Towards Smart Health Monitoring System for Elderly People

Achraf Ben Ahmed, Yumiko Kimezawa, Abderazek Ben Abdallah
The University of Aizu,
Graduate School of Computer Science and Engineering,
Adaptive Systems Laboratory,
Aizu-Wakamatsu 965-8580, Fukushima, Japan.

Abstract—Recent technological advances in sensors, low-power microelectronics, and wireless networking enabled the proliferation of wireless sensor networks for wide applications. One of the most promising applications of sensor networks is for health monitoring.

In this work, we present an interactive real-time (IRT) interface integrated with a multi-lead period-peak detection (PPD) algorithm for ECG processing. The ultimate goal of our system (BANSMOM\(^1\)) is accurate monitoring and study of cardiovascular and other complex biomedical signals for elderly people.

We describe the system’s main components architecture with special focus on the interactive real-time monitoring interface in a fair amount of details. We also present hardware and software prototyping results of the proposed system.

Keywords—Elderly Monitoring; Interactive real-Time Interface, Parallel Processing; Prototyping

I. INTRODUCTION

Despite the decreased mortality rate, heart diseases and their associated complications are one of the main causes of death around the world. As a result, detection of irregularities in the rhythms of the heart is a growing concern in medical research.

Using embedded health monitoring system, a wide range of parameters must be available and processed. Thus, multiple tasks must be performed in order to obtain accurate diagnosis. In most cases, complex computation is required because of the applied detection algorithm. When real time diagnosis needs to be done a single medium-performance processor is generally adopted. However, the development of such systems faces several challenging tasks since they need to often address conflicting requirements for performance, size, and accuracy.

Electrocardiography is an essential practice in heart-related diseases. It faces many computational challenges, especially when 12 or more lead signals should be analyzed in parallel, at real-time, and under high sampling frequencies. Another challenge is the analysis of huge amounts of data that may grow depending on the recording range time.

Recent techniques deployed for monitoring heart activity is the 12-lead ECG, which uses data coming from twelve ECG leads serially. The leads produce huge amounts of data, especially when used for a long number of hours. The most important points for the ECG signal are the peaks: P, Q, R, S, T, and U. Each peak is related to an important heart action in the medical analysis. Many ECG analysis methods use the three peaks Q, R, S and the corresponding intervals between these three peaks. In biomedical terms, this interval from Q to R to S is known as the QRS complex [1], [2]. The well known QRS Pan-Tompkins algorithm locates R-peaks in the ECG signal and calculates the heart period [3].

Traditionally, personal medical monitoring systems, such as ambulatory electrocardiography devices [4], have been used to record data. Data processing and analysis are performed off line, making such devices impractical for continual monitoring and early detection of medical disorders, especially for patients needing immediate medical interventions.

A number of recent research efforts focus on hardware implementations of health monitoring systems. Christos presented a hardware implementation of the Pan and Tompkins QRS detection algorithm [5]. The system achieved a speed up of 250% compared to the software implementation. Multiprocessor system on chip (MPSoC) are high performance devices that incorporate multiple building blocks from multiple sources. An MPSoC can contain general or special purpose fully programmed processors, co-processors, DSPs, dedicated hardware, memory blocks, etc. These systems are becoming a common design alternative in portable devices because it is possible to manufacture a silicon chip including only necessary elements. Certain medical applications require devices capable of providing very accurate information about monitored patients in places where complex clinical systems are not available and where also parameters such as electrocardiogram (ECG) characteristics are important to be determined.

In this paper, we present a new monitoring system based on an interactive real-time (IRT) interface and a multi-lead period-peak detection (PPD) algorithm for ECG processing [6]. The ultimate goal of this work is to develop a smart embedded system to monitor elderly people and promote well-being by introducing smart in-body sensors that allow medical professionals to monitor elderly health and initiate interventions at the home environment.

---

\(^1\)This project is partially supported by Competitive research funding, Ref.P36-37. http://www.bansmom.org
II. OVERVIEW OF PPD ALGORITHM

The Period-Peak-Detection Algorithm [6] detects first the period and then looks for all peaks. We detect first the period before finding peaks because there is a high degree of randomness in the ECG signals. The randomness makes finding peaks an erroneous process. What we would get by doing this is the level of correlation that these signals have.

Our PPD algorithm computes the required parameters: heart period, typical peaks (P, Q, R, S, T, and U), and inter-peak time spans (R-R interval). Peak height and inter-peak time ranging outside normal values, indicating different kinds of diseases, are detected with our algorithm. The algorithm consists of two execution flows: one finds the period using the auto-correlation function, and the other one finds the number, amplitude and time interval of the peaks.

