Elsevier Editorial System(tm) for Computers in Biology and Medicine
Manuscript Draft

Manuscript Number: CBM-D-16-00721R1

Title: Custom FPGA Processing for Real-Time Fetal ECG Extraction and Identification

Article Type: Full Length Article

Keywords: Embedded systems, fetal ECG, Field Programmable Gate Array (FPGA), Biomedical instrumentation

Corresponding Author: Professor. Francesco Leporati, PhD

Corresponding Author's Institution: University of Pavia

First Author: Emanuele Torti, PhD

Order of Authors: Emanuele Torti, PhD; Dimitris Koliopoulos, Dr.; Mirko Matraxia, Dr.; Giovanni Danese, PhD; Francesco Leporati, PhD

Abstract: Monitoring the fetal cardiac activity during pregnancy is of crucial importance for evaluating fetus health. However, there is a lack of automatic and reliable methods for Fetal ECG (FECG) monitoring that can perform this elaboration in real-time. In this paper, we present a hardware architecture, implemented on the Altera Stratix V FPGA, capable of separating the FECG from the maternal ECG and to correctly identify it. We evaluated our system using both synthetic and real tracks acquired from patients beyond the 20th pregnancy week.
This work is part of a project aiming at developing a portable system for FECG continuous real-time monitoring. Its characteristics of reduced power consumption, real-time processing capability and reduced size make it suitable to be embedded in the overall system, that is the first proposed exploiting Blind Source Separation with this technology, as the best of our knowledge

CONFLICTS OF INTEREST DISCLOSURE

The authors (who participated in the research and in the article preparation jointly and on a equal basis) hereby certify that no personal or financial relationships exist with people or any other organization that could inappropriately influence the work.

Francesco Leporati

GENERAL AUTHORS AGREEMENT

All authors have seen and approved the final version of the manuscript being submitted. They warrant that the article is the authors' original work, hasn't received prior publication and isn't under consideration for publication elsewhere.

Francesco Leporati

*Revision Notes

Response to Reviewers' comments to manuscript CBM-D-16-00721

The authors gratefully acknowledge the Editors and the Anonymous Reviewers for their detailed and highly constructive criticisms, which greatly helped us to improve the quality and presentation of our manuscript. In the following, we provide detailed, item-by-item, point-by-point responses to all the very interesting issues raised by the Anonymous Reviewers. We would also like to emphasize that, in order to simplify the review of our manuscript, we have highlighted the main modifications introduced in the revised manuscript in blue color to help the Associate Editor and the Anonymous Reviewers in finding the changes made with regards to the previous version. We are indebted to them for their careful assessment and outstanding suggestions for improving our manuscript, which have been really helpful in order to enhance its presentation and technical quality.

Response to Comments by Reviewer 2

========
In General
========
The work presents a hardware system for FECG monitoring implemented on a FPGA. The article should be improved. Some comments/suggestions follow:
The authors would like to take this opportunity to gratefully thank the Reviewer for his/her very accurate summary of our work. The comments and suggestions for improvement provided by the Reviewer are well-taken and greatly appreciated. Below we provide an item-by-item response to these interesting comments and suggestions.

1.Introduction:
* Authors comment that: "The monitoring of the fetal cardiac activity is a standard examination used, together with sonogram…". Really, fetal cardiac activity monitoring is not a standard activity because "… there is a lack of automatic and reliable methods for Fetal ECG (FECG) monitoring", as mentioned by authors in the Abstract (only FHR monitoring using Doppler methods is widely used as a standard).
The Reviewer's comments are well taken. We have modified the paper accordingly to the Reviewer's remark highlighting that only methods based on Doppler approach can be considered standard. We very much appreciate the outstanding suggestions provided by the Reviewer, which greatly enhanced the presentation of our approach and its evaluation with regards to other approaches

* "We choose to use a high-end FPGA device because we need high performance in terms of elaboration speed but also low power consumption". The Stratix V FPGA used cannot be considered a low power device.
This is a very good point that helps us to avoid a probable misunderstanding. With the term "low power" we meant "low with respect to" (or at least comparable) the solutions for FECG extraction that can be found in literature, not "absolutely low". In the revised version of the manuscript we changed this reference modifying it in "reduced power" meaning that all the solutions implemented are optimized for power restraint as also Altera claims in its web site.

* "At the best of … this is the first wearable device exploiting the FPGA technology for FECG monitoring". It

is questionable that a system based on a Stratix V FPGA, with 1064 pins could be considered a wearable device. Moreover, it requires additional circuitry, together with an analogic stage for acquiring the required bio-signals, leading to a significant oversize. Perhaps, the system could be considered "portable" rather than "wearable". The use of FPGAs for FECG monitoring has been proposed in other works such as [ref1], although with different features.
This is really true and we fully agree with the Reviewer in his/her remarks. In order to address this issue, we have replaced throughout the paper the word “wearable” with “portable” that is the really target of our work. Concerning the paper by Morales et al., we sincerely thank the Reviewer for her/his advice that improves the analysis of the state of the art. The principle followed in this work is to perform a subtraction of the noise on the acquired signal that is then processed through a no BSS algorithm. The number of the tracks is considerably lower than our ones and the elaboration speed is greater than us although compliant with the real time requirement (probably justified by the lower working frequency). Although the proposed solution is less efficient than our one we add it in the state of the art and (more important) we modified our claim in “the first exploiting Blind Source Separation exploiting FPGA technology …”).

2. Section VI:
*What is the content of the "fetal channel" presented in Figure 9? Is this signal perfectly separated from the mother ECG?
The Reviewer’s comments are well taken and we thank her/him for giving us the possibility to clarify. As mentioned in the new version of the paper “The output of this phase is the identifier of the channel containing the fetal signal (indicated as fetal channel in figure 9)” thus this points to the extracted fetal signal after the Infomax processing. The signal is perfectly separated from the mother’s one depending on the quality of the acquisition. As we will say in the next answer the typical SNR for these acquisitions is 50-60 dB and in these cases (that are the most ones) the separation works perfectly. Of course when the acquired tracks are particularly “noisy” Infomax could fail.

*What is the SNR of this signal? Typically 50/60 dB.

*A figure showing examples of the signal in the fetal channel would be illustrative.
The Reviewer’s comments are well taken and we thank her/him for giving us the possibility to clarify. Fig. 3 shows the channels as they were produced by Infomax that means they are separated. We added in the caption an explicit mention to the channel containing the FECG which is the number 5. On the other hand, the reader can easily identify this, looking at figure 7 where the channel (blu color) not discarded and with the highest number of turning points is indeed the fifth. Consider that in this figure the channels discarded by the noise identification phase are colored in red while those sent to the K-means classification algorithm are colored in blue.

*Can the FHR be extracted from this signal? It so, how the system performs this detection?
The Reviewer’s comments are well taken and we thank her/him for giving us the possibility to clarify. Yes we have the possibility to perform the FHR evaluation, since we have the peak number and the acquisition duration. We have not implemented this calculation since we consider it as one of the possible future very easy developments.

* What information about the FECG signal (S-T interval, etc) can or not be extracted from this signal?
This is a very good point. The information that can or cannot be extracted by the signal depend of course on the quality of the acquisition that preserves or not all the information contained. Assuming a good SNR,

every kind of morphologic analysis can be conceived on the signal and this is the reason for which it is important for us to save enough space and resources on the FPGA to allow this directly on the signal in real time and not offline. We have included this consideration in the revised version of the manuscript as well.

3. Section VII:

*Experimental results are not adequately presented. Table I should include resource usage of the different systems under comparison. Resource usage for the proposed implementation is mentioned in the text, but it should be given in terms of LEs (Logic Elements), and not in percentages of the device. Also, a comparison about operating frequency of the different architectures could be interesting for readers.
The Reviewer's comments are well taken and we modified the Table I adding the working frequency. In the text, moreover, the number of Logic Elements was explicitly mentioned for our project whilst in case of DSP based solutions it is not possible to mention or in case of other FPGA based solutions simply it was not provided by authors.

4. Conclusions
* Power estimations are obtained from simulations are restricted to the FPGA device. It requires additional circuitry, and it is not clear that the complete system could be "compatible with the constraints given by a wearable device".
This is an important remark and as we have already said in a previous answer, we have tried to avoid any misunderstanding by replacing "wearable" with "portable". On the other hand, we think that the Reviewer's comment open to us the possibility of enriching the paper providing more information. Thus we made an estimation of the system autonomous duration assuming it will be equipped with TR1865 lithium batteries: this will guarantee about 3 hours and 30 minutes of autonomy working at 95 MHz. This estimation has been done considering that the FPGA work at its maximum frequency (if the frequency is lower the power consumption will accordingly diminish), the acquisition module is equipped with a STR711 microprocessor supplied at 3.3 V and the WiFi hypothesized module is a Texas WL1807MOD with a 54 Mbps transmission rate. We consider this a good proposal although not the definitive one and a good step toward the instrument total portability preserving its real time processing capability. We add all these considerations in the final section of the paper. We thank the Reviewer for this important and very "stimulant" comments.

* "Moreover, the resource usage is low, allowing to implement future algorithms on the same FPGA". The resource usage is not low; you have additional space for future implementations because you are using a large FPGA device.
The Reviewer is completely right, we cancelled the "low" word in the text and simply we claimed that the "usage is compliant with future algorithms and implementations".
On the other hand we explicitly mentioned that other FPGA models were considered with less resources so as to reduce the overall power consumption. In particular, the Altera Cyclone V device was taken into account but it would not allow to implement all the previously mentioned functionalities and this would prevent future expansions of the design.

- A general flow diagram of the FECG identification system should be included for clarity, together with an explanation of the task performed by each stage.
The Reviewer's comments are well taken and we think that figure 4 (related to the classification phase) figure 5 (concerning the filtering) and figure 8 (concerning the Infomax algorithm) could already give a good

idea of the overall elaboration performed. Of course they are performed in sequence and if it is necessary a further figure connecting them we will be glad to produce it.

- There are some typos and grammatical errors in the text that should be corrected.

All the typos indicated by the Reviewer were identified and corrected moreover we completely revised the paper to improve its readability and the writing English style used. Thanks to the Reviewer for pointing out this issue that considerably improves the quality of the proposed paper.

Last but not least, we would like to take this opportunity to gratefully thank the Reviewer again for his/her assessment of our manuscript and for his/her comments and suggestions for improving our work, which have greatly helped us to raise up the technical quality and presentation of our manuscript.

Response to Comments by Reviewer 3

The authors would like to take this opportunity to gratefully thank the Reviewer for his/her very accurate summary of our work. The comments and suggestions for improvement provided by the Reviewer are well-taken and greatly appreciated. Below we provide an item-by-item response to these interesting comments and suggestions.

1) Typos:

- Table I: The Author of reference [39] is Hatai and not Hasan
- Table I: I guess the authors mean reference [25] instead of [39]
- Page 8: FPGA has been exploited ...

All the typos indicated by the Reviewer were identified and corrected moreover we completely revised the paper to improve its readability and the writing English style used. Thanks to the Reviewer for pointing out this issue that considerably improves the quality of the proposed paper.

