ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/323195733

Detect QRS complex in ECG

Conference Paper · June 2017

DOI: 10.1109/ICIEA.2017.8283170

CITATIONS
0

READS
169

4 authors, including:

Ngan V. T. Nguyen
Texas Tech University

10 PUBLICATIONS 0 CITATIONS

SEE PROFILE

Tuan Van Huynh
Ho Chi Minh City University of Science

8 PUBLICATIONS 22 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project EEG analysis and detection View project

All content following this page was uploaded by Ngan V. T. Nguyen on 21 November 2018.

The user has requested enhancement of the downloaded file.

Detect QRS complex in ECG

Ngan Vuong Thuy Nguyen
Physics and Computer Science dept.
Physcis and Engineering Physics faculty
University of Science
Ho Chi Minh city, Vietnam
Email: nvtngan@hcmus.edu.vn

Long Duc Tran
Physics and Computer Science dept.
Physcis and Engineering Physics faculty
University of Science
Ho Chi Minh city, Vietnam
Email: trduclong@gmail.com

Tuan Van Huynh
Physics and Computer Science dept.
Physcis and Engineering Physics faculty
University of Science
Ho Chi Minh city, Vietnam
Email: hvtuan@hcmus.edu.vn

***Abstract*—Thanks to the Electrocardiography (ECG) signal and its features, some serious heart diseases can be diagnosed and treated at the early stage. In this paper, we focus on the detecting QRS complex of ECG, more specifically, the R wave. While Pan - Tompkins's algorithm effectively extracts R peak, our new algorithm bases on Pan – Tompkins's model giving the same results but providing possibility of detecting separately Q, R and S features. In order to verify the performance of our method, we compared extracted features (R feature) of the new algorithm built on Simulink with the results from the original Pan – Tompkins's algorithm reassembled in Matlab. Besides, the delay and error analysis of the new algorithm on real-time system (TMS320C6713 DSK) had been done and gain some preliminary results.**

***Index Terms*—Detection algorithms, Electrocardiography, Filtering algorithms.**

I. Introduction

Electrocardiography (ECG) is the recording of the electrical activity of the heart. Traditionally, this is in the form of a transthoracic (across the chest) interpretation of the electrical activity of the heart over a period of time [2].

Abnormal ECG results may be a sign of: Damages or changes to the heart muscle, changes in the amount of sodium or potassium in the blood, congenital heart defect, enlargement of the heart, fluid or swelling in the sac around the heart, inflammation of the heart, past or current heart attack, poor blood supply to the heart arteries, abnormal heart rhythms (arrhythmias) [4]. Some heart problems that can lead to changes on an ECG test include: Atrial fibrillation/flutter, heart failure, multifocal atrial tachycardia, paroxysmal supraventricular tachycardia, sick sinus syndrome, Wolff–Parkinson–White syndrome.

Many analysing ECG solutions had been provided such as Pan-Tompkins algorithm, Low Pass Differentiation (LPD), Multichannel ART - Based Neural Network (MART), Wavelet Transform (WT), etc [5]. Those algorithms aimed at determining R-R interval (heart-rate), QRS duration, PR interval, etc which play special role in diagnostic heart diseases.

In this paper, we discuss about detecting QRS complexes in ECG basing on Pan – Tompkins algorithm [1] with a changing in Moving-window Integration stage. By replace Moving-window Integration, our effort is finding a fast and stable solution for pointing the position of R peak in QRS complex. Firstly, we firstly apply the original Pan-Tompkins algorithm to the offline (static) version of ECG signal in Matlab. And then the new algorithm is run on Simulink with online (dynamic) ECG signal. By comparing results of two algorithm versions, to verify the algorithm's reliability. Besides, we have experiment in the real-time system using Texax Instrument DSP starter kit (TMS320C6713 DSK) with new version of Pan – Tompkins algorithm.

II. Model of Pan – Tompkins for QRS detection

A. Pan – Tompkins algorithm

Pan – Tompkins is one of common algorithms in biomedical signal processing to detect QRS complexes. Figure 1 illustrates the basic model of this algorithm.

