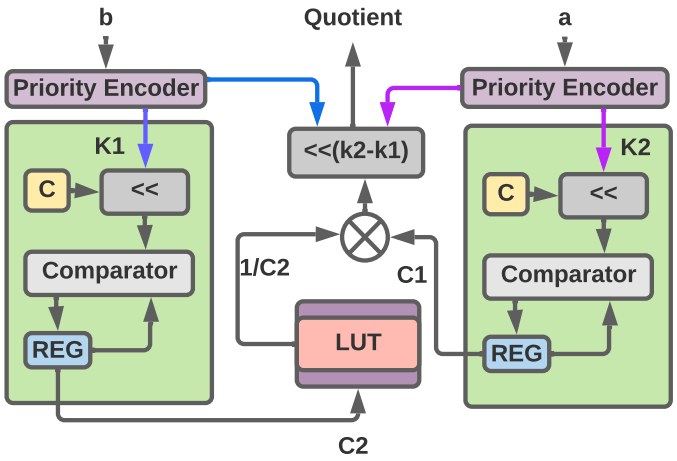


Figure 4.3: Circuit of the Proposed Multiplicative Divider

of R peaks. Therefore, once an isoelectric line is determined, Q peaks magnitude is also calculated, which is included in the classification feature set as shown in Figure 4.2.

4.3.3 Classification Module

Based on the extracted *markers*, an *if-else* rule based myocardial infarction stages classifier is proposed in this chapter, which allows the classification of ECG beats into five classes without using any complex machine learning algorithm. This classifier is designed employing behaviour modelling in the hardware as explained below. In the first stage, it removes the beats, which belong to non-ischemic diseases. There are various diseases, such as, left ventricular hypertrophy, hyperkalemia etc., which may have similar characteristics as MI. However, they show abnormal variations in QRS complex as well, which are not significant during MI stages. Therefore, if the extracted QRS complex width is in the normal interval of $60 - 100ms$ [11], then the beat is sent to the next stage for an analysis; else, if the QRS width is more than $100ms$, then it is considered as non-ischemic.

During early MI, changes in T waves start occuring along with slight changes in the ST segment. If the magnitude of T wave with respect to isoelectric line is greater than $0.5mV$, or if the T waves are inverted, then MInSC considers a beat as EMI, which is also known as Ischemia. With the progression of MI, necrosis starts causing pathological Q waves in the AMI stage and originates considerable ST elevations and T wave. Therefore, if T peaks are inverted and ST segment elevation is more than $0.07mV$, a beat is considered as AMI. Also, if T waves are normal, but ST segment elevation is greater than $0.1mV$, then also a beat is considered as AMI. Later, if Q waves are greater than 25% of R peak's magnitude, and T wave is $< 0.5mV$, and ST segment deviation is $< 0.07mV$, then a beat is classified

as CMI. If none of the above criteria are met, then a beat is classified as normal.

4.4 Experimental Results of Proposed Design

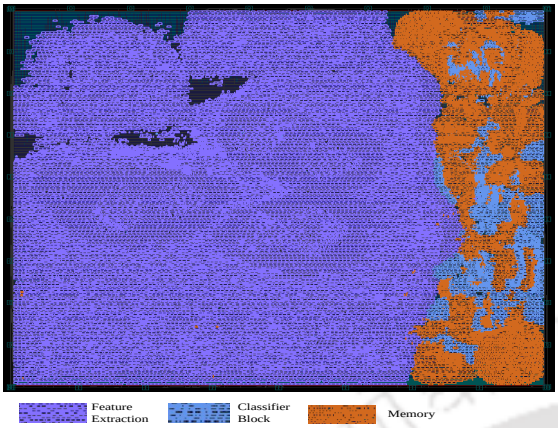
In this section, a detailed discussion is presented on the outcome of the proposed design, when it is validated with various ECG excerpts. ECG records of EMI patients are taken from STAFF III database [113]. The patients in PTB dataset [41] with infarct age less than $24Hrs$ are considered for AMI, and patients with infarct age greater than $24Hrs$ are marked for CMI [114]. Both normal and non-MI ECG excerpts are taken from PTB-XL dataset [115]. The ECG excerpts obtained for the verification of our proposed design are first downsampled at $250Hz$ and then employed for the classification.

The complete design is implemented using Verilog HDL and is synthesized on $XilinxVirtex-7$ FPGA board [116] to validate its performance by employing multiple ECG testcases. The resource utilization of the proposed architecture on the FPGA board is shown in Table 4.1, which exhibits that 0.516% of the total available resources are being utilized by the proposed design.

Resource	**Available**	**Utilization**	**Percentage Utilization**(%)
LUT	303600	3447	1.13
SLICE-REG	607200	1986	0.32
MUX	151800	33	0.0217
DSP	2800	3	0.11
IO's	600	39	6.5
BRAM	1030	1.50	0.15
BUFGCTRL	32	1	3.12

Table 4.1: FPGA Implementation of Complete Myocardial Ischemia Classifier Architecture

The proposed architecture is implemented as an ASIC using SCL $180nm$ Bulk CMOS PDKs and Synopsys design compiler and IC compiler tools. The complete design utilizes $1.38mm^2$ area and consumes $5.12\mu W$ power at $1.98V$ and $8kHz$. Since we utilize ECG excerpts having a sampling frequency of $250Hz$, it is observed that an operating frequency of $8khz$ is suitable for the proposed design to analyze ECG beats in the real-time. Table 4.2 presents the comparision of MInSC with other state-of-the-art methods. It can be seen from Table 4.2 that the average sensitivity exhibited by MInSC is 86.18%, which is higher than [117] and [118], but less than [108, 109]. Further, the specificity of our proposed design is 96.5%, which is more than [108,117,118], but less than [109]. This is because of these



CMOS Process	SCL 180nm
Area	1.38 mm^2
Voltage	1.98 V
Frequency	8 kHz
Dynamic Power	3.005 μW
Static Power	2.1163 μW

Figure 4.4: The Layout Photograph of MInSC Architecture and its Specifications

methods are implemented on the software for two class classification only whereas proposed design performs five class classification. Further, these methods employ complex deep learning algorithms, such as, convolutional neural networks etc. The high performance metrics of the literature makes other state-of-the-art methods suitable for in-hospital setups. However, wearable devices has strict area and power constraints, therefore, despite of remarkable classification performance, the computationally complex methodologies, viz. [108, 109, 117, 118], make them unsuitable for the low power and area constrained wearable devices.

Further, the design proposed in [111] is implemented on the hardware and has higher sensitivity performance metrics as well as low area and power requirements than our proposed design. However, our design is more suitable for wearable devices as compared to [111] due to the following reasons. First, MInsc can classify progression stages of MI whereas [111] classifies ECG beats into MI and Non-MI classes. Second, [111] utilizes only two features, ST segment and QRS width, to classify MI, whereas the proposed design analyzes ECG signals using the markers(as described before) utilized by medical professionals to perfom diagnosis. Third, design in [111] is tested on ECG excerpts with only ST segment variation whereas MInsc is tested on a large variation of data excerpts which incorporates all the possible variations that may occur during MI making our design robust. Due to incporporation of larger number of feature set and design of multiclass classifier our design has more hardware requirements than [111]. However, these features are very important for real world classification of myocaridal ischemia/infarction. Thus, it can be inferred that the proposed design is more suitable for wearable device applications as compared to the state-of-the-art methods.

Parameter	[108]	[109]	[117]	[118]	[111]	Proposed
Implementation	Software	Software	Software	Software	ASIC	ASIC
Database	PTB, PTB-XL	PTB	PTB	STAFF III	Long Term ST	PTB, PTB-XL, STAFF III
Sensitivity(%)	91.59	99.97	85.33	83.3	96.43	EMI=84.42%
						AMI=85.6%
						CMI=80.2%
						Non-MI=88.22%
						Normal=92.46%
Specificity(%)	85.89	99.54	84.09	91.7	96.88	EMI=97.3%
						AMI=96.7%
						CMI=97%
						Non-MI=95%
						Normal=96.3%
Operating Frequency	250-1000Hz	250 Hz	250 Hz	1000 Hz	250 Hz	8 kHz
Classes	2	2	2	2	2	5
Voltage	NA	NA	NA	NA	1.8V	1.98V
Power	NA	NA	NA	NA	0.274 μ W	5.12 μ W
Area	NA	NA	NA	NA	0.137 mm^2	1.38 mm^2

Table 4.2: Performance Comparison of MI Classification with Prior Work

4.5 Summary and Scope of Present Work

This chapter presents an edge enabled area and power efficient VLSI architecture of MInSC to classify progression MI stages from healthy and non-MI beats. MInsc has following novelties over previous reported method. It is the first ASIC which can classiy the different stages of MI from non-MI and normal beats. Second, it employs a simple and optimized ECG feature extraction algorithm. Third, it uses a detailed set of markers as per medical standards to classify MI. It also proposes an optimized divider for low power estimation of markers on hardware. Finally, it employs simple if-else based algorithm which is hardware efficient and suitable for use in wearable healthcare devices.Thus, it can be stated, MInSC may help to reduce mortality rate due to MI enabling patients to get preventive treatment on time. The low power and area requirements and multiclass classification capability of the proposed design make it suitable to be used in wearable devices.

5

ECG Data Compression

Contents

Objective

Continuous monitoring of the electrical activity of heart signals using wearable Internet of Healthcare Things (IoHT) devices plays a crucial role in decreasing mortality rates. In the previous chapters different architectures were proposed which can detect different CVD using a single lead ECG data. However, in general, a 12-lead ECG is considered over extended periods of approximately 12-24 hours by cardiologists for more accurate diagnosis of cardiac diseases. This continuous monitoring using an electrocardiogram generates huge clinical data. Moreover, these devices are constrained in terms of on-chip storage, data transmission capacity, and power. Thus, handling a large amount of data is difficult with these devices, making it necessary to compress this data for storage and transmission. Lossless data compression solves this problem, ensuring that no relevant physiological/clinical information is lost in the compression process. Therefore, low-power, resource-efficient, and lossless VLSI architectures are proposed in this chapter to compress multi-channel ECG data.

5.1 Introduction

Cardiac ailments are one of the biggest causes of mortality, causing millions of deaths worldwide every year. Electrocardiogram signal (ECG) manifests the heart's electrical activity and is widely used by clinicians to detect these heart abnormalities. However, in general, a 12-lead ECG is considered over extended periods of approximately 12-24 hours by cardiologists for the diagnosis of cardiac diseases. In the IoHT, ECG data is acquired using sensors and then processed on a chip. Later, the data and the probable diagnosis is sent to cloud storage to enable telehealth monitoring by the doctors in a hospital. Furthermore, doctors analyze the results of different tests, including ECG data, before making a diagnosis. To improve diagnostic efficiency and reduce hospital visits, it is imperative to design low-power VLSI architectures that enable multi-lead ECG acquisition at home and transmit the data to cardiologists through a secure cloud network. By implementing such architectures, healthcare providers can enhance diagnostic accuracy and provide early detection and treatment of various cardiovascular diseases. The development of such devices could transform cardiac care, making it more accessible, affordable, and convenient for patients, ultimately improving their quality of life. As we know, transmitting large amounts of ECG data can be power-hungry, which is why it is imperative to have low-power compression architectures for data transmission. This can be achieved through efficient compression techniques that can reduce the amount of data sent to the cloud for analysis.

