REAL-TIME SUBTRACTION PROCEDURE FOR ELIMINATING POWER-LINE INTERFERENCE FROM ECG

Slavy Georgiev Mihov*, Ratcho Marinov Ivanov, Angel Nikolaev Popov

*Faculty of Electronic Engineering and Technologies (FETT), Technical University – Sofia, Kl. Ohridsky str. No 8, bl. 1, 1797, Sofia, Bulgaria, phone: +39529653281, e-mail: smihov@tu-sofia.bg

Presented is the algorithm of the subtraction procedure for removing powerline interference from ECG signals. The existing MatLab prototype version is adapted for real time execution on DSP and programmable logic (FPGA) platforms. The algorithm is ported and experimented in case of odd sample number in one period of the interference. The corresponding formulas for even sample number are given too. Key moment is the use of the ongoing interference values, which are stored in a temporary buffer and then used for recalculating the filter's coefficients. Test results of the prototype MatLab code and its two implementations (for DSP and FPGA) show that the presented real time algorithms successfully compensate the presence of powerline frequency interference the ECG signal. Given are the changes in the DSP port of the algorithm and the generated structure by the FPGA configuration code.

Keywords: Powerline interference, ECG signal, Subtraction procedure

1. INTRODUCTION

The subtraction procedure for eliminating powerline (PL) interferences in electro cardio graph (ECG) signals [1] shows very promising results and is subject to intense investigation and refinement [2]. Its structure (Fig. 1) comprises three major stages.

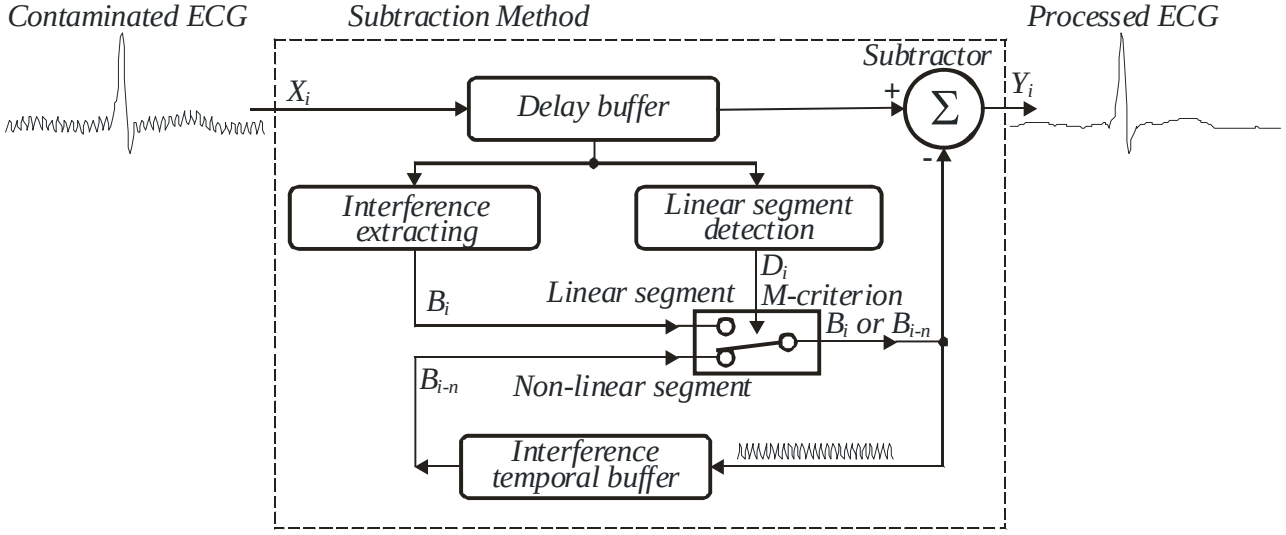

Fig. 1. Generalized structure of the subtraction method

1. Every point of the electro cardio signal is checked for belonging to a linear or non-linear section with introduced interference (mathematically, the second derivative of the signal should be less than a particular minimum). The check is done with the condition $|D_i| < M$ (criterion for linearity), where M is a practically chosen threshold. The most popular criterion for linearity

$$D_i = \left(X_{i-n} - X_i\right) - \left(X_i - X_{i+n}\right) = X_{i-n} - 2X_i + X_{i+n}, \quad (1)$$

is second difference of the signal samples' values. First differences are taken among the samples, with distance in between them equal to the PL interference period, thus eliminating its influence in the estimate for linearity of the section. X_i stands for the ongoing input sample.

