Methods for Detecting Pacemaker Pulses in ECG Signal: A Review

Valentin Viktorovich Tsibulko, Ivo Tsvetanov Iliev and Irena Ilieva Jekova

Abstract - **This paper reviews some different methods for detecting the pulses created by an implanted cardiac pacemaker in the ECG signal. It includes a hardware method implemented in a Texas Instruments circuit, software method proposed by Herleikson and patented by Hewlett-Packard, an upgrade of Herleikson's algorithm reported by Polpetta and Banelli and an software pace pulse detection embedded in the analog front end module of Analog Devices – ADAS1000.**

Keywords – **pacemaker, pace pulse detection, ECG**

I. Introduction

Correct detection of pacemaker pulses in the electrocardiogram (ECG) is crucial for proper evaluation of the effect of a pacemaker on the cardiac rhythm.

The ECG signal recorded from a patient with implanted cardiac pacemaker consists of three parts: the natural ECG signal, pacemaker pulses and noise. The informative content of the ECG signal lies in the frequency band (0 – 150) Hz and its dynamic range is usually up to 2mV. On the other hand, the pacing pulses have a typical duration from 0.1 ms to 2 ms [1-3], and amplitude higher than 0.5 mV. They have very fast rising and falling edges - the rising edge duration could be 100ns measured at the pacemaker leads, appearing widened to 10μs on the surface of the human body [1].

The detection of pacing artifacts is important, since they indicate the presence of a pacemaker and help to evaluate the reaction of the heart. There are different medical standards with variable requirements regarding the height and width of the pace pulse that has to be captured and indicated on the screen of the device. According to ANSI/AAMI EC11 [3] the features of the pacemaker pulses that should be obligatory detected are as follows:

- duration - 0,1ms to 2ms
- amplitude - 2mV to 250mV
- frequency - up to 100 impulses per minute
- rising edge duration - less than 100ms

The IEC60601-2-27 standard [4] states different requirements towards the duration (0.5 ms to 2.0 ms) and the amplitude (2mV to 700 mV) of the pulses.

Modern pacemakers could generate smaller pacing pulse amplitudes that could fall below the requirements set in the standards and lead to complications in the algorithms for pacing pulses detection [5].

V. Tsibulko and I. Iliev are with the Department of Electronics and Electronics Technologies, Faculty of Electronic Engineering and Technologies, Technical University - Sofia, 8 Kliment Ohridski blvd., 1000 Sofia, Bulgaria,
e-mail: valentin.tsibulko@gmail.com, izi@tu-sofia.bg
I. Jekova, is with the Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences, Acad. G. Bonchev str., bl. 105, 1113 Sofia, Bulgaria,
e-mail: irena@biomed.bas.bg

Another challenge faced by the methods for pacing pulses recognition is the noise part of the signal that consists of:

- baseline wander due to the patient respiration and movement - 0.05 Hz to 1 Hz;
- power-line interference - 50/60 Hz, and the respective harmonics;
- electromyographic (EMG) noise, caused by the electrical activity of the muscles.

The baseline wander and the power line interference are with relatively low frequency and do not disturb the detection of the pacing pulses in the ECG. However, the frequency band of the EMG is reported to be up to 5000 Hz [6] and depending on the sampling rate it can overlap with the pace pulses frequencies, thus causing some serious difficulties for the correct pace pulses recognition.

A pacing pulse that has passed thought a low-pass filter, can be significantly widened, and when it has passed to a high-pass filter, a tail at the end of the pulse can be created. Considering the above listed problems, the algorithm for pacing pulse detection should be applied on high-resolution ECG [7, 8] that preserves the frequency content of the pacing pulses.

The aim of this paper is to present some existing hardware and software methods for pace pulses detection.

II. Methods for Detection of Pace Pulses

The pacing pulses detection could be hardware, software or a combination of both.

