Paweł IRZMAŃSKI, Adam JÓŚKO

Warsaw University of Technology

Research tools for evaluation, development and optimisation of the QRS detection algorithm

Abstract. *The article presents the Author's approach to testing and evaluation of the QRS detection algorithm. The main goal is to make those two tasks easy, both for engineers and physicians. The specialized environment for the QRS detection quality analysis, in an convenient and effective way, was developed. The paper describes problems in the field that were solved, presents the Author's application which was created and shows how it fulfils the normative documents and physician requirements.*

Streszczenie. *W artykule przestawiono przykładowe podejście do testów i oceny jakościowej algorytmów detekcji zespołów QRS. Głównym celem pracy jest znaczne uproszczenie powyższych etapów składowych. Powstała w wyniku aplikacja umożliwia wygodny i efektywny sposób przeprowadzania testów normatywnych. Artykuł opisuje problemy związane z implementacją uwzględniającą wymagania restrykcyjnych norm bezpieczeństwa wymaganych przez środowisko medyczne. (**Narzędzia do realizacji badań w zakresie oceny działania, rozwijania i optymalizacji działania algorytmu detekcji zespołów QRS**)*

Keywords: ECG signal automated analysis, QRS complex detection, QRS detection testing and evaluation stage, EN 60601-2-51
Słowa kluczowe: Automatyczna analiza sygnału EKG, Detekcja zespołów QRS, Testy jakościowe detektorów, EN 60601-2-51

Introduction

The approach to a brand new modular application for testing evaluation of the QRS detector is presented in the paper. No software engineering process or methodology can guarantee reliable work of any software. Especially medical embedded systems have to be in especial secure. Thanks to presented software user is able to prepare written QRS detector code to move it to the next level of usability.

The environment concerns QRS and cardiac pacemaker (PM) detector modules, initially implemented in Matlab environment [1]. Application is equipped with additional testing signal generators tailored exactly to allow to meet the normative requirements stated by European Industry. The main intention is to move those products to real hardware platform in an easy way. Without respecting mentioned rules one cannot start a final product certification process to allow usage of such devices on medical market.

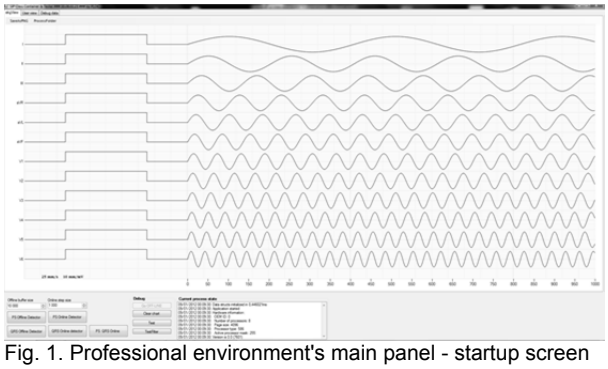

Fig. 1. Professional environment's main panel - startup screen

Background

The highly accurate QRS complexes detector and artificial pacemaker (PM) impulses detector scripts were checked with more than 1700 10s recordings taken from The Common Standards of Electrocardiography (CSE) Database - a set of test signals representing records of the most known heart pathologies. Those records were performed using a standard set of 12 leads, completed by the pseudo orthogonal Frank leads [3].

A PM module was implemented because described environment uses it prior to the main task (QRS processing) to inform the end user (operator or physician) on the PMI (Pacemaker Implantation) activity. Additionally the PM module allows for removing injected impulses from the input signal of essential analysis. The basic view of detection process has been shown in the figure 2. As assumed, the sequence must give possibility to perform tests of developed methods (including the new cases of use) under the requirements stated by European Industry.

The EN 60601-2-51 [5] specifies requirements for the safety including essential performance of a medical equipment and it also recommends calibrated ECGs and analytical ECGs for testing quality of detection process. Utilised ECG recordings are stored under database's specified format which has been developed during the first CSE Workshop [8]. Some of demands are also applicable directly to implementation like specified calibration and test data sets. Authors use these recipes to generate an extra ECG data files which can be used during algorithm robustness evaluation (the orange process block in the Fig. 2).