2. The PL interference is removed with a non-recursive symmetric digital FIR filter 'moving average' [3]

$$Y_i = \frac{1}{n}\left[\sum_{j=-m}^{m} X_{i+j} - \frac{2m+1-n}{2}\left(X_{i-m} + X_{i+m}\right)\right] \quad (2)$$

and only the useful signal Y_i remains. Here n is the number of samples in one period of the PL interference. At an 'odd multiplicity' $n = 2m+1$ while at an 'even multiplicity' $n = 2m$ .At the same time, simple subtraction between the filtered and non-filtered signals gives the momentary B_i value of the interference

$$B_i = X_i - Y_i . \quad (3)$$

which is stored in a temporary buffer.

3. If the signal sample being processed does not belong to a linear section, its value is taken from the temporary buffer

$$B_i = B_{i-n} . \quad (4)$$

which is identical in phase with the ongoing one. This value compensates the signal

$$Y_i = X_i - B_i . \quad (5)$$

(being subtracted from the signal) and is stored again in the temporary buffer.

The temporary buffer keeps n preceding values of the PL interference $B_{i-1}, B_{i-2}, \ldots, B_{i-n}$. This delay buffer is used to compensate the phase shift of criterion for linearity and moving averaging, because they are non-casual (physically unrealizable) filters.

2. PROTOTYPE ALGORITHM (MATLAB)

The prototype of the subtraction method for eliminating PL interference in ECG signals [1] is developed and tested in MatLab. Real ECG records of patients are used for processing. The sampling frequency used for recording these signals is 250 Hz (standard for this type of medical equipment). The primary purpose of the investigated algorithm is to remove interferences with frequency of 50 Hz caused by the PL. The exact multiplicity between the sampling frequency and the interference is what the investigated algorithm owes its functionality to. The real biometric signals used for testing the algorithm have magnitude in the range ±2.5 mV, recorded in the scale of 20 μV/bit. The magnitude of the interfering signal is comparable (in the same range), which makes its removal extremely difficult.

The prototype of the subtraction filtering algorithm is implemented in MatLab script. It is formed as a procedure operating out of real time, which processes a buffer of input signal sample values. The PL interference is generated as a 50 Hz sinusoid and added to the ideal biometric signal by the software. The so obtained contami-

nated signal is subjected to processing and estimates by the filtering algorithm. Part of it with length of 2.8 s is stored in a buffer, 700 samples long. This buffer is processed cyclically by taking a sample of the input signal sequence, performing the necessary computations and producing a resultant value in the output. For the purpose of visualizing the results, the output values are stored in another buffer. This allows making comparison between the obtained result and the initial ideal ECG signal.

Accurate processing of the signal is based on preliminary detection and estimate of linear and non-linear sections in it. Depending on this criterion, the processing continues in two different ways. The digital filtered released by this algorithm needs the pre-history and the post-history of the current signal sample being processed. That's why the incoming signal is stored in a delay buffer, where a series of sequential samples around the moment of interest are available.

The results of operation of the so realized digital filter are shown graphically in Fig. 2. The first diagram (Fig. 2a) contains part of a real electro cardio signal, free of any interference. After synthesizing and adding interference to it the signal takes the form from the second diagram (Fig. 2b). This second signal is used as an input of the algorithm, for testing its capabilities in removing the interference.