The data compression schemes can be of two types: lossy and lossless. Lossy data compression results in permanent loss of information during the processing steps, whereas lossless compression techniques ensure that no information loss occurs during the processing steps in the compression [119], [120]. As we know, biomedical signals contain critical pathological details that need to be preserved during the compression process; therefore, lossless compression is preferred over lossy techniques. Although users can rely on lossy techniques for ECG compression, lossless compression is needed for better clinical accuracy for many reasons. Further, worldwide medical regulations also advocate for the lossless techniques of ECG data compression. Also, with advanced diagnostics, ECG variations that were earlier treated as random heart activity can now be of great medical relevance. Utilizing lossy algorithms may lead to removing specific data segments that might be irrelevant to a clinician. However, advanced machine learning algorithms might get relevant information in these data segments, improving the diagnosis. Further, the reconstruction error due to lossy techniques may lead to misinterpretation of crucial ECG features, such as QRS complex amplitudes, ST segment slopes, and T and P wave amplitudes. Therefore, ECG signals require lossless compression so that the clinical information of these signals is not lost during the process. An exact copy of the data can be reproduced with the help of compressed data whenever required for the diagnosis. The compression systems are designed to have a large compression ratio with negligible error in the reconstructed data. Further, it is essential to mention that this data transmission to the cloud/remote location is a power-consuming process. As we know, IoHT-enabled wearable devices are constrained in terms of memory and power capacity. Thus, it becomes imperative to compress the ECG data to cater to the wearable devices' low power and area requirements. This chapter proposes low-power VLSI architectures for ECG data compression and decompression, which can be employed at the sender and receiver ends. Multi-lead data compression and decompression techniques for ECG signals are presented in this chapter. Further, the hardware implementation is showcased, aiming to achieve low power, small area, and real-time data compression and decompression so that it can be employed as a sub-system in the wearable IoHT devices.

5.2 Prior Works

Several state-of-the-art methods are available that target lossless ECG compression. A conventional ECG compression system is divided into prediction and entropy coding modules. Linear pre-

diction methods, such as discrete pulse code modulation (DPCM) [121] and forward prediction-based approach [122] to reduce signal prediction errors, are the most common prediction techniques. Further, entropy coding is essential to lossless ECG compression algorithm. Some previous works reported in the literature explore various entropy encoding schemes. The authors in [123], [122], [124] employ Huffman and prediction error coding schemes as entropy coding techniques in their algorithm. Further, [125] present a detailed review of different lossless methods. Variation separation and zero run-length coding schemes are explored for compression by Chen et al. [126]. The proposed design is also implemented on FPGA.

Dora et al. [127] propose another FPGA-based implementation, which uses the modified convolution and discrete wavelet transform. Another VLSI architecture is proposed in [128], which employs a new adaptive rending predictor and a two-stage entropy encoder. This ECG encoder utilizes simple arithmetic units for the realization. Further, an adaptive predictor based on fuzzy decision control and Huffman coding is employed in the design proposed by Luo et al. in [129]. Moreover, N. Attarmoghaddam [130] presents a design based on compressed sensing, illustrating a VLSI architecture using clock gating to reduce power consumption. A resource-efficient and low-power architecture of a compression algorithm is reported in [131], which utilizes shifting operations to replace different arithmetic operations. A lossless compression hardware design is depicted in [132] for multi-channel ECG based on multi-channel adaptive linear prediction and Golomb Rice coding (GRC) schemes, having low resource utilization and power consumption. Note that most of the designs in the literature propose only single lead compression. Furthermore, the methods stated above can be optimized by utilizing efficient hardware design techniques of prediction and encoding.

Most of the methods presented in the literature employ complex Golomb Rice and Huffman encoding schemes, having higher memory and hardware resource utilization. As we know, optimizing hardware would minimize the area and power requirements of the compression and decompression systems. Therefore, this chapter presents an optimized lossless ECG compression algorithm and its low-power and area-efficient VLSI architecture compared to the previous state-of-the-art methods. Based on the design metrics, it can be stated that our proposed design can be a good candidate for wearable healthcare IoHT applications.

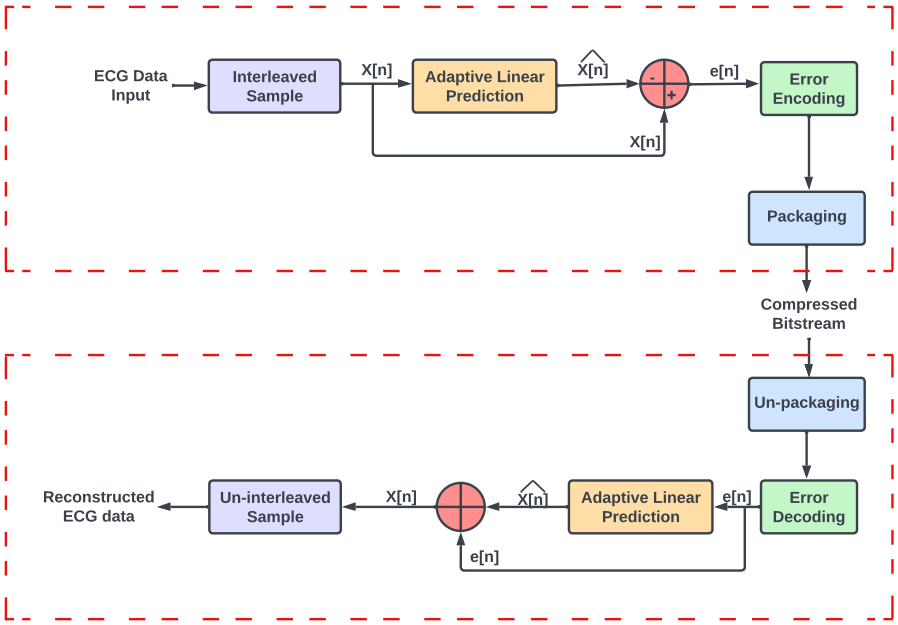


Figure 5.1: Basic block diagram of Compression/Decompression system

5.3 Proposed Methodology

This section presents the hardware design methodology of the proposed architecture for lossless data compression techniques for multi-channel ECG.

Figure 5.1 shows a basic block diagram of the proposed compression system. The system uses adaptive linear prediction, error calculation, error encoding, and packaging to perform compression. The first step is an adaptive linear prediction, which outputs an approximation of the current sample with a minimal error in prediction, followed by error calculation. A good prediction of the current sample leads to a small error value, which can be encoded with the help of less number of bits using variable-length entropy encoding. The encoded data are then packed to generate the final compressed bitstream. Packaging facilitates easy and faithful decompression of compressed bitstream. Here we propose simple encoding and packaging schemes, which require less hardware and memory resources. Their novelties are depicted as follows:

- **Serial Architecture:** An optimized technique to interleave the samples of multi-lead ECG is employed, eliminating the need for parallel architecture and saving area and power resources.
- **Modified adaptive linear prediction block:** A multiplier and divider-less adaptive linear prediction block is proposed. The architecture is designed to optimize all the mulitplication

and division operations as shift operations, making it area and power efficient. The same architecture is utilized as a sub-system for compression and decompression.

- **Simple error encoding scheme:** In the realization of the proposed method, a simple linear encoding scheme is employed, which is implemented using comparators. Further, optimization to the original samples is performed so that the error obtained can be encoded using fewer bits.
- **Novel packaging scheme for encoded error data:** The proposed work employs a simple variable length entropy encoding scheme, which involves selecting a fixed number of bits as per the range of encoded error. The packed data is generated by concatenating the encoding length and selected bits of the predicted error. This technique does not require complex computaions, making it hardware efficient and maintaining a high compression ratio.

5.3.1 Proposed Algorithm

Conventionally, ECG is recorded as 12 leads of continuous data. However, the electrical information of these 12 leads can be estimated mathematically with the help of only four reference leads [133], i.e., I, II, $V1$, and $V5$, as shown in the equations below.

$$III = II - I \tag{5.1}$$

$$aVR = \frac{-(I + II)}{2} \tag{5.2}$$

$$aVF = \frac{(II + III)}{2} \tag{5.3}$$

$$aVL = \frac{(I - III)}{2} \tag{5.4}$$

$$V2 = (0.088733 \times I) - (0.09116 \times II) + (1.57862 \times V1) + (0.230214 \times V5) \tag{5.5}$$

$$V3 = (0.245068 \times I) + (0.447773 \times II) + (1.14726 \times V1) + (0.609744 \times V5) \tag{5.6}$$

$$V4 = (0.111111 \times I) + (0.064849 \times II) + (0.465706 \times V1) + (1.07423 \times V5) \tag{5.7}$$

$$V6 = (0.202721 \times I) + (0.038811 \times II) - (0.176913 \times V1) + (0.59492 \times V5) \quad (5.8)$$

The complete 12-lead information can be built from the information of these suggested four leads by the clinicians. Therefore, 4-lead ECG data can be compressed instead of 12-lead data to be communicated over a channel. This reduces the hardware and memory requirements of the wearable devices. The complete ECG compression algorithm is explained in Algorithm 5. The interleaved samples of 4-lead ECG fed as input to the compression system. A 4-lead data compression system is presented in [132]. The data of four leads come parallelly from the sensors in any conventional system. However, processing these leads in parallel generally requires parallel architetcure [132]. As it is evident that parallel architectures increase hardware requirements, we utilize sequential implementation by employing a simple technique to interleave samples of all four leads in this work. Initially, the first sample of the first lead is taken as an input and compressed. Later, the first samples of the other three leads are processed sequentially. The operating frequency of the design is set in such a way that all four samples of every lead are processed before the second set of samples starts processing. This enables us to optimize hardware resources considerably and achieve low power and area-efficient design for multi-lead ECG compression.

Algorithm 5.