2) Novelty:

- The authors claim on page 2 that to the best of their knowledge this is the first device exploiting the FPGA technology for FECG monitoring. However, they have in Table I a comparison to the reference [25] where a FPGA solution was already presented in 2013. Can the authors better explain this matter? In section III about the state-of-the-art, the authors even claim, that the solution presented by [25] is the first implementation of such on a Stratix II FPGA. Without further explanation, this is a contradiction.

This is a very important point and we thank the Reviewer to give us the opportunity of trying to be clearer. The real novelty of the work is in the implementation of a Blind Source Separation algorithm like Infomax on a programmable logic device. This is not done in the work of Hasan that employs other algorithms and also in the work of Morales that we recently discovered although they both proposed FPGA implementations of techniques for FECG extraction. We modified the paper accordingly to these considerations and we upgraded the analysis of the state of the art although highlighting how it is very difficult to perform a real comparison with these works since they do not provide information on power consumption and the test cases are very different and less with respect us. In the case of Morales the elaboration speed is also less than us. Thus the reader could conclude that the novelty is in the algorithm, in the type of the device used, in the width of the test performed and in the completeness of the results presented.

3) Power consumption:

- The authors claim to have a wearable device with low power consumption (1 W of power is consumed in the FPGA). Is that really low power and if so, then relative to what exactly? I guess the battery lifetime of the wearable device is short. What is the power consumption of the wireless communication device (WiFi)?
This is a very good point that helps us to avoid a probable misunderstanding. With the term “low power” we meant “low with respect to” (or at least comparable) the solutions for FECG extraction that can be found in literature, not “absolutely low”. In the revised version of the manuscript we changed this reference modifying it in “reduced power” meaning that all the solutions implemented are optimized for power restraint as also Altera claims in its web site.
We think also that the Reviewer’s comment open to us the possibility of enriching the paper providing more information. Thus, we made an estimation of the autonomous system duration assuming it will be equipped with TR1865 lithium batteries: this will guarantee about 3 hours and 30 minutes of autonomy working at 95 MHz. This estimation has been done considering that the FPGA work at its maximum frequency (if the frequency is lower the power consumption will accordingly diminish), the acquisition module is equipped with a STR711 microprocessor supplied at 3.3 V and the WiFi hypothesized module is a Texas WL1807MOD with a 54 Mbps transmission rate. We consider this a good proposal although not definitive one and a good step toward the instrument total portability preserving its real time processing capability. We add all these considerations in the final section of the paper. We thank the Reviewer for this important and very “stimulant” comments.

- In section III about the state-of-the-art, the authors claim, that the DSP solution presented in [30] is complete and has a high power consumption of max 200 mA. Assuming a supply voltage of 3.3 V, the power consumption would be 0.66 W. I can't really see, how the 0.66 W can be high, when the authors themselves burn about 1 W in their FPGA solution. The paper needs more explanation on the power consumption and its comparison to the state-of-the-art.
The Reviewer’s comments are well taken. Concerning the DSP solution, in the paper on page 8 we say that “For what concerns the implementation carried out in [30] elaboration times are faster, but the adopted technique is not a BSS approach, so the separation quality is lower. Moreover, the power consumption is 1 W, since authors of [30] claimed a current absorption of 200 mA and the component is supplied with 5 V”. So the power consumption is equal (not higher) to our proposal for what concerns the processing unit while no other details are given on acquisition and WiFi transmission. Instead, the precision is lower since they do not use BSS and do not provide information about number of cases analysed and accuracy. For what concerns explanation on the power consumption we think that the previous answer could clarify the issue

- On page 8, the authors claim that the power consumption of 1 W is compatible with the constraints of wearable systems. What are the constraints of wearable systems - I guess these constraints are highly dependent on the battery size in use? Can the authors give a reference on these constraints?
This is another very good point and as we already said in another answer to avoid misunderstandings we replaced the term “wearable” with “portable” that is more adequate to the nature of our project. For what

concerns the batteries that can be used, power consumption and duration (that is one of the very critical constraints) we have already answered in the previous point. Thanks for helping us in making the concept of the device we want to propose more comprehensible to the reader.

- Since the resource usage on the FPGA is low, wouldn't it be desirable to take a smaller FPGA in order to reduce its power consumption?

The Reviewer is completely right, and we really considered other FPGA models with less resources so as to reduce the overall power consumption and encumbrance. In particular the Altera Cyclone V device was taken into account but it would not allow to implement all the foreseen functionalities related to morphological analysis and this would prevent future expansions of the design. In any case this would require a complete re-design of the actual implementation since portability is not assured.

4) Precision:

- The authors claim to use the wearable data acquisition device designed by the Polytechnic of Milan. A reference is missing in the paper on page 5. A description of the analog front-end would be necessary in order to understand the noise performance of the device. What is the relationship between noise of the analog front-end and the precision of the proposed algorithm?

The authors completely agree with the Reviewer's remark. A reference to an already present publication [23] has been moved on page 5, moreover further details concerning the ADC accuracy, the used microcontroller, the power consumption and the SNR were added while referring the reader to the article [23] for the complete description of the acquisition system to avoid an excessive paper extension.

- On page 6, the authors claim to use 32-bit precision for the calculations. First of all, what is the precision of the internal ADC in the data acquisition device? Since filters are used in the design and they need either too small or too large coefficients, isn't it more preferable to use an exponent for the number representation in the FPGA (a pseudo floating point representation)?

As we said in the previous answer we added several details concerning the acquisition system included the required ADC precision (16 bit).

Concerning the filtering the choice was suggested by the Matlab Fixed Point Toolbox we used in our simulations that demonstrated (in terms of Minimum Square Error) that the fixed point was equivalent to the floating point one in terms of results accuracy. Since the FPGA fixed point blocks are cheap in terms of encumbrance and power consumption we preferred this last choice.

- Dividers cost in terms of circuit complexity and power. The authors claim on page 6 to use more than one divider. I guess the matrix inversion needs division operations. Nowadays, Newton Raphson is used to approximate 1/b which is multiplied with a in order to perform the division a/b. The Newton Raphson method is less complex and highly dedicated for hardware implementation. Can the authors give an explanation, why complex dividers were used in the architecture instead of the Newton Raphson method? Later on page 6, it is claimed that the Newton's method is used only to calculate the inversion of equation (21).

This is a very outstanding comment and it is true everything depends on equation 21. In this equations the "b" parameter (divisor) can range over a wide set of values. When the divisor is small basically the calculation with NR converges with a small number of iterations (less than 10). When the divisor is big this is not assured and the requested number of iterations can be very high and in particular not fixed . This means potentially a very complex control logic. This brought us to use NR with small divisors and other methods with big divisors.

Last but not least, we would like to take this opportunity to gratefully thank the Reviewer again for his/her assessment of our manuscript and for his/her comments and suggestions for improvement of our work, which have greatly helped us to improve the technical quality and presentation of our manuscript.

Response to Comments by Reviewer 6

The mathematical approach is really interesting and has been tested on a wide database composed by both synthetic and real data. The most critical aspect from my point of view is the lack of indication about precision and recall of their algorithms. In the paper authors stated that the proposed system correctly separates and classified both the simulated and the real track, without false negatives, but without further details. This could be a lack for a critical analysis of the results, especially when there are different threshold that can affect the overall result. Some more detail about such results could be appreciated.

This is a very good point. In the revised version of the manuscript we specified that the relevant feature recognized by our implementation is the peak number on the elaborated channels. If these channels loose some peaks they are considered by the classification stage as “noisy” and discarded. On the other hand within 4 seconds of FECG acquisition 8 peaks are on average present: the implemented technique recognizes the peaks and classifies them as belonging to a fetal track. In this case the precision is 8/8 and also the recall as well. All these considerations were added to the paper and we hope that now is “accuracy” in terms of critical analysis f the results is acceptable. We sincerely thank the Reviewers for helping us to improve the paper from this point of view.

From the hardware point of view, the authors really stressed the problem of real-time analysis, whilst in such application, and typically in telemonitoring scenario is not so critical. Some second of delay from a possible abnormal FECG and its identification is not a problem, especially considering application, where hare often reported delays ten of seconds in remote transmission and further examination by the clinicians could be scheduled within minutes or hours, depending on the service. I could suggest for future improvement to consider find a better tradeoff between real-time constraints and power consumption, which is still too high for a portable (i.e. battery-powered) device. Although not so performing in terms of parallel-computing, Cortex® M3 and M4 MCU could be a possible solution to analyze.

We completely agree with the Reviewer whose comments are outstanding. The aim of the paper was to propose a design implementing a very reliable classification algorithm for FECG on programmable logic devices exploring consumption, processing rate, portability issues. The performed experimentation showed that this kind of technology could be valid for this kind of application although architectures like those proposed by the Reviewer could be compliant as well with the not trivial advantage of a very easy (compared to FPGAs) programmability. It is our intention to continue in this exploration and in the future developments we made an explicit reference to the ideas suggested by the Reviewer.

Last but not least, we would like to take this opportunity to gratefully thank the Reviewer again for his/her assessment of our manuscript and for his/her comments and suggestions for improvement of our work, which have greatly helped us to improve the technical quality and presentation of our manuscript.

Custom FPGA Processing for Real-Time Fetal ECG Extraction and Identification

E. Torti[1], D. Koliopoulos[2], M. Matraxia[3], G. Danese[1], F Leporati[1]
[1]University of Pavia, Dept. of Electrical, [2]Positech Consulting, [3]Johnson & Johnson
Corresponding author: Francesco Leporati, Computer and Biomedical Engineering, Via Ferrata 5, Pavia, Italy phone: (+39) 0382985678-373 mail: leporati@unipv.it

***Abstract*—Monitoring the fetal cardiac activity during pregnancy is of crucial importance for evaluating fetus health. However, there is a lack of automatic and reliable methods for Fetal ECG (FECG) monitoring that can perform this elaboration in real-time. In this paper, we present a hardware architecture, implemented on the Altera Stratix V FPGA, capable of separating the FECG from the maternal ECG and to correctly identify it. We evaluated our system using both synthetic and real tracks acquired from patients beyond the 20th pregnancy week.**
This work is part of a project aiming at developing a portable system for FECG continuous real-time monitoring. Its characteristics of reduced power consumption, real-time processing capability and reduced size make it suitable to be embedded in the overall system, that is the first proposed exploiting Blind Source Separation with this technology, as the best of our knowledge.

