FPGA Based Compressed Heart Rate Changeability Recoding Algorithm

[1]S.Jeniton, [2]Karthick L S, [3]A.Suganya, [4]A.G.Paranthaman, [5]P.Latha, [6]B.Manojkumar

[1,2,4,5,6]Department of Electronics and Communication Engineering, [3]Department of Computer Science and Engineering

[1,6]Karpagam Institute of Technology, Coimbatore, India, [2]Rathinam Technical Campus, Coimbatore, India, [3]Dr. Mahalingam College of Engineering and Technology, Pollachi, India, [4]NPR College of Engineering and Technology, Natham, India, [5]V.S.B College of Engineering Technical Campus, Coimbatore, India

Abstract:-The paper mentioned describes a system for measuring a person's heart rate using Field-Programmable Gate Array (FPGA) technology. Here's a breakdown of the key points from the information provided:Heart Rate Measurement**: The system measures a person's heart rate by monitoring their heartbeats. It does this by processing a signal acquired by operational amplifiers (OPAMPs), specifically the LM328P model. Algorithm with Level Crossing Techniques: The system employs an algorithm that utilizes level crossing techniques to analyze the acquired signal and predict the heart rate. MIT-BIH Arrhythmia Database: The system was tested using the MIT-BIH Arrhythmia Database, which is a widely used dataset for heart rate analysis. In these tests, the system achieved impressive results:

- Average detection accuracy: 99.08%
- Sensitivity: 93.33%
- Positive prediction: 98.23%

Wearable and Compact: The algorithm is designed to extract the necessary information from the photoplethysmography (PPG) signal acquired from patients. This makes it suitable for creating a wearable and compact heart rate variability measuring system that is cost-effective. FPGA Specifications: The FPGA used in the system has a maximum operating frequency of 50 MHz. It can process the input and deliver the output in a very short time, with a minimum processing period of 19.617 nanoseconds. This makes it suitable for real-time applications. Low Power Requirement: The algorithm is designed to be power-efficient. It does not require multipliers or other power-hungry processing elements, resulting in a low power requirement. Resource Utilization: The FPGA's resource utilization is very efficient. The number of Slice Flip Flops utilized is only 1%, and the number of occupied slices is only 10% of the FPGA device. This means the system is highly resource-efficient, leaving plenty of room for other functions within the FPGA. In summary, this paper presents a heart rate monitoring system that uses FPGA technology and an algorithm based on level crossing techniques to accurately predict heart rates. It is designed to be power-efficient, suitable for wearable applications, and capable of delivering real-time results. The system's performance on the MIT-BIH Arrhythmia Database is impressive, with high accuracy, sensitivity, and positive prediction rates.

Keywords*: Electrocardiogram (ECG) signal, Level Crossing (LC) method, FPGA, MATLAB*

I. Introduction

The text you provided outlines key points from a paper discussing a novel algorithm for measuring heart rate, particularly using an asynchronous level-crossing sampling scheme. Below is a summary of the main topics covered in the text:Cardiovascular Diseases and Heart Rate: Cardiovascular diseases are a leading cause of death worldwide. Proper care and monitoring, especially regarding behavioral risk factors like hypertension, diabetes, and hyperlipidemia, can help prevent these diseases. Heart rate is an essential parameter in the cardiovascular system and can indicate a person's health and fitness. It varies with age and activity levels. Sampling Scheme: The paper describes an "activity-dependent" sampling scheme in which a new sample is taken only when there's a significant change in the input signal. This approach is particularly suited for sparse, burst-like signals like ECG signals. It's mentioned that this approach offers advantages over traditional Nyquist sampling, especially in data compression. Electrocardiograph (ECG) Signal: The electrical activity of the heart can be recorded using an electrocardiograph device, which produces an ECG signal. The paper focuses on the QRS complex in the ECG signal, with the R-peak being a crucial feature. The time interval between consecutive R-peaks, known as the RR interval, is used to calculate the heart rate and detect irregularities like arrhythmia. QRS-Detection Algorithms: Various algorithms have been developed for detecting the QRS complex, and many rely on amplitude-based techniques or more complex approaches, like wavelet-based detection and neural networks. However, these algorithms often involve complex computations that may not be suitable for body sensor networks (BSNs).Level-Crossing Sampling: To address the limitations of complex algorithms and power consumption in BSNs, the paper introduces an asynchronous level-crossing sampling scheme. This scheme optimizes the sampling rate based on signal activity, reducing power wastage. Algorithm for Heart Rate Measurement: The paper proposes a novel algorithm for measuring RR intervals from level-crossing sampled data. This algorithm is designed to improve power efficiency, data compression, and detection accuracy. MIT-BIH Arrhythmia Database: The system's performance is evaluated using data from the MIT-BIH Arrhythmia Database.Paper Structure: The text outlines the organization of the paper, including sections dedicated to explaining the asynchronous operation, introducing the heart rate measuring algorithm, discussing power consumption, performance comparisons, FPGA implementation, and concluding remarks.

In summary, the paper seems to present an innovative approach for measuring heart rate by using an asynchronous level-crossing sampling scheme, which is particularly well-suited for body sensor networks. The proposed algorithm aims to address issues of power efficiency and data compression while ensuring accurate heart rate measurements.

Ii. Asynchronous Sampling Technique

The text describes the key differences between synchronous and level-crossing sampling techniques and highlights the advantages of level-crossing sampling for certain types of signals, particularly sparse and burst-like signals. Here's a breakdown of the key points:

1. **Synchronous Sampling vs. Level-Crossing Sampling**: In synchronous sampling, a reference clock with a constant period (Ts) is used, resulting in regularly spaced samples with a fixed time interval between them. In contrast, level-crossing sampling is an irregular or asynchronous sampling method. It divides the signal's amplitude into quantization levels ($2M-1$ levels, where M is the number of bits or resolution of the algorithm), and a sample is taken only when the input signal crosses one of these quantization levels.

2. **Proportional Sampling Rate**: Level-crossing sampling's unique characteristic is that it adjusts the sampling rate based on changes in the input signal's amplitude. The sampling rate is proportional to the activity of the signal. This property can be advantageous for two types of signals:

- Sparse Signals: These are signals that are relatively constant most of the time but become active in small time intervals. Examples include speech, pressure, temperature, and various physiological signals like heart rate and neural signals. With level-crossing sampling, when the signal experiences minimal variations and no level-crossing occurs, the system effectively enters a sleep mode, conserving power by not converting and sampling data. This approach can significantly improve system efficiency.