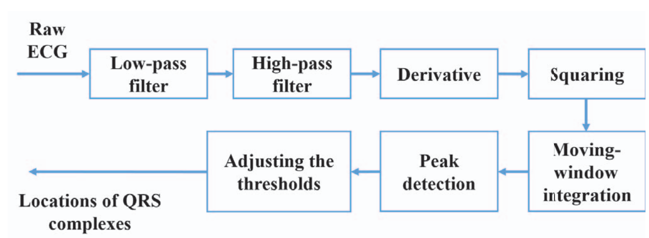

Figure 1: Seven basic stages of QRS Detection Algorithm (Pan–Tompkins 1985)

The purpose of the first 4 stages in this method is removing noise and highlighting features in ECG signal.

1) Low-pass filter: The transfer function of the second-order low-pass filter (where the high cutoff frequency is about 11Hz) is

$$H(z) = \left(\frac{1 - z^{-6}}{1 - z^{-1}}\right)^2 \tag{1}$$

The difference version of this filter is

$$y(n) = 2y(n-1) - y(n-2) + x(n) - 2x(n-6) + x(n-12) \tag{2}$$

The delays of this stage is 6 samples.

2) High-pass filter: The transfer function of the high-pass filter (where the low cutoff frequency is about 5Hz) is

$$H(z) = \frac{-\frac{1}{32} + z^{-16} + z^{-17} + \frac{1}{32}z^{-32}}{1 - z^{-1}} \tag{3}$$

978-1-5090-6161-7/17/$31.00 ©2017 IEEE

The difference version of this filter is

$$y(n) = y(n-1) - \frac{1}{32}x(n) + x(n-16) - x(n-17) + \frac{1}{32}x(n-32) \quad (4)$$

This processing causes delaying 16 samples.

3) Derivative: After filtering, the signal is differentiated to provide the QRS-complex slope information. The transfer function is

$$H(z) = \frac{1}{10}(-2z^{-2} - z^{-1} + z^{1} + 2z^{2}) \quad (5)$$

The difference version of this filter is

$$y(n) = \frac{1}{8}(-2z^{-2} - z^{-1} + z^{1} + 2z^{2}) \quad (6)$$

At this stage, the output of signal is delayed 2 samples

4) Square function: After differentiation, the signal is squared point by point. The equation of this operation is

$$y(n) = [x(n)]^2 \quad (7)$$

This makes all data points positive and does nonlinear amplification of the output of the derivative emphasizing the higher frequencies.

5) Moving-Window Integration: Moving-window integration obtains wave-form feature information and the slope of the R wave. The equation of this operation is

$$y(n) = \frac{1}{N}[x(n-(N-1)) + x(n-(N-2)) + \ldots + x(n)] \quad (8)$$

Where N is the width of the integration window and depend on the number of samples. For the samples rate 200 samples/second, the window is 30 samples wide (150 ms).

6) Adjusting the Thresholds: The algorithm separates the R wave of QRS complexes in real time, so that the value of thresholds among R waves is different. Therefore, it need to be adjusted.

$$\begin{aligned}
&Peak > ThrSig \Rightarrow SigLev = \frac{1}{8}Peak + \frac{7}{8}SigLev \\
&ThrNoise < Peak < ThrSig \\
&\qquad\qquad \Rightarrow NoiseLev = \frac{1}{8}Peak + \frac{7}{8}NoiseLev \\
&ThrSig = 1/4SigLev + 3/4NoiseLev \\
&ThrNoise = 1/2ThrSig
\end{aligned}$$

Where all the variables refer to the integration waveform:

$Peak$	*is the overall peak.*
$ThrSig$	*is the threshold applied to the signal peak.*
$SigLev$	*is the running estimate of the signal peak.*
$ThrNoise$	*is the threshold applied to the noise peak.*
$NoiseLev$	*is the running estimate of the noise peak.*

B. New algorithm based on Pan – Tompkins algorithm

We built up a new algorithm which has ability to adapt dynamic signal in real-time using DSP starter kit (TMS320C6713 DSK). We kept the most stages of Pan - Tompkins algorithm, except the Moving-window Integration. It is replaced with a new stage that we call it Timing-filter.