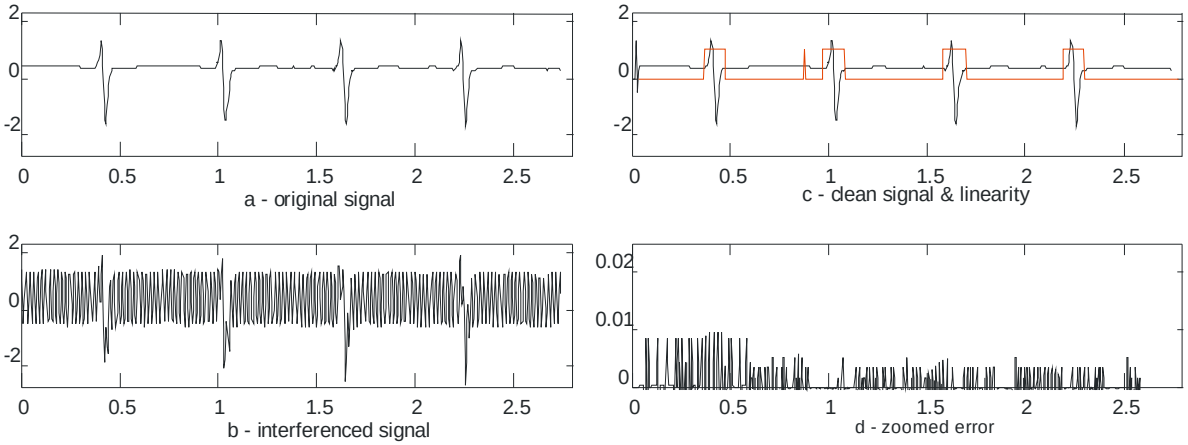

Fig. 2. Subtraction procedure operational results

The third graphic (Fig. 2c) shows the output signal, obtained after eliminating the PL interference from the input one. Notable is the exact match with the initial one. The same diagram contains also, the criterion for linearity, which takes only two values and classifies the corresponding section as linear or non-linear. Storing the input and output signals in buffers allows to make comparison between them, with ease. The difference (absolute value) between their sample values gives the error made by the operation of the processing algorithm when filtering the signal. This error is displayed in the last diagram (Fig. 2d). Notable is the scale of the error that is below 10 μV or 0.5 samples of the ADC. The last result is especially demonstrative for the operation of the so realized digital filter and its precision.

3. Real Time DSP Implementation

After confirming the functionality of the subtraction method for removing PL interferences from electro cardio signals theoretically in MatLab, software realization

of the algorithm was made in C for real time execution by DSP [4]. The development platform used is Analog Devices' BF537-STAMP operating with processor Blackfin-537. The system is complemented with an ADC and a DAC used as I/O devices of the system. The interconnection with the main board is done via the serial port. When receiving input data from the ADC, a serial port interrupt on receive is triggered, which starts the subroutine for processing the input sample just received. Upon completing the computations, the output sample is sent back the serial line to the DAC for producing the output (filtered) signal. The mere processing takes less time than the length of the interval between two successive serial port interrupts on receive, which allows this procedure to be executed in real time. The software initializes the clock frequency that leads the serial data transfer, thus regulates the length of the interval between the interrupts, setting the desired input signal sampling rate to 250 Hz. The development platform used (BF537-STAMP) operates on 500 MHz and completes the necessary computations in 7 μs, which is significantly shorter time than the limit of 4 ms between the arrivals of two input samples, for real time execution. This suggests that the same filtering algorithm can be executed by slower processors with ease.

During the real time tests of the execution of the program, the results are observed visually on the oscilloscope. Below are given the results from the simulation of device's operation, made in Analog Devices' IDE VisualDSP++. The simulation results (Fig. 3) of this concrete realization of the algorithm are identical with those of the theoretic experiments in MatLab.