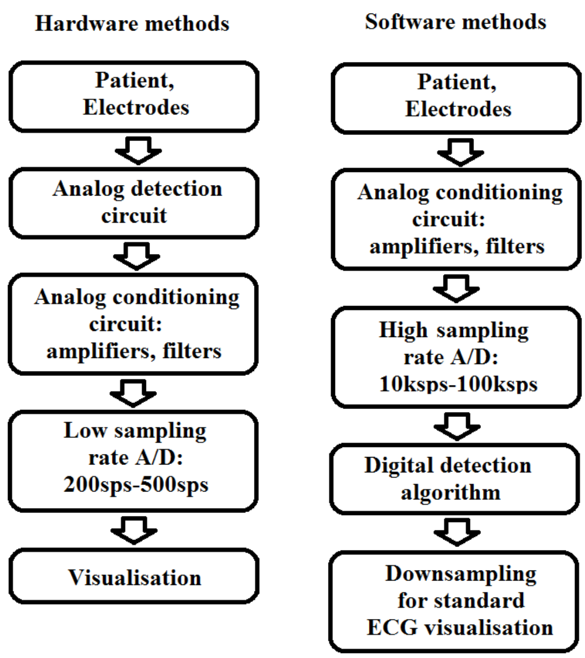

Fig. 1. Block-diagrams of hardware and software pace pulse detection methods

The hardware detection is faster, mostly real-time, uses dedicated circuitry, and consumes less power. Its drawback is that it is not flexible, it is hard to be tuned, and only simple algorithms can be realized this way.
Software (or digital) detection is versatile – the complexity of the algorithms is limited only by the available computation resources and it is possible to adjust and adapt the implemented methods. The digital detection could consume more power because of the high sampling rate of the ADCs required for accurate pace pulses recognition and the power-consuming DSP algorithms. The principles embedded in the hardware and software pace pulses detection are presented by the block-diagrams in Figure 1.

Hardware pace pulse recognition using slope detection

Design proposed by Texas Instruments [9]. Requirements:

- minimum pace signal width - 100μs
- minimum pace signal amplitude - 2mV
- monitor for slope of pacemaker signal while ignoring ECG signal

The first block of the design is a circuit for slope detection (Figure 2). It looks for the leading edge of the pacemaker signal and attenuates the QRS complexes by means of a differentiator circuit. This circuit takes advantage of the characteristic of current across a capacitor and uses the feedback resistor R1 to develop a proportional voltage. The output from the circuit is a signal relative to the slope, dV/dt, of the input. The resistor R1 and the capacitor C1 form a high pass filter (HPF), which is set to monitor for the higher frequency components from the pacemaker signal. Thus, the output is proportional to the slope amplitude and duration while ignoring the P-QRS-T segment of the ECG. According to the patent, C1 is chosen to be 1μF to set the HPF cut off of the differentiator circuit to attenuate the unwanted QRS complex. Setting R1 to 392kΩ and C2 to 10nF help maximize the gain, while keeping the circuit stable. R2 is short circuited to 0Ω but included for design flexibility.

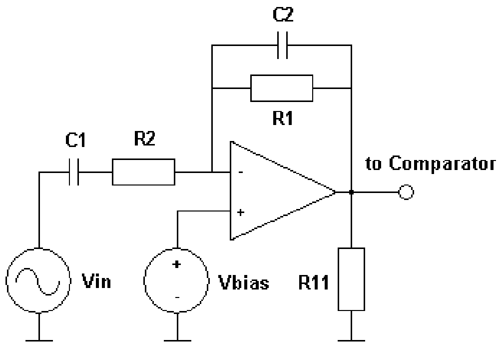

Fig. 2. Differentiator Circuit with Compensation.
The passive components set the poles and zeroes for the transfer function. A second capacitor and resistor - C2, R2 help to stabilize the circuit.

A second capacitor C2 is placed in parallel with the resistor in the feedback loop to stabilize the circuit, along with a second resistor R2 in the input.

The second block of the design is a window comparator circuit (Figure 3). The output of the differentiator circuit produces a pulse when a pace event occurs. This pulse afterwards should be recognized to register an alert. By design, the pulse from the output of the differentiator could range from a few hundred mV to a couple of Volts depending on the pace pulse characteristics. The pulse is captured by using a window comparator circuit designed to output a logic low signal when the comparator input pulse exceeds either a high or low threshold voltage.

When the output from the differentiator circuit, exceeds the pre-set limits V1 or V2, the comparator output triggers by pulling the comparator output line low.

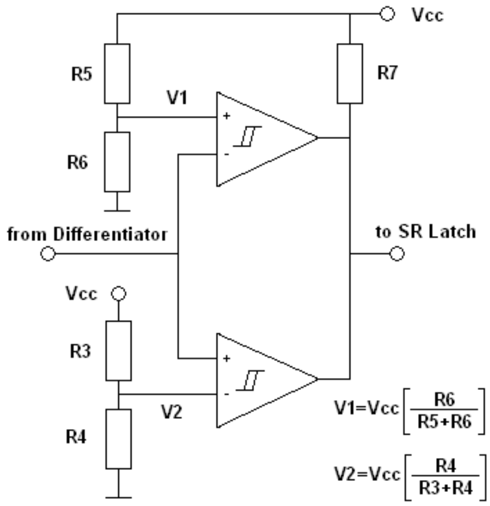

Fig. 3. Window Comparator Circuit

The third block of the design is a SR Latch using two NAND gates (Figure 4) connected to a general purpose input-output of the microcontroller. The width of the comparator output pulse is determined by the time that Vin exceeds the window comparator boundaries. It could be as small as a few milliseconds and may be missed if not latched. The SR latch performs two functions: latching the window comparator output signal and serving as an inverter to create an active high alert output signal.