Input: ECG Samples from lead I,II,V1 and V5
Register Initialization
if $reset == 1$ **then**
    Initialize $i = 0$
    Initialize $registers$, $slope\_parameters$ & $first\_sample\_flag(SF) -> low$
    //Perform gain adjustment
    Initialize x1[n-1]=x1[n-2]=x1[n-3]=$x1[n]$ >>> 5
else
    if $SF -> high$ **then**
        Perform gain adjustment for all leads
    end if
end if
Adaptive Linear Prediction
for $i <= 3$ **do**
    $d_{12_i} = x_i[n-1] - x_i[n-2]$
    $d_{23_i} = x_i[n-2] - x_i[n-3]$
    $\hat{x1}_i[n] = x_{1_i}[n]$
    $\hat{x2}_i[n] = 2x_{1_i}[n-1] - x_{1_i}[n-2]$
    $\hat{x3}_i[n] = 3x_{1_i}[n-1] - 3x_{1_i}[n-2] + x_{1_i}[n-3]$
    Average Error Calculation
    $\bar{e_{2_i}} = \frac{e_{2_i}[n-1]+e_{2_i}[n-2]+e_{2_i}[n-3]}{3}$ //Similarly $\bar{e_{3_i}}$ is calculated
    if $d_{12_i} < th$ and $d_{23_i} < th$ **then**
        $pred\_val = \hat{x1}_i[n]$
    else

```
        pred_val = ((x2^[n]·W2[n]+x3^[n]·W3[n]))/((W2[n]+W3[n]))
    end if
end for
if SF− > high then
    first_sample = 1′b1
    pred_error = (ecg_val >>> 5) − pred_val
else
    first_sample = 1′b0
    pred_error = (ecg_val >>> 5) − pred_val
end if
Error Encoding
if −2 <= pred_error <= 1 then
    encode_length = 0
else if −4 <= pred_error <= 3 then
    encode_length = 1
else if −8 <= pred_error <= 7 then
    encode_length = 2
else
    encode_length = 3
end if
Packaging
if encode_length == 0 then
    encode_error = pred_error[1 : 0]
    packed_data = {2′b00, pred_error[1 : 0]}
else if encode_length == 1 then
    encode_error = pred_error[2 : 0]
    packed_data = {2′b01, pred_error[2 : 0]}
else if encode_length == 2 then
    encode_error = pred_error[3 : 0]
    packed_data = {2′b10, pred_error[3 : 0]}
else
    encode_error = pred_error
    packed_data = {2′b11, pred_error[3 : 0]}
end if
```

end

Next, these interleaved samples are divided by a value "T" chosen adaptively to remove an overflow error in the computations and are input to the adaptive linear prediction block. In this block, the first (I), second (II), and third (III) order linear prediction of the current sample are performed. The previous three samples are employed to estimate the slope parameter and the weighted average of II and III order linear predictions [132]. Note that we have assumed the first three samples to lie in the flat region of an ECG signal to simplify the algorithm. This assumption does not introduce any error in the compressed ECG signal. In the first step, the I, II, and III order linear predictions of the current sample are estimated using equations 5.9, 5.10, and 5.11. $\hat{x_1}[n]$, $\hat{x_2}[n]$, and $\hat{x_3}[n]$ are I, II, and III order linear predictions of the current sample, respectively.

$$\hat{x_1}[n] = x[n-1] \tag{5.9}$$

$$\hat{x_2}[n] = 2 \cdot x[n-1] - x[n-2] \tag{5.10}$$

$$\hat{x_3}[n] = 3 \cdot x[n-1] - 3 \cdot x[n-2] + x[n-3] \tag{5.11}$$

Further, slope parameters are estimated using equations 5.12 and 5.13.

$$d_{12}[n] = x[n-1] - x[n-2] \tag{5.12}$$

$$d_{23}[n] = x[n-2] - x[n-3] \tag{5.13}$$

If $d_{12}[n]$ and $d_{23}[n]$ are small compared to the threshold value of the slope, it is assumed that the current sample is lying in a flat region; thus, the final prediction is made with the help of I order linear prediction using equation 5.9. However, if any of the two slope parameters breaches the threshold, the final prediction is the weighted average of II and III order linear predictions as per equation 5.14. Note that the threshold value of 10 is chosen adaptively by considering the tradeoff between accuracy and compression ratio.

$$\hat{x[n]} = \frac{(\hat{x_2}[n] \cdot W_2[n] + \hat{x_3}[n] \cdot W_3[n])}{(W_2[n] + W_3[n])} \tag{5.14}$$

The weights $W_2[n]$ and $W_3[n]$ are calculated according to equation 5.15.

$$W_i[n] = K^{(M - e_i\bar{[n]})} \tag{5.15}$$

where K is taken as 2 for the ease of hardware implementation, and M is determined after multiple iterations. $e_i\bar{[n]}$ is the average error calculated for the previous three samples for a particular order of linear prediction as per equation 5.16.

$$e_i\bar{[n]} = \frac{(e_i[n-1] + e_i[n-2] + e_i[n-3])}{3} \tag{5.16}$$

The error term in weight calculation ensures that the final prediction after the weighted average is closer to the linear prediction with fewer errors than II and III order linear predictions. The adaptive linear prediction is followed by the final prediction error calculation, which is estimated using equation 5.17.

$$e[n] = x[n] - \hat{x[n]} \tag{5.17}$$

The next step of the compression process is encoding the error data. In the previous works, complex encoding techniques, such as Huffman encoding schemes [134] and Golomb Rice Coding [132, 135], are

used. However, these encoding schemes involve complex multiplication and division operations, which are compute and resource-intensive. Therefore, a variable length entropy encoding [136] is used in the proposed work, which requires only comparing operations and can be implemented with minimal hardware resources on silicon, as shown in Algorithm 5. In our proposed work, we encode the samples in fewer bits, evident in Algorithm 5 due to the optimization performed in preprocessing the interleaved samples. This further enables us to optimize the hardware requirements. The encoding length for the error depends on the range in which the prediction error falls as shown in Algorithm 5. The final step of the encoding process is packaging an encoded error, which facilitates easy decompression of the compressed data. Some of the previous works [136], [132] utilize complex packaging techniques, which require more information to be sent over the channel and more registers and counters to store the data before transmission. This leads to a lesser compression ratio and the utilization of more hardware resources. Therefore, we employ a simple packaging technique in this work, which selects a fixed number of bits as per the encoded sample error range, shown in Algorithm 5. In the proposed compression algorithm, the first sample of every ECG lead is not encoded; therefore, no packaging is required for these first samples. However, the prediction error for subsequent samples of all four ECG leads is also packed with the encoding length information. Finally, this packed encoded error data is obtained as a compressed bitstream and is sent for telemonitoring or stored on-chip or cloud.

Later, the decompression system is designed, as illustrated in Figure 5.1. During decompression, initially, decoding of the packed encoded error is performed, which is followed by an adaptive linear prediction of the current sample. The predicted sample is corrected for the error to obtain the reconstructed data. The following section presents the hardware implementation details of the proposed algorithm.

5.3.2 Hardware Implementation

The proposed hardware for multi-lead ECG compression is shown in Figure 5.2. The complete architecture is designed as a finite state machine. The architecture has serial data as input where the samples of individual ECG leads are interleaved to form the input data stream. Each sample of this input data stream is presented in 2's complement using a 16-bit representation. Furthermore, lead separation is performed where samples from the four chosen ECG leads are stored in the registers with the help of a counter. Unlike the method in [132], which presents a parallel architecture for processing four leads, our proposed lead separation method enables us to reuse the same architecture

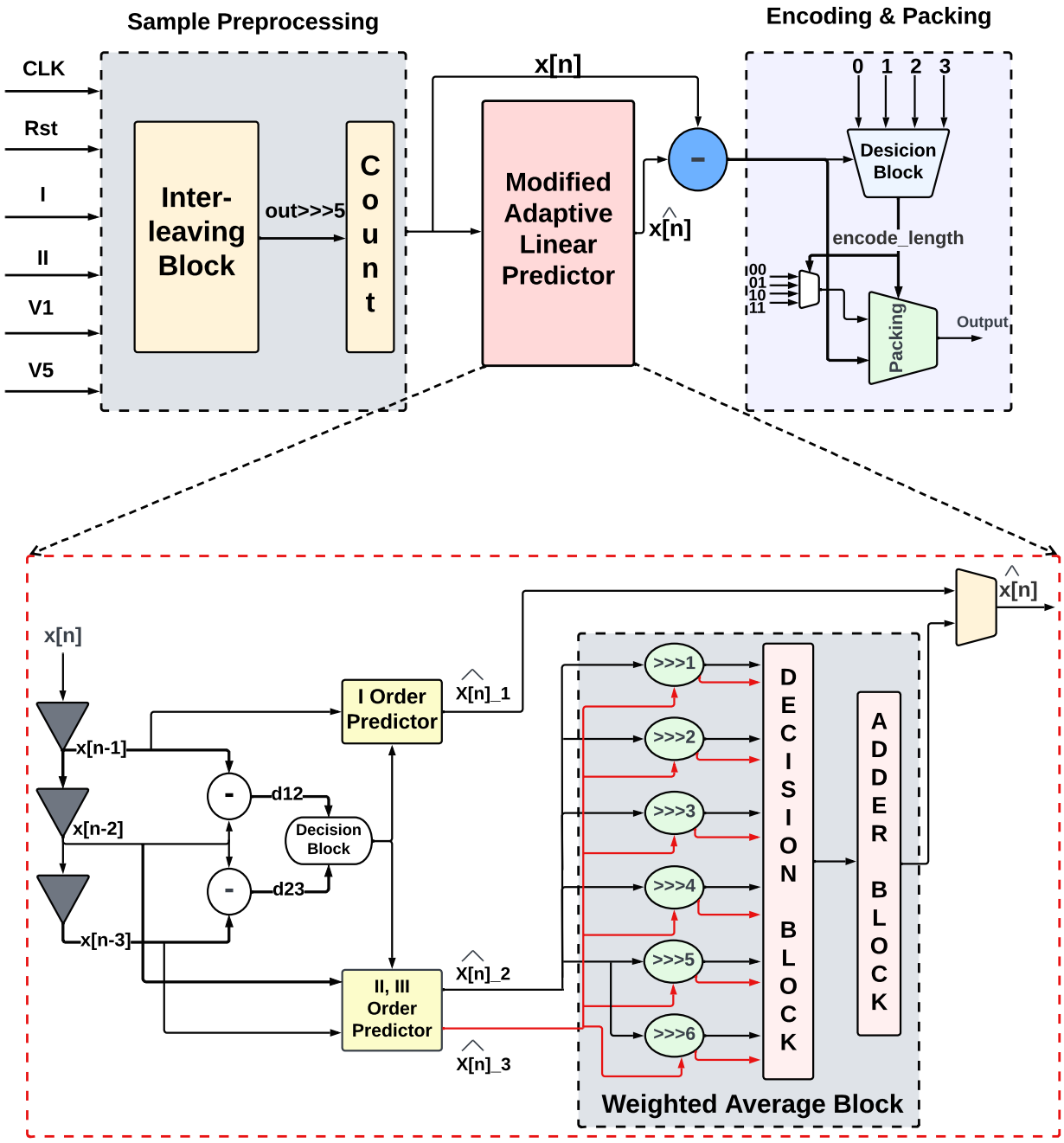


Figure 5.2: Architecture of Compression System

for compressing all the leads by utilizing just a few additional registers to store some samples of other leads. This serial scheme reduces the hardware resources drastically. These interleaved samples are divided by 32 (T) to reduce the gain value of the samples. This value is chosen adaptively to avoid any overflow during multiplication operations. Note that the division operation is performed in hardware using simple shift operations, and dividing the input ECG samples by 32 does not cause any error in the compressed data. Later, adaptive linear predictions of I, II, and III order are implemented as data-shifting operations for every individual lead. These predicted values are then processed to get prediction errors using a simple subtractor. Next, the leads are merged and processed for slope parameter estimation, followed by adaptive linear prediction of the current sample based on threshold comparison.