***Index Terms*—Embedded systems, fetal ECG, Field Programmable Gate Array (FPGA), Biomedical instrumentation**

I. Introduction

The monitoring of the fetal cardiac activity using Doppler methods is a standard examination used, together with sonogram, for evaluating fetus health.
Fetal monitoring is often performed in medical centers, and the analysis of the cardiac tracks is simply performed by eye inspecting the recordings. This means that the recognition relies on the experience of the doctors, with a possible low level of reliability [1]. These considerations lead to the need of a portable system that can autonomously extract and identify the FECG in real-time, eventually storing it in a local memory or sending to a dock-unit through a Wi-Fi connection. This allows a continuous monitoring, not always possible with the traditional systems. Moreover, recent advances in the field of "textile wearable devices" [2] [3] [4] [5] make possible to develop a wearable unit that can acquire and examine the signals and send them to a remote diagnostic center.
The acquired ECG signal is the result of the superimposition of electrical activities corresponding to maternal and fetal hearts. In addition, there are noisy contributions due to electrodes, maternal breath and involuntary movements [6].
Noise can be reduced using a suitable filtering stage, so the system must be able to separate the FECG from the maternal one. Separation quality is improved if the acquisition involves multiple ECG channels (not less than 4).
FECG extraction has been widely investigated in the last years and many approaches emerged, exploiting wavelet transformations [7] [8], typical artificial intelligence techniques [9] [10] [11] and purposely devoted algorithms [12] [13] [14].
Among these solutions, we performed a careful analysis, after which Blind Source Separation (BSS) algorithms such as Infomax [15] and JADE [16] emerged as the best techniques, both in terms of reliability and in terms of documentation for implementing the same steps on a dedicated hardware circuit like the one we want to carry out.
These two methods produce a set of signals that need to be classified for FECG recognition. The JADE algorithm produces a set of outputs where portion of the fetal signal can be subdivided in different output signals and should then be recomposed. For what concerns Infomax, instead, this technique does not suffer from this problem, since the fetal signal is fully contained in at least one of the output channels.
It must be noticed that, in both cases, the output channels containing cardiac signals preserve the morphological characteristics, such as QRS peaks and P and T waves. We chose to design an architecture based on Infomax, since we want to identify only one channel containing the fetal track.
In particular, we chose a nonlinear Infomax network, which produces sets of filters that are usually applied to data streams containing mixed information sources with the aim of separating them [15]. Those filters produce outputs that are as independent as possible, so it is possible to say that the chosen network performs an Independent Component Analysis (ICA).
Moreover, the literature shows that Infomax outperforms, in terms of extraction reliability, similar approaches, such as Fast-ICA, Pearson-ICA and Sequential Analysis.
After FECG extraction, we need to identify which channels correspond to mother (maternal signal), to fetus signal, or to noise; also this step should be done automatically.
This issue has been addressed in different ways, such as nonlinear transforms [17], genetic algorithms [18], wavelet transforms [19] and filter banks [20].
We identify suitable filter banks for highlighting typical features of the cardiac signal [20][28] introducing some useful modifications. Those filters are applied to each output channel produced by Infomax. The resulting channels can be separated in two groups: noisy channels and cardiac channels. The first ones are discarded while the others are classified through a suitable clustering algorithm described further in the paper.
We choose to use a high-end FPGA device because we need high performance in terms of elaboration speed but also limited power consumption. We are exploring this technology

as an alternative to the literature, which is typically based on Digital Signal Processors (DSP).

The proposed system has been validated using both synthetic and real tracks, outperforming existing monitoring systems in terms of elaboration speed and detection accuracy. Moreover, our system is real-time compliant and has a reduced power consumption, enabling its use for the development of a portable monitoring system.

This project involves a collaboration among the dept. of Electrical, Computer and Biomedical Engineering at the University of Pavia, the Bioengineering dept. of the Polytechnic of Milan and the Polyclinic research unit of the University of Naples and has been funded by the Italian government within the National Interest Research Projects Program. At the best of our knowledge, this is the first portable device exploiting BSS on the FPGA technology for FECG monitoring.

The remainder of the paper is organized as follows: section II describes the fetal ECG signal and its main characteristics. Section III presents the state of the art of automatic FECG extraction and identification. Section IV and V respectively describe the Infomax network and the classification algorithm adopted in our work. Section VI presents the FPGA design and section VII describes experiments conducted for validating the architecture. Section VIII concludes the paper with some remarks and comparisons with the state of the art.

II. THE FETAL ELECTROCARDIOGRAM

The ECG is a track representing the heart's electrical activity. A typical ECG record shows a cardiac cycle which is made up of three parts: the P wave (related to the atrial depolarization), the QRS complex (related to ventricles depolarization) and the T wave (related to new ventricle polarization).

The heart rate is an important parameter, which is defined as the time between two consecutive R peaks; the inverse of the heart rate is called cardiac frequency. This frequency, in an adult, is typically in the range [60-100] beats per minute (bpm), while the fetal one is in the range [110-150] bpm. For what concerns fetal cardiac frequency, fast modifications of this frequency are considered normal, but their absence or a frequency out of the previous mentioned range are considered indicators of possible anomalies. Moreover, FECG is correlated with the maternal body response. For example, accelerations of the cardiac frequency with respect to uterine contractions (UC) rates is an indicator of non-correct abdominal venous circulation [21].

Nowadays, fetal cardiac monitoring is mainly conducted in an invasive form and could happen only during labor. In this case an electrode is placed on the fetus head, but it is an extremely delicate procedure. For what concerns noninvasive techniques, the most common exploit ultrasound Doppler acquisition and fetoscope, but, unfortunately, they do not allow a continuous monitoring [22].

The ECG signal can also be acquired using surface electrodes positioned on the mother's abdomen. As said before, the electrodes acquire, together with the FECG, the maternal ECG and noise. An example of those mixed signals is depicted in figure 1.

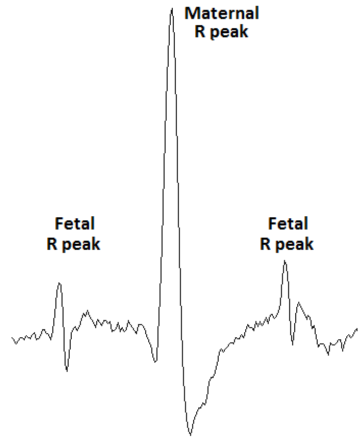

Figure 1 - Superimposition of FECG, maternal ECG and noise.

III. STATE OF THE ART

Fetal cardiac signal separation from mother's one has been widely investigated and, as already highlighted, the best techniques are based on BSS. However, different techniques among the ones previously mentioned have been evaluated, in particular, the most recent and performing proposals are:

- the paper of Fanelli et al. [23] implementing the Marten's algorithm (a PCA enhancement of the QRS complex with averaging and filtering);
- the two papers of Karvounis et al. [7][24] featuring a three stage mixed approach (parabolic fitting, spatial and time frequency multivariate analysis and wavelets). They try to identify maternal peaks and eliminate mother QRS complexes, while using histogram based techniques for finding fetal R peaks);
- the paper of Ming et al. [8] based on a two stage time frequency BSS and wavelet noise filtering.

All these proposals feature very interesting accuracies but are not real-time nor provide indications about the feasibility of a real-time implementation. On the other hand, Hasan et al [25] [26], Pani et al. [27] [28], Xuan-Ang et al. [29], and Arias Ortega et al. [30] proposed implementations into which online elaboration seems possible. The most interesting one is [25] into which a neural network based FECG extraction is implemented on a Stratix II FPGA but no results about elaboration time and working frequency (and then about real-time feasibility) are provided. The project presented in [26] on the other hand presents similarities with our solution, but seems still in its preliminary phase and the paper is focused only on the acquisition and networking implementation: in any case it does not feature portability/wearability nor local elaboration capability since the signal is acquired through a microcontroller while elaboration and extraction seems to be performed by a host computer.

Pani [27] [28] proposed a 300 MHz DSP implementation of JADE taking about 140 Mcycles/block for elaboration while Xuan-Ang et al. [29] features a similar DSP implementation of PCA but without giving accuracy and elaboration times.

The most complete work is the one by Arias Ortega et al. [30]

who implemented a LMS adaptive filtering on a dsPIC30F6014A in 550 μsec with 93.1% accuracy and 87% of sensitivity. However, the power consumption is high (average 90 mA, max 200 mA) and the adopted algorithm is less accurate than BSS.

IV. The Infomax algorithm

Infomax is an ICA-based algorithm introduced by J. Bell and T. J. Sejnowski in 1995 [15] for audio signal elaboration. However, it has been successfully used also for ECG signal elaboration and FECG extraction [31].

The problem is described by the equation:

$$x(t) = As(t) + n(t) \tag{1}$$

where $x(t)$ is the acquired signal, $s(t)$ represents the different sources which must be weighted with the values of the matrix A and $n(t)$ is the noise. In particular, A is a *M*x*N* matrix where *M* is the dimension of $x(t)$ and *N* is the dimension of $s(t)$. If we assume that the sources are statistically independent it is possible to separate them.

The Infomax algorithm estimates the matrix W which is defined as the inverse of the mixing matrix A.

In our case, the $x(t)$ measurements are the acquired abdominal ECG signals. If the signal to noise ratio is high enough, the noise $n(t)$ can be neglected, so it is possible to estimate the independent signal as:

$$\hat{s}(t) = Wx(t) = PSx(t) \tag{2}$$

where the matrix W is factorized by the permutation matrix P and the scaling matrix S. It is possible to choose those matrices so that $PS = I$, where I is the identity matrix. The order of the estimated sources is not a-priori known. Moreover, the amplitude information is lost, but this is not a critical issue, since the signal shape is sufficient for evaluating FECG cardiac frequency.

Infomax estimates the W matrix using iterative approximations, trying each time to minimize the mutual information between distinct sources:

$$I(\hat{s}_i, \hat{s}_j) = 0 \tag{3}$$

It is possible to minimize $I(\hat{s}_i, \hat{s}_j)$ by elaborating the joint entropy $H(\hat{s}_i, \hat{s}_j)$, and to do so we can consider the mutual information between system outputs, which is defined as:

$$I(\hat{s}, x) = H(\hat{s}) - H(\hat{s} \mid x) \tag{4}$$

where $H(\hat{s})$ is the joint entropy of the outputs and $H(\hat{s} \mid x)$ is the conditional entropy, mainly due to noise. For minimizing $H(\hat{s})$ and, consequently, $I(\hat{s}, x)$, it is possible to differentiate the equation (4) with respect to a characteristic parameter of the network, that we will indicate with w. If we assume that the relation between inputs and outputs can be well represented by logistic function:

$$\hat{s} = (1 + e^{-(w_x + w_0)})^{-1} \tag{5}$$

it is possible to define a learning rule to update weights and make an iterative estimation of the matrix W:

$$\Delta w \propto w^{-1} + x(1 - 2\hat{s}) \tag{6}$$

In the vector space equations (6) becomes:

$$\Delta W \propto [W^T]^{-1} + (1 - 2\hat{s})x^T \tag{7}$$

At the first iteration, W is set to the value $W_0 = 1$; the iterative process stops when the square of the difference between two consecutive approximations of W is sufficiently low. This threshold is a critical issue, since a big value involves a low quality separation, while a small threshold involves a high number of iterations, which are not compatible with a real-time implementation.

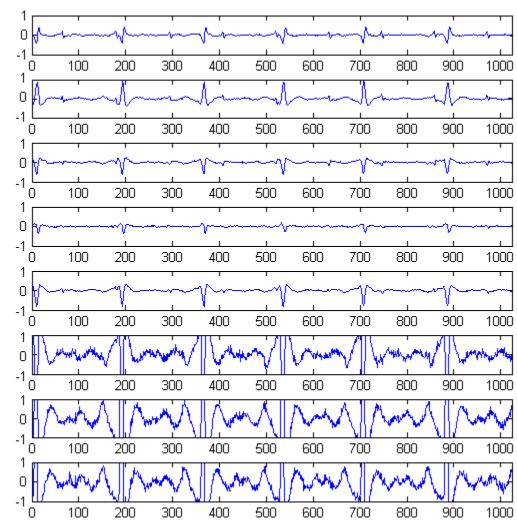

Figure 2 - Input channels. On the x axis there is the sample number, while on the y axis there is the amplitude, measured in mV.

