

Digital ECG and its Analysis

Vidur Arora, Rahul Chugh, Abhishek Gagneja and K. A. Pujari

Abstract--Cardiac problems are considered to be the most fatal in medical world. Conduction defects in the heart leads to different cardiac arrhythmias. In this project work a method is proposed to classify ECG arrhythmias accurately through a combination of Digital Signal Processing (DSP) using DSK6713 and Artificial Neural Networks (ANN). The ability of DSK6713 to decompose signal at various resolutions allows accurate extraction/detection of features from non-stationary signals like ECG. In this approach a set of discrete features from the ECG signal which contains information about the arrhythmia are selected from the time-domain analysis done on DSK6713. These features in addition to the information about RR interval (the difference between the QRS peak the present and previous beat) are fed to the back-propagation neural network which classifies the ECG signal as 'Normal' or 'Abnormal'. Arrhythmia information is represented in terms of amplitude and position of P, Q, R, S, T peaks in this approach. In the present work the ECG data is taken from standard MIT-BIH Arrhythmia database. The proposed method is capable of distinguishing the normal sinus rhythm and 4 different arrhythmias. The overall accuracy of classification proposed approach is 96.7976%. The results of the analysis were found in very less time and were more accurate than those of the existing methods. Three types of noise, i.e., muscle noise, power line interference and base-line wander, were removed with SNR values ranging from 0 dB to 10 dB to the signal and the accuracy was found to be well within the clinical limits. It was observed that the effect of base-line wander on the accuracy of detection was less than the other disturbances. In all conditions DSK 6713 approach is having good accuracy and showing faster results than other existing methods.

I. INTRODUCTION

HEART diseases are the leading cause of death in the world. Therefore, there is a need for accurate diagnosis of various cardiac diseases suffered by a patient for proper medication. Heart diseases are caused due to abnormal propagation of impulses through the specialized cardiac conduction system. These abnormalities are called cardiac arrhythmias and they are caused due to the conduction blocks in the bundle branch and atrial or ventricular irregular de-polarizations. Cardiac arrhythmias are alterations of cardiac rhythm that disturb the normal synchronized contraction sequence of the heart and reduce pumping efficiency. Frequency of occurrence and type of arrhythmias provide an important indication of the electrical stability of the heart. Since arrhythmias can be suppressed by anti-arrhythmic drugs, early recognition is important. Fortunately, majority of cardiac arrhythmias are preventable

and controllable with timely monitoring, diagnosis and treatment.

A number of biomedical signals are recorded and used for patient monitoring or diagnostic purpose. The Electrocardiogram (ECG), which is recorded in a noninvasive manner from the surface of the chest, plays a key role in monitoring and diagnosis of cardiac diseases. Long-term monitoring in the intensive care unit (ICU) or on patients in ambulance, one or two ECG leads are monitored or recorded to investigate life-threatening disturbances in the rhythm of the heartbeat. This approach is called arrhythmia analysis. This analysis is used for detection of infrequent arrhythmias, evaluation of patients at risk for sudden cardiac-death, monitoring patients after myocardial infarction, documenting the effectiveness of anti-arrhythmic drug therapy, and assessing the functioning of implanted cardiac pacemakers. Visual analysis of long-term (24 hours) ECG is tedious and operator dependent. Computer techniques have been developed in order to facilitate the visual analysis, e.g. by condensed print-outs of various signals and trends. With this type of presentation the operator usually can analyze a 24-hours recording in 20-40 minutes provided that no complex arrhythmias exist. It is obvious that automated systems for detection of arrhythmias considerably reduce the amount of time the operator needs to spend. Automated analysis of ECG signal is essential for timely diagnosis and treatment of cardiac diseases, and thus for saving lives.

A. Design Approach

For the portable arrhythmia monitor, the two major software design tasks are

1. QRS detection and
2. Arrhythmia analysis.

The QRS detection must be nearly perfect, otherwise the arrhythmia analysis algorithm will be fooled too often by false reports of beats that are not really there (i.e. false positive) or lack of reporting of beats that are missed (i.e. false negative). Some time an additional software design task important in these devices with limited memory is a data reduction algorithm.