- Burst-Like Signals: Signals like ECG, EEG, and EMG exhibit burst-like behavior, where there are short bursts of activity within longer periods of inactivity. Level-crossing sampling is well-suited for these signals as it allows for improved power efficiency by using a smaller hardware bit depth, resulting in lower data rates and reduced hardware complexity.

3. **Accuracy and Hardware Complexity**: Unlike conventional synchronous ADCs, the accuracy of level-crossing sampling is not limited by the number of quantization levels. In applications with burst-like signals, the

approach's power efficiency can be enhanced by using fewer hardware bits, reducing data rates, and simplifying hardware design.

4. **Illustration**: Figures 1 and 2 are likely used to visually represent the difference in the number of samples taken by the two sampling methods in active and inactive signal regions. The text indicates that for burst-like signals such as ECG, level-crossing sampling is particularly advantageous, as it reduces the number of samples taken during signal inactivity, conserving power and data transmission resources.

In summary, the text highlights the suitability of level-crossing sampling for signals with burst-like or sparse characteristics, such as ECG, due to its ability to adapt the sampling rate based on signal activity, thereby improving power efficiency and reducing data transmission requirements.

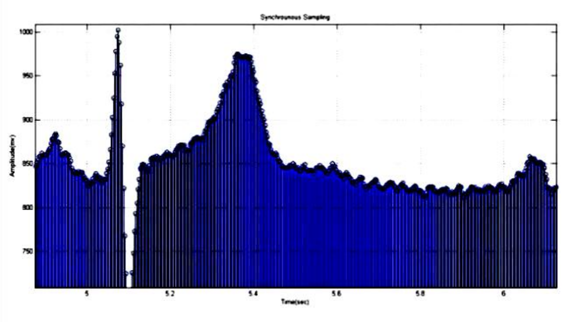

Fig.1 With the record (117): synchronous sampling

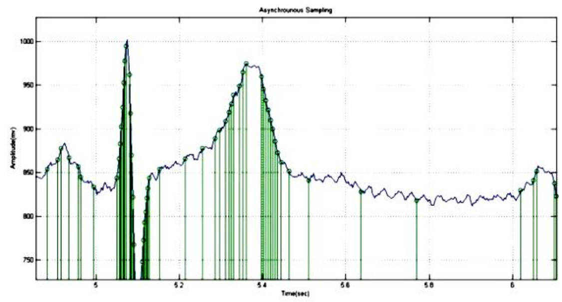

Fig.2 With the record (117): level-crossing sampling.

The text provides additional insights into the operation and advantages of the level-crossing ADC (Analog-to-Digital Converter) technique. Here are the key points:

1. **Dual Operation to Synchronous ADC**: In the level-crossing ADC technique, the operation is described as the "dual" of a synchronous ADC. In synchronous ADC, the sampling instants are precisely known, and the data values are quantized. However, in level-crossing ADC, the situation is reversed: the sampling instants are quantized, while the data values are precisely known.

2. **Measuring Time Intervals**: To measure the time interval between adjacent samples in level-crossing ADC, a local timer is used. This timer can be implemented as a counter with a constant period (TTimer), or it can be a counter that dynamically adjusts its period based on the input sample instances.

3. **Minimal Noise Introduction**: Unlike synchronous ADCs, level-crossing ADCs do not introduce inherent noise to the data values due to the limited number of quantization levels. The primary factor that can affect the accuracy of a level-crossing ADC is the resolution of the timer used.

4. **Asynchronous Signal Processing**: The text mentions the use of asynchronous signal processing methods for handling irregular level-crossing samples of value and time. In this context, asynchronous signal processing refers to techniques that can work with non-uniformly sampled data, such as the level-crossing samples. These methods are used to process the data, which includes both value and time information, to achieve specific goals. In this paper, these methods are applied to detect QRS complexes from ECG signals.

5. **Simplified Circuitry and Processing**: The use of the level-crossing ADC technique simplifies the design and avoids the need for additional circuitry to convert non-uniformly sampled data into uniform data. Additionally, there is no requirement for a special continuous-time digital signal processor. This simplification can reduce the complexity and cost of the overall system.

In summary, level-crossing ADC is described as a technique where the sampling instants are quantized, and the data values are precisely known. It can offer advantages in terms of noise reduction, simplification of circuitry, and the use of asynchronous signal processing methods to work with non-uniformly sampled data, making it particularly suitable for applications like QRS complex detection in ECG signals.

Iii. Heart Ratemeasuringsystem

The text discusses the uneven distribution of samples in level-crossing sampling compared to synchronous sampling. In level-crossing sampling, the sampling instants are not evenly spaced, and they depend on how fast the input signal changes. In addition to recording the signal values, it is necessary to record the sampling instants.

To handle this non-uniformly sampled data, various methods have been proposed. One such method is to quantize and represent the time intervals between samples (Dti) with a limited number of bits. This allows for the data to be transported and stored using a common synchronous system.

The text then introduces a new approach for R-peak detection in the context of Heart Rate Variability (HRV) systems. Traditional HRV systems typically involve several components, including electrodes, amplifiers, filters, ADCs, and processing units to detect R-peaks and calculate the patient's heart rate. However, the proposed approach is based on an asynchronous level-crossing ADC and non-uniformly spaced data processing.

Key points regarding this new approach include:

1. **Reduced Data Volume**: The new approach aims to considerably reduce the volume of data to be processed. This reduction in data volume can lead to lower power consumption, especially when transmitting data wirelessly. Since the system is signal-dependent, it focuses on the important data in the signal, reducing power consumption related to less critical data.

2. **Improved Detection Accuracy**: The proposed system is claimed to significantly improve the accuracy of R-peak detection compared to previous methods that use level-crossing (LC) techniques for QRS detection. This enhancement in detection accuracy is a noteworthy benefit, as accurate R-peak detection is essential for precise heart rate measurement and HRV analysis.