A. Period Detection

As indicated in Figure 1, our PPD algorithm consists of 4 phases: (1) Data reading, (2) Derivation, (3) Auto-correlation, and (4) Find intervals.

The derivation phase finds the discrete derivative of the ECG signal. The derivative function is the best function we can run that can aid in amplifying signal peaks. Therefore, after reading the data from the signal y with samples from memory, a very helpful step is to calculate the derivative of the signal y(t). The following equation is the derivative function used by this algorithm:

\[
\frac{\partial y}{\partial t}(t) \approx \frac{y[n + 1] - y[n]}{(n + 1) - n} = y[n + 1] - y[n]
\]

With this derivative, a peak will be amplified relatively to the samples before it. If the values of y[n] and y[n + 1] are close to each others (i.e. no peaks), then the difference will look relatively small on the new derivative graph. The advantage of taking the derivative is that the fluctuations taking place in the signal, especially those around the peaks, would be reduced to a near-zero-value. In addition, performance overhead associated with derivative calculation of the ECG signal is negligible compared to the rest of the algorithm. The autocorrelation phase finds the period of ECG signals. This phase uses autocorrelation function (ACF). The ACF, shown in (2), is a statistical method used to measure the degree of association between values in a single series separated by some lags. The fixed length of ACF is defined by (3). By running the ACF on the function y over the recorded data sample, we can get the coefficients of the ACF. The following equation is the used autocorrelation function:

\[
R_y[k] = \sum_{n=-\infty}^{\infty} y[n] \times y[n - k]
\]

\[
R_y[L] = \sum_{n=0}^{N} y[n] \times y[n - L]
\]

where \(R_y\) is the autocorrelation function, \(y[n]\) is the filtered ECG signal. \(L\) is a positive natural number related to the number of times needed for the calculations to get the period. It is the same as the number of lags of the autocorrelation. Finally, the find interval phase finds interval point in ECG signals based on results of the autocorrelation phase.

B. Peaks Detection

The peaks detection phase consists of 3 main phases: Extraction, Discrimination and Store results. The extraction phase discriminates significant peaks from calculated interval information in period detection flow. The discrimination phase finds 6 peak points (P, Q, R, S, T and U) from the extracted peaks [7], [8]. Finally, the Store results phase stores interval and peak information in a buffer that is reused for calculations period detection.

III. SYSTEM ARCHITECTURE

Figure 2 shows the BANSMOM system architecture. Signal processing is performed in four major phases: (1) signal reading, (2) filtering, (3) analysis, and (4) display.

A. Signal reading

The number of the data inputs from the sensors is extensible to 15 or more. The size of data is included in data read from the sensors. Analog Digital Converter (ADC) converts analog data into a digital one. So a contiguous ECG signal is converted into a discrete ECG signal. As a result, filter processing and analysis processing can easily be done.

B. Filtering

Noise filtering uses a bandpass filter that is based on the Finite Impulse Response (FIR) filter. The bandpass filter reduces the influence of muscle noise, 50 Hz interference, baseline wander, and T-wave interference.

Digital filters process digitized or sampled signals. A digital filter computes a quantized time-domain representation of the convolution of the sampled input time function and a representation of the weighting function of the filter. They are realized by an extended sequence of multiplications and additions carried out at a uniformly spaced sample interval.
The digitized input signal is mathematically influenced by the DSP program. These signals are passed through structures that shift the clocked data into summers (adders), delay blocks and multipliers. These structures change the mathematical values in a predetermined way: the resulting data represents the filtered or transformed signal.

Digital filters are very important parts of DSP. They have two uses: signal separation and signal restoration. Signal separation is needed when a signal has been contaminated with interference, noise, or other signals. For example, imagine a device for measuring the electrical activity of a baby’s heart (EKG) while still in the womb. The raw signal will likely be corrupted by the breathing and heartbeat of the mother. A filter might be used to separate these signals so that they can be individually analyzed. Signal restoration is used when a signal has been distorted in some way. For example, an audio recording made with poor equipment may be filtered to better represent the sound as it actually occurred.

Finite Impulse Response (FIR) filter is a basic type of digital filter. FIR filters have no non-zero feedback coefficient in the general form of the digital filter difference equation. That is, the filter has only zeros, and once it has been excited with an impulse, the output is present for only a finite N number of computational cycles. The FIR filter uses noise rejection and waveform extraction for the ECG algorithm. The data from analog/digital converter is finite and is a discrete digital signal; therefore, our system uses FIR filter. This filter is popular among liner digital filters and the most safety in another filter within finite data. The FIR filter is composed of three parts: delay element, multiplier, and adder. The following equation is the difference equation for FIR filter that is defined by the relationship between the input signal and the output signal.

$$y[n] = a_0 x_n + a_1 x_{n-1} + \cdots + a_N x_{n-N}$$  \hspace{1cm} (4)  

$$y[n] = \sum_{i=0}^{N} a_i x_{n-i}$$  \hspace{1cm} (5)

N is the filter’s order that corresponds to the number of taps. $x_n$ are current or previous filter’s inputs. $y[n]$ is the current filter output. $a_i$ are the filters’ coefficients that correspond to the impulse response. A FIR filter works by multiplying an array of the most recent n data samples by an array of constants, and summing the elements of the resulting array. The filter then inputs another sample of data and repeats the process.