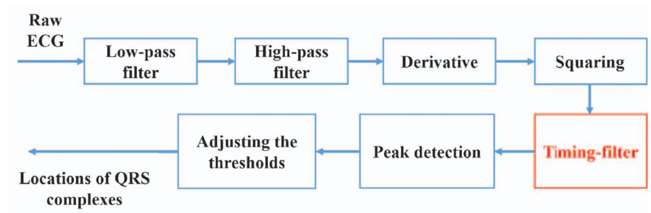

Figure 2: Stages of new algorithm based on Pan-Tomkins algorithm

Moving-window integration can be replaced by this filter to split the signal into separating QRS complexes. The idea came from the minimum distance between two R peaks (200 ms). In other words, if One R peak is detected, no valid R peak will appears during next 200ms. In detail, by comparing the value of two nearest peaks and the time during two peaks, timing-filter prevents the time of two nearest peaks lower than 200 ms. It is calculated from:

$$y(n) = \begin{cases} x(n-0.1/Ts) & ,w(n-1) \le x(n-0.1/Ts) \\ 0 & ,w(n-1) > x(n-0.1/Ts) \end{cases} \quad (9)$$

$$w(n) = \begin{cases} w(n-1)a(n) & ,x(n-1) \le w(n-1) \\ x(n-1)a(n) & ,x(n-1) > w(n-1) \end{cases} \quad (10)$$

$$a(n) = \begin{cases} 1 & ,b(n) \le 0.2 \\ 0 & ,b(n) > 0.2 \end{cases} \quad (11)$$

$$b(n) = \begin{cases} Ts + b(n-1) & ,w(n-1) \le x(n-0.1/Ts) \\ 0 & ,w(n-1) > x(n-0.1/Ts) \end{cases} \quad (12)$$

Where the Ts is sampling period. Obviously, this segment lead to 0.1 sec delay with any sample time (Ts). Note that a peak detection stage need to be done before this filter.

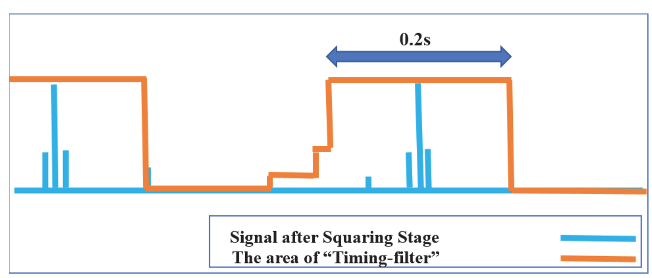

Figure 3: Illustration of Signal in Timing-filter. The blue line demonstrate for the QRS trap zone to detect the highest value in 0.2s.

The total sample delay of this method can calculate through equation:

$$delayall = 6 + 16 + 2 + floor(0.1/Ts) \tag{13}$$

By applying this stage 2 more times with some modifies, we can also point out the Q and S positions approximately. It will cost 0.1/Ts samples delay more.

III. RESULT

A. Simulation result in Matlab using new algorithm

We using the dataset from PhysioBank Databases [6]. Total 12 difference types of ECG signal have been concern in this project: Normal sinus, Congestive Heart Failure, Arrhythmia, Ventricular Tachyarrhthmia, etc. We just cover some examples in this paper.

1) Normal sinus rhythm (offline ECG1): Simulation parameter: Sample rate 128Hz, Offline ECG1 of normal sinus rhythm.