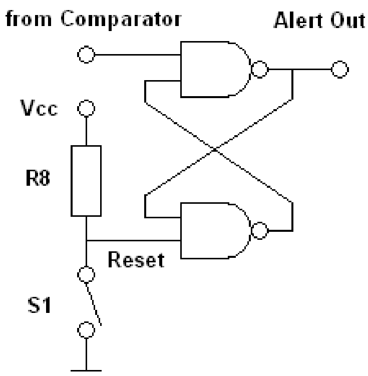

Fig. 4. SR Latch to GPIO

Software pace pulse detection Herleikson's algorithm

This algorithm is proposed by Herleikson, and is patented by Hewlett-Packard - US patent 5682902 [10]. It uses sampling frequency of 4000 Hz. The input signal is filtered by a band-pass filter to separate the pace pulses from the natural ECG and the low-frequency noise:

$$Fi = Filter(x) = x_i+x_{i-1}-x_{i-2}-x_{i-3} \quad (1)$$

which is actually an average of two consecutive two-step finite differences. The information about the presence of a pacing pulse resides mostly on the two samples located at the rising and falling edges of the pulse, which are typically characterized by opposite amplitude and by a distance equal to the pulse [11]. The amplitude-frequency response of the filter is presented in Figure 5 (k=0, the lowest trace).

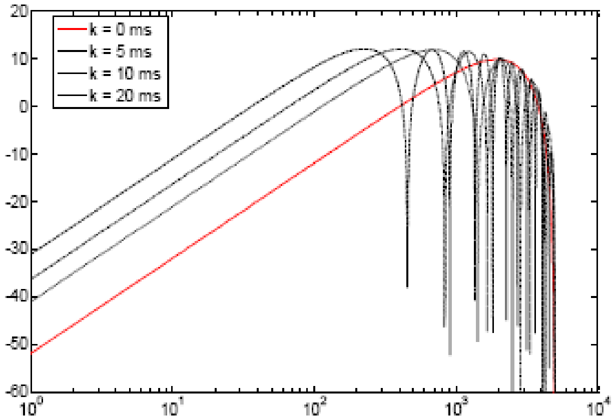

Fig. 5. Amplitude-frequency response of the linear digital filter applied in [10, 11]

For each value of the filtered signal F the absolute value $F_{abs} = |F|$ is calculated. An adaptive threshold is set to be equal to the highest F_{abs} value in a time window of 64 ms before the current sample. It is then multiplied with the coefficient $G_T = 3$.

$$Threshold\ (F_{abs}, W_o, W_l, G_T) =$$

$$G_T{*}max(F_{abs\ i-Wo-1}, F_{abs\ i-Wo-2},\ldots, F_{abs\ i-Wo-Wl}) \quad (2)$$

- W_o is the offset of the time window corresponding to the sample X_i, for which the threshold is determined;
- W_l is the length of this window (64 ms)

The algorithm searches for pace pulse in the time interval (112 – 64) ms before the current sample. This part of F_{abs} is compared with the adaptive threshold. The first value above the threshold is considered as a candidate for pace pulse edge. The next samples within the analyzed part of F_{abs} are searched for another edge that crosses the threshold and is in opposite direction to the found candidate for pace pulse edge. If such edge is found then a pace pulse is detected.

Software pace pulse detection using a non-linear filtering approach

This algorithm is proposed by Polpetta and Banelli [11], and is an upgrade of the Herleikson's algorithm.

The first step is linear high-pass filtering. It removes the natural ECG signal using simple differential filtering. [10,11] and enhances the rising and falling edges of the pacing pulse.

$$F_{HPi} = Filter^k(x,k) = x_i+x_{i-1}-x_{i-k-2}-x_{i-k-3} \quad (3)$$

The difference with the filter in the Herleikson's algorithm is that Polpetta and Banelli introduce a coefficient **k**, which specifies the size of the window in which the filter is working. The amplitude-frequency response of the filter for different values of K is presented in Figure 5.