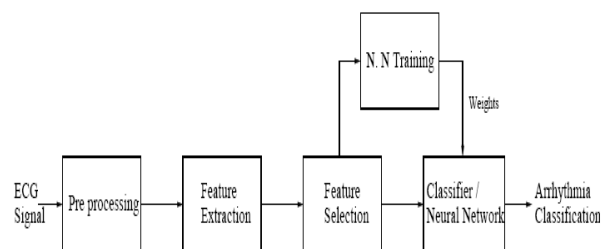


Figure 1. Block Diagram of ECG classifier

To implement the software design for the ECG classifier shown above two types of approaches have been followed:

1) DSP approach:

In this approach the use of DSK 6713 is made. The basic algorithm is the same as shown above with the exception of neural networks. The algorithm begins with the data acquisition from the MIT-BIH database. The signal is then differentiated to find out the highest slope present in it. It is followed by squaring of signal to remove any negative peaks. Threshold application follows in the above mentioned algorithm to find out the R-peaks. Similar process is followed to detect the rest of the peaks. Finally, arrhythmias are detected and heart rate is calculated. For implementation of above procedure we use CCS (Code Composer Studio Version 3.1).

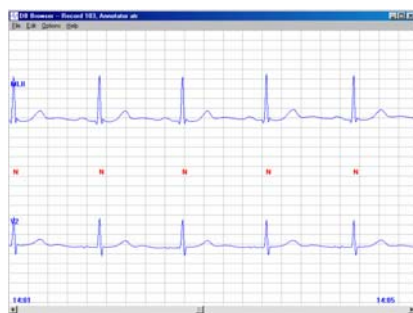


Figure 2. Sample of the ECG Waveforms contained in Record 106

The ECGs are selected using the timing information provided. During this interval five ECGs are displayed. This gives a rate of 1.25 ECGs per second or a heart rate of seventy five beats per minute, which is quite normal. These examples of “good” ECGs show the PQRST complex clearly and that there is very little noise present.

The unhealthy or “bad” ECGs are selected from files of which 102, in the *mitdb* directory, is an example. The ECG waveforms produced at electrode position V5 have correct arrhythmia. Again five beats in four seconds gives a rate of seventy five beats per minute which is within the normal heart rate. However it is clear that the PQRST complex is not that of a healthy person.

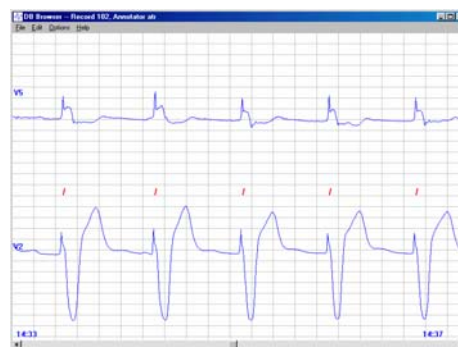


Figure 3. Sample of the ECG Waveforms contained in Record 102

The tool *MATLAB* is used to read the ECG signal files for the required timing period and to decode the signal samples to decimal numbers.

The data is read into Matlab and stored as a data (mat) file. Following is a Matlab figure of a good ECG. The ECG is 450 samples long and displays the T-wave of the previous ECG.

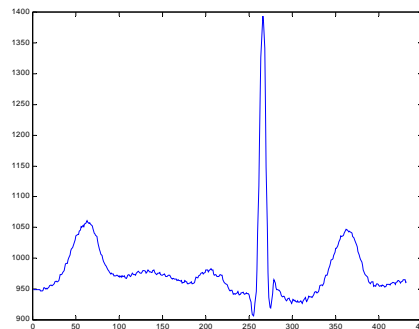


Figure 4. Un-pre-processed ECG of 450 Samples

B. ECG Normalisation and Standardisation

Each ECG sample is pre-processed as follows: The ECGs obtained using the *Matlab* tool consists of files which are in the form of 3 columns of circa 350 samples in each column. The required data is in column two only. The other columns contain timing and annotation data which is of no interest in this project. When the files are read into Matlab and stored, column two only is selected. The location of the maximum value is located. The ECG is then centred on the maximum value and limited to 30000 samples. The resulting normalised ECG is then saved in a Data directory. This process is repeated for each ECG in the training and test database. The data is converted to binary files using the *fileconv* command. This enabled the data to be read using a ‘for’ loop and matrices of training and test data are subsequently generated.