Moreover, it is possible to improve performance by pre-processing the discrete input signals $x(k)$ in order to make them uncorrelated. To this purpose, we use a de-correlation matrix W_z:

$$W_z = 2(\sqrt{Var[x(k)]})^{-1} \tag{8}$$

Figure 2 shows a dataset of cardiac tracks proposed by De Moor et al. [35], which is largely used for validating FECG extraction algorithms, while figure 3 shows how those tracks have been separated by Infomax.

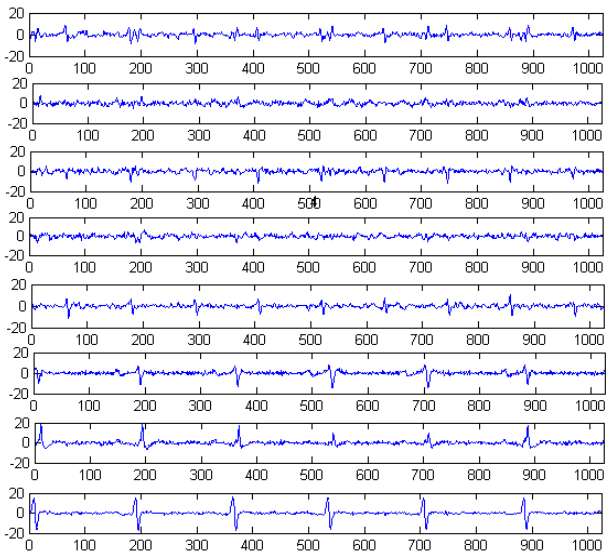

Figure 3 - Output channels produced by Infomax. On the x axis there is the sample order number, while on the y axis there is the amplitude, measured in arbitrary units. The fetal signal is in channel five.

V. THE CLASSIFICATION ALGORITHM

The output channels produced by Infomax must be classified for clearly pointing out which one contains the fetal track. Our technique is based on a preliminary filter that affects all the Infomax output channels for highlighting the peaks, typical in the QRS complex of cardiac signals. After that, a phase of noise channels identification takes place. This stage is required to send to the successive clustering phase those channels that contain cardiac activity only from the fetus and from the mother.

The block diagram of this algorithm is depicted in figure 4.

In our case, N is equal to 8 and M is the number of noise channels and depends on the output signals produced by Infomax.

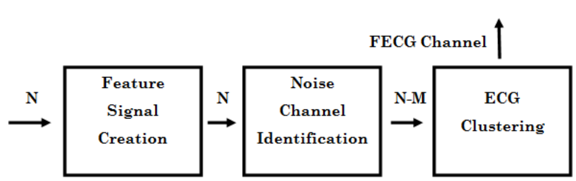

Figure 4. Block diagram of the classification algorithm.

A. Feature signals creation

This phase exploits a series of FIR filters for highlighting the peaks related to the QRS complex.

Standard filters for QRS detection have been widely exploited [17] [32] [33].

We decided, according to [34], to use:

- a first order derivative filter;
- a second order derivative filter;
- a Multiplication Of Background Difference (MOBD) filter;
- a weighted moving average filter.

The first order derivative filter is a high-pass filter and is used for highlighting the fast amplitude variations, which typically occur in QRS events.

Several implementations of this filter are provided in literature such as [32]:

$$y[n] = x[n+1] - x[n-1] \tag{9}$$

$$y[n] = 2x[n+2] + x[n+1] \tag{10}$$

$$y[n] = x[n] - x[n-1] \tag{11}$$

where x denotes the input sample vector.

We chose to use a different first order derivative filter, based on absolute values, to avoid negative values:

$$y[n] = \frac{1}{2}\left(| x[n+1] - x[n-1] | \right) \tag{12}$$

For what concerns the second order derivative filter, it is implemented as:

$$y_2[n] = \frac{1}{4}(x[n+2] - 2x[n] + x[n-2]) \tag{13}$$

which is further combined with the aforementioned first order derivative filter after a weighted moving average and a MOBD filter:

$$u[n] = y_1[n] + \frac{1}{2} z[n] \tag{14}$$

where y_1 is the output of the first order derivative filter, and z is the result of the second order derivative filter, the moving average filter and the MOBD filter.

We implemented a MOBD filter given by:

$$z[n] = y_3[n] * y_3[n-1] \tag{15}$$

where y_3 is the output of the second order derivative filter and the moving average filter.

The adopted filtering schema is shown in figure 5.

Figure 6 shows the feature signals obtained by the output channels depicted in figure 3.

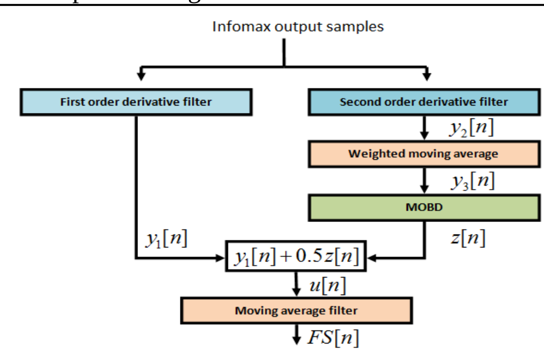

Figure 5 - Adopted filtering schema

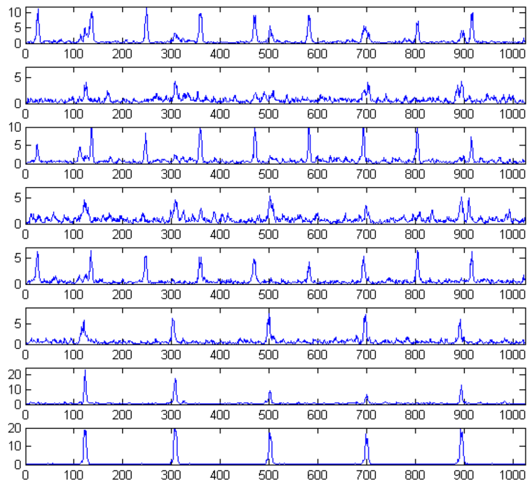

Figure 6 - Feature signals. On the x axis there is the sample number, on the y axis there is the amplitude, measured in arbitrary units.

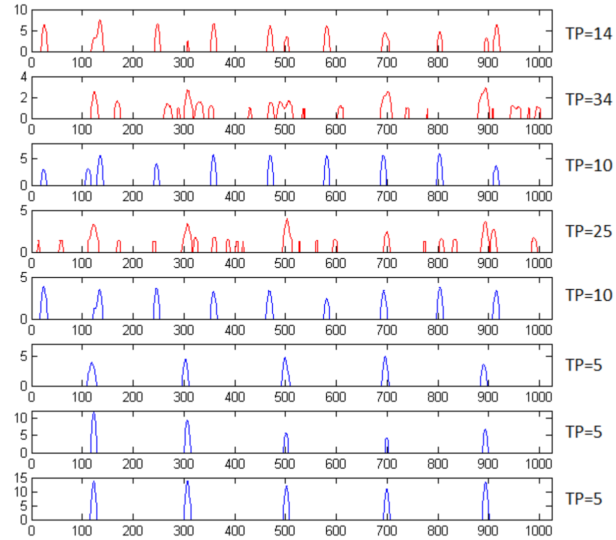

Figure 7 - Noise channel identification outputs. On the x axis there is the sample number, while on the y axis there is the amplitude, measured in arbitrary units.

B. Noise channels identification

The feature signals obtained using the filtering stage are the inputs of the noise channel identification stage. This phase is needed for sending a limited number of channels to the clustering stage, accelerating the classification phase, to avoid working on signals with no clinical interest.

For each channel, a suitable threshold is computed as function of the maximum amplitude, to exclude uninteresting signals. This is done to remove the baseline noise.

In our implementation, this threshold has been set to 30% of the maximum value, so we can remove the baseline noise without removing plausible QRS peaks.

After that another first order derivative filter, like the one described in the previous section, is applied, for highlighting the transactions from positive to negative derivative of each channel.

Those transactions are defined as the *turning points* (TP), related to the number of QRS events of each signal.

The number of the detected turning points is used to evaluate if a channel should be sent to the clustering stage or not.

Our architecture must work with ECG records of about 4 seconds, as they come from the wearable acquisition system designed by the Polytechnic of Milan [23]. This system has been developed using a STR711RF2 microcontroller, which is equipped with a 16-bit A/D converter. The number of turning points can be used for calculating the cardiac frequency. We choose to send to the clustering phase all those channels that contain a number of turning points belonging to the range [43-190] bpm, so those channels may contain mother and fetal signals.

We also generalized this threshold for working with different record lengths, implementing the empirical formula:

$$3 * \frac{nums}{1024} \leq TP \leq 13 * \frac{nums}{1024} \qquad (16)$$

where $nums$ is the number of samples to be processed and 1024 is the number of samples processed by default in our implementation (250 Hz sampling frequency with a 1 KB buffer). In this phase, beyond the number of turning points, the average distance between two adjacent turning points and the standard deviation of the distances is elaborated. Those three values are needed by the clustering phase.

Figure 7 shows the output of the noise channel identification phase, assuming as input the signals shown in figure 6. The red ones are the channel labelled as noise, while the blue ones are the signals used as input for the clustering stage.

C. Clustering

The authors of [34] proposed an agglomerative clustering based on the Pearson correlation index for FECG identification. However, we chose to test different clustering techniques in order to improve the performance.

Instead of using the Pearson correlation index for evaluating the distance between two clusters, we used a single linking cluster working on the number of turning points.

We also adopted a K-means clustering method, testing two different ways of partitioning data. The first one uses only the number of turning points as parameter to perform the partitioning, while the second one uses also the standard deviation between all the turning points distance and the average between the distances. Those values are computed by the previous noise channel identification stage.

We tested four clustering algorithms (agglomerative clustering from [31], single linking clustering and the two K-means clustering described before) with our dataset and we found that the best accuracy is obtained using the K-means method based only on the number of the turning points (accuracy of 100%). The agglomerative clustering proposed in [31] shows an accuracy of 80%, while the single linking clustering an accuracy of 95% and, finally, the K-means based on two parameters shows an accuracy of 60%.

After those experiments, we chose to work with the K-means clustering based only on turning point numbers. This algorithm automatically classifies data into two distinct groups, creating a fetal cluster and a maternal cluster using as information only the number of turning points calculated by the previous phase.

K-means clustering results depend on the initialization of the algorithm and it is not guaranteed that the obtained result is the optimal one, but it can be a sub-optimal solution. This is due to a non linear objective function that is minimized in the clustering stage, that ends when a minimum is reached. However it is not assured that it is a global minimum.

To avoid this problem we chose to run five different instances of the K-means clustering algorithm with different initializations. When all the five algorithms ended, the values of the objective functions are compared and the minimum one is chosen as optimal one. The corresponding partition is selected as output of the clustering stage.

In the K-means algorithm, the number of groups to be produced as output must be set. We set this value to two, because we foresee to have only maternal and fetal signals as input of this stage. Experiments have pointed out that some noise channels can pass the previous filtering stages and enter in the clustering stage. Those channels are typically classified in the fetal cluster due to the number of their turning points. To avoid this problem, when in the fetal cluster there is more than one signal, the one is selected whose distances between the turning points feature the lower standard deviation. This choice has been made because a cardiac signal is always more regular than noise.