As depicted earlier, the first-order prediction equation is used if the current sample is located in a flat region. However, if the error exceeds the threshold, the weighted average of II and III order linear prediction is used for the compression. As evident from equation 5.14, the calculation of the weighted average requires both multiplication and division operations, which are compute-intensive and require more hardware resources. Therefore, in this work, we optimize the weighted average using shifting operation, as explained in equation 5.18.

$$\begin{aligned} pred\_val &= \frac{(\hat{x_2}[n] \cdot 2^{c-\bar{e_2}[n]} + \hat{x_3}[n] \cdot 2^{c-\bar{e_3}[n]})}{(2^{c-\bar{e_2}[n]} + 2^{c-\bar{e_3}[n]})} \\ &= \frac{(\hat{x_2}[n] \cdot 2^{c-\bar{e_2}[n]})}{(2^{c-\bar{e_2}[n]} + 2^{c-\bar{e_3}[n]})} + \frac{(\hat{x_3}[n] \cdot 2^{c-\bar{e_3}[n]})}{(2^{c-\bar{e_2}[n]} + 2^{c-\bar{e_3}[n]})} \\ &= \frac{(\hat{x_2}[n])}{(1 + 2^{\bar{e_2}[n]-\bar{e_3}[n]})} + \frac{(\hat{x_3}[n])}{(1 + 2^{\bar{e_3}[n]-\bar{e_2}[n]})} \\ &= \frac{(\hat{x_2}[n])}{(1 + 2^{\bar{\triangle e}})} + \frac{(\hat{x_3}[n])}{(1 + 2^{\bar{\triangle e}})} \end{aligned} \tag{5.18}$$

where, $\bar{\triangle e} = -\triangle e$. If $\triangle e = 0$, then the prediction value is estimated as equation 5.19

$$\begin{aligned} pred\_val &= \frac{(\hat{x_2}[n])}{(1 + 2^0)} + \frac{(\hat{x_3}[n])}{(1 + 2^0)} \\ &= \frac{\hat{x_2}[n] + \hat{x_3}[n]}{2} \\ &= \hat{x_2}[n] >>> 1 + \hat{x_3}[n] >>> 1 \end{aligned} \tag{5.19}$$

If $\triangle e = 1$, then the prediction value is estimated as equation 5.20

$$\begin{aligned} pred\_value &= \frac{(\hat{x_2[n]})}{(1+2^1)} + \frac{(\hat{x_3[n]})}{(1+2^{-1})} \\ &= \frac{\hat{x_2[n]}}{3} + \frac{\hat{x_3[n]}}{1.5} \\ &= \hat{x_2[n]} >>> 2 + \hat{x_2[n]} >>> 4 + \hat{x_2[n]} >>> 6 \\ &+ \hat{x_3[n]} >>> 1 + \hat{x_3[n]} >>> 3 + \hat{x_3[n]} >>> 5 \end{aligned} \tag{5.20}$$

If $\triangle e = 2$, then the prediction value is estimated as equation 5.21

$$\begin{aligned} pred\_value &= \frac{\hat{x_2[n]}}{5} + \frac{\hat{x_3[n]}}{1.25} \\ &= \hat{x_2[n]} >>> 2 + \hat{x_2[n]} >>> 4 + \hat{x_2[n]} >>> 7 \\ &+ \hat{x_3[n]} >>> 1 + \hat{x_3[n]} >>> 2 + \hat{x_3[n]} >>> 5 \end{aligned} \tag{5.21}$$

If $\triangle e = 3$, then the prediction value is estimated as equation 5.22

$$\begin{aligned} pred\_value &= \frac{\hat{x_2[n]}}{9} + \frac{\hat{x_3[n]}}{1.125} \\ &= \hat{x_2[n]} >>> 4 + \hat{x_2[n]} >>> 5 + \hat{x_2[n]} >>> 6 \\ &+ \hat{x_3[n]} >>> 1 + \hat{x_3[n]} >>> 2 + \hat{x_3[n]} >>> 3 \end{aligned} \tag{5.22}$$

If $\triangle e >= 4$, then the prediction value is estimated as equation 5.23. Equation 5.23 is optimized, considering $2^{\triangle e} >> 1$ and $2^{\bar{\triangle e}} << 1$

$$\begin{aligned} pred\_val &= \frac{(\hat{x_2[n]})}{(1+2^{\triangle e})} + \frac{(\hat{x_3[n]})}{(1+2^{\bar{\triangle e}})} \\ &= \frac{(\hat{x_2[n]})}{2^{\triangle e}} + \frac{(\hat{x_3[n]})}{1} \\ &= \hat{x_2[n]} >>> \triangle e + \hat{x_3[n]} \end{aligned} \tag{5.23}$$

For the negative values of $\triangle e$, shifting operations of $\hat{x_2[n]}$ and $\hat{x_3[n]}$ are interchanged. It can be observed from the equations mentioned above, the division operations are optimized as shift operations saving hardware resources and power. The architecture of the modified adaptive linear block is presented in figure 5.2. The final predicted value of the current sample is then used to calculate the absolute prediction error. The error data is further encoded with the help of variable length entropy encoding, which requires comparators for the implementation. Finally, the encoded error is packaged, appending the length information of the current encoded error sample with the encoded error samples. The resulting packed encoded error data as a compressed bitstream can be stored or

transmitted as needed. Once the complete process is completed, the registers are updated for the new samples, and a similar process is repeated for the entire input bitstream. On the other hand, the input of the decompression system contains the error information of the successive samples. The first step of decompression is unpacking and decoding the error data, followed by lead separation, similar to compression. Further, I, II, and III order linear prediction, prediction error calculation, lead merging, slope parameter calculation, and adaptive linear prediction are performed to retrieve the original ECG.

5.4 Experimental Results of Proposed Design

The proposed method is validated using MATLAB and is realized as an ASIC. Its implementation details and analysis are presented below.

5.4.1 Performance Evaluation

The proposed compression design is validated using the PTB-DB database sampled at 250 Hz [41]. Each input ECG sample is 16 bits and is taken from a variety of age ranges and gender. Further, each signal is analyzed for a 3-second duration at a time. The efficacy of designs is evaluated by compression ratio (CR) and Percentage Root Mean Square Difference (PRD). Compression ratio, CR, is defined mathematically as the ratio of the size of data before compression to the size of data after compression. Note that a high CR value is desirable for a compression system.

$$CR = \frac{Original\_data\_size}{Compressed\_data\_size} \tag{5.24}$$

Further, PRD should be as low as possible for better performance, which can be expressed as equation 5.25.

$$PRD(100\%) = 100 \times \sqrt{\frac{\sum_{n=1}^{N}((X(n) - X_r(n))^2}{\sum_{n=1}^{N}(X(n))^2}} \tag{5.25}$$

Table 5.1 presents the complete validation of the proposed algorithm on the PTB dataset. The CR and PRD of the proposed design are estimated to be 3.857 and $0.46 - 11.1\%$, respectively, for four lead ECG data compression. It is observed that irrespective of age and gender variations, the proposed scheme achieves a good CR.

S.No.	Record Name	Compression Ratio	PRD(%)	S.No.	Record Name	Compression Ratio	PRD(%)
1	s0014lrem	3.7587	5.3168	29	s0054lrem	3.8546	2.0916
2	s0015lrem	3.8396	2.6777	30	s0055lrem	3.8567	0.9907
3	s0016lrem	3.7481	2.5185	31	s0056lrem	3.8449	2.252
4	s0017lrem	3.8621	3.5505	32	s0057lrem	3.8364	0.9871
5	s0020arem	3.84	2.8448	33	s0058lrem	3.822	2.2582
6	s0020brem	3.8442	3.8151	34	s0059lrem	3.8371	2.8051
7	s0021arem	3.8729	1.0488	35	s0060lrem	3.852	3.6382
8	s0021brem	3.845	3.2914	36	s0061lrem	3.8941	0.503
9	s0022lrem	3.8724	2.3001	37	s0062lrem	3.8943	0.6593
10	s0025lrem	3.8554	4.705	38	s0063lrem	3.8804	0.4608
11	s0026lrem	3.8579	6.1186	39	s0064lrem	3.8752	1.839
12	s0027lrem	3.8496	2.6324	40	s0065lrem	3.8308	0.5881
13	s0028lrem	3.8798	1.0785	41	s0066lrem	3.8852	3.1787
14	s0029lrem	3.839	1.5319	42	s0067lrem	3.8657	6.0609
15	s0031lrem	3.8544	1.3668	43	s0068lrem	3.858	2.9357
16	s0035_rem	3.8095	6.9846	44	s0069lrem	3.8677	2.919
17	s0036lrem	3.8857	1.2797	45	s0070lrem	3.8219	1.332
18	s0037lrem	3.886	5.5317	46	s0071lrem	3.8551	2.9742
19	s0038lrem	3.8321	6.4803	47	s0072lrem	3.8581	0.8459
20	s0039lrem	3.8252	6.7656	48	s0073lrem	3.8536	0.6429
21	s0042lrem	3.9064	1.246	49	s0074lrem	3.8758	1.2128
22	s0043lrem	3.8819	0.7066	50	s0075lrem	3.8481	4.4311
23	s0044lrem	3.8708	5.3216	51	s0076lrem	3.8759	2.1705
24	s0045lrem	3.8591	3.9437	52	s0078lrem	3.9028	5.074
25	s0046lrem	3.8419	4.4191	53	s0079lrem	3.8753	1.9554
26	s0049lrem	3.8765	4.3605	54	s0080lrem	3.8473	1.11
27	s0050lrem	3.8792	2.1679	55	s0081lrem	3.8683	4.5996
28	s0051lrem	3.844	4.1238	56	s0082lrem	3.8831	2.7574
-	-	-	-	57	s0083lrem	3.8715	5.087

Table 5.1: Performance Evaluation of Compression Design

Resources	Total Available	Utilized Resources	
		ECG Compression	ECG Decompression
LUT	53200	922	944
Reg	106400	508	477
MUX F7	26600	8	7
MUX F8	13300	0	0
IOB	200	52	56
BUFGCTRL	32	1	1

Table 5.2: FPGA Resource Utilization

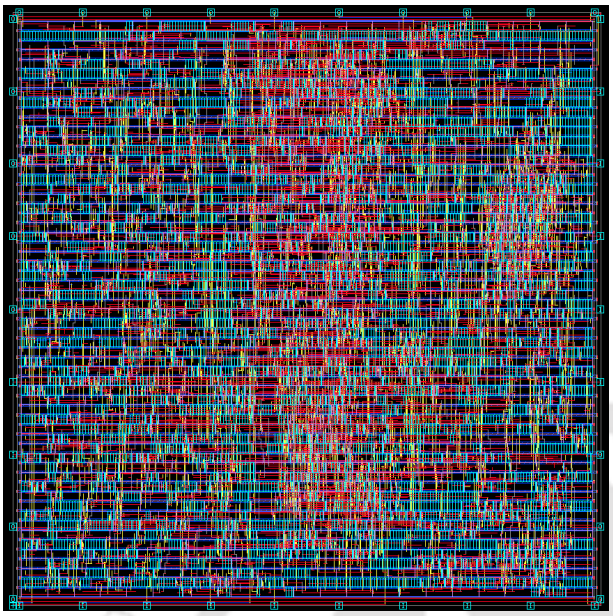
Parameters	[130]	[132]	Proposed
Method	Lossy	Lossless	Losless
Platform	Virtex-6	Virtex-6	Zedboard
LUT	1332	38179	922
FF	NA	6182	508
DSP	0	72	0