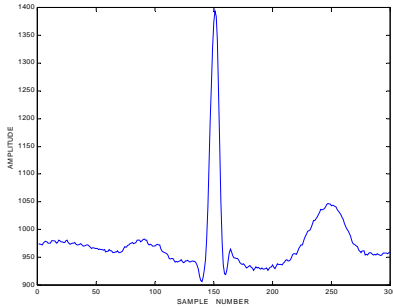


Figure 5. Example of Normalised ECG

C. Pre-Processing

The form of temporal pre-processing employed in this project is a process whereby the energy of the ECG signal which exists in a number of time windows, is calculated. The theoretical temporal windows are set out in the summary of intervals in Fig. as follows:

P-wave duration:	60 ms to 110 ms
P-R interval:	120 ms to 200 ms
QRS duration:	60 ms to 100 ms
Q-T interval:	340 ms to 420 ms
T-wave duration:	120 ms to 180 ms

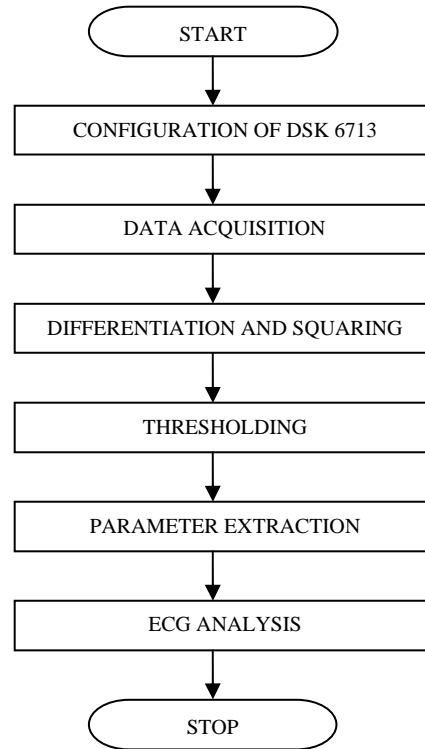
II. SOFTWARE APPROACHES

A. DSP Approach

Initially the DSK kit is configured for our project. The configuration consists of memory allocation.

The compiler produces re-locatable blocks of code and data. These blocks, called sections, are allocated in memory in a variety of ways to conform to a variety of system configurations.

The compiler creates two basic kinds of sections: initialized and uninitialized. This table summarizes the sections.



Configuration of TMS320VC6713

In our program the memory locations are as follows:

```

MEMORY
{
  IVECS:      org = 0h, len = 0x220
  IRAM:       org = 0x00000220, len = 0x0002FDE0 /*
0x0002FDE0*/
  SRAM_EXT:  org = 0x80000000, len = 0x01000000
  FLASH:     org = 0x90000000, len = 0x00020000
}
    
```

Sections Created by the Compiler

(a) Initialized sections

Name	Contents
.cinit	Tables for explicitly initialized global and static variables :> IRAM
.const	Global and static const variables that are explicitly initialized and that are string literals :> IRAM
.pinit	Table of constructors to be called at startup. Not used in our program.
.switch	Tables for implementing switch statements. :> IRAM
.text	Executable code and constants. :> IRAM

(b) Uninitialized sections

Name	Contents
.bss	Global and static variables. :> IRAM
.stack	Stack. :> IRAM
.system	Memory for malloc functions. :> IRAM

When the program is linked, specify where to allocate the sections in memory. In general, initialized sections are linked into ROM or RAM; uninitialized sections are linked

into RAM. With the exception of .text, the initialized and uninitialized sections created by the compiler cannot be allocated into internal memory.

1) Data Acquisition

'data.h' contains the ECG data which is converted from the original format given by MIT database into the format supported by Code Composer Studio Version 3.1. The data is then stored in a buffer for further processing. Total no. of sample taken for processing is 10800.

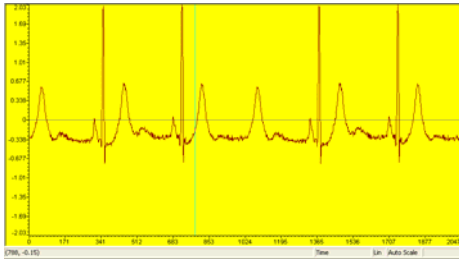


Figure 6. Original data

2) Differentiation

It is a standard technique for finding the high slopes that normally distinguish the QRS complexes from the ECG waves. Since it is basically a high-pass filter, the derivative amplifies the higher frequencies characteristic of the QRS complex while attenuating the lower frequencies of the P and T waves. This algorithm has the advantage that it produces a pulse which is proportional in width to the QRS complex. However, a disadvantage is that it is particularly sensitive to higher- frequencies noise.