C. Data analysis and display

Correlation is calculated between the acquired segment and a pattern which has been previously obtained. For each analyzed patient, the pattern segment is assumed to contain
regular ECG signals where a signal’s QRS complex is contained so that any further ECG pulses can be correlated with it. High correlation values correspond to the pulse detection. Pattern extraction is performed by the main processor. Once obtained, it is transferred and stored to the off-chip memory. The off-chip memory starts to receive data samples directly from the ADC. Data samples are stored in a Queue (FIFO). The correlation pattern is calculated, and the pulse alignment is evaluated. When the input signals in the Queue are aligned with the pattern, a high correlation value will be obtained, and a signal indicating the presence of a new pulse is generated.

External monitor outputs the analyzed results. The output data are peaks of each typical wave (P, Q, R, S and T), heart rate and entire waveform. While running the system, the external monitor outputs the results at real-time.

Table II shows the test results for several over several leads. On average, our algorithm achieves about 69% accuracy. We found that these results are promising and satisfy our real-time processing requirement when considering our leads reading speed.

We used real sample data from PhysioBank database[9] for testing the correctness and the accuracy of our system.

**Figure 3. PPD Processing Module High-Level View Block Diagram.**

**Figure 4. Get live data (a) and get previous data (b).**

**IV. PROTOTYPING AND EVALUATION RESULTS**

The system was implemented in Verilog HDL and prototyped on a commercial FPGA board. Figure 3 shows the block diagram of the PPD main processing module block diagram. The master module consists of a Nios II processor, four on-chip memories (used for raw ECG data storage, processor memory, shared memory and virtual external memory), an interrupt timer, a graphics LCD controller, a LED controller and a JTAG UART (used for connection with host PC). The PPD module consists of 12 reconfigurable Nios II cores, on-chip memory and an interrupt timer. The FIR filter module is generated by the Altera MegaCore Function. The specification of this filter is as follows: filter step is 51, sample rate is 128, and cutoff frequency is from 5-15 Hz. Table I shows the hardware complexity results of the BNSMOM system. In 1-lead system, the logic utilization is about 14%. The total block memory bits is about 21%. The total power dissipation is about 677mW, and the system speed is 97.89MHz.

We used real sample data from PhysioBank database[9] for testing the correctness and the accuracy of our system.

**V. MONITORING SOFTWARE DESCRIPTION**

The monitoring part is crucial for the real time diagnosis that we want to perform in our BANSNOM system. This importance came from the fact that in order to make clinical studies or heart diseases diagnosis, researchers and doctors are required to be in immediate proximity to patients. Therefore, conventional monitoring approaches are impractical since patients and research facilities are often geographically distributed. The existing methods of ECG monitoring are characterized by a manually-intensive work flow for data acquisition, formatting and visualization. Besides, they are most often relying on multiple serial processes and several software packages. All these reasons make ECG monitoring requiring a robust system for the collection, visualization, and analysis of ECG signals.

In this section we give a novel approach of physiological data visualization and monitoring, especially for the ECG signals. This novelty can be seen in the fact that our visualization tool is a web based application. This allows
Table I

<table>
<thead>
<tr>
<th>System Model</th>
<th>Logic utilization</th>
<th>Total block memory bits</th>
<th>Speed (MHz)</th>
<th>Power (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-lead</td>
<td>9,769 ALUTs</td>
<td>11,669</td>
<td>14%</td>
<td>1,207,312 (21%)</td>
</tr>
<tr>
<td>2-lead</td>
<td>17,169 ALUTs</td>
<td>21,297</td>
<td>26%</td>
<td>1,810,384 (32%)</td>
</tr>
<tr>
<td>3-lead</td>
<td>24,592 ALUTs</td>
<td>30,947</td>
<td>38%</td>
<td>2,413,840 (43%)</td>
</tr>
<tr>
<td>4-lead</td>
<td>32,047 ALUTs</td>
<td>40,566</td>
<td>50%</td>
<td>3,016,976 (54%)</td>
</tr>
</tbody>
</table>