Table 5.3: Comparison of FPGA Implementation

5.4.2 Implementation Results

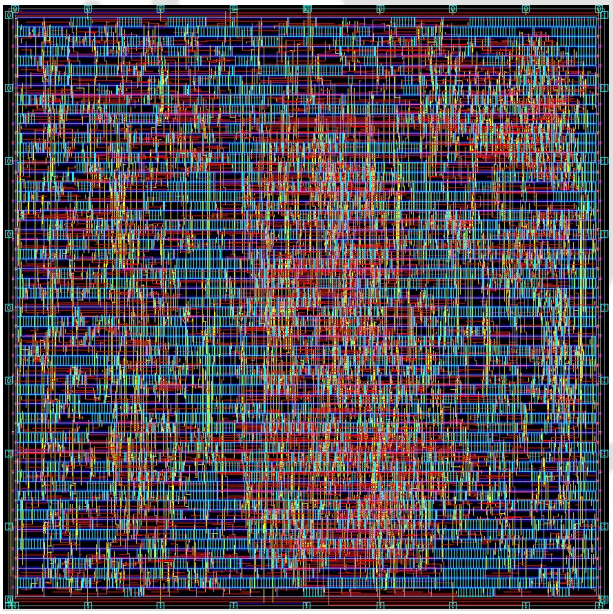
The compression and decompression systems are validated using MATLAB, employing ECG records from the PTB-DB database [41]. Further, an RTL design of the proposed method is realized using Verilog-HDL and is synthesized and verified on an FPGA. The overall resource utilization of all four methods performing ECG data compression and decompression is presented in Table 5.2. Further, ECG compression and decompression architectures are compared with the methods available in the literature, which are implemented on an FPGA. It can be observed from Table 5.3 that the resource utilization of our proposed architectures is the least among all the contemporary designs. Further, our designs are realized as an ASIC using SCL 180nm PDKs. The functionality of the proposed architecture is tested with the Synopsys VCS. Synopsys Design Compiler is employed to generate a gate-level netlist and the placement and routing are performed on this netlist using ICC.

The operational frequency of ECG compression and decompression systems is 36kHz. Note that our proposed methods can perform ECG in real-time at the low frequencies mentioned above, enabling low power operations. Figure 5.3 and figure 5.4 depicts the post-placement and routing layout of the proposed ECG compression and decompression architectures. The power consumption at 1.98 V supply voltage and 36kHz frequency is 2.102μW. Further, the areas of ECG compression and decompression modules are 0.0813mm^2 and 0.0801mm^2.



CMOS Process	SCL 180nm
Area	$0.0813mm^2$
Voltage	1.98V
Frequency	36kHz
Dynamic Power	1.9874 uW
Static Power	0.1157uW

Figure 5.3: The Layout Photograph of Compression Module and its Specifications



CMOS Process	SCL 180nm
Area	$0.08018mm^2$
Voltage	1.98V
Frequency	36kHz
Dynamic Power	1.799 uW
Static Power	0.133uW

Figure 5.4: The Layout Photograph of ECG Decompression Module and its Specifications

Parameters	[132](2021)	[137](2022)	[138](2022)	Proposed	
Signal	ECG	ECG	ECG	ECG	
#Leads	4	1	2	4	
Function	Compression	Compression	Compression	Compression	Decompression
Methodology	ALP+Golomb Rice Encoding	Run_length encoding +Golomb Rice Encoding	Huffman Encoding	ALP+ Linear Encoding	
Type	Lossless	Lossless	Lossless	Lossless	
Database	PTB-DB	MIT-BIH	PTB-DB	PTB-DB	
Technology(nm)	180	90	180	180	
CR	4.067	2.91		3.86	
Voltage(V)	1	1.2	1.8	1.8	
Frequency(Hz)	1k	100M	16M	36k	
Area(mm^2)	16.4	0.0051	-	0.0813	0.0801
Gate Count	475.9k	0.4k	9.5k	2442	2412
Power Consumption(uW)	69.18	18.78	12.7	2.102	1.913

Table 5.4: Comparison with State-of-the-art Methods

The detailed results are presented in Table 5.4 and are compared with the contemporary methods. It can be seen from Table 5.4 that only [132] reports the compression of four lead ECG data. The ECG compression scheme of the proposed design utilizes 201.72× less area and 32.91× less power than [132]. The gate count of the proposed method is considerably less than [132] as well. Further, the CR of the proposed ECG compression method is close to [132] and is 1.05× less. Our proposed implementations are more efficient than the single lead [137] and double lead [138] ECG compression architectures. To the best of our knowledge, there is no data available for ECG decompression architectures in the literature. Low area utilization and power consumption makes our proposed methods a suitable candidate for IoT-enabled wearable healthcare devices.

5.5 Summary and Scope of Present Work

This chapter proposes hardware realization of multi-channel lossless ECG compression and decompression systems. The compression and decompression methods include adaptive linear prediction and variable length entropy encoding schemes. The PTB-DB database has been utilized for the verification of the proposed method. The compression ratios obtained for ECG is 3.8576. The proposed ECG designs are area and power efficient compared to the contemporary methods reported in the literature, making them suitable for low power IoT-enabled wearable healthcare applications.

6

Conclusions and Future Directions

Contents

In this chapter, the contributions of this thesis toward developing low power VLSI architectures for ECG feature extraction, cardiovascular disease detection, and ECG data compression are depicted. A future research roadmap to further the present work is also outlined.

6.1 Summary

In recent years, due to unhealthy and irregular lifestyle, the mortality rate due to cardiac diseases have increased drastically. Introducing wearable healthcare devices has catalyzed remote personalized patient monitoring and care. The wearables for cardiac monitoring face several challenges and must be more advanced to perform different cardiovascular disease detection. Therefore, this thesis proposes low-power VLSI architectures for ECG signal analysis and cardiovascular disease (CVD) detection using machine learning models, which are suitable for wearable devices.

As per the conventional system for wearable healthcare devices for ECG analysis, we proposed optimized VLSI architectures to detect and predict Cardiac Arrhythmia and Myocardial Infarction. To detect any CVD, extracting the fiducial points of ECG is imperative. Therefore, in Chapter 2, low complexity algorithms and their low-power VLSI architectures to extract a complete feature set, i.e., P-QRS-T peaks and their respective onset and offset, are proposed. An average sensitivity of 99% is achieved on the QT dataset by the proposed algorithms. The algorithm utilized simple difference and comparison operations, avoiding all floating-point calculations to delineate the features making it computationally inexpensive. Once the basic features of ECG are delineated, several secondary features and ECG beats can be estimated for detecting different cardiovascular diseases. Chapters 3 and 4 focus on developing classifier blocks that utilize deep neural networks and if-else-based classifiers to detect CVDs.

Two different VLSI architectures that can detect different arrhythmia beats and predict ventricular arrhythmia are discussed in Chapter 3. The first design is a low-power, patient-independent, generic, and adaptive ECG cardiac arrhythmia co-processor, which can classify all five types of ECG beats according to AAMI standards. It employs a simple ECG pre-processing method and uses a complete ECG beat as an input to DNN, allowing the classifier to capture all the morphological changes in the arrhythmia beats. Using a simplified novel MAC unit, resource sharing and optimization of softmax function make our co-processor area and power efficient. It consumes 8.75 μ W at 12 kHz with 91.6% accuracy when realized with SCL 180nm CMOS technology. Secondly, a low-power ECG

co-processor for predicting cardiac arrhythmia is proposed, which can predict ventricular arrhythmia 15 min before its occurrence. The proposed design utilizes an optimal feature set and novel DNN classifier architecture, enabling it to operate at a minimum frequency of 12.5kHz, consuming 4.69 μ W of power.

Later, Chapter 4 proposes an edge-enabled area and power-efficient VLSI architecture (MInSC) to classify progression MI stages from healthy and non-MI beats. MInSC is the first ASIC to classify the different stages of MI from non-MI and normal beats. It employs a simple and optimized ECG feature extraction algorithm and uses a detailed set of markers per medical standards to classify MI. It also proposes an optimized divider for low-power estimation of markers on hardware. Furthermore, it employs a simple if-else-based algorithm that is hardware efficient. Finally, in Chapter 5, VLSI implementation of multi-channel ECG compression and decompression systems have been proposed. These schemes utilize optimized hardware implementation of adaptive linear prediction and variable length entropy encoding schemes. The proposed architecture has a compression ratio of 3.8576 when tested on the PTB database.

The proposed VLSI architectures operate at low power in the range of microwatts and have low area and resource requirements when implemented as an ASIC, making them suitable for low-power machine learning-enabled wearable devices.

6.2 Directions for Future Work

Based on the outcomes of this thesis, this section provides possible future research directions for the development of advanced biomedical signal processing devices.

(i) To develop clinically accurate cardiovascular devices, multi-lead ECG data analysis can be used. Generally, medical professionals analyze all 12 leads of ECG to diagnose any cardiac anomalies. Therefore, algorithms and their corresponding VLSI architectures should be designed to perform feature extraction efficiently on all 12 leads of ECG data.

(ii) A wide variety of ECG data from different demographic regions can be used to train CVD classifiers. As the ECG signal variations are different in different geographical regions, a classifier trained on such a wide variety of ECG data will be more accurate for a generic wearable device.

(iii) The classification of several cardiovascular diseases can be integrated into one classification module. A multi-disease unified classifier will be more beneficial as it can diagnose different

CVDs, making it more robust and resource-efficient.

(iv) In this thesis, we aimed to detect different types of CVDs. However, a design that can predict diseases, such as MI, different types of arrhythmia, and heart attacks much before their occurrence will be more beneficial for a wearable application.

(v) The deep neural network should be optimized further to increase accuracy while maintaining low power. Furthermore, advanced ML models can be explored to increase the diagnosis accuracy of cardiovascular diseases.

List of Publications

Journal Publications

- Published Papers:
  1. Janveja, Meenali, Rushik Parmar, and Gaurav Trivedi. "MInSC: A VLSI Architecture for Myocardial Infarction Stages Classifier for Wearable Healthcare Applications." IEEE Transactions on Circuits and Systems II: Express Briefs (2022).
  2. Janveja, Meenali, et al. "A DNN-Based Low Power ECG Co-Processor Architecture to Classify Cardiac Arrhythmia for Wearable Devices." IEEE Transactions on Circuits and Systems II: Express Briefs 69.4 (2022): 2281-2285.
  3. Janveja, Meenali, and Gaurav Trivedi. "An Area and Power Efficient VLSI Architecture for ECG Feature Extraction for Wearable IoT Healthcare Applications." Integration 82 (2022): 96-103.
- Manuscripts Communicated
  1. Janveja, Meenali, et al., "A Low Power Co-processor Architecture to Predict Ventricular Arrhythmia for Wearable Healthcare Devices"
  2. Janveja, Meenali, et al., "An Optimized Low Power VLSI Architecture for ECG/VCG Data Compression for IoHT Wearable Device Application"

Conference and Workshop Publications

1. Janveja, Meenali, Rushik Parmar, Gaurav Trivedi, Pidanic Jan, and Zdenek Nemec. "An Energy Efficient and Resource Optimal VLSI Architecture for ECG Feature Extraction for Wearable Healthcare Applications", In 2022 32nd International Conference Radioelektronika (RADIOELEKTRONIKA), pp. 1-6. IEEE, 2022.
2. Janveja, Meenali, Mayank Tantuway, Ketan Chaudhari, and Gaurav Trivedi. "Design of Low Power VLSI Architecture for Classification of Arrhythmic Beats Using DNN for Wearable Device Applications", In 2021 IEEE Nordic Circuits and Systems Conference (NorCAS), pp. 1-6. IEEE, 2021.
3. Janveja, Meenali, et. al "Design of Efficient AES Architecture for Secure ECG Signal Transmission for Low-power IoT Applications." In 2020 30th International Conference Radioelektronika (RADIOELEKTRONIKA), pp. 1-6. IEEE, 2020.