In summary, the text highlights a novel approach for R-peak detection in HRV systems that leverages asynchronous level-crossing ADC and non-uniformly spaced data processing. This approach aims to reduce data volume, lower power consumption, and improve detection accuracy, making it a potentially valuable advancement in heart rate monitoring technology.

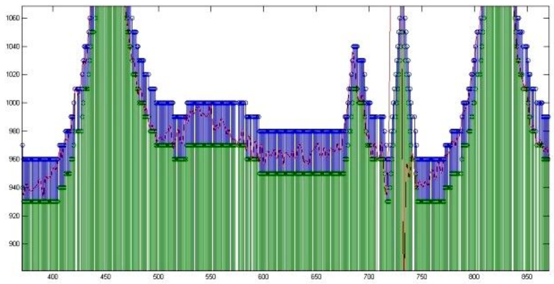

Fig.3 Operation of Level Crossing Algorithm

A. Usage of algorithm for R-peak Detection after compression

The text provides an in-depth discussion of several aspects related to the use of level-crossing sampling for QRS detection in ECG (Electrocardiogram) signals. Here's a breakdown of the key points:

Challenges in QRS Detection:

- **Noise Sources**: Detecting the QRS complex in ECG signals is challenging due to various noise sources, including muscle contractions, power-line interference, baseline drift caused by respiration, electrode contact noise, and motion artifacts. Additionally, large P and T waves in the ECG signal can interfere with the detection process.

- **Variability**: The basic morphological features of the QRS complex can vary from patient to patient, making it more challenging to develop a universal detection algorithm.

Filtering:

- Many QRS detection algorithms use filtering to attenuate unwanted parts of the ECG signal. Filtering can be implemented in the analog domain before the ADC or in the digital domain after the ADC. However, implementing filters in the analog domain can increase power consumption, and in the digital domain, it can burden the system with additional computation.

Level-Crossing Sampling and Quantization:

- Level-crossing sampling results in unevenly distributed samples, and the sampling instants are determined by the rate of change of the input signal.

- The levels at which quantization occurs are adjusted based on the input signal's crossing behavior.

- A quantization level is defined based on the ADC's input voltage amplitude range (AFS) and the ADC's resolution (M).

Adaptability and Drift Correction:

- The level-crossing method may require an initial time (e.g., 5 seconds) to settle and operate properly.

- Adapting the threshold value is essential to account for baseline drift. This adaptability can be achieved by considering the signal range in the previous second or by averaging the signal range over a few seconds.

Increasing the Gap (k) Between Quantization Levels:

- The gap (k) between quantization levels is typically set to 1 LSB (Least Significant Bit) to avoid missing level-crossings. However, a problem arises if the input signal experiences noise at the crossing point, leading to multiple, unwanted output samples.

- One solution is to increase the gap (k) to more than 1 LSB. This means samples are taken only when the signal changes more than 1 LSB in one direction or k-1 LSB in the opposite direction.

- Increasing k can reduce the number of sample points and improve noise filtering. Different values of k can be applied by loading proper values into registers, and there's no need for additional circuitry.

R-Peak Detection Using Level-Crossing Samples:

- Previous methods for QRS detection using level-crossing samples often used amplitude thresholding or conversion to uniform samples.

- Amplitude thresholding can be affected by baseline wandering or sudden amplitude changes in the signal, potentially reducing accuracy.

- Conversion to uniform samples requires additional conversion circuitry or algorithms, which may not be power-efficient.

- The proposed algorithm aims to detect R-peaks using non-uniformly sampled data generated by the level-crossing ADC. It is described as simple and accurate and can process non-uniform samples effectively.

The flowchart of the proposed R-peak detection algorithm is illustrated in Figure 4.

In summary, the text delves into the challenges of QRS detection in ECG signals, the use of level-crossing sampling, and the advantages of the proposed R-peak detection algorithm, particularly in handling non-uniformly sampled data. The ability to adapt the threshold and adjust the gap between quantization levels provides flexibility and accuracy in QRS detection.

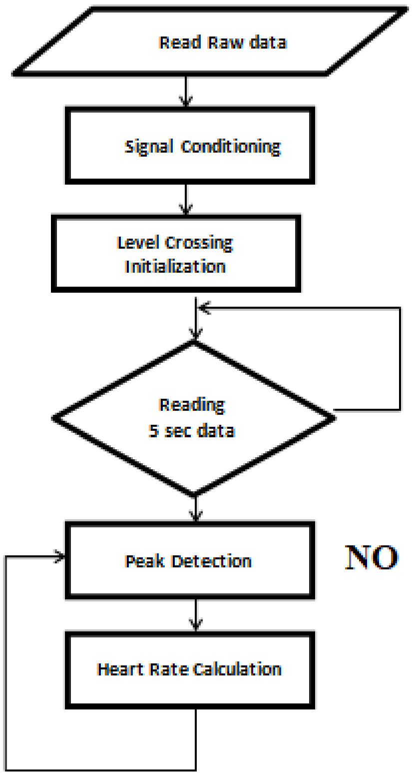

Fig.4 Flow chart of the LC Algorithm

The text outlines the specific steps and methodology involved in the proposed algorithm for R-peak detection in ECG signals using level-crossing sampled data. Here's a summary of the key steps and concepts:

1. **Direction-Based Detection**: The algorithm focuses on the change in the direction of the ECG signal (upward or downward) rather than the absolute signal values. This change in direction is instrumental in identifying signal peaks.

2. **Peak Detection**: The algorithm's second step involves detecting the peaks of the signal. This is accomplished by determining the maximum value within the current interval after identifying the lower and upper instant points.

3. **R-Peak Identification**: The main and critical step of the algorithm is identifying whether the detected peak is an R-peak. This is accomplished through thresholding, and it relies on making the threshold adaptive for each patient.

4. **Threshold Calculation**: The threshold for each patient is calculated based on the first 300 sampling instances (i.e., the first heart beat) of the ECG signal. To ensure adaptability, the next 300 sampling instances are also analyzed.

5. **Reference Time Intervals**: The time intervals (Dti) between two consecutive samples are used as a representation of the gradient (rate of change) of the input signal. Smaller time intervals indicate a steeper gradient. The time intervals of detected peaks from upcoming beats are noted and serve as a reference for finding the R-peaks.