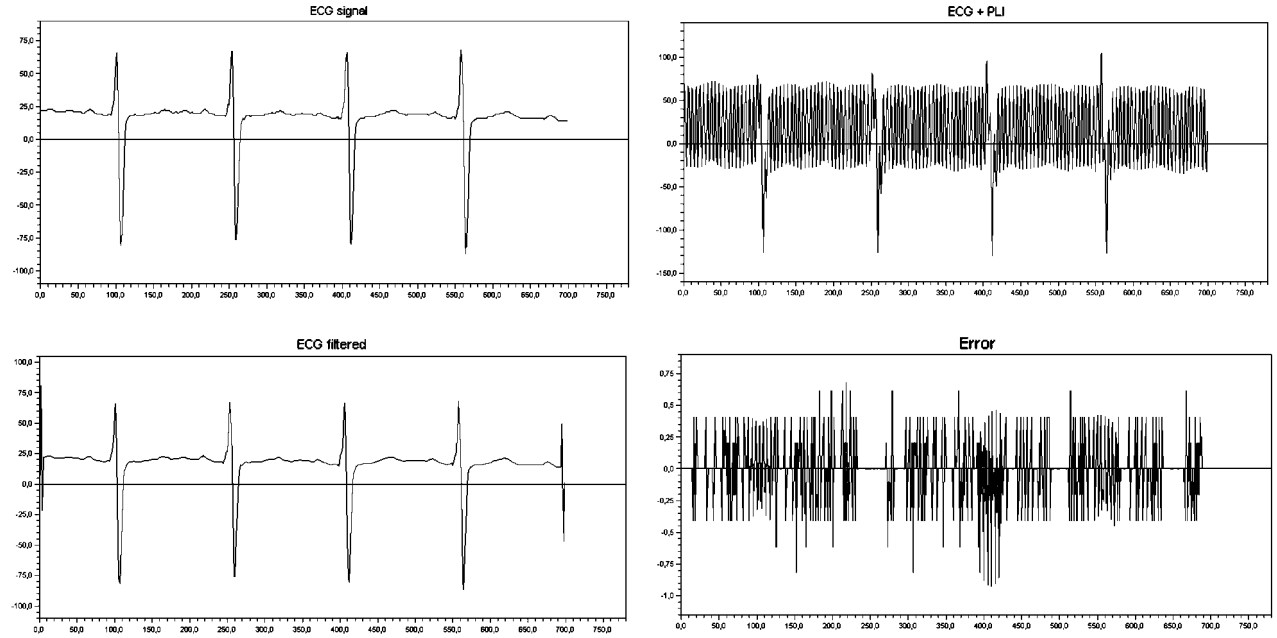

Fig. 3. Real time DSP implementation results

The notable difference is only in the scale of the vertical axis. This discrepancy is due to preliminary scaling the input signals (magnify the magnitude 50 times), for the purpose of reducing the computational errors when working with floating point signal sample values. As a result, the vertical axis unit becomes sampling numbers. Abso-

lutely evident is the variation of the error (Fig. 3d), which is kept in the interval less than ±0.5 samples.

The obtained accuracy in the practical realization of the filtering algorithm is significant and matches exactly the theoretic model studied with MatLab.

4. PROGRAMMABLE LOGIC IMPLEMENTATION (FPGA)

The subtraction method for removing PL interferences from electro cardio signals proves its functionality in real time in theory and practice. The third phase of this algorithm development consists of hardware implementation of all the computational procedures [5]. It is chosen an approach of work with programmable logic and a description language – VHDL. In this particular case, without restricting the generality of investigation, as a target platform is used a Xilinx development kit based on Spartan-3A 1400K.

The description of device's architecture repeats exactly the structure of the main processing algorithm in Fig. 1. The topmost level of the description is structural, where the main functional blocks are declared and connected. The description of the particular modules is behavioral, organized as lower levels in the hierarchy.

In general, the main functional blocks in the structure of the algorithm are implemented as:

- shift register – stores the input signal
- arithmetic block – performs the computations for determining the interference in the input signal
- delay line – stores the detected interference signal
- arithmetic block – computes the criterion for linearity
- multiplexer – switches the interference signal source, controlled by the linearity criterion
- subtractor – forms the difference between the input signal and the estimated interference

In order to simplify the hardware implementation, two modifications in the processing algorithm are introduced. As first, all computations are done in integer arithmetic. We assume that the sample values of the operational signals are scaled in such a way, that all the fraction parts disappear. For example, instead of working with floating point values in the interval ±100 samples, we can use 10^6 times larger numbers without fraction part, from the interval $\pm 10^8$. The mere scaling of the samples can be easily done by arithmetic bit shift left or right. For avoiding computational errors due to loss of accuracy, all signals in the VHDL description of the structures are defined 32-bit long.

The second modification done is for the purpose of eliminating operation division from all computations. This is possible only when the sampling rate of analog signal capture is exact multiple of the interference frequency being removed. In this particular case, the equations take simpler form where no division is present. The issue with averaging N signals samples is solved by simple sum producing N-times larger signal in the output. This is not a big deal, as it can easily be scaled, if need, when convert-

ing back to analog form, which is out of the scope of any filter processing.