Table II

<table>
<thead>
<tr>
<th>Recode (No.)</th>
<th>Detected RR Interval ( # of interval)</th>
<th>Failed Detection ( # of interval)</th>
<th>Execution Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10283</td>
<td>14</td>
<td>7 (50%)</td>
<td>6.787</td>
</tr>
<tr>
<td>10273</td>
<td>15</td>
<td>3 (23%)</td>
<td>6.309</td>
</tr>
<tr>
<td>16420</td>
<td>14</td>
<td>5 (36%)</td>
<td>6.791</td>
</tr>
<tr>
<td>16773</td>
<td>10</td>
<td>1 (10%)</td>
<td>6.311</td>
</tr>
<tr>
<td>16786</td>
<td>10</td>
<td>3 (30%)</td>
<td>6.324</td>
</tr>
<tr>
<td>17052</td>
<td>9</td>
<td>2 (22%)</td>
<td>6.182</td>
</tr>
<tr>
<td>18177</td>
<td>15</td>
<td>5 (33%)</td>
<td>8.316</td>
</tr>
<tr>
<td>18184</td>
<td>8</td>
<td>3 (38%)</td>
<td>4.860</td>
</tr>
</tbody>
</table>

us to deal with the high requirements of the bio-medical data monitoring, such as the real time constraint in addition to the interaction and the synchronization issues between the medical staff’s side and the patients’ side.

Our visualization tool uses PHP [10] as a server side and Mysql [11] as a data base management system. The ECG waveform will be displayed with a free library based on Javascript [12]. Moreover, the dynamic update is ensured by the Ajax technology to allow the user to interact directly with the data coming from the local storage without needing to refresh the page. The incoming data is stored in a dictated table in Mysql database. Each node has its own table that contains all data classified by capture date. This classification allows the medical staff to visualize any specific data at any desired date. In addition, by building the visualization tool as a web application, we improve the mobility of the monitoring task without distributing it or installing any specific software. The medical staff can consult the ECG graph at real time from anywhere through the Internet browser, and can also easily interact with patients at any time as long as the Internet connection is established. Our monitoring tool includes three main parts: data capture, data display and data analysis.

A. Data capture

The output of our processing part is the coordinates of each peak for each corresponding node. To continue on the same way of mobility, the processing data coming from DSP are transmitted to the database through the Internet and the corresponding table in MySQL will be updated automatically Figure 5 (a) shows the DSP board used for our system prototyping.

B. Data display and analysis

The coordinates of the ECG graph from each node are first transmitted through the Internet to be stored in the database. This process is done as long as there are new incoming processed data. The continuity of the update process gives us a wide range of ECG data classified by capture date. The real time charting library implemented in our web visualization tool will manipulate this huge amount of data. So, the medical staff can consult the latest incoming data at real time (a marker is implemented to show that there are new incoming data) or it can re-consult previous data in case of clinical studies, where the diagnosis is made on a large period of time.

Figure 5(b) shows the implemented features of the interactive real-time interface (IRI). It contains four main blocks. The most important one is the ECG viewer itself with the capability to display the data from three leads at the same time (we can extend the ECG viewer to display other additional leads if needed). The second block displays the users nodes which have stored data in the database. The remaining parts of our visualization tool are dictated to show the information related to the node (patient) being checked. These parts consist of the clinical demographics informations and the medical history. The user of our visualization tools should select one node, after that the corresponding part will be filled from the database. Concerning the wave form, and as mention before the user can consult previous data in this case the ecg peaks will be fetched from the database directly through a php script as shown in Fig.4 (b). In the case of a live incoming data, to get the ECG peaks an Ajax call is executed from the chart and return the last peaks that have been added to the database every second as shown in Fig.4(a). Finally, due to the privacy of data that we are handling, the security issue becomes extremely important for our visualization tool. In fact we restricted the access to our interface only for the concerning medical staff and more precisely only for the patient’s doctor. Also displaying...
private information can be done only with the agreement of the concerned patient. This restriction will be implemented when accessing this interface and also at the database level, and it aims to preserve the privacy between the patient and only his doctor.

VI. CONCLUSION

In this work, we presented an optimized architecture of our earlier proposed elderly monitoring system based on an interactive real-time interface and new ECG parallel processing algorithm, called PPD. The systems was prototyped in hardware using commercial CAD tools. An on-line real-time interface was also developed. The key performance idea of our ECG processing algorithm is based on parallel processing technique. Our solution paves the way for real-time processing diagnosis of heart-related diseases especially for elderly people.

The hardware prototyping result shows that the complexity is about 14% of logic utilization of the target FPGA. The PPD algorithm was tested with only normal ECG signals, then it has a poor support for abnormal or noisy ECG signals. Our future work should be related to improving the accuracy of the PPD algorithm, refining our prototype system and the visualisation tools and performing real test on humans.

REFERENCES