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 04 (2023)

6. **Spike Duration and Recognition**: The counter value represents the duration of the detected spike. If the detected peak height is equal to or greater than a certain threshold (TH1) and the width of the peak is equal to or lesser than another threshold (TH2), the peak is identified as an R-peak. This is based on the understanding that R-peaks have certain characteristic features, including a relatively high amplitude and a short duration with a rapid change.

7. **ECG Signal Processing**: The text mentions that the algorithm has been tested on a real ECG signal, specifically a section of Tape 117 from the MIT-BIH Arrhythmia Database. The operation of the algorithm on this signal is demonstrated in Fig.5.

In essence, the algorithm leverages the gradient of the ECG signal, captures adaptive thresholds based on initial signal segments, and identifies R-peaks based on specific height and width criteria. This methodology allows for accurate R-peak detection, an essential component in ECG analysis for heart rate measurement and arrhythmia detection.

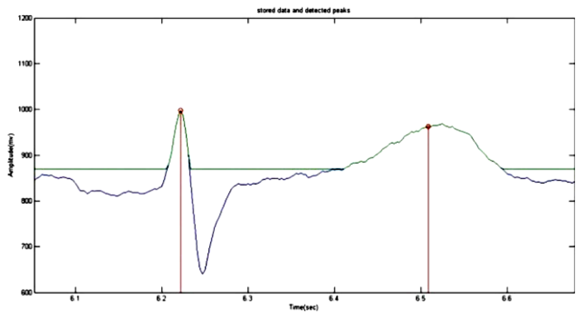

Fig.5R-peak detection

This portion of the text discusses further details about the algorithm's adaptability and the determination of parameter values for efficient QRS detection in level-crossing sampled data from ECG signals. Here are the key points:

1. **Adaptive Threshold Tuning**:

- The slope of R-waves and the amplitude of the QRS complex can vary among different patients and over time. To address this, the range of TH1 and TH2 values, which are used for peak detection, can be adjusted.

- TH1 is tuned adaptively during algorithm operation based on previously detected R-peaks, without using a multiplier operator. This adaptability reduces the sensitivity of the algorithm to threshold values.

2. **Dead-Time Zone for T-Waves**:

- To prevent false R-peak detection caused by fast high T-waves, a dead-time zone is established adaptively. This zone rejects any QRS detection occurring too close to the previous one.

3. **RR Interval Measurement**:

- The RR intervals, which represent the time between successive R-peaks, are crucial for heart rate analysis. This part of the algorithm can be realized by comparing the time interval between the current sample and the previous R-peak with a threshold (TH2).

- The value of TH2 is also selected based on the widths of previous R-peaks.

4. **Utilization of Level-Crossing Samples**:

- The proposed algorithm detects QRS complexes using unique features inherent in level-crossing samples, such as the gradient and time intervals between samples.

- Importantly, no additional circuitry is required to convert non-uniformly sampled data to uniform data, reducing the complexity of the algorithm and making it more efficient.

5. **Required Level-Crossings and Resolution**:

- The algorithm's proper operation depends on capturing a minimum number of W level-crossings from the R-wave. The number of samples taken from the R-wave is directly related to the algorithm's resolution (M).

- For simulation purposes, the text describes applying ten-second data from the MIT-BIH Arrhythmia Database to a MATLAB-modeled LC-ADC with various values of M and k (the gap between quantization levels).

- The goal is to determine the minimum required value of M for QRS detection. The text indicates that to keep missed beats within 0.05%, M should be greater than 7 bits for $k \geq 3$ LSB. A 7-bit LC-ADC can achieve lower average sampling frequency at the same k values compared to its 8-bit counterpart, making it preferable for noise immunity.

In summary, the text emphasizes the adaptability of the algorithm's parameters, particularly threshold values, and highlights the importance of selecting suitable values for M and k to ensure efficient QRS detection. The algorithm's performance and accuracy are assessed through simulation using data from the MIT-BIH Arrhythmia Database.

Iv Discussion On Power Consumption

The quantity of data samples collected can be used to support the low power consumption claim made for our method. As opposed to the standard Nyquist rate sampling technique, an adaptive asynchronous sampling technique produces about 40% fewer samples than the ordinary asynchronous sampling technique [6]. Additionally, the average sampling rate of the input signal is decreased by almost 5 times (67 Hz) [10]. This demonstrates the actual 5-times data compression of the signal. Additionally, the signal's compression has grown without any quantization mistakes when the base value of the signal is decreased during storage. Despite producing more samples than the adaptive asynchronous technique, the input-feature-correlated asynchronous sampling technique produces significantly fewer samples than the ordinary asynchronous technique.

1)Data generation power:

In contrast to the synchronous approach, which creates samples at a frequency higher than the Nyquist rate, the asynchronous approach has no external clock and relies only on input activity for sample creation. As previously stated, we produce less samples with our method than with the Nyquist rate ADC. Therefore, compared to the standard Nyquist rate sampling approach for data creation, the input-feature-correlated asynchronous sampling technique consumes relatively little power because there is no clock and less activity due to fewer output samples.

2) Storage and transmission power:

The sampled output data are saved, after which they are sent to the processing station for processing. The amount of samples directly relates to the storage and transmission power. The proposed method is anticipated to outperform the synchronous sampling method in terms of storage and transmission power because it generates fewer samples than the Nyquist rate sampling approach.

3) Processing power:

In our method, we produce indications for the distinctive properties of the signal at the ECG electrode and amplifier output directly. This can give the doctor early indications of the patient's status. The doctor can also examine the generated output digital samples to learn more about the signal's shape and other characteristics. In this scenario, there is no need for signal reconstruction, and our approach performs significantly better because there are less samples to process.

V Performance Evaluation

The proposed QRS detector is modeled and simulated in MATLAB. In order to evaluate the system functionality, the first channels of the ten second data of 48 half-hour ECG recordings of MIT-BIH Arrhythmia Database are used. The signals are passed through 7-bit LC algorithm with upper and lower quantization levels of 10LSB and then evaluated by the proposed algorithm. The performance of the algorithm, dealing with different abnormal ECG signals, is shown in Figs.6.1–6.3.