Bibliography

[1] C. Bock, "ECG Signal Analysis based on the Wavelet Transform," https://www.jku.at/fileadmin/gruppen/183/Docs/Finished_Theses/Master_Thesis_Boeck.pdf, 2015.

[2] S. K. Berkaya, A. K. Uysal, E. S. Gunal, S. Ergin, S. Gunal, and M. B. Gulmezoglu, "A Survey on ECG Analysis," *Biomedical Signal Processing and Control*, vol. 43, pp. 216–235, 2018.

[3] T. Jun, H. Nguyen, D. Kang, D. Kim, D. Kim, and Y. Kim, "ECG Arrhythmia Classification using a 2-D convolutional Neural Network. arXiv 2018," *arXiv preprint arXiv:1804.06812*.

[4] V. Kumar, A. K. Abbas, N. Fausto, and J. C. Aster, *Robbins and Cotran Pathologic basis of Disease, Professional Edition e-book*. Elsevier health sciences, 2014.

[5] Polaris, "Wearable Medical Devices Market Share, Size, Trends, Industry Analysis Report, By Device Type," https://www.polarismarketresearch.com/industry-analysis/wearable-medical-devices-market, 2020.

[6] ScienceSoft, "Wearable Technology in Healthcare," https://www.scnsoft.com/healthcare/medical-devices/wearable, 2022.

[7] W. H. Organization, *World health statistics 2015*. World Health Organization, 2015.

[8] M. P. Heron, "Deaths: leading causes for 2016," 2018.

[9] K. Bayoumy, M. Gaber, A. Elshafeey, O. Mhaimeed, E. H. Dineen, F. A. Marvel, S. S. Martin, E. D. Muse, M. P. Turakhia, K. G. Tarakji *et al.*, "Smart Wearable Devices in Cardiovascular Care: Where We are and How to Move Forward," *Nature Reviews Cardiology*, vol. 18, no. 8, pp. 581–599, 2021.

[10] G. S. Wagner, *Marriott's Practical Electrocardiography*. Lippincott Williams & Wilkins, 2001.

[11] M. Thaler, *The Only EKG book you'll Ever Need*. Lippincott Williams & Wilkins, 2017.

[12] W. H. Organization, *World health statistics 2015*. World Health Organization, 2015.

[13] S. R. Steinhubl and E. J. Topol, "Moving from Digitalization to Digitization in Cardiovascular Care: Why is it Important, and What Could it Mean for Patients and Providers?" *Journal of the American College of Cardiology*, vol. 66, no. 13, pp. 1489–1496, 2015.

[14] J. Pan and W. Tompkins, "Real Time Algorithm Detection for QRS," *IEEE Trans. Eng. Biomed Eng*, vol. 32, no. 3, pp. 230–236, 1985.

[15] P. S. Hamilton and W. J. Tompkins, "Quantitative Investigation of QRS detection Rules Using the MIT/BIH Arrhythmia Database," *IEEE transactions on biomedical engineering*, no. 12, pp. 1157–1165, 1986.

[16] V. Kalidas and L. Tamil, "Real-time QRS Detector using Stationary Wavelet Transform for Automated ECG Analysis," in *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering (BIBE)*. IEEE, 2017, pp. 457–461.

[17] J. Pan and W. Tompkins, "Real Time Algorithm Detection for QRS," *IEEE Trans. Eng. Biomed Eng*, vol. 32, no. 3, pp. 230–236, 1985.

[18] E. B. Mazomenos, D. Biswas, A. Acharyya, T. Chen, K. Maharatna, J. Rosengarten, J. Morgan, and N. Curzen, "A Low-Complexity ECG Feature Extraction Algorithm for Mobile Healthcare Applications," *IEEE journal of biomedical and health informatics*, vol. 17, no. 2, pp. 459–469, 2013.

[19] N. Vemishetty, P. Patra, P. K. Jha, K. B. Chivukula, C. K. Vala, A. Jagirdar, V. Y. Gudur, A. Acharyya, and A. Dutta, "Low Power Personalized ECG based System Design Methodology for Remote Cardiac Health Monitoring," *IEEE Access*, vol. 4, pp. 8407–8417, 2016.

[20] A. I. Kalyakulina, I. I. Yusipov, V. A. Moskalenko, A. V. Nikolskiy, A. A. Kozlov, N. Y. Zolotykh, and M. V. Ivanchenko, "Finding Morphology Points of Electrocardiographic-Signal Waves using Wavelet Analysis," *Radiophysics and Quantum Electronics*, vol. 61, no. 8-9, pp. 689–703, 2019.

[21] T. Tekeste, H. Saleh, B. Mohammad, A. Khandoker, and M. Ismail, "A Nano-Watt ECG Feature Extraction Engine in 65-nm Technology," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 8, pp. 1099–1103, 2017.

[22] S. K. Jain and B. Bhaumik, "An energy efficient ECG signal processor detecting cardiovascular diseases on smartphone," *IEEE transactions on biomedical circuits and systems*, vol. 11, no. 2, pp. 314–323, 2016.

[23] 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, 2017.

[24] T. Tekeste, H. Saleh, B. Mohammad, and M. Ismail, "Ultra-Low Power QRS Detection and ECG Compression Architecture for IoT Healthcare Devices," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 2, pp. 669–679, 2018.

[25] Y. Zou, J. Han, X. Weng, and X. Zeng, "An Ultra-low Power QRS complex Detection Algorithm based on Down-Sampling Wavelet Transform," *IEEE Signal Processing Letters*, vol. 20, no. 5, pp. 515–518, 2013.

[26] T. Tekeste, N. Bayasi, H. Saleh, A. Khandoker, B. Mohammad, M. Al-Qutayri, and M. Ismail, "Adaptive ECG Interval Extraction," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2015, pp. 998–1001.

[27] B. Zhang, L. Sieler, Y. Morère, B. Bolmont, and G. Bourhis, "A Modified Algorithm for QRS complex Detection for FPGA Implementation," *Circuits, Systems, and Signal Processing*, vol. 37, no. 7, pp. 3070–3092, 2018.

[28] T. Tekeste, H. Saleh, B. Mohammad, A. Khandoker, and M. Ismail, "A Nano-Watt ECG Feature Extraction Engine in 65-nm Technology," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 8, pp. 1099–1103, 2017.

[29] D. Castells-Rufas and J. Carrabina, "Simple Real-Time QRS Detector with the MaMeMi Filter," *Biomedical Signal Processing and Control*, vol. 21, pp. 137–145, 2015.

[30] B. Subramanian, "ECG Signal Classification and Parameter Estimation using Multiwavelet Transform," 2017.

[31] S. M. Noor, E. John, and M. Panday, "Design and Implementation of an Ultralow-Energy FFT ASIC for Processing ECG in Cardiac Pacemakers," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 4, pp. 983–987, 2018.

[32] M. Hammad, S. Zhang, and K. Wang, "A Novel two-dimensional ECG Feature Extraction and Classification Algorithm based on Convolution Neural Network for Human Authentication," *Future Generation Computer Systems*, vol. 101, pp. 180–196, 2019.

[33] R. Rao, "Wavelet transforms," *Encyclopedia of Imaging Science and Technology*, 2002.

[34] R. Stojanović, D. Karadaglić, M. Mirković, and D. Milošević, "A FPGA System for QRS Complex Detection based on Integer Wavelet Transform," *Measurement Science Review*, vol. 11, no. 4, pp. 131–138, 2011.

[35] S. Selvaraj and S. Sundaravaradhan, "Challenges and Opportunities in IoT Healthcare Systems: a Systematic Review," *SN Applied Sciences*, vol. 2, no. 1, pp. 1–8, 2020.

[36] J. P. Martínez, R. Almeida, S. Olmos, A. P. Rocha, and P. Laguna, "A Wavelet-based ECG Delineator: Evaluation on Standard Databases," *IEEE Transactions on biomedical engineering*, vol. 51, no. 4, pp. 570–581, 2004.

[37] S. A. Guidera and J. S. Steinberg, "The Signal-Averaged P wave Duration: A Rapid and Non-Invasive Marker of Risk of Atrial Fibrillation," *Journal of the American College of Cardiology*, vol. 21, no. 7, pp. 1645–1651, 1993.

[38] C. Saritha, V. Sukanya, and Y. N. Murthy, "ECG Signal Analysis using Wavelet Transforms," *Bulg. J. Phys*, vol. 35, no. 1, pp. 68–77, 2008.

[39] R. B. Northrop, *Non-invasive Instrumentation and Measurement in Medical Diagnosis*. CRC press, 2017.

[40] L. H. Crockett, R. A. Elliot, and M. A. Enderwitz, *The Zynq Book Tutorials for Zybo and Zedboard*. Strathclyde Academic Media, 2015.

[41] A. L. Same Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, June 2000. [Online]. Available: http://circ.ahajournals.org/content/101/23/e215

[42] F. Rincón, J. Recas, N. Khaled, and D. Atienza, "Development and Evaluation of Multilead Wavelet-Based ECG Delineation Algorithms for Embedded Wireless Sensor Nodes," *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, no. 6, pp. 854–863, 2011.

[43] L. Y. Di Marco and L. Chiari, "A Wavelet-based ECG Delineation Algorithm for 32-bit Integer Online Processing," *Biomedical engineering online*, vol. 10, no. 1, p. 23, 2011.

[44] A. for the Advancement of Medical Instrumentation *et al.*, "Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms," *ANSI/AAMI EC38*, vol. 1998, 1998.

[45] H. Chatterjee, M. Mitra, and R. Gupta, "Real–time Detection of Electrocardiogram Wave Features using Template Matching and Implementation in FPGA," *International Journal of Biomedical Engineering and Technology*, vol. 17, no. 3, pp. 290–313, 2015.

[46] X. Wang, Y. Zhu, Y. Ha, M. Qiu, and T. Huang, "An FPGA-based Cloud System for Massive ECG Data Analysis," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 3, pp. 309–313, 2016.

[47] D. Alhelal, K. A. Aboalayon, M. Daneshzand, and M. Faezipour, "FPGA-based Denoising and Beat Detection of the ECG Signal," in *2015 Long Island Systems, Applications and Technology*. IEEE, 2015, pp. 1–5.