The second step of the algorithm is non-linear filtering. The authors consider the F_{HP} signal in a window of N previous samples, applying a support window p to improve the detection robustness against noises

$$F_{N,p(-)}(i) = [F_{HPi-p-N}, F_{HPi-p-N+1},\ldots F_{HPi-p}, F_{HPi}] \quad (4)$$

and define associated variation series as the sorted version of the vector – $V_{N,p(-)}(i) = sorted(F_{N,p(-)}(i))$.

The vector signal $F_{N,p(-)}(i)$ and $V_{N,p(-)}(i)$ permit to compare the current sample F_{HPi} with the past ones and consequently to enhance the rising edges of a pacing pulse with respect to the noise. In order to avoid a false pulse detection induced by a sudden voltage rise caused by an electrode-skin contact loss, the authors consider also the vectors $F_{N,p(+)}(i)$ and $V_{N,p(+)}(i)$ composed by the current sample and other N samples in the future. Further the authors defined in both direction (-/+) $R(V_{N,p}(i),n)$ as the position (rank) of the sample F_{Hpi-n} in $V_{N,p}(i)$ and $Val(V_{N,p}(i),m)$ as the value of the sample whose rank is m in $V_{N,p}(i)$. Instead of considering only the signal derivative in the time domain, the authors analyze the derivative in the sorted domain. They compare the amplitude of the current sample F_{HPi} with the closest neighbor in the variation series $V_{N,p}(i)$. Thus, a differential rank signal of a vector V - $RD(V)$ is defined as follows:

- if $R(V,0)>(N+1)/2$, $RD(V) = F_{Hpi} - Val(V, R(V,0)-1)$;
- if $R(V,0)<(N+1)/2$, $RD(V) = F_{Hpi} - Val(V, R(V,0)+1)$;
- otherwise $RD(V) = 0$.

For pace pulse detection the $RD(V)$ signal is compared to a predefined threshold. The value of the threshold is again a compromise between false detection and failing to detect a real pulse. After detecting a valid pace pulse, the comparison is inhibited for certain duration to avoid detecting a single pulse multiple times.

The size of the windows k and p should be chosen carefully .If the window is too short and there is only a single noise spike in it, the noise can be erroneously classified as a valid pace impulse. If the size of the window is increased the probability to get only a single noise spike in it decreases and the probability for a false detection decreases too. On the other hand, if the window is too long a noise spike with amplitude similar with the pace impulse can be collected, thus causing again a false pace pulse detection.

Software pace pulse detection implemented in ADAS100

The device is a front-end ECG module, including a digital pacemaker artifact detection algorithm that detects pacing artifacts with widths in the range (0.1 – 2) ms, and amplitudes in the range (0.4 – 1000) mV. The pace-detection algorithm runs on three of four possible leads (I, II, III, or aVF). This is due to the fact that different pacemaker leads do not have the same vectors, and align better with some ECG leads than with others.

- Right atrium – Lead II or one of the chest leads;
- Right ventricle – Lead II;

Left ventricle is actually placed out of it and is best captured by Lead II or one of the chest leads;

Pacing leads of implantable defibrillators and resynchronization devices are sometimes placed in areas of the heart that do not have an infarction, and it may be hard to choose the correct ECG Lead.

The block-diagram of the method embedded in ADAS1000 [1] is presented in Figure 6. The pace detection algorithm searches for pulses by analyzing samples in the 128 kHz/16-bit ECG data stream. The first step is to search for a valid leading edge. Once a candidate edge has been detected, the algorithm begins searching for a second, opposite-polarity edge that meets with pulse width criteria and passes the (optional) noise filters. Only the pulses that meet all the criteria are flagged as valid pace pulses.