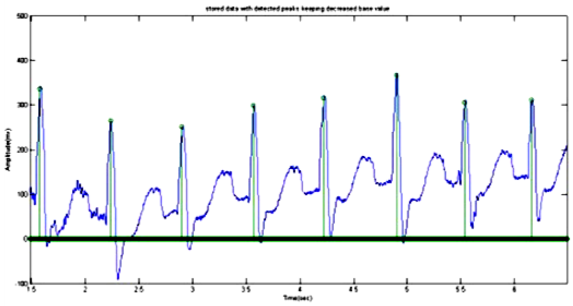

Fig.6.1 R-Peak detection of Record 109

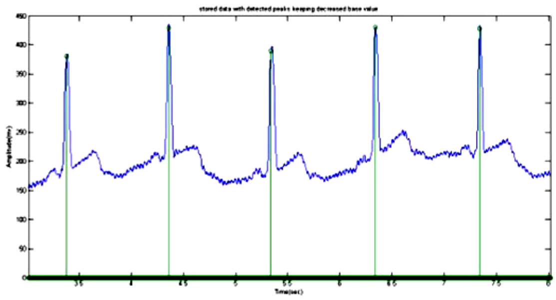

Fig.6.2 R-Peak detection of Record 121

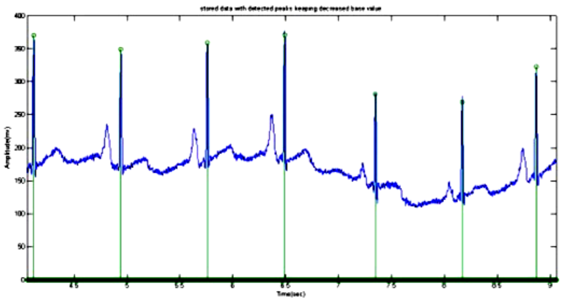

Fig.6.3 R-Peak detection of Record 222

In Figs.7, a detection example of the QRS using Tape 117 is shown. In Fig.7.1, the variation of quantization levels is clearly seen it provides the direction of the incoming signal. And above threshold when the signal crosses these two quantization levels that particular instant and the amplitude values are noted. Thus it indirectly compresses the signal by avoiding the noisy signal parts or signal with little information.

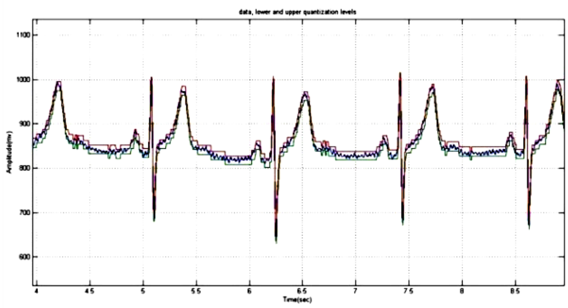

Fig.7.1 Variations of Quantization Levels

Fig.7.2 shows the original signal and the necessary part which is essential to detect the Heart Rate of the patient. Fig.7.3 shows the signal with its R-Peak identification with low base value. As it can be seen, nearly all the samples are taken from the QRS regions and no power is wasted to sample the silent parts of the signal with small variations.

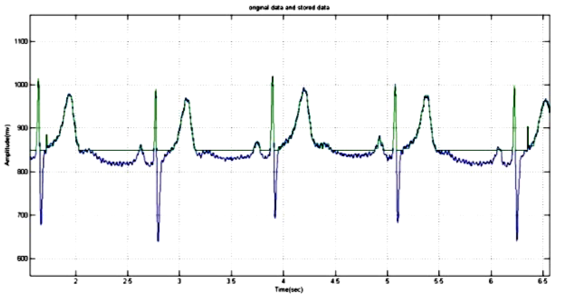

Fig.7.2 Stored values of ECG signal

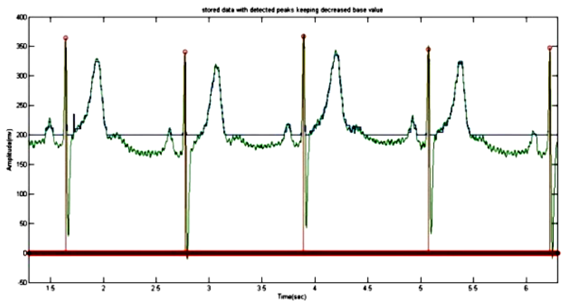

Fig.7.3 Detected R-Peaks Highlighted

These demonstrate the ability of the algorithm to handle large P or T waves and irregular ECG signals, respectively. From all examples, it can be seen that the proposed system correctly detects the QRS complexes of the ECG signal, even under the presence of baseline drift, severe noise, large P or T waves, and irregular ECG waveform. The no of samples required to store the given signal get reduced. All the unwanted samples which are within these upper and lower levels are ignored. From the stored signal peaks are detected. This algorithm produces an average compression ratio of less than 50 percent. Fig.8 shows the Heart Rate of the record 100 of 10seconds. If there exist any large deviation then there will be disorder in the Heart.

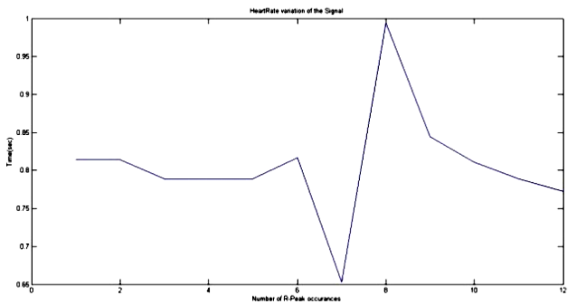

Fig.8 Heart Rate variation of Record 1 (10 seconds)

In order to evaluate the performance of the QRS-detection algorithm, the numbers of failings to detect an R-peak (false negative or FN) and false beat detection (false positive or FP) , Total Beats presented in 10sec data, TP(True Positive- actual R-Peaks detected by algorithm) are calculated using the standard database. The results are reported in Table I. Using these values, three other parameters of sensitivity (Se), positive prediction (+P), and detection error rate (DER) are calculated and reported, which are defined as follows:

$$Se = \frac{TP}{TP + FN}$$

$$+P = \frac{TP}{TP + FP}$$

$$DER = \frac{FP + FN}{TotalBeat}$$

The no of bits actually required to decide the heart rate is the QRS samples and its instant. This method stores only the important portions of the given signal. As an example for record 100 the actual samples are 3600 for ten second data as it is sampled at 360 samples per second. But the algorithm compresses the record 100 from 3600 to 109 samples and for record 117 from 3600 to 412. And it produces an average compression ratio of less than 40 percent. Thus Data Compression is also good in this algorithm.