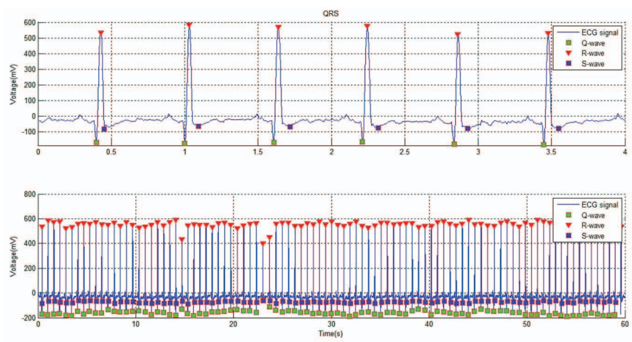

Figure 4: Simulation result of offline ECG1 of normal sinus rhythm

Table I: The amplitudes and positions of Q R S in ECG1

Q_a	-171	-175	-169	-167	-181	-183	-149	-171
Q_p	51	128	206	283	363	441	520	597
R_a	535	585	571	577	523	531	559	563
R_p	55	132	210	287	366	445	524	601
S_a	-83	-67	-69	-75	-79	-79	-61	-67
S_p	58	140	220	297	375	454	533	609

RR interval (s)	0.6296	**Heart rate (beat/m)**	95
R amplitude (mV)	554.41	**QRS interval (s)**	0.085

Where all the variables:
- Q_a is the amplitude of Q peak
- Q_p is the position of Q peak
- R_a is the amplitude of R peak
- R_p is the position of R peak
- S_a is the amplitude of S peak
- S_p is the position of S peak

2) Arrhythmia (offline ECG2): Simulation parameter: Sample rate 360Hz, Offline ECG2 of normal sinus rhythm.

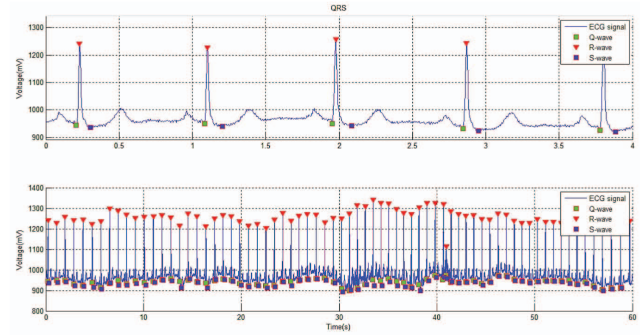

Figure 5: Simulation result of offline ECG2 of Arrhythmia

Table II: The amplitudes and positions of Q R S in ECG2

Q_a	944	950	950	931	926	938	914	934
Q_p	74	390	703	1024	1360	1703	2015	2342
R_a	1241	1228	1257	1243	1244	1224	1237	1299
R_p	83	397	712	1033	1369	1713	2037	2350
S_a	937	943	946	933	923	928	906	939
S_p	104	414	733	1052	1390	1729	2053	2371

RR interval (s)	0.8498	**Heart rate (beat/m)**	71
R amplitude (mV)	1258.2	**QRS interval (s)**	0.0782

3) Congestive heart Failure (offline ECG3): Simulation parameter: Sample rate 250Hz, Offline ECG1 of normal sinus rhythm.

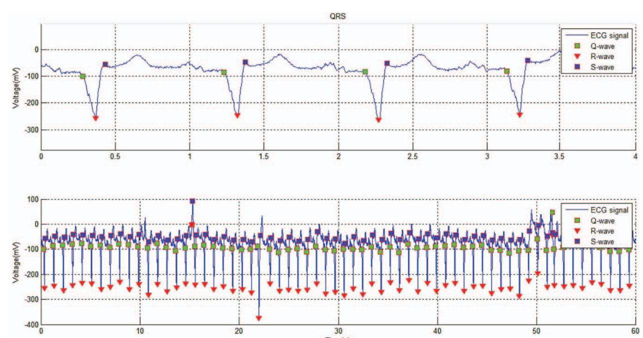

Figure 6: Simulation result of offline ECG3 of congestive heart failure

Table III: The amplitudes and positions of Q R S in ECG3

Q_a	-100	-85	-84	-80	-76	-88	-84	-92
Q_p	71	309	546	784	1023	1260	1496	1733
R_a	-255	-246	-263	-244	-235	-237	-259	-251
R_p	93	311	569	806	1045	1281	1519	1755
S_a	-56	-48	-52	-41	-44	-46	-49	-45
S_p	109	345	583	819	1060	1295	1533	1768

RR interval (s)	0.9082	**Heart rate (beat/m)**	66
R amplitude (mV)	-247.2	**QRS interval (s)**	0.1423

According to the results above, the algorithm of Pan-Tompkin works fine with the static signal and produces precisely value. Therefore, we choose those detected R peak values being out standard.