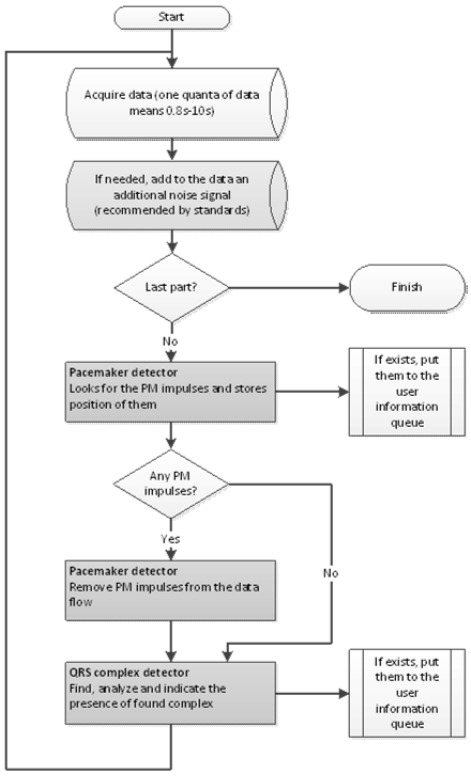

Fig. 2. Basic view of detection process working with acquired in real time data (data blocks) [3]

EN ISO 60601-1-4 [6] delivers the methodology on procedures of implementation and testing stage of the written code. It targets directly to its behaviour in case of undocumented use and other not foreseen situations. The major outcome for developers is FMEA document (Failure Mode and Effect Analysis) generated during the development and evaluation of detectors. Environment is prepared to help perform any user defined cases of problematic use or incorrect data passed to detector methods.

Both normative documents create procedures that allow providing pre-compliant and safe to use product. Both papers have Polish counterparts.

Proposed solution

C++ Windows application was developed under Visual Studio C++ with use of a third party static code analysis tool. Due to indisputable need of move tested algorithm implementation to real platform, the environment and detectors were developed with respect to selected MISRA C++ rules too. It is well known that none software engineering process can guarantee secure code, but following the right coding guidelines can dramatically increase the mobility and reliability of developed code. Authors use for crucial parts of implementation a web based PClint parser software (freely available demo on the Gimpel Company webpage).

Presented application consists of the following three logical modules (Fig. 2.).

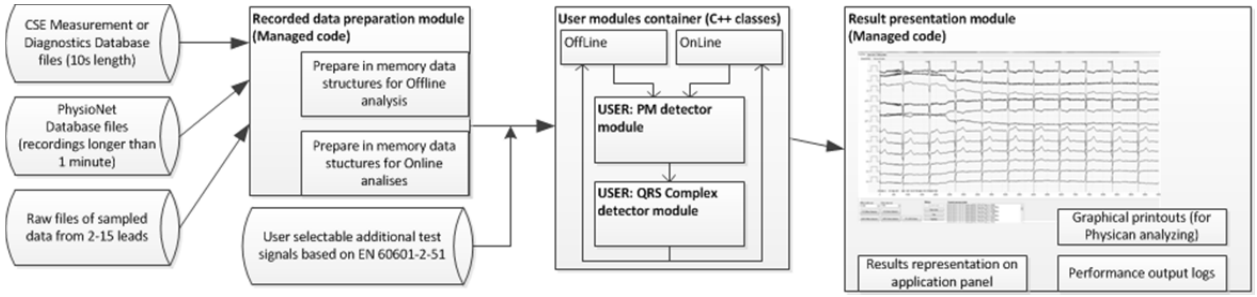

Fig. 2. Testing application structure (including user modules placement)

Data preparation module is intended to open ECG recording from various data sources. Authors present all result taken from the CSE database in a DCD file format, in the paper. Initially it was designed for the exchange of data based on magnetic tapes and was oriented towards research only but now it is also used as ECG records reference standard family EN 60601. Additionally ECG data has included patient record identification structures also used by application. The second module ability is to provide robust data flow in two modes.

The online mode simulates real time ECG data acquisition from 12 leads by generating 0.8-10s pieces delivered continuously - it can work like on real hardware or provide data from databases simulating such process.