$$CompressionRatio = \frac{Storedsize}{Actualsize}$$

Table 1: Mit-Bih Ecg Atrial Fibrilation Database (10sec) [1]

No	Tape	Total Beat	TP	FN	FP	Se (%)	+P (%)	DER (%)	Compression	
									Samples	Ratio
1	100	13	13	0	0	100	100	0	109	3.02
2	101	11	11	0	0	100	100	0	143	3.97
3	102	11	11	0	0	100	100	0	495	13.75
4	103	11	11	0	0	100	100	0	480	13.33
5	104	13	12	1	0	92.3	100	0.0769	857	23.80
6	105	14	14	0	0	100	100	0	439	12.19
7	106	10	10	0	0	100	100	0	719	19.97
8	107	12	11	1	0	91.66	100	0.0833	1164	32.33
9	108	11	6	5	1	54.5	85.7	0.5454	229	6.36
10	109	16	16	0	1	100	94.1	0.0625	550	15.27
11	111	12	12	0	1	100	92.31	0.0833	823	22.86
12	112	14	14	0	0	100	100	0	258	7.16
13	113	9	9	0	0	100	100	0	633	17.58
14	114	9	9	0	2	100	81.82	0.2222	628	17.44
15	115	10	10	0	0	100	100	0	98	2.72
16	116	13	13	0	2	100	86.67	0.1538	1552	43.11
17	117	9	9	0	0	100	100	0	412	11.44
18	118	12	12	0	0	100	100	0	171	4.75
19	119	10	8	2	0	80	100	0.2	455	12.63
20	121	10	10	0	0	100	100	0	352	9.77
21	122	15	14	1	0	93.3	100	0.0666	241	6.69

No	Tape	Total Beat	TP	FN	FP	Se (%)	+P (%)	DER (%)	Compression	
									Samples	Ratio
22	123	8	8	0	0	100	100	0	261	7.25
23	124	8	7	1	1	87.5	87.5	0.125	540	15.00
24	200	8	8	7	0	53.3	100	0	368	10.22
25	201	14	14	0	0	100	100	0	394	10.94
26	202	9	9	0	0	100	100	0	124	3.44
27	203	18	16	2	2	88.8	82.2	0.0222	1354	37.60
28	205	15	14	1	0	93.3	100	0.066	137	3.80
29	207	5	4	1	0	80	100	0.2	42	1.16
30	208	16	15	1	0	93.75	100	0.0625	1228	34.11
31	209	15	14	1	0	93.33	100	0.0666	101	2.80
32	210	15	14	1	0	93.33	100	0.0666	1125	31.25
33	212	15	14	1	0	93.33	100	0.0666	1316	36.55
34	213	18	17	1	0	94.44	100	0.0558	638	17.72
35	214	12	11	1	0	91.66	100	0.0833	844	23.44
36	215	18	17	1	0	93.33	100	0.0558	1025	28.47
37	217	12	12	0	0	100	100	0	1227	34.08
38	219	13	13	0	0	100	100	0	176	4.88
39	220	12	11	1	0	91.66	100	0.0833	1473	40.91
40	221	13	13	0	0	100	100	0	621	17.25
41	222	13	10	3	0	76.92	100	0.2307	54	1.5
42	223	13	13	0	0	100	100	0	1600	44.44
43	228	12	12	0	0	100	100	0	665	18.47
44	230	14	13	1	0	92.85	100	0.0714	450	12.50
45	231	10	10	0	0	100	100	0	1493	41.47
46	232	8	7	1	0	87.5	100	0.125	547	15.19
47	233	17	13	4	0	76.47	100	0.2352	337	9.36
48	234	15	14	1	0	93.3	100	0.0666	151	4.19
TOTAL		588	548	40	10	93.19	98.20	0.0850	29099	16.83

The above table inferred that the Sensitivity of the algorithm is about 94percent. And the Positive prediction is about 98 percent. The minimum Detection Error Rate is about .08 only. Compared to other algorithms this LC algorithm produces good positive prediction and DER ratio. Also the Compression Ratio is also high in the order of 16. As the samples are stored with decreased base level, it actually requires 9 bits per sample instead of

11bits. As in asynchronous sampling the 7 bit timer value also needs to be stored, an average of 8 times the no of bits get reduced totally.

Vifpga Implementation Of The Algorithm

The ECG Pulse Acquisition is taken in real time using PPG principles as change in blood volume is directly proportional to the Heart Rate of the patient. For acquiring this signal from finger the LM324 Driver IC is used as an Instrumentation Amplifier see Fig.9 to build up the small variations occur in the blood vessels as an output voltage of 5v i.e. whenever a pulse is detected by the amplifier section it provides a pulse output and at times other than the detection time it provide constant dc voltage.

The number of the samples taken from the R-wave is directly related to the resolution of the algorithm (*M*). Using the LC sampled output data, the possible locations of the beats are estimated utilizing simple peak detection and the numbers of missed beats are measured using beat-by-beat comparison.