B. Simulation result in Simulink using new algorithm

The following results acquired from our model Fig.7 in Simulink. All results are recalculated before showing in summary table.

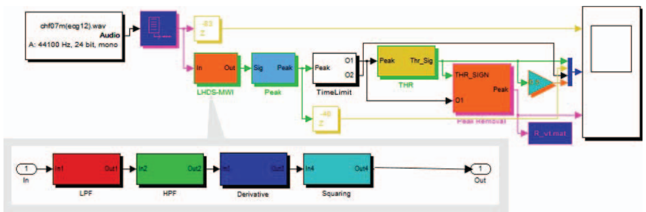

Figure 7: Our model in Simulink.

1) Normal sinus rhythm (online ECG1): Simulation parameter: Sample rate 128Hz, Online ECG1 of normal sinus rhythm

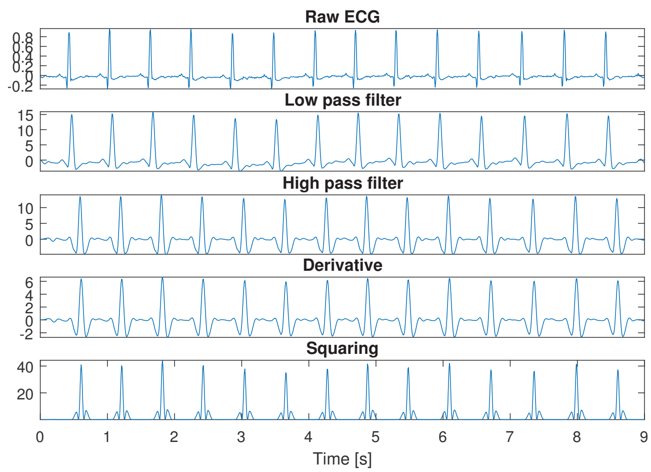

Figure 8: Simulation result of online ECG1 of normal sinus rhythm according to 4 first-stages

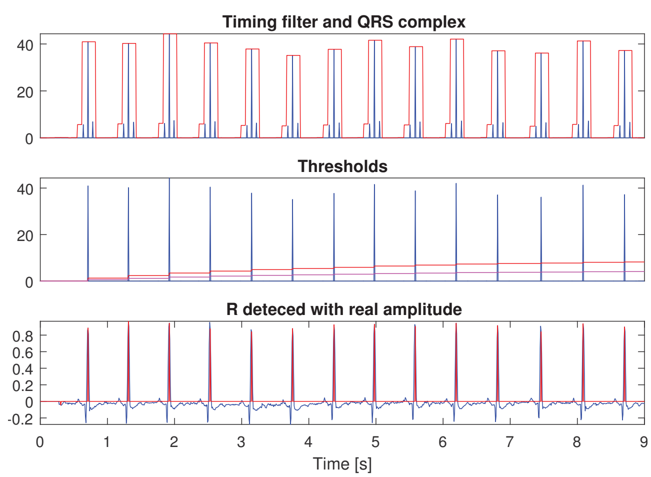

Figure 9: Simulation result of online ECG1 of normal sinus rhythm

The Fig.9 illustrates the signal and the detected peak result with our model. The comparison is also showed at Table IV. Our model detected the R positions, in this case, have 0.2813s delay and no R peak is missed.

2) Arrhythmia rhythm (online ECG2): Simulation parameter: Sample rate 360Hz, Online ECG2 of Arrhythmia rhythm.