As a result of all modifications made, the VHDL description of the filtering algorithm generates the following hardware structure, which matches exactly the theoretic method structure from Fig. 1.

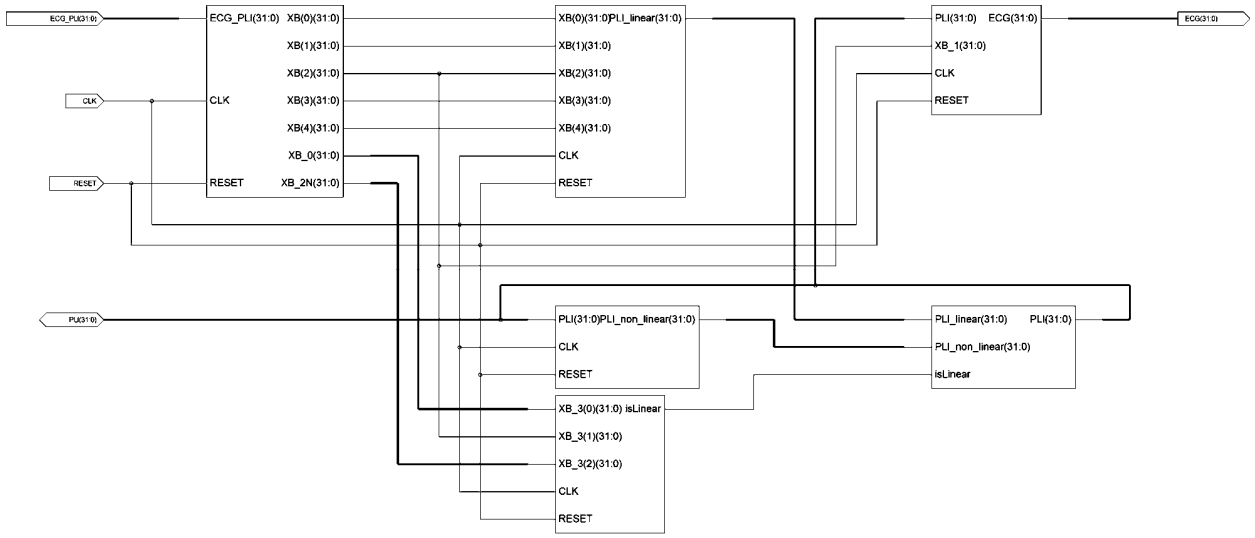

Fig. 4. Programmable logic structure of the filter

5. CONCLUSIONS

In the present paper are shown two possible implementations of one very promising method for removing PL interferences from ECG signals. The theoretic base of the method is realized and tested for consistency in MatLab environment. Obtained are demonstrative proofs for the functionality of the chosen method.

The developed theoretic model is implemented in C programming language, for real time execution on DSP. The test results of the physical realization of the filtering algorithm and the simulations done, confirm the results for processing accuracy of the theoretic study. Third approach for hardware implementation of the investigated filter is completed in the structure of FPGA. For simplifying the description are introduces some minor changes in the computational procedures.

As a whole, the investigated algorithm proves to be functional and obtaining extreme accuracy.

6. REFERENCES

[1] Levkov, C., G. Mihov, R. Ivanov, I. Daskalov. *Subtraction of 50 Hz Interference from the Electrocardiogram*. Medical & Biological Engineering & Computing 22, pp. 371-373, 1984.

[2] Levkov, C., G. Mihov, R. Ivanov, Ivan K. Daskalov, I. Christov, I. Dotsinsky. *Subtraction Method for Powerline Interference Removing from ECG*. 13-th International Scientific and Applied Science Conference Electronics ET'2004, b. 1, pp. 3-14, Sozopol, 2004.

[3] Mihov, G., R. Ivanov, C. Levkov. *Casual Filter Applications in the Subtracting Method for Power-Line Interference Removing from ECG*. (in this issue), 2008.

[4] Adams, M., M. Hayes. *Blackfin Data Acquisition System*. Electronics New Zealand Conference 2006, New Zealand, pp. 228-233, 2006.

[5] Hwang, J. *FPGA Code Portability for Signal Processing in Software Defined Radios*. Military and Aerospace Programmable Logic Devices Conference MAPLD-2004, Washington D.C., pp. 178-181, 2004.