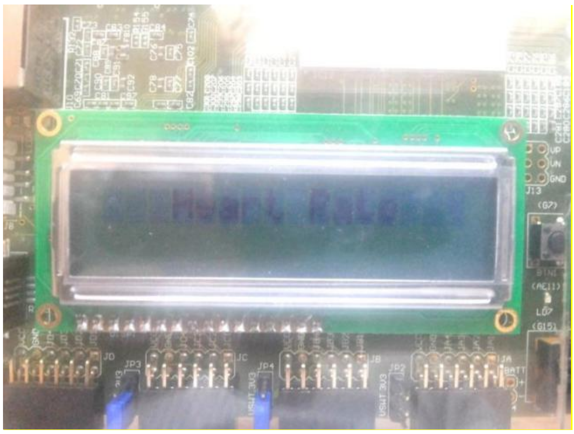

Fig.9 ECG Pulse Acquisition Unit

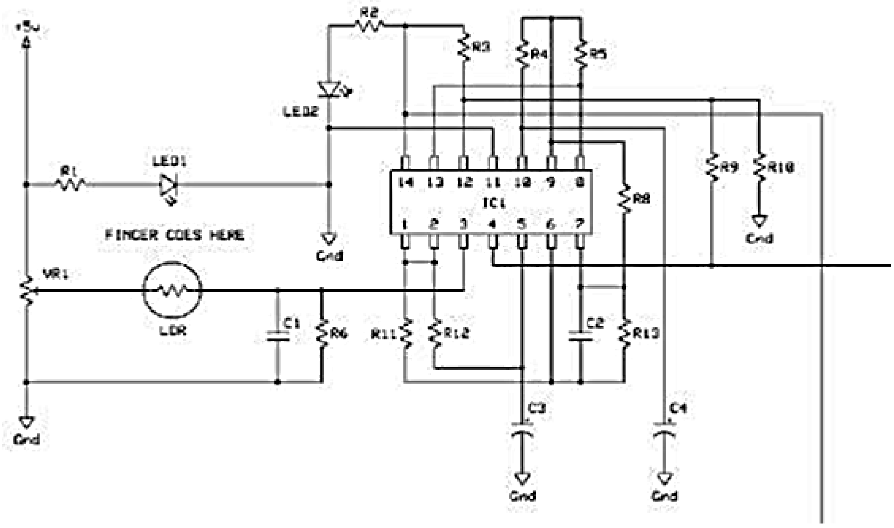

Fig.10 FPGA LCD Module

This setup provides output for every five seconds with overlapping of 55seconds from the last session so that the current Heart Rate will be the updated one with respect to time. Fig.10 depicts the initial display content of the prototype design through the LCD module of the vertex 5 FPGA kit displaying “Heart Rate”. After 5 seconds, the LCD module displays the Heart Rate Variation values e.g. Fig.11 shows the binary value of 23.

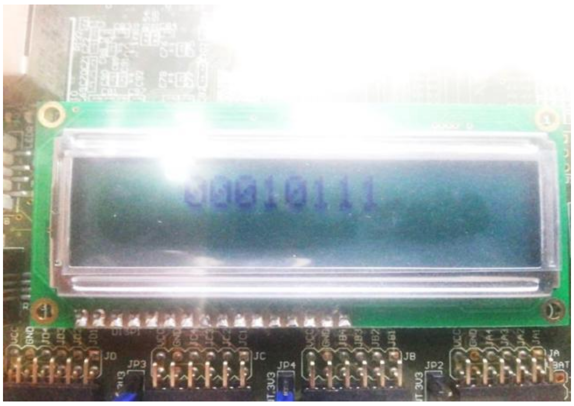

Fig.11 Heart Rate value in binary format (for 23)

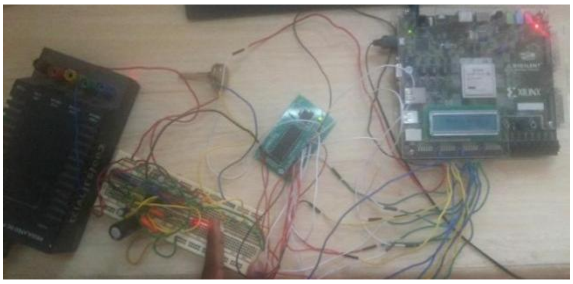

Fig.12 Overall view of Prototype Design

The overall prototype setup is shown in the Fig.12 and it consist of the power supply, Signal Acquisition, ADC module and FPGA Virtex 5 kit.

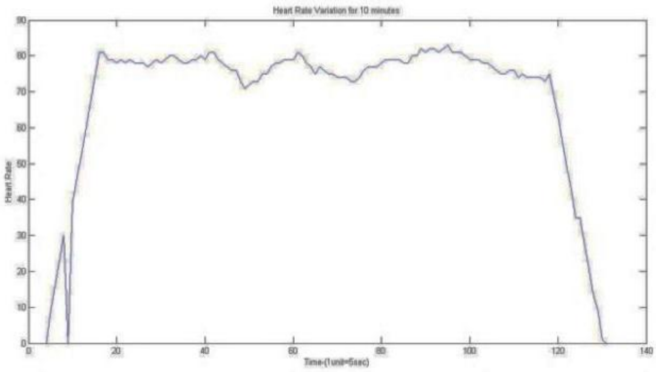

Fig.13 Heart Rate Variation for 10 Minutes

Fig.13 infers that 360 discrete samples are chosen for fixing the threshold at first. Then it is followed by tracing the upcoming signals using Quantization Lower and Upper Levels for detection of level crossing. The instant at which the signal crosses the threshold is updated for every 5 sec by the algorithm. It is checked for providing the Peak of that particular level crossing. Using the general constraint of the R-peak, the peak is concluded as whether it is R-peak or not. The respective values are viewed through ISIM Simulator provided by Xilinx.

Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 04 (2023)