Moreover, our experiments showed that the K-means algorithm performs better than agglomerative clustering proposed in [34] (see section VII).

VI. System Implementation on FPGA

The whole FECG extraction and identification system has been implemented on an Altera Stratix V 5SGXEA7N2F45C2 FPGA. We chose this device because Stratix is the high-end FPGA developed by Altera.

Since floating point calculations are expensive in terms of resource usage and execution time, we used a fixed point data representation. Each sample is represented with a 32 bit word, where the 12 most significant bits are used for storing the integer part, while the remaining 20 are used for the decimal one.

First of all, we developed a fixed point library for performing addition, subtraction, multiplication and division. The blocks have been connected using the pipeline philosophy, to maximize the working frequency of the architecture.

The division block requires a latency and a number of resources greater than the others, so we used it only when strictly required. For all those operations where the divisor is a constant value, we obtain the result using a series of shifts and additions, with a consequent resource saving.

For what concerns Infomax, the architecture can be divided into three main parts. The first one computes the product:

$$U = W_{old}x \qquad (17)$$

where W_{old} is the matrix estimated at the previous iteration. After that we compute the learning rule, according to (7), given by:

$$\Delta W_k = L\left(BI + \left(1 - 2\left(\frac{1}{1+e^{-U}}\right)\right)U^T\right)W_k \qquad (18)$$

where L is a suitable constant, B is the number of samples acquired for each channel and k is the considered iteration. In our case, B is equal to 1024; this value is related to the records length (about 4 seconds) and the sampling frequency (250 Hz). Equation (18) is used for computing the new separation matrix:

$$W_{k+1} = W_k + \Delta W_k \qquad (19)$$

The third stage of the Infomax architecture computes the termination criteria given by:

$$change = \sum_{i=1}^{N}\sum_{j=1}^{M}\left(W_k^{i,j} - W_{k-1}^{i,j}\right)^2 \qquad (20)$$

where the threshold is chosen for stopping the iterative process when the mean squared error is less than 2.5%. The most complex stage of the architecture is the computation of equation (18), which is made up of four main blocks. The first one computes the C matrix, given by:

$$C = 1 - 2\left(\frac{1}{1+e^{-u}}\right) \qquad (21)$$

where the exponential function is implemented using the corresponding Taylor series truncated at the fifth term. Moreover, the inversion is computed using Newton's method, which is an iterative method. We experimentally found that seven iterations are sufficient to reach a suitable precision for our architecture. After that, the architecture computes the D matrix, given by:

$$D = L(BI + Cu^T) \qquad (22)$$

The next two steps are a multiplication and a sum for computing the new separation matrix. It is important to emphasize that all matrix multiplications have been performed in parallel for achieving better performance. While the new separation matrix is computed, we also calculate equation (20) to establish if the algorithm converges. If this criteria is not fulfilled, the control logic saves the new separation matrix into a suitable RAM and, subsequently, starts another iteration of Infomax, otherwise the separated samples are computed and saved in suitable FIFO memories, which are the input of the identification and classification phases.

The schema of this elaboration chain is depicted in figure 8.

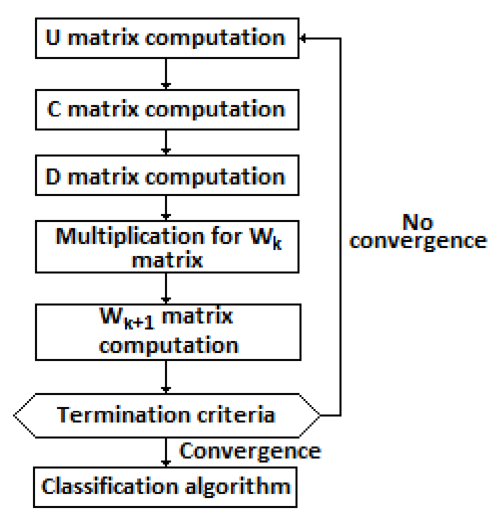

Figure 8 - Schema of the Infomax implementation

The first operation of the classification algorithm is *feature signals creation*. As shown by Figure 5, the first order derivative filter and the series of second order derivative filters, the moving average filter and the MOBD filter can be performed in parallel.

We instantiated eight different blocks for feature signals creation that are able to elaborate in parallel all the Infomax output channels.
The second phase, the *noise channel identification*, requires a buffer for storing the samples during the threshold evaluation. This buffer is implemented through a simple 1 KB FIFO memory. When the input samples are produced by the previous phase, the maximum value is calculated while storing the data inside the FIFO. After the maximum value has been computed, the threshold is calculated and the FIFO is read. The output sample is set to zero if the read value is below the threshold otherwise it is kept as it has been acquired.
While transferring the correct output, this phase also calculates the number of turning points (i.e. the average of the distance). The standard deviation is not directly computed, because it requires to perform a square root operation, which is heavy in terms of FPGA usage. For this reason, we chose to compute the cheaper variance.
For each feature signal, one of those blocks is created. Those eight blocks are connected to another block used for establishing if each feature signal is noise or not.
After that, the selected turning points numbers of the feature signals, the average of distance and the standard deviations are stored in separated FIFOs.
Those data are the inputs of K-means block.
As before mentioned, K-means requires an initialization, that must be a random value for the best performance. We implemented a Galois linear feedback shift register [36], for producing a pseudorandom 32 bit value.
This circuit also needs a seed, which can be given using an input pin, so at each execution the initialization can be changed.
We created five K-means blocks, that work in parallel on the same data; each K-mean block has a different initialization value. Each one computes the *objective* function given by:

$$SSE = \sum_{i=1}^{2} \sum_{x \in Cl_i} \left(m_i - x\right)^2 \qquad (23)$$

where x is an element of the cluster Cl_i and m_i is the average value of the same cluster.
When all the K-means blocks end their elaboration, all the SSEs are processed and the results related to the minimum are selected as output.
If the selected cluster contains more than a channel, the output is chosen by comparing the average distance of the turning points and the standard deviation of the distance as described in the previous section.
The framework of the implementation is shown in figure 9. Please note that each channel is elaborated in parallel for what concerns the feature signal creation and the turning points count. Also the *noise identification* phase is performed on each channel in parallel, but at the end of this block there is a synchronization logic that waits the end of operations for all the channels. This is done because the clustering phase needs to work on the full dataset, which is ready after the end of the noise identification on all the channels. The output of this phase is the identifier of the channel containing the fetal signal (indicated as fetal channel in figure 9).

VII. Experimental Results and Discussion

First of all, we tested the proposed elaboration chain using the well-known Physionet database [37]. The tests show that our algorithm is capable of correctly separate and identify the fetal signal.
We tested the proposed architecture both with synthetic data and with real data. The first dataset is the well-known one by De Moor et al. [35], while the latter is a dataset of 343 tracks acquired from pregnant volunteers at the Polytechnic of Milan and at the Naples hospital “Vincenzo Cardarelli”. The volunteers were informed on the nature of the research and about the use of the acquired data and expressed their consent to the purpose of their use. Those real tracks have been acquired using the wearable prototype described in [38]. We stored the input samples, both the synthetic and the real, on a suitable on-chip RAM.
The proposed system correctly separates and classifies both the simulated and the real tracks.
After separation of each track into eight channels, there is always at least one channel containing the fetal signal. In all the considered cases the fetal track is correctly distinguished by the architecture, so it is possible to say that, for the considered dataset, there are no false negatives.

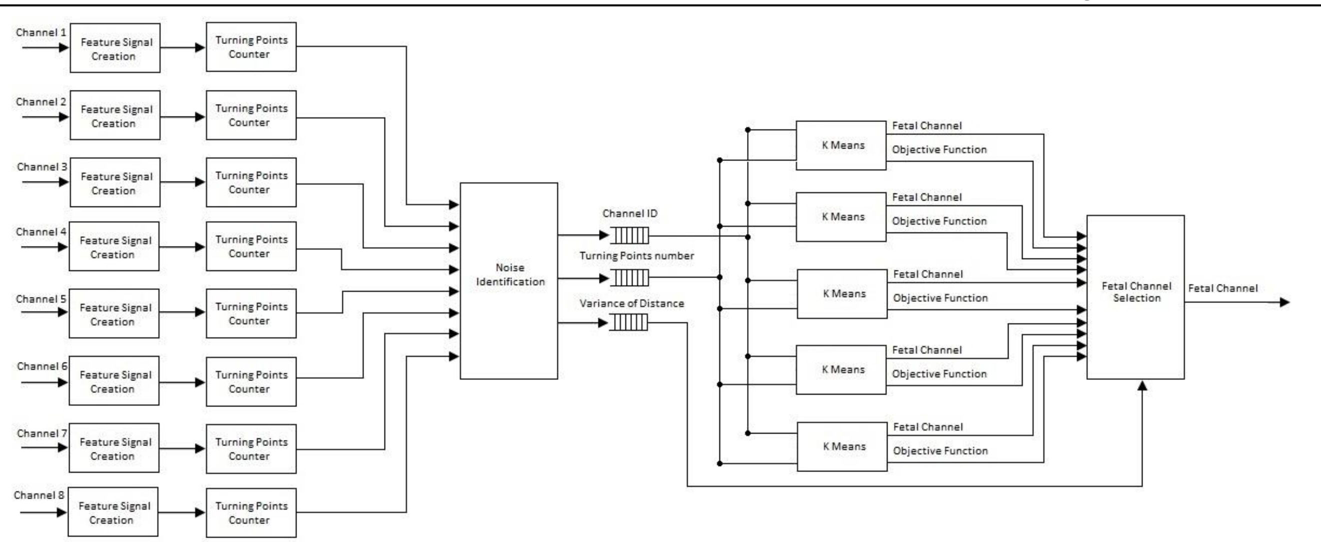

Figure 9 - Schema of the classification algorithm implementation

TABLE I – Comparisons between the proposed work and the state of the art

	Device	Technique	Working frequency [MHz]	Time [ms]	Power consumption [W]
Ortega et al. [30]	dsPIC30F6014A	LMS (No BSS)	30	0.55	0.45-1
Pani et al. [34]	OMAP L-137 DSP	OL-JADE	300	120	1
Hasan et al. [25]	FPGA	Neural Network	N. A.	N. A.	N. A.
Morales et al. [40]	FPGA	Adaptive filtering (NO BSS)	50	600	N. A.
Proposed work	FPGA	Infomax + K_Means	95	1.8-27	1
Proposed work	FPGA	Infomax + K_Means	50	3.4-54	0.55

Moreover, there are also no false positives, since the clustering stage works considering two attributes: number of turning points and variance of distance between two consecutive turning points. Theoretically, a false positive, is a noisy channel classified as fetal signal. For being classified as fetal, a noisy channel must exhibit a correct number of turning points with regular distances. This is an improbable situation and is not present in our datasets. Finally, if a fetal channel reaches the clustering phase, it is always correctly classified, so the recall is 100%.

We have also estimated the 95% confidence interval using a binomial test (*binofit* Matlab function), obtaining as result the interval (98.9%-100%).

It should be emphasized, moreover, that a careful filtering has to be done to obtain accurate results. In our case, we filtered the signals using an high pass FIR with a cut-off frequency of 5 Hz, for removing slow signals such as movements and breath.