The offline mode - decodes and prepares single solid twelve channels structure. This functionality applies when working with large ECG records and it is desirable to have the most accurate outcome with no priority on speed of processing.

The example data from PhysioNet - especially PhysioBank is a large and growing archive of biomedical signals. Authors decided to add its data import possibility also to improve scope of possible test areas for end users - not described in this paper. To meet stability tests required by EN 60601-2-51 user is able to add additional noise signals previously prepared and consistent with appendix HH of mentioned document [7,4].

User module container enclosures user detector classes (in this case called SIP_EKGFiltractionPack) written in pure C++ (Fig. 3). These members were equipped with standardised prototype and a list of possible output states to meet EN ISO 60601-1-4 requirements. Each subclass works with online and offline data. The SIP_DetektorKS is a PM detector and impulses remover and the SIP_DetektorQRS is a QRS complex detector. Mostly both of them are used together (Fig. 1).

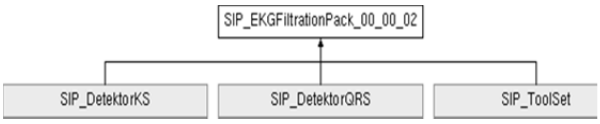

Fig. 3. SIP_EKGFiltractionPack container structure

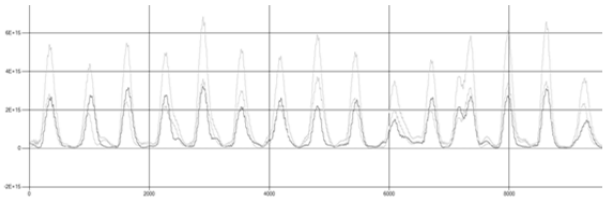

Fig. 4. D_00576.DCD record file feature detection function (10s). The three input parameters was shown

Most of available computing time should be reserved for this processing part. Every change in algorithm or secure addon connected with EN document gave an extra amount of cycles to be performed by a medical computer. Typical platform is based on Intel Atom processor so it is importat to find a ballance beetwen a reliability demanded by the normative documents and execution per data step time. For the enviroment based on the Intel i7 platform in application debug mode Authors obtained the following sample execution times:

- *D_00001.DCD processed in 14,58ms*
- *D_00002.DCD processed in 10,99ms*
- *D_00003.DCD processed in 10,97ms*

The time between data acquisition steps is not greater than 800ms in the worst case. Tests performed on slower platform showed the longer typical execution time up to 200ms on the Intel Celeron 1.7GHz processor (also in the debug mode). The amount of the system memory is not

critical but the minimal planned size is 1GB due to operating system requirements (Windows XP Embedded or CE). It is a fact that the during initial step of the algorithm in the online mode the computing time of the first data quanta is typically 40% longer that the rest of them. This phenomena was not observed during test on the hardware platform. It will be investigated during the next stages of the environment developing.

The major function of this module is to determine the usefulness of prepared medical software classes. The clear container module implementation with EN 60601-1-4 rules in mind gives chance to perform professional testing of developed code on a desk.

Result presentation module is a panel window for each working part of presented software unit. A developer is able to look into particular input data parts stopping execution between them, analyse a feature detection function (Fig. 4) and observe how it changes during following steps of data acquisition, also the detector's internal parameters can be taken under consideration via a separate non modal window. And finally one can put output flow to the previously selected folder in a batch mode besides of displaying it on the professional app panel.

Typically, a developer can use an application panel to determine the quality of detection and do non-standard invokes of detectors function observing the results on screen and in log files. In turn, a physician is rather interested in the effectiveness of finding R waves in polluted records. Both modes are extremely easy to perform and no Matlab environment is needed.