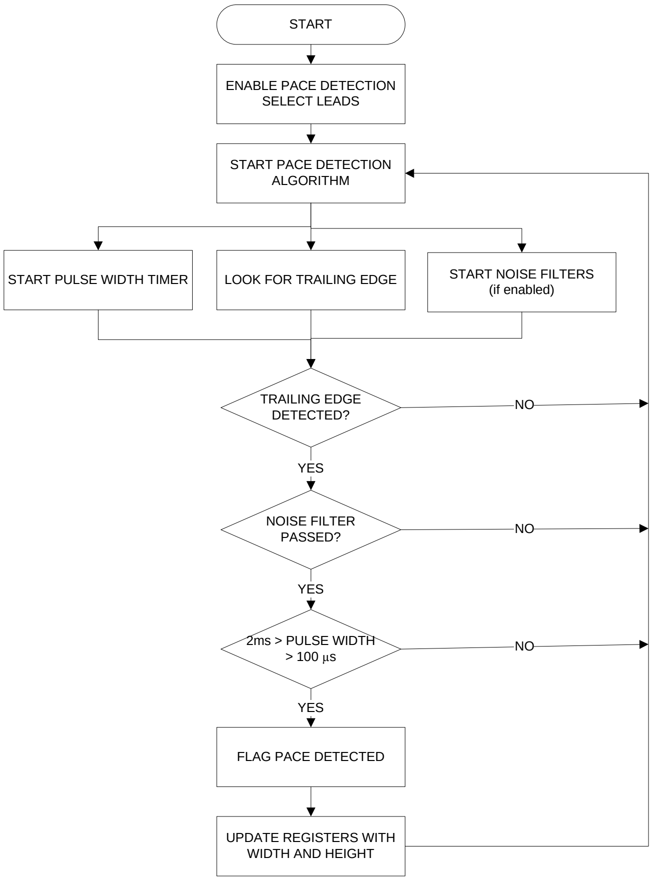

Fig. 6. Block-diagram of ADAS1000 pace detection algorithm

The artifact detection system has programmable threshold levels so it can be tuned:

- Pace Amplitude Threshold would typically be set to be the minimum expected pace amplitude:

$$PACEAMPLTHRsetting = \frac{N.V_{REF}}{GAIN.2^{16}} \quad (5)$$

- Pace Edge Threshold is used to find a leading edge, denoting the start of a pace pulse:

$$PACEEDGETHRsetting = \frac{N.V_{REF}}{GAIN.2^{16}} \quad (6)$$

- Pace Level Threshold is used to find the leading edge peak:

$$PACELVLTHRsetting = \frac{N.V_{REF}}{GAIN.2^{16}} \quad (5)$$

where: N = 0 to 255 (8 bits);. GAIN = 1.4, 2.1, 2.8 or 4.2 (programmable) and VREF = 1.8 V.

Some pacemakers use minute-ventilation pulses with length from 15 µs to 100 µs to detect respiration rates and control the pacing rate (in rate-responsive pacemakers). The ADAS1000 has a minute-ventilation filter built into its algorithm. It also has algorithms for filtering the noise and the heartbeats. There is a specially designed “pace width” filter, which searches for an edge of opposite polarity to the leading edge that has at least half of its magnitude. The second edge duration must be between 100µs to 2ms from the original edge. When a valid pace width is detected, the width is stored.

III. Conclusion

This paper presents a number of methods for pace pulse detection in the ECG, some of which implemented in real ECG modules. It could be used as a background for development of a new algorithm for pacing artifact detection.

Acknowledgement

This work has been supported by R&D Fund of the TU-Sofia, Grant 142ПД0005-03

References

[1] J. M. Kruse, C. Redmond, *"Detecting and Distinguishing Cardiac Pacing Artifacts"*, Analog Dialogue, vol. 46, no. 4, pp. 13-18, 2012.
[2] M. Akay, *“Wiley Encyclopedia of Biomedical Engineering”*, Wiley 2006.
[3] American National Standard *Diagnostic Electrocardiographic Devices*, ANSI/AAMI EC11, 2007
[4] American National Standard. *Medical electrical equipment — Part 2-27: Particular requirements for the basic safety and essential performance of electrocardiographic monitoring equipment*, ANSI/AAMI/IEC 60601-2-27, 2011.
[5] V. Tsibulko, I. Iliev, I. Jekova, *"A Review on Pacemakers: Device Types, Operating Modes and Pacing Pulses. Problems Related to the Pacing Artifacts Detection"*, International Journal of Bioautomation, vol. 18(2), pp. 89-100, 2014.
[6] И. Даскалов, И. Стамболиев, *“Електро-медицинска диагностична техника”*, Издателство “Техника”, 1978.
[7] S. Petrutiu, A. Sahakian, A. Ricke, B. Young, S. Swiryn, *“High Resolution Electrocardiography Optimized for Recording Pulses from Electronic Pacemakers: Evaluation of a New Pacemaker Sensing System”*, Computers in Cardiology, vol. 34, pp. 197-200, 2007.
[8] S. Luo, P. Johnston, W. Hong, *“Performance Study of Digital Pacer Spike Detection as Sampling Rate Changes”*, Computing in Cardiology, vol. 35, pp. 349-352, 2008.
[9] T. Calabria, *“Hardware Pace using Slope Detection”*, Texas Instruments Precision Designs: Verified Design, 2007
[10] E. C. Herleikson, "ECG pace pulse detection and processing", US5682902 A04-Nov-1997.
[11] A. Polpetta, P. Banelli, *"Fully digital pacemaker detection in ECG signals using a non-linear filtering approach"*, Proc. 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS’2008, pp. 5406-5410, 2008.