We implemented this system on an Altera Stratix V 5SGXEA7N2F45C2 FPGA, equipped with 622k logic elements (LEs), 512 18x18 bit multipliers, 256 27x27 bit multipliers, 6.25 MB of memory and 1064 pins.

The resource usage is less than 15% for what concerns Logic Elements (29780 LEs), less than 3% of total memory (1668714 blocks) and about 44% of embedded multipliers (122 DSPs). The working frequency is 95 MHz. Power consumption has been estimated using Altera Quartus II Power Play Analyzer and is less than 1 W, so it is compatible with the constraints given by a portable system.

The elaboration time depends on the number of Infomax iterations, on the number of turning points and on the iteration needed on the clustering phase.

We tested the architecture using Altera DSP builder, which gives a suitable interface between the targeted FPGA and the PC. The results provided by the FPGA are the same shown in figures 3, 6 and 7. In general, the results provided by the FPGA are the same of the software simulation.

De Moor signals required only 2 Infomax iterations and a single clustering iteration, while, real signals required 30 Infomax iterations and 5 clustering iterations in the worst case. At the working frequency of 95 MHz, this corresponds to an elaboration time from 1.8 ms to 27 ms. The real-time constrain is satisfied, since the used tracks are of about 4 s.

The work proposed in [34], implementing OL-JADE on an OMAP L-137 DSP took about 120 ms to separate sources, so our implementation is faster than this one. The power consumption of the solution described in [34] is about 1 W, which is comparable with our power consumption. We also performed tests halving the working frequency of our architecture; the power consumption decreases to the value of about 550 mW, with a maximum elaboration time of about 54 ms, values that outperform the solution proposed in [34] while remaining still real-time compliant.

For what concerns the implementation carried out in [30] elaboration times are faster, but the adopted technique is not a BSS approach, so the separation quality is lower. Moreover, the power consumption is 1 W, since authors of [30] claimed a current absorption of 200 mA and the component is supplied with 5 V.

FPGA has been exploited in [11] but authors give no results about elaboration time and working frequency, so it is not possible to carry out a comparison. Another work based on FPGA is [39], which is real time compliant, but validation have been conducted only using synthetic tracks. Moreover, the adopted technique is based only on filtering and maternal ECG elimination, which is less reliable than modern BSS algorithms.

Finally, FPGA has been also exploited in [40], but the adopted technique is based on adaptive filtering, so it does not provide a good separation since it is not a BSS method. Elaboration time is about 600 ms working at 50 MHz, so the system is slower than our proposal. About power consumption, it is not possible to make comparisons since authors of [40] did not provide this information.

Table I gives a summary of these comparisons.

VIII. Conclusions

In this paper, we present a novel architecture for FECG extraction and identification. A suitable dataset, made up of both synthetic and in vivo signals, has been correctly classified by the system. Estimated power consumption is compatible with the constraints given by a portable device, overall target of this project.

The proposed architecture outperforms the elaboration times of the other works in literature who implement similar algorithms for successfully separating the tracks.

Moreover, the resource usage is compliant with the implementation of future algorithms on the same FPGA. For example, it is possible to improve the system by adding diagnostic functions, such as morphological analysis of the fetal track, which can also be performed in real-time. Another possibility is the development of a module for storing the separated tracks on an SD/SDHC card, making possible an off-line analysis of daily-acquired data.

It is also possible to integrate a Wi-Fi module for sending the fetal tracks to a dock-unit, which is responsible of data storage and eventually transmit them to a medical center. In particular, the Texas Instruments' WL1807MOD Wi-Fi module has a power consumption during transmission of 2.8 W, assuming a bit rate of 54 Mbps. In our application, this bit rate is too high, so it is reasonable to adopt a slower transmission, with the aim of reducing power consumption.

If we assume to power the whole system (the acquisition board, our FPGA and the Wi-Fi module) using a single TR1865 lithium battery, this will guarantee about 3 hours and 30 minutes of autonomy working at 95 MHz.

At last, it is possible to add a cryptography function for protecting data before transmission, since those data are strictly personal.

Another possibility is to use a smaller FPGA, such as an Altera Cyclone V, which is equipped with less logic resources. This device has a lower power consumption than the one considered by us. This choice will further reduce the power consumption of our system. However, this device is suitable for housing only our system, without the possibility to expand its functions with the features described above.

Moreover, we will explore solutions based on ARM Cortex M3 and M4 MCU which are less performant devices than our FPGA, but will hopefully guarantee a lower power consumption.

IX. Conflicts of Interest Disclosure

The authors (who participated in the research and in the article preparation jointly and on an equal basis) hereby certify that no personal or financial relationships exist with people or any other organization that could inappropriately influence the work.

X. Acknowledgements

The authors gratefully thank prof. Andrea Fanelli, prof. Danilo Pani and prof. Giovanni Magenes for their useful advices.

This work was funded by the Italian Ministry for the Health (PRIN 2009).

References

[1] M. Ferrario, M. G. Signorini, G. Magenes, "*New indexes from Fetal Heart Rate Analysis for the identification of severe intra uterine growth restricted fetuses*", Engineering in Medicine and Biology Society, 2006. EMBS '06, 28th Annual Conference of the IEEE, 30 August-3 September 2006, pp. 1458-1461.

[2] M. Di Rienzo, P. Meriggi, F. Rizzo, E. Vaini, A. Faini, G. Merati, G. Parati, P. Castiglioni, "*A wearable system for the seismocardiogram assessment in daily life conditions*", Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE, 30 August-3 September 2011, pp. 4263-4266.

[3] P. K. Jain, A. K. Tiwari "*Heart monitoring systems – A review*", Comp. in Biology and Medicine, vol. 54, no. 1, August 2014, pp.1-13.

[4] H. Han, J. Kim, "*Artifacts in wearable photoplethysmographs during daily life motions and their reduction with least mean square based active noise concellation method*", Comp. in Biology and Medicine, vol. 42, no. 4, April 2012, pp.387-393.

[5] D. Bansal, M. Khan, A. K. Salhan "*A computer based wireless system for online acquisition, monitoring and processing of ECG waveforms*", Comp. in Biology and Medicine, vol. 34, no. 4, April 2009, pp. 361-367.

[6] S. M. M. Martens, C. Rabotti, M. Mischi, R. J. Sluijter, "*A robust fetal ECG detection method for abdominal recordings*", Physiological Measurement, vol. 28, no. 4, March 2007, pp. 373-388.

[7] E. C. Karvounis, M. G. Tsipouras, D. I. Fotiadis, K. K. Naka "*An Automated Methodology for Fetal Heart Rate Extraction From the Abdominal Electrocardiogram*", IEEE Transactions on Information Technology in Biomedicine, vol. 11, no. 6, November 2007, pp. 628–638.

[8] M. Ming, W. Ning, L. San-Ya, "*Extraction of FECG Based on Time Frequency Blind Source Separation and Wavelet Denoising*", Bioinformatics and Biomedical Engineering, ICBBE 2009, 3rd International Conference on, Beijing, 11-13 June 2009, pp. 1-3.

[9] C. Kezi Selva Vijila, P. Kanagasabapathy, S. Johnson, "*Adaptive Neuro Fuzzy Inference System for Extraction of fECG*", IEEE Indicon 2005 Conference, 11-13 December 2015, pp. 224–227.

[10] Y. Zeng, S. Liu, J. Zhang, "*Extraction of Fetal ECG Signal via Adaptive Noise Cancellation Approach*", Bioinformatics and Biomedical Engineering, ICBBE 2008. The 2nd International Conference on, 11-16 May 2008, pp. 2270 - 2273.

[11] M. A. Hasan, M. B. I. Reaz, M. I. Ibrahimy, "*Fetal electrocardiogram extraction and R-peak detection for fetal heart rate monitoring using artificial neural network and correlation*", Neural Networks (IJCNN), The 2011 Conference on, 31 July-5 August 2011, pp. 15-20.

[12] Z. Shi, C. Zhang, "*Blind Source Extraction Using Generalized Autocorrelations*", IEEE Transactions on Neural Networks, vol. 18, no. 5, September 2007, pp. 1516–1524.

[13] Y. Yang, C. Guo, "*Blind Extraction for Noisy Fetal Electrocardiogram by Using Gaussian Moments*", 4th International Conference on Natural Computation, 18-20 October 2008, pp. 44–47.

[14] Y. Ye, J. Zeng, K. Lu, "*An Extraction Algorithm Based on Second-order Statistics for Noisy Mixtures*", Fuzzy Systems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conference on, 18-20 October 2008, pp. 281-285.

[15] J. Bell, T. J. Sejnowski, "*An information maximization approach to blind separation and blind deconvolution*", Neural Computation, vol. 7, no. 6, November 1995, pp. 1129–1159.

[16] J. F. Cardoso, A. Souloumiac, "*Blind beamforming for non-Gaussian signals*", Radar and Signal Processing, IEE Proceedings-F, vol. 140, no. 6, December 1993, pp. 362-370.

[17] S. Suppapola, Y. Sun, "*Nonlinear Transforms of ECG Signals for Digital*", IEEE Transactions on Biomedical Engineering, vol. 41, n. 4, April 1994, pp. 397 - 400.

[18] R. Poli, S. Cagnoni, G. Valli, "*Genetic design of optimum linear and nonlinear QRS detectors*", IEEE Transactions on Biomedical Engineering, vol. 42, no. 11, November 1995, pp.1137-1141.

[19] S. Kadambe, R. Murray, G. Faye Boudreaux-Bartels, "*Wavelet transform-based QRS complex detector*", IEEE Transactions on Biomedical Engineering, vol. 46, no. 7, July 1999, pp. 838-848.

[20] V. X. Afonso, W. J. Tompkins, T. Q. Nguyen, S. Luo, "*ECG beat detection using filter banks*", IEEE Transactions on Biomedical Engineering, vol. 46, no. 2, February 1999, pp. 192-202.

[21] H. P. Van Geijn, "*Cardiotocograpy. Textbook of Perinatal Medicine*"; Parthenon Publishing, vol. 2, 2006, pp. 1424-1428.

[22] M. A. Hasan, M. B. I. Reaz, M. Ibrahimy, M. S. Hussain, J. Uddin, "*Detection and Processing Techniques of FECG Signal for Fetal Monitoring*", Biological Procedures Online, vol. 11, no. 1, March 2009, pp. 263-295.

[23] A. Fanelli, M. G. Signorini, M. Ferrario, P. Perego, L. Piccini, G. Andreoni, G. Magenes, "*Telefetalcare: A first prototype of a wearable fetal electrocardiograph*", Engineering in Medicine and Biology Society, EMBC 2011, 2011 Annual International Conference of the IEEE, 30 August-3 September 2011, pp. 6899-6902.

[24] E. C. Karvounis, M. G. Tsipouras, D. I. Fotiadis, "*Detection of Fetal Heart Rate Through 3-D Phase Space Analysis From Multivariate Abdominal Recordings*", IEEE Transactions on Biomedical Engineering, vol. 56, no. 5, May 2009, pp. 1394-1406.

[25] M. A. Hasan, M. I. Ibrahimy, M. Reaz, J. Uddin, M. S. Hussain, "*VHDL modeling of FECG extraction from the composite abdominal ECG using Atificial Intelligence*", Industrial Technology, ICIT 2009, IEEE International Conference on, 10-13 February 2009, pp. 1-5.