In case of different data and noise scenarios one is able to store all test automatically with the following format and further analyze them fulfilling FMEA papers:

D_00001.DCD processed in 14,58ms HR=57
Peaks [ms] on 451 1500 2536 3573 4577 5579 6643 7698 8719 9739

D_00002.DCD processed in 10,99ms HR=64
Peaks [ms] on 106 1025 1911 2809 3726 4635 5546 6477 7401 8311 9244

D_00003.DCD processed in 10,98ms HR=46
Peaks [ms] on 304 1557 2870 4075 5411 6745 7951 9261

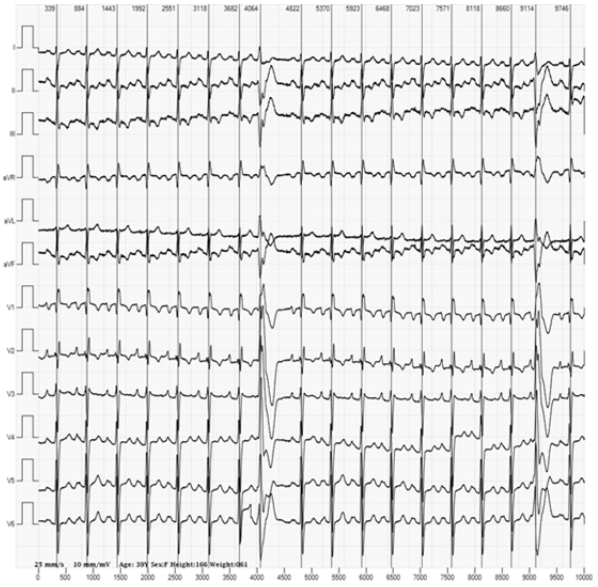

Fig. 5. D_00643.DCD record file with detected R waves

The main screen as well as ECG record file stored, after detection (Fig. 5), in graphic format on hard disk contain the HR (Heart Rate) value computed for the last data part and the wave R positions marked. The values in the log files can be used to compare the effectiveness of detection method with selected noise profile, improper input ECG data level (ADC value per real 1mV signal) or incorrect invoking of pure C++ method.

Results

The presented application helps testing and allows to import detector scripts from Matlab to C++ code with conserving full compatibility to external platform based on Intel Atom like microprocessor or faster with efficiency in mind. Thanks to the wide input data format the user is able to check modules on almost any kind (format) of ECG record and has got many ways to simulate and perform additional signals described in the normative documents.

Depended on the analysis mode user is able to simulate many different cases of disrupted or polluted ECG signals that are imposed by standards, with results and feature detection function shapes preview including the long processing process. The observed deterioration in the detection quality allows to apply changes decreasing error susceptibility directly in the source code.

There are still many functionalities pending on next release tasks list however the presented version of the professional environment was very helpful during preparing a SIP ECG Classes by the team.

The mentioned research was carried out in connection with the project being conducted by FARUM Sp. z o.o. Warsaw.

REFERENCES

[1] Jośko A., *Quantitative analysis of the QRS complex detection in ECG*, Computational Problems of Electrical Engineering CPEE 2010 Conference Lazne, Czech Republic, 13-16 September, (2010).
[2] Jośko A., Rak R.J., Effective simulation of signals for Testing ECG Analyzer**,** *IEEE Transactions on Instrumentation and Measurement*, June (2005).
[3] Augustyniak P. Miszczak D., *Testing the quality of the algorithms for automated electrocardiogram analysis*, Congress of the Polish Society of Medical Physics, Kraków Sept. 15-18, 1995
[4] J.H. van Bemmel and J.L. Williams, "*Standardization and validation of medical decision support systems: The CSE project,*" Methods Inform. Med., vol. 29, pp. 261-262, 1990.
[5] European Standard EN 60601-2-51. 2003.
[6] European Standard EN 60601-1-4. 1999.
[7] Hargittai S., Measurements standards and test signals in QRS boundary determination, *Computers in Cardiology 2007*
[8] The CSE Working party, Recommendations for measurement standards in quantitative electrocardiography. European Heart Journal 1985;6:815-825.

***Authors**:*
Paweł IRZMAŃSKI M.Sc.,pawel.irzmanski@ee.pw.edu.pl
Adam JÓŚKO PhD.,adam.josko@ee.pw.edu.pl
Warsaw University of technology, Institute of Theory of Electrical Engineering, Measurement and Information Systems, 75 Koszykowa Str., 00-662 Warsaw.