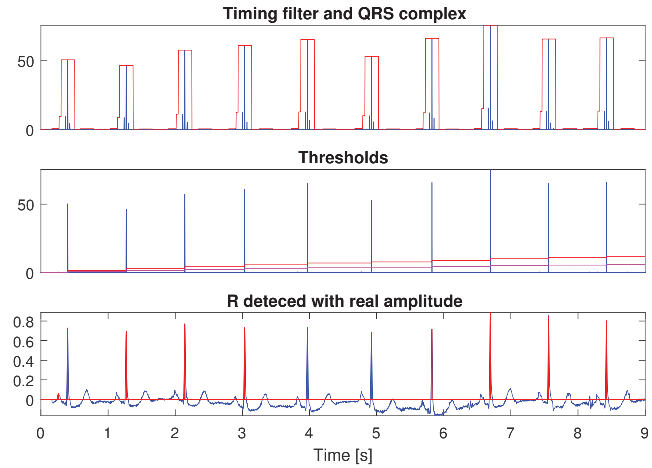

Figure 10: Simulation result of online ECG3 of Arrhythmia Rhythm

The Fig.10 shows the signal and the detected peak result with our model. The comparison is also showed at Table V. Our model detected the R positions, in this case, have 0.1667s delay and no R peak is missed.

3) Congestive Heart Failure Rhythm (online ECG3): Simulation parameter: Samples rate 250Hz, Online ECG3 of normal sinus rhythm

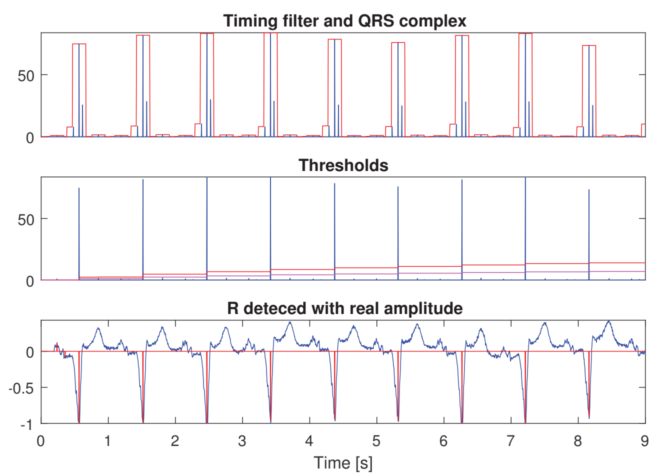

Figure 11: Simulation result of online ECG3 of Congestive Heart Failure Rhythm

The Fig.11 shows the signal and the detected peak result with our model. The comparison is also showed at Table VI. Our model detected the R positions, in this case, have 0.1960s delay and no R peak is missed. In this example, R value have

Table IV: Summary of simulation results of Normal Sinus Rhythm ECG1

Position of R wave of offline ECG1, samples rate 128Hz								
R_{pos}	55	132	210	287	366	445	524	601
(second)	0.4297	1.0313	1.6406	2.2422	2.8594	3.4766	4.0938	4.6953
RR interval (s)		0.6016	0.6094	0.6016	0.6172	0.6172	0.6172	0.6016
Position of R wave of online ECG1, samples rate 128Hz								
R_{pos}	92	169	247	325	404	482	561	638
(second)	0.7188	1.3203	1.9297	2.5391	3.1563	3.7656	4.3828	4.9844
R_{fix}(s)	0.4297	1.0313	1.6406	2.2500	2.8672	3.4766	4.0938	4.6953
Error (s)	0.0000	0.0000	0.0000	0.0078	0.0078	0.0000	0.0000	0.0000
RR interval (s)		0.6016	0.6094	0.6094	0.6172	0.6094	0.6172	0.6016
Error RR (s)		0.0000	0.0000	0.0078	0.0000	-0.0078	0.0000	0.0000