[26] M. I. Ibrahimy, K. A. S. Al-Khateeb, M. A. Hasan, S. M. A. Motakabber, "*An Emergency Medical Care Network System for Fetal ECG Monitoring*", Electronic Design, Test and Application (DELTA), 2011 Sixth IEEE Int. Symposium on, 17-19 January 2011, pp. 176-179.

[27] S. Mucelli, D. Pani, L. Raffo, "*Real-time fetal ECG extraction with JADE on floating point DSP*", Electronics Letters, vol. 43, no. 18, August 2008, pp. 963-965.

[28] D. Pani, S. Argiolas, L. Raffo, "*A DSP algorithm and system for real-time ECG extraction*", 35th annual conference on Computers in Cardiology, 14-17 September 2008, pp. 1065-1068.

[29] L. Xuan-Ang, M. Xiao-Min, D. Ying-Qiang, Q. Zhi-Hui, C. En-Qing, "Research on a Fetal Electrocardiogram Separation System Based on DSP", International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS 2010, 6-8 December 2010, pp. 1-4.

[30] R. Arias-Ortega, M. J. Gaitan-Gonzalez, O. Yanez-Suarez, "*Implementation of a real-time algorithm for maternal and fetal heart rate monitoring in a digital signal controller platform*", 32nd Annual International Conference of the IEEE, EMBS 2010, 31 August-4 September 2010, pp. 2354-2358.

[31] J. Bell, T. J. Sejnowski, "*The "independent components" of natural scenes are edge filters*", Vision Research, vol. 37, no. 23, December 1997, pp. 3327–3338.

[32] B. U. Köhler, C. Hennig, R. Orglmeister, "*The Principles of Software QRS Detection*", IEEE Engineering in medicine and biology,vol. 21, no. 1, February 2002, pp. 42-57.

[33] J. Pan, W. J. Tompkins, "*A Real-Time QRS Detection Algorithm*", IEEE Transactions on Biomedical Engineering, vol. 32, n. 3, March 1985, pp. 230-236.

[34] D. Pani, A. Dessi, B. Cabras, L. Raffo, "*A Real-Time Algorithm for Tracking of Foetal ECG Sources Obtained by Block-on-Line BSS Techniques*", Computing in Cardiology, 9-12 September 2012, p. 65−68.

[35] B. De Moor, L. De Lathauwer, J. Vandewalle, "*Fetal Electrocardiogram Extraction by Blind Source Subspace Separation*", IEEE Transactions on Biomedical Engineering, vol. 47, no. 5, May 2000, pp. 567–572.

[36] A. K. Panda, P. Rajput, B. Shukla, "*FPGA Implementation of 8, 16 and 32 Bit LFSR with Maximum Length Feedback Polynomial using VHDL*", Communication Systems and Network Technologies (CSNT), 2012 International Conference on, 11-13 May 2012, pp. 769-773.

[37] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, H. E. Stanley, "*PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals*". [http://circ.ahajournals.org/cgi/content/full/101/23/e215], 2000 (June 13).

[38] A. Fanelli, M. G. Signorini, P. Perego, G. Magenes, G. Andreoni, "*A Smart Wearable Prototype for Fetal Monitoring*", Advances in Human Aspects of Healthcare, CRC Press. 2012, pp. 33-39.

[39] I. Hatai, I. Chakrabarti, S. Banerjee, "*FPGA Implementation of a Fetal Heart Rate Measuring System*", Advances in Electrical Engineering (ICAEE), 2013 International Conference on, 19-21 December 2013, pp. 160-164.

[40] D. P. Morales, A. Garcia, E. Castillo, M. A. Carvajal, L. Parrilla, A. J. Palma, "*An application of reconfigurable technologies for non-invasive fetal heart rate extraction*", Medical Engineering and Physics, vol. 35, no. 7, July 2013, pp. 1005-1014.

*Revised manuscript (clean)

Custom FPGA Processing for Real-Time Fetal ECG Extraction and Identification

E. Torti[1], D. Koliopoulos[2], M. Matraxia[3], G. Danese[1], F Leporati[1]

[1]University of Pavia, Dept. of Electrical, [2]Positech Consulting, [3]Johnson & Johnson

Corresponding author: Francesco Leporati, Computer and Biomedical Engineering, Via Ferrata 5, Pavia, Italy phone: (+39) 0382985678-373 mail: leporati@unipv.it

***Abstract*—Monitoring the fetal cardiac activity during pregnancy is of crucial importance for evaluating fetus health. However, there is a lack of automatic and reliable methods for Fetal ECG (FECG) monitoring that can perform this elaboration in real-time. In this paper, we present a hardware architecture, implemented on the Altera Stratix V FPGA, capable of separating the FECG from the maternal ECG and to correctly identify it. We evaluated our system using both synthetic and real tracks acquired from patients beyond the 20th pregnancy week.**

This work is part of a project aiming at developing a portable system for FECG continuous real-time monitoring. Its characteristics of reduced power consumption, real-time processing capability and reduced size make it suitable to be embedded in the overall system, that is the first proposed exploiting Blind Source Separation with this technology, as the best of our knowledge.

***Index Terms*—Embedded systems, fetal ECG, Field Programmable Gate Array (FPGA), Biomedical instrumentation**

I. Introduction

The monitoring of the fetal cardiac activity using Doppler methods is a standard examination used, together with sonogram, for evaluating fetus health.

Fetal monitoring is often performed in medical centers, and the analysis of the cardiac tracks is simply performed by eye inspecting the recordings. This means that the recognition relies on the experience of the doctors, with a possible low level of reliability [1]. These considerations lead to the need of a portable system that can autonomously extract and identify the FECG in real-time, eventually storing it in a local memory or sending to a dock-unit through a Wi-Fi connection. This allows a continuous monitoring, not always possible with the traditional systems. Moreover, recent advances in the field of "textile wearable devices" [2] [3] [4] [5] make possible to develop a wearable unit that can acquire and examine the signals and send them to a remote diagnostic center.

The acquired ECG signal is the result of the superimposition of electrical activities corresponding to maternal and fetal hearts. In addition, there are noisy contributions due to electrodes, maternal breath and involuntary movements [6].

Noise can be reduced using a suitable filtering stage, so the system must be able to separate the FECG from the maternal one. Separation quality is improved if the acquisition involves multiple ECG channels (not less than 4).

FECG extraction has been widely investigated in the last years and many approaches emerged, exploiting wavelet transformations [7] [8], typical artificial intelligence techniques [9] [10] [11] and purposely devoted algorithms [12] [13] [14].

Among these solutions, we performed a careful analysis, after which Blind Source Separation (BSS) algorithms such as Infomax [15] and JADE [16] emerged as the best techniques, both in terms of reliability and in terms of documentation for implementing the same steps on a dedicated hardware circuit like the one we want to carry out.

These two methods produce a set of signals that need to be classified for FECG recognition. The JADE algorithm produces a set of outputs where portion of the fetal signal can be subdivided in different output signals and should then be recomposed. For what concerns Infomax, instead, this technique does not suffer from this problem, since the fetal signal is fully contained in at least one of the output channels.

It must be noticed that, in both cases, the output channels containing cardiac signals preserve the morphological characteristics, such as QRS peaks and P and T waves. We chose to design an architecture based on Infomax, since we want to identify only one channel containing the fetal track.

In particular, we chose a nonlinear Infomax network, which produces sets of filters that are usually applied to data streams containing mixed information sources with the aim of separating them [15]. Those filters produce outputs that are as independent as possible, so it is possible to say that the chosen network performs an Independent Component Analysis (ICA). Moreover, the literature shows that Infomax outperforms, in terms of extraction reliability, similar approaches, such as Fast-ICA, Pearson-ICA and Sequential Analysis.

After FECG extraction, we need to identify which channels correspond to mother (maternal signal), to fetus signal, or to noise; also this step should be done automatically.

This issue has been addressed in different ways, such as nonlinear transforms [17], genetic algorithms [18], wavelet transforms [19] and filter banks [20].

We identify suitable filter banks for highlighting typical features of the cardiac signal [20][28] introducing some useful modifications. Those filters are applied to each output channel produced by Infomax. The resulting channels can be separated in two groups: noisy channels and cardiac channels. The first ones are discarded while the others are classified through a suitable clustering algorithm described further in the paper.

We choose to use a high-end FPGA device because we need high performance in terms of elaboration speed but also limited power consumption. We are exploring this technology as an alternative to the literature, which is typically based on Digital Signal Processors (DSP).

The proposed system has been validated using both synthetic and real tracks, outperforming existing monitoring systems in terms of elaboration speed and detection accuracy. Moreover, our system is real-time compliant and has a reduced power consumption, enabling its use for the development of a portable monitoring system.

This project involves a collaboration among the dept. of Electrical, Computer and Biomedical Engineering at the University of Pavia, the Bioengineering dept. of the Polytechnic of Milan and the Polyclinic research unit of the University of Naples and has been funded by the Italian government within the National Interest Research Projects Program. At the best of our knowledge, this is the first portable device exploiting BSS on the FPGA technology for FECG monitoring.

The remainder of the paper is organized as follows: section II describes the fetal ECG signal and its main characteristics. Section III presents the state of the art of automatic FECG extraction and identification. Section IV and V respectively describe the Infomax network and the classification algorithm adopted in our work. Section VI presents the FPGA design and section VII describes experiments conducted for validating the architecture. Section VIII concludes the paper with some remarks and comparisons with the state of the art.

II. The fetal electrocardiogram

The ECG is a track representing the heart's electrical activity. A typical ECG record shows a cardiac cycle which is made up of three parts: the P wave (related to the atrial depolarization), the QRS complex (related to ventricles depolarization) and the T wave (related to new ventricle polarization).

The heart rate is an important parameter, which is defined as the time between two consecutive R peaks; the inverse of the heart rate is called cardiac frequency. This frequency, in an adult, is typically in the range [60-100] beats per minute (bpm), while the fetal one is in the range [110-150] bpm. For what concerns fetal cardiac frequency, fast modifications of this frequency are considered normal, but their absence or a frequency out of the previous mentioned range are considered indicators of possible anomalies. Moreover, FECG is correlated with the maternal body response. For example, accelerations of the cardiac frequency with respect to uterine contractions (UC) rates is an indicator of non-correct abdominal venous circulation [21].

Nowadays, fetal cardiac monitoring is mainly conducted in an invasive form and could happen only during labor. In this case an electrode is placed on the fetus head, but it is an extremely delicate procedure. For what concerns noninvasive techniques, the most common exploit ultrasound Doppler acquisition and fetoscope, but, unfortunately, they do not allow a continuous monitoring [22].

The ECG signal can also be acquired using surface electrodes positioned on the mother's abdomen. As said before, the electrodes acquire, together with the FECG, the maternal ECG and noise. An example of those mixed signals is depicted in figure 1.

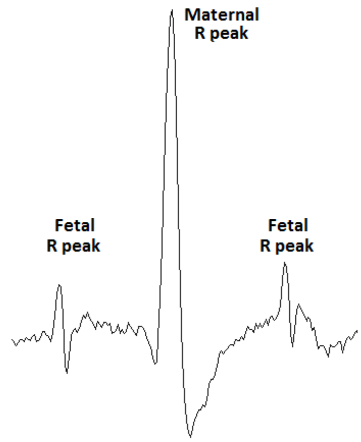

Figure 1 - Superimposition of FECG, maternal ECG and noise.