After the signal has been filtered, it is then differentiated to provide information about the slope of the QRS complex. A five point derivative has the transfer function

$$H(z) = 0.1(2+z^{-1}-z^{-3}-2z^{-4})$$

This derivative is implemented with the differential equation:

$$Y(nT) = (2x(nT)+x(nT-T)- x(nT-3T)- 2x(nT-4T))*(1/8)$$

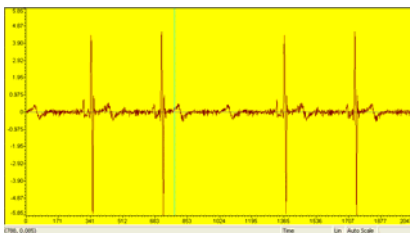


Figure 7. Data after differentiation

Code for differentiation:

```
/*differentiating the signal */
DIFF_data[p] = 2*buffer [p+4] +buffer [p+3] - buffer [p+1]
- 2 * buffer[p];
```

The derivative has a filter delay of 2T (or 10 ms). The amplitude response approximates that of a true derivative

up to 20 Hz. This is the important frequency range since all the higher frequencies are significantly attenuated by the band pass filter.

3) Squaring

The previous process i.e. differentiation is linear processing parts of the QRS detector. The squaring function that the signal now passes through is a non linear operation. This transformation serves to make all the data positive prior to subsequent integration, and also accentuates the higher frequencies in the signal obtained from the differentiation process. The equation that implements this operation is

$$Y(nT) = [x(nT)]^2$$

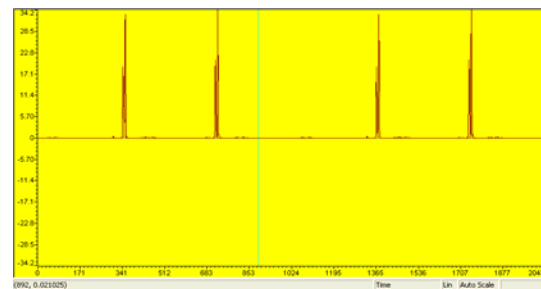


Figure 8. Squaring

This operation makes all data points in the processed signal positive, and it amplifies the output of the derivative process nonlinearly. It emphasizes the higher frequencies in the signal, which are mainly due to the QRS complex. The output of this stage should be hardlimited to a certain maximum level corresponding to the number of bits used to represent the data type of the signal.

Code for squaring:

```
/*squaring */
SQR_data[p]=DIFF_data[p]*DIFF_data[p];
```

4) Thresholding

It is basically a peak detection algorithm which finds QRS peaks. Signal peaks are defines as those of the QRS complex, while noise peaks are those of the T waves, muscle noise, etc. After the ECG signal has passed through the bandpass filter stages, its signal-to-noise ratio increases. This permits the use of thresholds that are just above the noise peak levels. Thus, the overall sensitivity of the detector improves.

Code for thresholding:

```
/*threshold */
if(SQR_data[p]>=3)
{
tim[t]=p;
}
```

a) Data Acquisition

The first 2 bytes of the header file gives the number of signals and sample rate of data. The next 5 bytes give us other information which helps us decode the dat file. In all

30,000 samples are read and stored in a binary format. The binary values are checked for their sign and accordingly the samples are normalized and stored in the workspace. Then the signals can be saved in a mat file.

TABLE I
FORMAT OF HEADER OF DAT FILE

Byte Number	Significance
1	Number of signals in dat file
2	Sample rate of data
3	Format of data (here only 212 is allowed)
4	Number of integers per mV
5	Bit resolution
6	Integer value of ECG zero point
7	First integer value of signal

b) Data Filtering

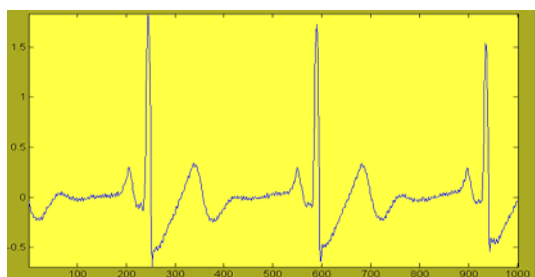


Figure 9. Pre-processed data

This removes the baseline wander by passing it through a Butterworth high-pass filter. The power line interference is also removed by passing it through an 11 Hz band pass filter.