Table V: Summary of simulation results of Arrhythmia Rhythm ECG2

Position of R wave of offline ECG2, samples rate 360Hz								
R_{pos}	83	397	712	1033	1369	1713	2037	2350
(second)	0.2306	1.1028	1.9778	2.8694	3.8028	4.7583	5.6583	6.5278
RR interval (s)		0.8722	0.8750	0.8917	0.9333	0.9556	0.9000	0.8694
Position of R wave of online ECG2, samples rate 360Hz								
R_{pos}	145	458	773	1094	1430	1774	2098	2411
(second)	0.4028	1.2722	2.1472	3.0389	3.9722	4.9278	5.8278	6.6972
R_{fix}(s)	0.2333	1.1028	1.9778	2.8694	3.8028	4.7583	5.6583	6.5278
Error (s)	0.0028	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
RR interval (s)		0.8694	0.8750	0.8917	0.9333	0.9556	0.9000	0.8694
Error RR (s)		-0.0028	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

a negative value and it is an important evidence in diagnosis. Hence, beside R-R interval, delay time value $delayall$ need to be calculate correctly in order to get the real value of R peak. According to Table VI, our equation eq.13 for delay time gives a good delay estimation so far.

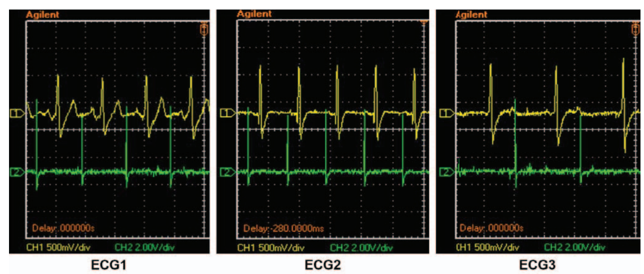

Figure 13: Experiment results observed on oscilloscope. Where the signal (1)-yellow line is raw ECG and signal (2)-green line is location of R peaks. The noise and the shape of the green line is causing by the probe and the DSP Kit and not relevant to the algorithm.

C. Experiment results using the new algorithms on DSP starter kit (TMS320C6713)

We embedded the new algorithm in DSP starter kit (TMS320C6713) with input is online ECG (.wav file). The experiment results show below is conclusion of Normal Sinus Rhythm (ECG1), Sudden Cadia Death Holter (ECG4) and Arrhythmia (ECG2).

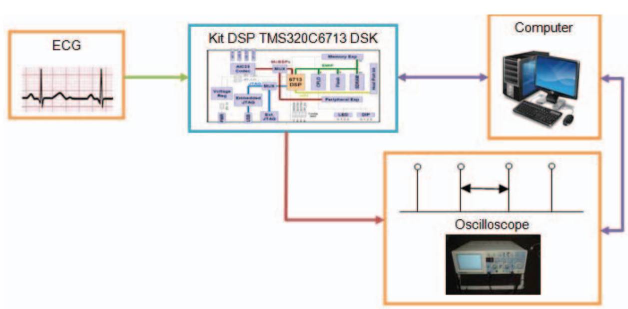

Figure 12: The real-time system of QRS detection. In this system, the computer has three function: transfers ECG signal to the kit through audio line, embeds our algorithm to the kit and acquires signal by connecting it to oscilloscope (Tektronix_MSO4032 (350 MHz, 5GSa/s))

Table VII: Summary of R detection

ECG type	RR interval of raw ECG	RR interval detected	Delay in experiment	Delay in simulation
ECG1 (ms)	622.4	625.85	174.8	281.3
ECG2 (ms)	836.73	836.73	170.07	166.7
ECG3 (ms)	928.57	931.97	173.47	196.0

Our real-time system has been set for running at Sample rate $360Hz$ ($Ts = 2.78ms$). Therefore if we use the eq.13, total delay equals 60 samples or $166.7ms$. Besides, some more delay (1-2 samples) we must add to the real-time version in order to over come the loop of algorithm and the different rate between the original ECG signal and the sample time. Consequently, the delay in experiment is slightly larger than the simulation one which show on the VII. According to table VII, R-R interval in real-time system and simulation system just has a slightly gap which prove our experiment