[48] G. G. C. Lee, Z.-J. Huang, C.-Y. Chen, and C.-F. Chen, "Implementation of Gabor Feature Extraction Algorithm for Electrocardiogram on FPGA," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2015, pp. 798–801.

[49] M. A. Kumar and K. M. Chari, "Efficient FPGA-based VLSI Architecture for Detecting R-Peaks in Electrocardiogram Signal by Combining Shannon Energy with Hilbert Transform," *IET Signal Processing*, vol. 12, no. 6, pp. 748–755, 2018.

[50] M. Karim *et al.*, "Novel Simple Decision Stage of Pan & Tompkins QRS Detector and its FPGA-Based Implementation," in *Second International Conference on the Innovative Computing Technology (INTECH 2012)*. IEEE, 2012, pp. 331–336.

[51] S. Agarwal, A. Rani, V. Singh, and A. P. Mittal, "Performance Evaluation and Implementation of FPGA Based SGSF in Smart Diagnostic Applications," *Journal of medical systems*, vol. 40, no. 3, p. 63, 2016.

[52] A. Page, A. Kulkarni, and T. Mohsenin, "Utilizing Deep Neural Nets for an Embedded ECG-based Biometric Authentication System," in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2015, pp. 1–4.

[53] H. Zairi, M. Kedir-Talha, S. Benouar, and A. Ait-Amer, "Intelligent System for Detecting Cardiac Arrhythmia on FPGA," in *2014 5th International Conference on Information and Communication Systems (ICICS)*. IEEE, 2014, pp. 1–5.

[54] D. Panigrahy, M. Rakshit, and P. Sahu, "FPGA Implementation of Heart Rate Monitoring System," *Journal of medical systems*, vol. 40, no. 3, p. 49, 2016.

[55] N. Vemishetty, P. Patra, P. K. Jha, K. B. Chivukula, C. K. Vala, A. Jagirdar, V. Y. Gudur, A. Acharyya, and A. Dutta, "Low Power Personalized ECG based System Design Methodology for Remote Cardiac Health Monitoring," *IEEE Access*, vol. 4, pp. 8407–8417, 2016.

[56] H. Kim, S. Kim, N. Van Helleputte, A. Artes, M. Konijnenburg, J. Huisken, C. Van Hoof, and R. F. Yazicioglu, "A Configurable and Low-Power Mixed Signal SoC for Portable ECG Monitoring Applications," *IEEE transactions on biomedical circuits and systems*, vol. 8, no. 2, pp. 257–267, 2013.

[57] P. Li, X. Zhang, M. Liu, X. Hu, B. Pang, Z. Yao, H. Jiang, and H. Chen, "A 410-nW Efficient QRS Processor for Mobile ECG Monitoring in 0.18-μm CMOS," *IEEE Transactions on biomedical circuits and systems*, vol. 11, no. 6, pp. 1356–1365, 2017.

[58] S.-Y. Lee, J.-H. Hong, C.-H. Hsieh, M.-C. Liang, S.-Y. C. Chien, and K.-H. Lin, "Low-power Wireless ECG Acquisition and Classification System for body sensor networks," *IEEE journal of biomedical and health informatics*, vol. 19, no. 1, pp. 236–246, 2014.

[59] T. T. Habte, H. Saleh, B. Mohammad, and M. Ismail, "IoT for Healthcare," in *Ultra Low Power ECG Processing System for IoT Devices*. Springer, 2019, pp. 7–12.

[60] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Same PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[61] A. for the Advancement of Medical Instrumentation *et al.*, "Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms," *ANSI/AAMI EC38*, vol. 1998, 1998.

[62] S. Dutta, A. Chatterjee, and S. Munshi, "Correlation Technique and Least Square Support Vector Machine Combine for Frequency Domain based ECG Beat classification," *Medical engineering & physics*, vol. 32, no. 10, pp. 1161–1169, 2010.

[63] H. H. Haseena, A. T. Mathew, and J. K. Paul, "Fuzzy Clustered Probabilistic and Multi Layered Feed Forward Neural Networks for Electrocardiogram Arrhythmia Classification," *Journal of Medical Systems*, vol. 35, no. 2, pp. 179–188, 2011.

[64] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, and R. San Tan, "A Deep Convolutional Neural Network Model to Classify Heartbeats," *Computers in biology and medicine*, vol. 89, pp. 389–396, 2017.

[65] S.-Y. Hsu, Y. Ho, P.-Y. Chang, C. Su, and C.-Y. Lee, "A 48.6-to-105.2 μW Machine Learning Assisted Cardiac Sensor SoC for Mobile Healthcare Applications," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 4, pp. 801–811, 2014.

[66] Z. Chen, J. Luo, K. Lin, J. Wu, T. Zhu, X. Xiang, and J. Meng, "An Energy-Efficient ECG Processor with Weak-Strong Hybrid Classifier for Arrhythmia Detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 7, pp. 948–952, 2017.

[67] P. Chollet, R. Pallas, C. Lahuec, M. Arzel, and F. Seguin, "A Sub-nJ CMOS ECG Classifier for Wireless Smart Sensor," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017, pp. 3840–3843.

[68] B. Larras, P. Chollet, C. Lahuec, F. Seguin, and M. Arzel, "A Fully Flexible Circuit Implementation of Clique-Based Neural Networks in 65-nm CMOS," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 5, pp. 1704–1715, 2018.

[69] M. A. Sohail, Z. Taufique, S. M. Abubakar, W. Saadeh, and M. A. B. Altaf, "An ECG Processor for the Detection of Eight Cardiac Arrhythmias with Minimum False Alarms," in *2019 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2019, pp. 1–4.

[70] Z. Chen, J. Luo, K. Lin, J. Wu, T. Zhu, X. Xiang, and J. Meng, "An Energy-Efficient ECG Processor with Weak-Strong Hybrid Classifier for Arrhythmia Detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 7, pp. 948–952, 2017.

[71] P. De Chazal, M. O'Dwyer, and R. B. Reilly, "Automatic Classification of Heartbeats using ECG Morphology and Heartbeat Interval Features," *IEEE transactions on biomedical engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.

[72] C. Ye, B. V. Kumar, and M. T. Coimbra, "Heartbeat Classification using Morphological and Dynamic Features of ECG Signals," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2930–2941, 2012.

[73] S. Raj and K. C. Ray, "Sparse Representation of ECG Signals for Automated Recognition of Cardiac Arrhythmias," *Expert systems with applications*, vol. 105, pp. 49–64, 2018.

[74] S. S. Xu, M.-W. Mak, and C.-C. Cheung, "Towards End-to-End ECG Classification With Raw Signal Extraction and Deep Neural Networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 4, pp. 1574–1584, 2019.

[75] Y. Zhao, Z. Shang, and Y. Lian, "A 13.34 uW Event-Driven Patient-Specific ANN Cardiac Arrhythmia Classifier for Wearable ECG Sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 2, pp. 186–197, 2020.

[76] N. Bayasi, T. Tekeste, H. Saleh, B. Mohammad, A. Khandoker, and M. Ismail, "Low-power ECG-based Processor for Predicting ventricular arrhythmia," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 5, pp. 1962–1974, 2015.

[77] S. Mandala, T. Cai Di, and M. S. Sunar, "ECG-based Prediction Algorithm for Imminent Malignant Ventricular Arrhythmias using Decision Tree," *Plos one*, vol. 15, no. 5, p. e0231635, 2020.

[78] S. M. Abubakar, W. Saadeh, and M. A. B. Altaf, "A Wearable Long-term Single-Lead ECG Processor for Early Detection of Cardiac Arrhythmia," in *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2018, pp. 961–966.

[79] W. J. Tompkins, "Biomedical Digital Signal Processing," *Editorial Prentice Hall*, 1993.

[80] Y. Zhao, Z. Shang, and Y. Lian, "A 13.34 μW Event-Driven Patient-Specific ANN Cardiac Arrhythmia Classifier for Wearable ECG Sensors," *IEEE transactions on biomedical circuits and systems*, vol. 14, no. 2, pp. 186–197, 2019.

[81] Y. Yin, S. M. Abubakar, S. Tan, H. Jiang, Z. Wang, U. Seng-Pan, and W. Jia, "A 17.7-pJ/Cycle ECG processor for Arrhythmia Detection with High Immunity to Power Line Interference and Baseline Drift," in *2020 IEEE Asian Solid-State Circuits Conference (A-SSCC)*. IEEE, 2020, pp. 1–4.

[82] D. K. Atal and M. Singh, "Arrhythmia Classification with ECG Signals based on the Optimization-Enabled Deep Convolutional Neural Network," *Computer Methods and Programs in Biomedicine*, vol. 196, p. 105607, 2020.

[83] Z. Chen, J. Luo, K. Lin, J. Wu, T. Zhu, X. Xiang, and J. Meng, "An Energy-Efficient ECG Processor with Weak-Strong Hybrid Classifier for Arrhythmia Detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 7, pp. 948–952, 2017.

[84] Z. Chen, H. Xu, J. Luo, T. Zhu, and J. Meng, "Low-power Perceptron Model based ECG Processor for Premature Ventricular Contraction Detection," *Microprocessors and Microsystems*, vol. 59, pp. 29–36, 2018.

[85] F. Ma, J. Zhang, W. Liang, and J. Xue, "Automated Classification of Atrial Fibrillation using Artificial Neural Network for Wearable Devices," *Mathematical Problems in Engineering*, vol. 2020, 2020.

[86] A. Laupacis, N. Sekar *et al.*, "Clinical Prediction Rules: A Review and Suggested Modifications of Methodological Standards," *Jama*, vol. 277, no. 6, pp. 488–494, 1997.

[87] M. E. Nojehdeh, L. Aksoy, and M. Altun, "Efficient Hardware Implementation of Artificial Neural Networks Using Approximate Multiply-Accumulate Blocks," in *2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2020, pp. 96–101.

[88] 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.

[89] "Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms ," *(AAMI Recommended Practice/American National Standard)*, 1998.

[90] J. H. Tan, Y. Hagiwara, W. Pang, I. Lim, S. L. Oh, M. Adam, R. San Tan, M. Chen, and U. R. Acharya, "Application of Stacked Convolutional and Long Short-term Memory Network for Accurate Identification of CAD ECG Signals," *Computers in biology and medicine*, vol. 94, pp. 19–26, 2018.

[91] T. J. Jun, H. J. Park, N. H. Minh, D. Kim, and Y.-H. Kim, "Premature Ventricular Contraction Beat Detection with Deep Neural Networks," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2016, pp. 859–864.

[92] U. R. Acharya, H. Fujita, O. S. Lih, Y. Hagiwara, J. H. Tan, and M. Adam, "Automated Detection of Arrhythmias using Different Intervals of Tachycardia ECG Segments with Convolutional Neural Network," *Information sciences*, vol. 405, pp. 81–90, 2017.

[93] Y.-C. Yeh, C. W. Chiou, and H.-J. Lin, "Analyzing ECG for Cardiac Arrhythmia using Cluster Analysis," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1000–1010, 2012.