The data is then passed through a low-pass filter to remove any high frequency noise caused by muscle jitters. The coefficients are calculated using Matlab and then used in the following formula:

$y(n) = b(1)*x(n) + b(2)*x(n-1) + \dots + b(nb+1)*x(n- nb) - a(2)*y(n-1) - \dots - a(na+1)*y(n-na)$ where $n-1$ is the filter order, and which handles both FIR and IIR filters. It is then passed through a high-pass filter to remove any low frequency noise. The results are as shown below:

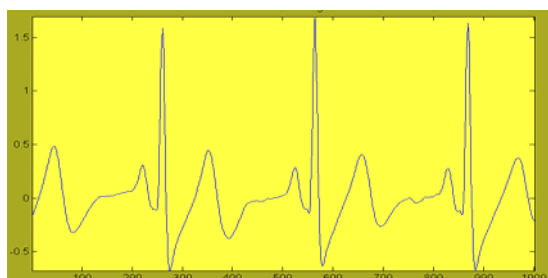


Figure 10. Data after low-pass filtering

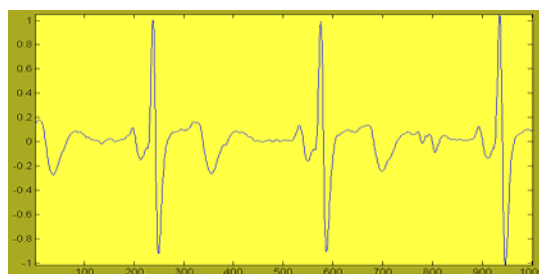


Figure 11. Data after high-pass filtering

c) Detection of R-peaks through thresholding

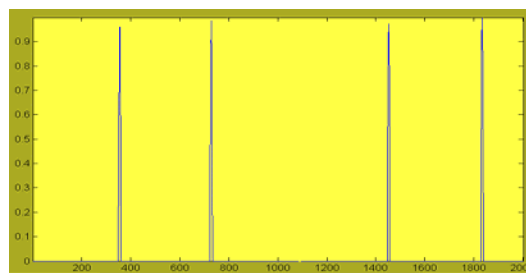


Figure 12. R peaks

As seen in the above figure a threshold is applied to detect the R-peaks. The threshold is set at 0.3V. The points at which R-peaks are detected are made zero. The points to the immediate left and right are made +1 and -1 according to their sign. Thus the zero-crossing gives the R-peaks.

Detection of other points:

The other peaks are detected in similar fashion. For Q point:

The signal is checked 25 locations to the left and right of itself. If zero-crossing is detected, it is classified as Q-point. Similar approach is followed in other points. Finally heart rate is calculated by using the time difference between two R-peaks (R-R interval).

B. ECG Analysis and Classification

The parameters obtained from the previous analysis are fed as vectors to the "Black-Box" of Artificial Neural Networks (ANN). The ANN is first given training to classify the signal as Normal or Abnormal. It does so by a variety of tools available in Matlab tool-box.

1) Graphic User Interface

The GUI is made to facilitate user for quick and easy analysis of the selected database. The GUI provides the user with the following control options:

- i. **Load Database:** This button opens a dialog box which prompts the user to select a database out of the number of databases available for analysis. Once the user selects a particular database it starts to get loaded in the memory.
- ii. **Select number of samples:** This pop-up menu prompts the user to select the number of samples to be displayed at a particular instance of time. The

- options available are 1000, 2000, 3000, 4000 and 5000 samples.
- iii. **Select graph:** This pop-up menu prompts the user to select which graph is to be displayed from the options- Pre-processed, LPF Pass, HPF Pass, Thresholds and the unmodified database.
- iv. **Run:** This check-box when checked starts to display the graph. To stop the running graph at any instance of time the user has to just uncheck this box.
- v. **Generate Report:** This button when clicked generates a report with all the various time-intervals calculated along with the heart rate of the patient. The report also suggests the disease which the patient might be suffering from.
- vi. **Exit:** This button asks the user to confirm her decision of closing the GUI and closes the GUI when confirmed.

Apart from these control buttons there are two status bars which continuously display the status of analysis.

- i. **Status:** This indicates when the database is being loaded and how many numbers of samples are selected. It basically tells the present status of analysis.
- ii. **Heart Rate:** As the name suggests it displays the heart rate of the patient continuously.