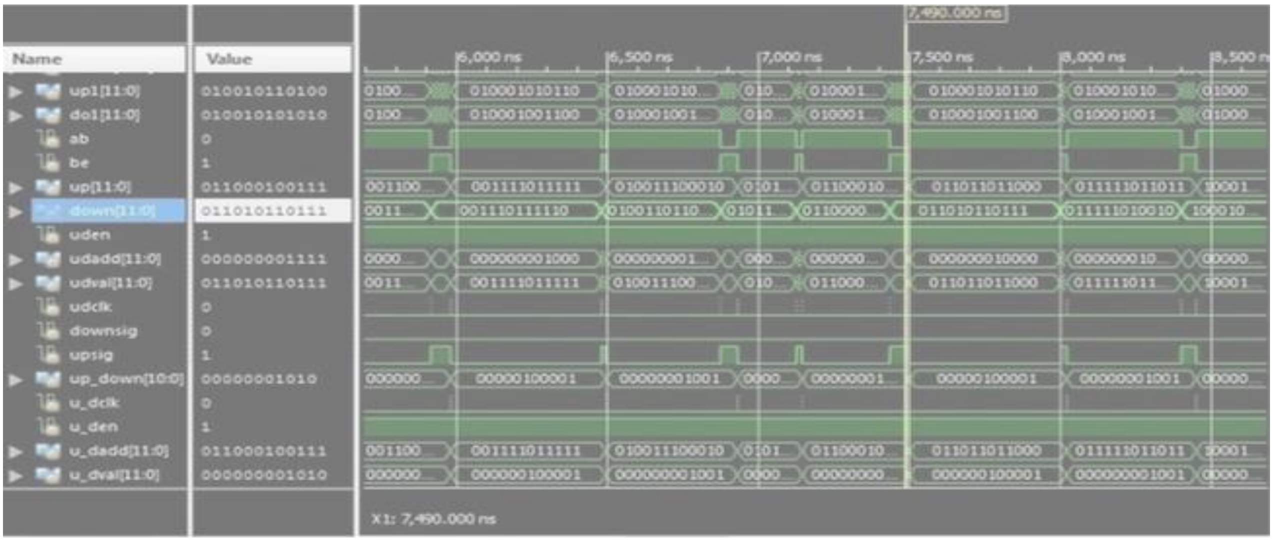

Fig.14 ISIM Output for storing the sample values above the threshold

```
Macro Statistics
# Adders/Subtractors                                  : 10
 11-bit adder                                         : 6
 11-bit subtractor                                    : 1
 12-bit adder                                         : 2
 14-bit subtractor                                    : 1
# Counters                                            : 8
 11-bit up counter                                    : 6
 12-bit up counter                                    : 1
 28-bit up counter                                    : 1
# Accumulators                                        : 1
 12-bit up accumulator                                : 1
# Registers                                           : 1
 11-bit register                                      : 1
# Latches                                             : 18
 1-bit latch                                          : 8
 11-bit latch                                         : 2
 12-bit latch                                         : 8
# Comparators                                         : 83
 11-bit comparator greatequal                         : 6
 11-bit comparator greater                            : 31
 11-bit comparator less                               : 6
 11-bit comparator lessequal                          : 35
 12-bit comparator greatequal                         : 1
 12-bit comparator less                               : 2
 14-bit comparator equal                              : 1
 14-bit comparator not equal                          : 1

---------------------------------------------------------------
```

Fig.15 Resource Utilization

The Resource Utilization of the algorithm for efficient R-Peak detection is shown in Fig.15. As it does not use any multipliers and any other hard processing elements, the power requirement of the algorithm gets reduced considerably. The maximum frequency of the FPGA is 50 MHz and the minimum period required for the input being processed and providing the output is 19.614ns only from Fig.16. In Body sensor network, circuits like this will really reduce the power and the time to process the input signal.

```
Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 19.617ns (Maximum Frequency: 50.976MHz)
   Minimum input arrival time before clock: 3.090ns
   Maximum output required time after clock: 6.492ns
   Maximum combinational path delay: No path found
```

Fig.16 Timing Summary of the design

Design Goal:	Balanced	•Routing Results:	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	•Timing Constraints:	All Constraints Met
Environment:	System Settings	•Final Timing Score:	0 (Timing Report)

Device Utilization Summary				[-]
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	99	9,312	1%	
Number of 4 input LUTs	703	9,312	7%	
Number of occupied Slices	470	4,656	10%	
Number of Slices containing only related logic	470	470	100%	
Number of Slices containing unrelated logic	0	470	0%	
Total Number of 4 input LUTs	871	9,312	9%	
Number used as logic	703			
Number used as a route-thru	168			
Number of bonded IOBs	26	190	13%	
Number of RAMB16s	10	20	50%	
Number of BUFGMUXs	2	24	8%	
Average Fanout of Non-Clock Nets	3.68			

Performance Summary			[-]
Final Timing Score:	0 (Setup: 0, Hold: 0)	Pinout Data:	Pinout Report

Fig.17 Design Summary of the Algorithm

The Whole Design Summary of the Prototype is given in Fig.17. The Number of Slice Flip Flops used is only 1%. And the Number of occupied slices is only 10% of the FPGA Device. The maximum number of RAM used i.e. 50%. The prototype is checked with 5 different persons and it shows good results.

Table 2: Average Heart Beat Of Different Samples

Name	Heart beat	
	Actual	Measured
Person 1(Kumar)	68	70
Person 2 Kannan)	73	75
Person 3 (Kalaiselvan)	66	67
Person 4(Kavin)	64	66

VII. Conclusion

For the resting and working heart, a computationally effective cardiac pulse measurement algorithm is described. The algorithm can be simply implemented without multipliers and does not require prior knowledge of the Heart signal's frequency spectrum. It is extremely appealing for wearable devices in BAN applications because of these qualities. It is demonstrated that the system may be implemented with less complexity and less power consumption than the traditional regular-sampling synchronous systems by lowering the processing rate to less than 5 Hz and using a small number of samples to make the choice. The system performs well when tested against the MIT/BIH database, with 99.61% detection rate, 99.81% sensitivity, and 99.80% positive prediction rates. The performance of the proposed algorithm might be enhanced by applying techniques such as search-back and adaptive thresholding, at the cost of more power consumption. Indirectly it compresses the signal and stores only the special features of the given signal in order to determine the R-peaks and Heart Rate.

References:

[1] *Ripoll, V.J., Wojdel, A., Romero, E., et al.: 'ECG assessment based on neural networks with pretraining', Appl. Soft Comput., 2016, 49, pp. 399–406*

[2] *Raman, P., Ghosh, S.: 'Classification of heart diseases based on ECG analysis using FCM and SVM methods', Int. J. Eng. Sci., 2016, 6, pp. 6739–6744*

[3] *R. ShanthaSelvaKumari, D. Raja Godwin, "Data Compression using Event Driven QRS Detection Algorithm on Fibrillation Database"IJAER,Vol. 10, no. 6,pp. 5335-5340, Feb. 2015.*

[4] *Mehta, S.S., Lingayat, N.S.: 'Comparative study of QRS detection in single lead and 12-lead ECG based on entropy and combined entropy criteria using support vector machine', J. Theor. Appl. Inf. Technol., 2007, 3, (2), pp. 8–18*

[5] *F. Maravasti, Nonuniform Sampling Theory and Practice. New York: Kluwer, 2001.*

[6] *Silipo, R., Marchesi, C.: 'Artificial neural networks for automatic ECG analysis', IEEE Trans. Signal Process., 1998, 46, (5), pp. 1417–1425*

[7] *J. Mark and T. Todd, "A nonuniform sampling approach to data compression," IEEE Trans. Commun., vol. 29, no. 1, pp. 24–32, Jan. 1981.*