Table VI: Summary of simulation results of Congestive Heart Failure Rhythm ECG3

Position of R wave of offline ECG3, samples rate 250Hz								
R_{pos}	93	331	569	806	1045	1281	1519	1755
(second)	0.3720	1.3240	2.2760	3.2240	4.1800	5.1240	6.0760	7.0200
RR interval (s)		0.9520	0.9520	0.9480	0.9560	0.9440	0.9520	0.9440
Position of R wave of online ECG3, samples rate 250Hz								
R_{pos}	142	380	618	855	1094	1330	1568	1804
(second)	0.5680	1.5200	2.4720	3.4200	4.3760	5.3200	6.2720	7.2160
R_{fix}(s)	0.3680	1.3200	2.2720	3.2200	4.1760	5.1200	6.0720	7.0160
Error (s)	-0.0040	-0.0040	-0.0040	-0.0040	-0.0040	-0.0040	-0.0040	-0.0040
RR interval (s)		0.9520	0.9520	0.9480	0.9560	0.9440	0.9520	0.9440
Error RR (s)		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

working correctly. Hence, we developed another system with our algorithm to detect not only R peaks but also Q and S.

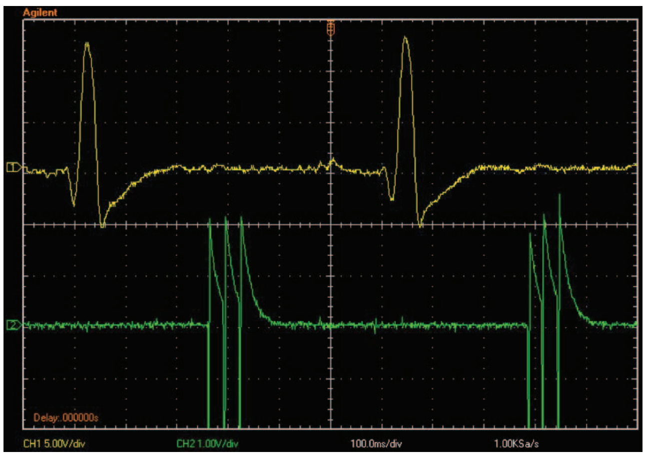

Figure 14: Experiment result of QRS detection. Where signal (1)-yellow line is raw ECG and signal (2)-green line is location of Q, R, S peaks. The noise and the shape of the green line is causing by the probe and the DSP Kit and not relevant to the algorithm.

Table VIII: Experiment results of Q, R, S detection

RR interval of raw ECG	RR interval detected	Delay in experiment	Delay in simulation
622.35 ms	622.45 ms	270.41 ms	268 ms

Fig.14 and Table VIII present the preliminary results of Q, R, S detection of Normal Sinus Rhythm observed on oscilloscope.

IV. Conclusion

In brief, there are some drawbacks of this system such as the old kit DSP, limit of ECG source, etc . Moreover, our additional filter isn't improve speed for the system: 0.1s delay with the timing-filter compares to 0.08s delay with the Moving-Window Intergration. However, by comparing the preliminary results from Matlab, Simulink and real-time system, there is possibility that our algorithm can detect QRS complex with high reliable result. With the new algorithm, we will develop a completed system which can point out other components like Q and S correctly such as our preliminary results.

References

[1] Jiapu Pan and Willis J. Tompkins, senior member, IEEE, *A Real-Time QRS Detection Algorithm*, IEEE Transactions on Biomedical Engineering, Vol. BME-32, No.3, March 1985.

[2] Bert-Uwe Kohler, Carsten Hennig, Reinhold Orglmeister, *The Principals of Software QRS Detection*, Department of Electrical Engineering, Biomedical Electronics Group, Berlin University of Technology, 2002.

[3] Texas Instruments, *TMS320C6713 Floating-Point Digital Signal Processor*, Texas Instruments Incorporated, 2005.

[4] Ganz L. Electrocardiography. In: Goldman L, Schafer AI, eds. *Goldman's Cecil Medicine* . 24th ed. Philadelphia, PA: Saunders Elsevier; 2011:chap 54.

[5] Bert-Uwe Köhler and Carsten Hennig and Reinhold Orglmeister, IEE, *The Principles of Software QRS Detection*, 2002.

[6] PhysioBank Databases, http:/www.physionet.org/physiobank/database/#ecg .

View publication stats