[94] S. K. Pandey and R. R. Janghel, "Automatic Detection of Arrhythmia from Imbalanced ECG Database using CNN Model with SMOTE," *Australasian physical & engineering sciences in medicine*, vol. 42, no. 4, pp. 1129–1139, 2019.

[95] Y. Xu, Z. Chen, F. Li, and J. Meng, "A Granular Resampling Method and Adaptive Speculative Mechanism-based Energy-Efficient Architecture for Multiclass Heartbeat Classification," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 38, no. 11, pp. 2172–2176, 2018.

[96] C. J. Garratt, C. Statter, and D. Statters, *Mechanisms and Management of Cardiac Arrhythmias*. BMJ Books London, 2001.

[97] D. P. Zipes, A. J. Camm, M. Borggrefe, A. E. Buxton, B. Chaitman, M. Fromer, G. Gregoratos, G. Klein, A. J. Moss, R. J. Myerburg *et al.*, "ACC/AHA/ESC 2006 Guidelines for Management of Patients with Ventricular Arrhythmias and the Prevention of Sudden Cardiac Death," *Europace*, vol. 8, no. 9, pp. 746–837, 2006.

[98] S. Mandala and T. C. Di, "ECG Parameters for Malignant Ventricular Arrhythmias: A Comprehensive Review," *Journal of medical and biological engineering*, vol. 37, no. 4, pp. 441–453, 2017.

[99] S. Mandala, T. Cai Di, and M. S. Sunar, "ECG-based Prediction Algorithm for Imminent Malignant Ventricular Arrhythmias using Decision Tree," *Plos one*, vol. 15, no. 5, p. e0231635, 2020.

[100] M. Janveja, R. Parmar, G. Trivedi, P. Jan, and Z. Nemec, "An Energy Efficient and Resource Optimal VLSI Architecture for ECG Feature Extraction for Wearable Healthcare Applications," in *2022 32nd International Conference Radioelektronika (RADIOELEKTRONIKA)*. IEEE, 2022, pp. 1–6.

[101] R. Parmar, M. Janveja, P. Jan, and G. Trivedi, "An Edge Enabled and Low Power VLSI Architecture to Classify Atrial Fibrillation for Wearable Devices," *Department of Eletronics and Electrical Enineering, IIT Guwahati*, India.

[102] J. N. Mitchell, "Computer Multiplication and Division using Binary Logarithms," *IRE Transactions on Electronic Computers*, no. 4, pp. 512–517, 1962.

[103] A. Laupacis, N. Sekar *et al.*, "Clinical Prediction Rules: A Review and Suggested Modifications of Methodological Standards," *Jama*, vol. 277, no. 6, pp. 488–494, 1997.

[104] W. H. Organization, *Fact sheet on CVDs*, 2017.

[105] E. Braunwald, A. S. Fauci, D. L. Kasper, S. L. Hauser, D. L. Longo, and J. L. Jameson, *Harrison's Principles of Internal Medicine*. McGraw Hill, 2001.

[106] V. Kumar, A. K. Abbas, N. Fausto, and J. C. Aster, *Robbins and Cotran Pathologic basis of Disease, Professional Edition E-book*. Elsevier health sciences, 2014.

[107] J. G. Canto, M. G. Shlipak, W. J. Rogers, J. A. Malmgren, P. D. Frederick, C. T. Lambrew, J. P. Ornato, H. V. Barron, and C. I. Kiefe, "Prevalence, Clinical Characteristics, and Mortality among Patients with Myocardial Infarction Presenting without Chest Pain," *Jama*, vol. 283, no. 24, pp. 3223–3229, 2000.

[108] H. Martin, U. Morar, W. Izquierdo, M. Cabrerizo, A. Cabrera, and M. Adjouadi, "Real-Time Frequency-independent Single-Lead and Single-Beat Myocardial Infarction Detection," *Artificial Intelligence in Medicine*, vol. 121, p. 102179, 2021.

[109] W. Liu, F. Wang, Q. Huang, S. Chang, H. Wang, and J. He, "MFB-CBRNN: A Hybrid Network for MI Detection using 12-Lead ECGs," *IEEE journal of biomedical and health informatics*, vol. 24, no. 2, pp. 503–514, 2019.

[110] F. Rincón, J. Recas, N. Khaled, and D. Atienza, "Development and Evaluation of Multilead Wavelet-based ECG Delineation Algorithms for Embedded Wireless Sensor Nodes," *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, no. 6, pp. 854–863, 2011.

[111] G. De La Fuente-Cortes, A. Diaz-Mendez, G. E. Flores-Verdad, and V. R. Gonzalez-Diaz, "On-Chip Fuzzy Logic Synthesis of a New Ischemic and Non-Ischemic Heartbeat Classifier," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 1, pp. 476–480, 2020.

[112] S. Obermann and M. Flynn, "Division algorithms and implementations," *IEEE Transactions on Computers*, vol. 46, no. 8, pp. 833–854, 1997.

[113] J. P. Martinez, O. Pahlm, M. Ringborn, S. Warren, P. Laguna, and L. Sornmo, "The STAFF III database: ECGs Recorded during Acutely Induced Myocardial Ischemia," in *2017 Computing in Cardiology (CinC)*. IEEE, 2017, pp. 1–4.

[114] V. Kumar, A. K. Abbas, N. Fausto, and J. C. Aster, *Robbins and Cotran Pathologic basis of Disease, Professional Edition E-book*. Elsevier health sciences, 2014.

[115] P. Wagner, N. Strodthoff, R.-D. Bousseljot, D. Kreiseler, F. I. Lunze, W. Samek, and T. Schaeffter, "PTB-XL, A Large Publicly Available Electrocardiography Dataset," *Scientific data*, vol. 7, no. 1, pp. 1–15, 2020.

[116] "Virtex-7 fpga family." [Online]. Available: https://www.xilinx.com/products/silicon-devices/fpga/virtex-7.html

[117] T. Reasat and C. Shahnaz, "Detection of Inferior Myocardial Infarction Using Shallow Convolutional Neural Networks," in *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*. IEEE, 2017, pp. 718–721.

[118] R. Magrans, P. Gomis, and P. Caminal, "Myocardial Ischemia Event Detection based on Support Vector Machine Model using QRS and ST Segment Features," in *2016 Computing in Cardiology Conference (CinC)*. IEEE, 2016, pp. 405–408.

[119] M. S. Manikandan and S. Dandapat, "Wavelet-based Electrocardiogram Signal Compression Methods and their Performances: A Prospective Review," *Biomedical Signal Processing and Control*, vol. 14, pp. 73–107, 2014.

[120] B. Singh, A. Kaur, and J. Singh, "A Review of ECG Data Compression Techniques," *International journal of computer applications*, vol. 116, no. 11, 2015.

[121] E. Chua and W.-C. Fang, "Mixed Bio-signal Lossless Data Compressor for Portable Brain-heart Monitoring Systems," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 267–273, 2011.

[122] C. J. Deepu and Y. Lian, "A Low Complexity Lossless Compression Scheme for Wearable ECG Sensors," in *2015 IEEE International Conference on Digital Signal Processing (DSP)*. IEEE, 2015, pp. 449–453.

[123] S.-L. Chen, G.-A. Luo, and T.-L. Lin, "Efficient Fuzzy-Controlled and Hybrid Entropy Coding Strategy Lossless ECG Encoder VLSI Design for Wireless Body Sensor Networks," *Electronics Letters*, vol. 49, no. 17, pp. 1058–1060, 2013.

[124] C. J. Deepu and Y. Lian, "A Joint QRS Detection and Data Compression Scheme for Wearable Sensors," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 1, pp. 165–175, 2014.

[125] A. Malik and R. Kumar, "Compression Techniques for ECG Signal: A Review," *Int. J. Modern Electron. Commun. Eng.*, vol. 4, no. 4, pp. 1–4, 2016.

[126] W.-J. Chen, C.-C. Lu, Y.-H. Jiang, and H.-W. Fang, "FPGA Design and Verification for Lossless ECG Coding Compression," in *2017 6th International Symposium on Next Generation Electronics (ISNE)*. IEEE, 2017, pp. 1–3.

[127] D. M. Ballesteros, D. M. Moreno, and A. E. Gaona, "FPGA Compression of ECG Signals by using Modified Convolution Scheme of the Discrete Wavelet Transform," *Ingeniare. Revista chilena de ingeniería*, vol. 20, no. 1, pp. 8–16, 2012.

[128] S.-L. Chen and J.-G. Wang, "VLSI Implementation of Low-power Cost-efficient Lossless ECG Encoder Design for Wireless Healthcare Monitoring Application," *Electronics Letters*, vol. 49, no. 2, pp. 91–93, 2013.

[129] G.-A. Luo, S.-L. Chen, and T.-L. Lin, "VLSI Implementation of a Lossless ECG Encoder Design with Fuzzy Decision and two-stage Huffman Coding for Wireless Body Sensor Network," in *2013 9th International Conference on Information, Communications & Signal Processing*. IEEE, 2013, pp. 1–4.

[130] N. Attarmoghaddam, S. H. Khayat, and M. Maymandi-Nejad, "FPGA Implementation of Power and Area Efficient Compressed Sensing for Multi-channel ECG Compression," in *2016 24th Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 2016, pp. 1027–1032.

[131] T.-H. Tsai and M. A. Hussain, "VLSI Implementation of Lossless ECG Compression Algorithm for Low Power Devices," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3317–3321, 2020.

[132] T.-H. Tsai, N.-C. Tung, and D.-B. Lin, "VLSI Implementation of Multi-channel ECG Lossless Compression System," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 8, pp. 2962–2966, 2021.

[133] J. Xue, "Adapting ECG Morphology Changes from Reduced-lead Set by Specifically Trained Algorithms for Acute Ischemia Detection," in *2007 Computers in Cardiology*. IEEE, 2007, pp. 709–712.

[134] A. Jas, J. Ghosh-Dastidar, M.-E. Ng, and N. A. Touba, "An Efficient Test Vector Compression Scheme using Selective Huffman Coding," *IEEE transactions on computer-aided design of integrated circuits and systems*, vol. 22, no. 6, pp. 797–806, 2003.

[135] T.-H. Tsai and W.-T. Kuo, "An Efficient ECG Lossless Compression System for Embedded Platforms with Telemedicine Applications," *IEEE Access*, vol. 6, pp. 42 207–42 215, 2018.

[136] S. Biswas and R. Biswas, "Data Encoding for IoHT ECG Application," in *2021 International Symposium on Devices, Circuits and Systems (ISDCS)*. IEEE, 2021, pp. 1–4.

[137] J. Sarma and R. Biswas, "A VLSI-Based Hybrid ECG Compression Scheme for Wearable Sensor Node," *IEEE Sensors Letters*, vol. 6, no. 4, pp. 1–4, 2022.

[138] Y.-F. Chen, W.-J. Huang, Y.-W. Zeng, H.-T. Hua, K.-F. Chang, and Y.-H. Chen, "Very-large-scale Integration of a Dual-lead Electrocardiogram Compression Chip with Modified Huffman Encoding," in *2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech)*. IEEE, 2022, pp. 427–428.

TM——$\mu\mu\mu\mu$