2) Testing

The following steps are performed for analysis and arrhythmia recognition of ECG:

- i. The dat file from MIT-BIH database is processed in Matlab to convert it into mat file and also in the format which Code Composer Studio understands.
- ii. The data from mat file is filtered in the Matlab approach and RR interval is calculated. Similarly other features are extracted.
- iii. Selected feature vectors are given as input to the neural network which has an optimal set of weights.
- iv. Classification of the input data set is carried out by the back propagation neural network.
- v. In DSP approach the data is used for calculation of various time intervals and features of ECG signal.

III. RESULT AND CONCLUSIONS

The following table shows the result of our analysis of one minute signal from various MIT-BIH database records:

Name of record	Heart beat span	Average heart rate	Predicted disease
100	55-84	68	Normal
101	58-65	62	Normal
102	62-70	66	Normal
103	59-66	63	Normal
105	37-129	72	Normal
106	52-80	63	Normal
209	81-92	87	Sinus Tachycardia
215	96-111	104	Sinus Tachycardia
217	61-70	66	Normal
221	35-109	59	Sinus Bradycardia
223	71-77	74	Normal
231	31-66	57	Sinus Bradycardia
233	68-158	94	Sinus Tachycardia
234	80-87	84	Sinus Tachycardia

The above table also shows the predicted disease on the basis of heart rate classification.

Shown below are the ECG signals of the database which were found to contain arrhythmias during our testing:

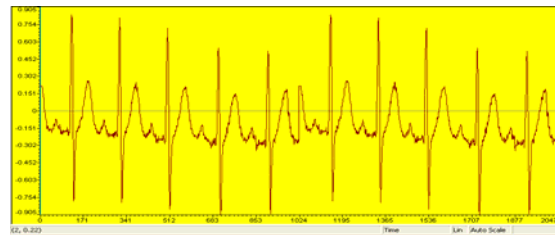


Figure 13. Graph of 100.dat

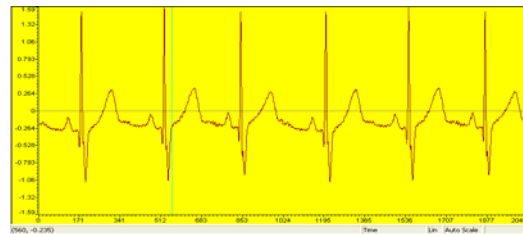


Figure 14. Graph of 215.dat

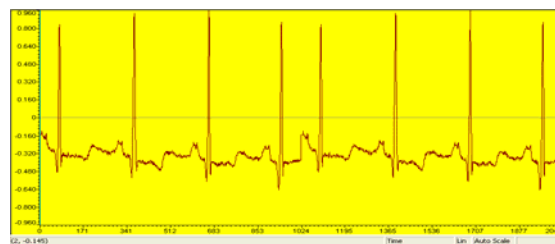


Figure 15. Graph of 231.dat

A high quality of feature set is undoubtedly the first important factor for good performance of ECG analysis algorithms. Here two approaches are proposed to extract features from the ECG signal- DSP approach and Matlab approach.

The robustness of the proposed methods against three types of noises, muscle noise, power-line interference and base-line wander, are investigated. The muscle noise and power-line interference are affecting both the classifiers performance, and in both approaches the accuracy increases with the increasing value of SNR value. The base-line wander is affected less as compared with other noises.

One should mention that the proposed DSP approach does not require filtering of the power line frequency in the preprocessing stage.

IV. REFERENCES

Journals:

- [1] "Key Technical Challenges and Current Implementations of Body Sensor Networks"- Benny P L Lo and Guang-Zhong Yang, Department of Computing, Imperial College London, UK, September, 2006
- [2] "Energy Profiling of DSP Applications, A Case Study of an Intelligent ECG Monitor"-Dejan Raskovic, Emil Jovanov, Thomas Martin, Shuaib Hanief, Pedro Gelabert , October, 2002
- [3] "Patient Monitoring Using Personal Area Networks of Wireless Intelligent Sensors"-Emil Jovanov, Dejan Raskovic, John Price, John Chapman, Anthony Moore, Abhishek Krishnamurthy, University of Alabama, January, 1997
- [4] "Using Technology to Identify Triggers for Chronic Conditions"- Anne M. Faber, Indiana University, March, 2004
- [5] "Wearable Wellness Monitoring Using ECG and Accelerometer Data"- Jennifer Healey, Beth Logan, Cambridge Research Laboratory, HP Laboratories, July 2005

Websites:

- [6] <http://www.medgadget.com>
- [7] <http://www.ti.com/>
- [8] <http://www.ieee.org/>