III. State of the art

Fetal cardiac signal separation from mother's one has been widely investigated and, as already highlighted, the best techniques are based on BSS. However, different techniques among the ones previously mentioned have been evaluated, in particular, the most recent and performing proposals are:

- the paper of Fanelli et al. [23] implementing the Marten's algorithm (a PCA enhancement of the QRS complex with averaging and filtering);
- the two papers of Karvounis et al. [7][24] featuring a three stage mixed approach (parabolic fitting, spatial and time frequency multivariate analysis and wavelets). They try to identify maternal peaks and eliminate mother QRS complexes, while using histogram based techniques for finding fetal R peaks);
- the paper of Ming et al. [8] based on a two stage time frequency BSS and wavelet noise filtering.

All these proposals feature very interesting accuracies but are not real-time nor provide indications about the feasibility of a real-time implementation. On the other hand, Hasan et al [25] [26], Pani et al. [27] [28], Xuan-Ang et al. [29], and Arias Ortega et al. [30] proposed implementations into which online elaboration seems possible. The most interesting one is [25] into which a neural network based FECG extraction is implemented on a Stratix II FPGA but no results about elaboration time and working frequency (and then about real-time feasibility) are provided. The project presented in [26] on the other hand presents similarities with our solution, but seems still in its preliminary phase and the paper is focused only on the acquisition and networking implementation: in any case it does not feature portability/wearability nor local elaboration capability since the signal is acquired through a microcontroller while elaboration and extraction seems to be performed by a host computer.

Pani [27] [28] proposed a 300 MHz DSP implementation of JADE taking about 140 Mcycles/block for elaboration while Xuan-Ang et al. [29] features a similar DSP implementation of PCA but without giving accuracy and elaboration times.

The most complete work is the one by Arias Ortega et al. [30] who implemented a LMS adaptive filtering on a dsPIC30F6014A in 550 μsec with 93.1% accuracy and 87% of sensitivity. However, the power consumption is high (average 90 mA, max 200 mA) and the adopted algorithm is less accurate than BSS.

IV. The Infomax algorithm

Infomax is an ICA-based algorithm introduced by J. Bell and T. J. Sejnowski in 1995 [15] for audio signal elaboration. However, it has been successfully used also for ECG signal elaboration and FECG extraction [31].

The problem is described by the equation:

$$x(t) = As(t) + n(t) \tag{1}$$

where $x(t)$ is the acquired signal, $s(t)$ represents the different sources which must be weighted with the values of the matrix A and $n(t)$ is the noise. In particular, A is a MxN matrix where M is the dimension of $x(t)$ and N is the dimension of $s(t)$. If we assume that the sources are statistically independent it is possible to separate them.

The Infomax algorithm estimates the matrix W which is defined as the inverse of the mixing matrix A.

In our case, the $x(t)$ measurements are the acquired abdominal ECG signals. If the signal to noise ratio is high enough, the noise $n(t)$ can be neglected, so it is possible to estimate the independent signal as:

$$\hat{s}(t) = Wx(t) = PSx(t) \tag{2}$$

where the matrix W is factorized by the permutation matrix P and the scaling matrix S. It is possible to choose those matrices so that $PS = I$, where I is the identity matrix. The order of the estimated sources is not a-priori known. Moreover, the amplitude information is lost, but this is not a critical issue, since the signal shape is sufficient for evaluating FECG cardiac frequency.

Infomax estimates the W matrix using iterative approximations, trying each time to minimize the mutual information between distinct sources:

$$I(\hat{s}_i, \hat{s}_j) = 0 \tag{3}$$

It is possible to minimize $I(\hat{s}_i, \hat{s}_j)$ by elaborating the joint entropy $H(\hat{s}_i, \hat{s}_j)$, and to do so we can consider the mutual information between system outputs, which is defined as:

$$I(\hat{s}, x) = H(\hat{s}) - H(\hat{s} \mid x) \tag{4}$$

where $H(\hat{s})$ is the joint entropy of the outputs and $H(\hat{s} \mid x)$ is the conditional entropy, mainly due to noise. For minimizing $H(\hat{s})$ and, consequently, $I(\hat{s}, x)$, it is possible to differentiate the equation (4) with respect to a characteristic parameter of the network, that we will indicate with w. If we assume that the relation between inputs and outputs can be well represented by logistic function:

$$\hat{s} = (1 + e^{-(w_x + w_0)})^{-1} \tag{5}$$

it is possible to define a learning rule to update weights and make an iterative estimation of the matrix W:

$$\Delta w \propto w^{-1} + x(1 - 2\hat{s}) \tag{6}$$

In the vector space equations (6) becomes:

$$\Delta W \propto [W^T]^{-1} + (1 - 2\hat{s})x^T \tag{7}$$

At the first iteration, W is set to the value $W_0 = 1$; the iterative process stops when the square of the difference between two consecutive approximations of W is sufficiently low. This threshold is a critical issue, since a big value involves a low quality separation, while a small threshold involves a high number of iterations, which are not compatible with a real-time implementation.

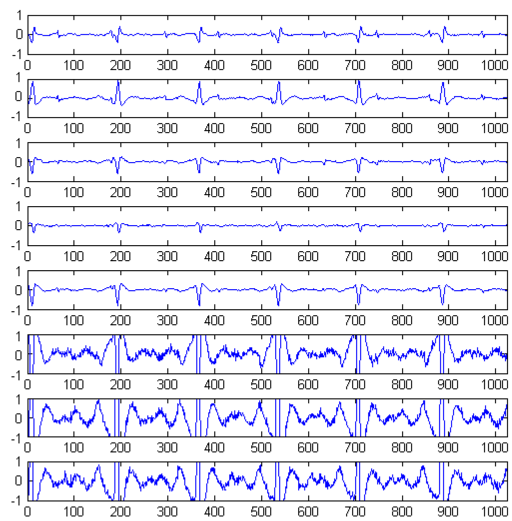

Figure 2 - Input channels. On the x axis there is the sample number, while on the y axis there is the amplitude, measured in mV.

Moreover, it is possible to improve performance by pre-processing the discrete input signals $x(k)$ in order to make them uncorrelated. To this purpose, we use a de-correlation matrix W_z:

$$W_z = 2(\sqrt{Var[x(k)]})^{-1} \tag{8}$$

Figure 2 shows a dataset of cardiac tracks proposed by De Moor et al. [35], which is largely used for validating FECG extraction algorithms, while figure 3 shows how those tracks have been separated by Infomax.

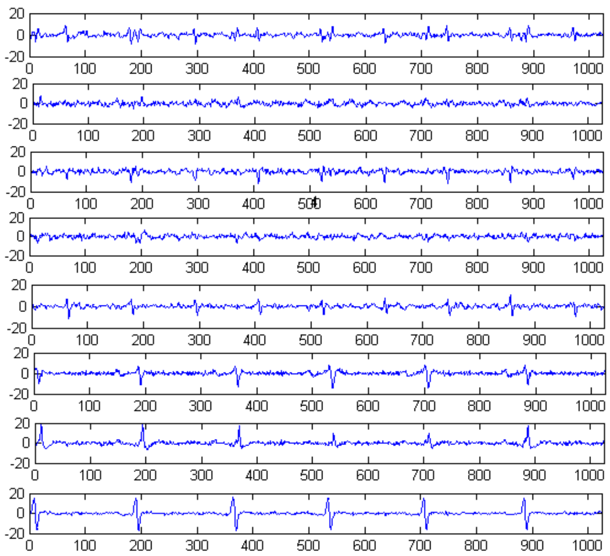

Figure 3 - Output channels produced by Infomax. On the x axis there is the sample order number, while on the y axis there is the amplitude, measured in arbitrary units. The fetal signal is in channel five.

V. The Classification Algorithm

The output channels produced by Infomax must be classified for clearly pointing out which one contains the fetal track. Our technique is based on a preliminary filter that affects all the Infomax output channels for highlighting the peaks, typical in the QRS complex of cardiac signals. After that, a phase of noise channels identification takes place. This stage is required to send to the successive clustering phase those channels that contain cardiac activity only from the fetus and from the mother.

The block diagram of this algorithm is depicted in figure 4.

In our case, N is equal to 8 and M is the number of noise channels and depends on the output signals produced by Infomax.

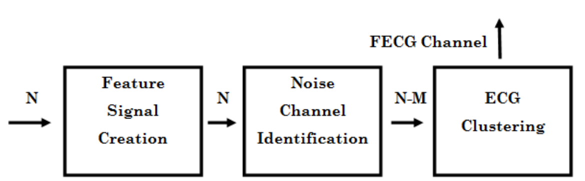

Figure 4. Block diagram of the classification algorithm.

A. Feature signals creation

This phase exploits a series of FIR filters for highlighting the peaks related to the QRS complex.

Standard filters for QRS detection have been widely exploited [17] [32] [33].

We decided, according to [34], to use:

- a first order derivative filter;
- a second order derivative filter;
- a Multiplication Of Background Difference (MOBD) filter;
- a weighted moving average filter.

The first order derivative filter is a high-pass filter and is used for highlighting the fast amplitude variations, which typically occur in QRS events.

Several implementations of this filter are provided in literature such as [32]:

$$y[n] = x[n+1] - x[n-1] \tag{9}$$

$$y[n] = 2x[n+2] + x[n+1] \tag{10}$$

$$y[n] = x[n] - x[n-1] \tag{11}$$

where x denotes the input sample vector.

We chose to use a different first order derivative filter, based on absolute values, to avoid negative values:

$$y[n] = \frac{1}{2}\left(|\, x[n+1] - x[n-1] \,|\right) \tag{12}$$

For what concerns the second order derivative filter, it is implemented as:

$$y_2[n] = \frac{1}{4}(x[n+2] - 2x[n] + x[n-2]) \tag{13}$$

which is further combined with the aforementioned first order derivative filter after a weighted moving average and a MOBD filter:

$$u[n] = y_1[n] + \frac{1}{2} z[n] \tag{14}$$

where y_1 is the output of the first order derivative filter, and z is the result of the second order derivative filter, the moving average filter and the MOBD filter.

We implemented a MOBD filter given by:

$$z[n] = y_3[n] * y_3[n-1] \tag{15}$$

where y_3 is the output of the second order derivative filter and the moving average filter.

The adopted filtering schema is shown in figure 5.

Figure 6 shows the feature signals obtained by the output channels depicted in figure 3.

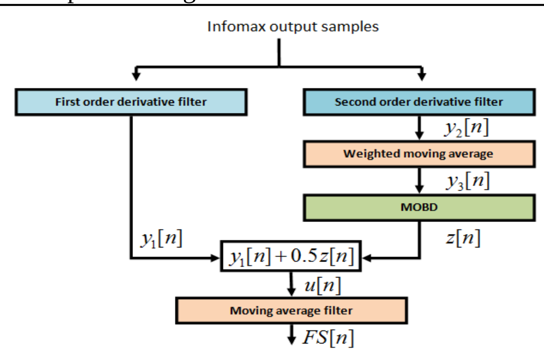

Figure 5 - Adopted filtering schema

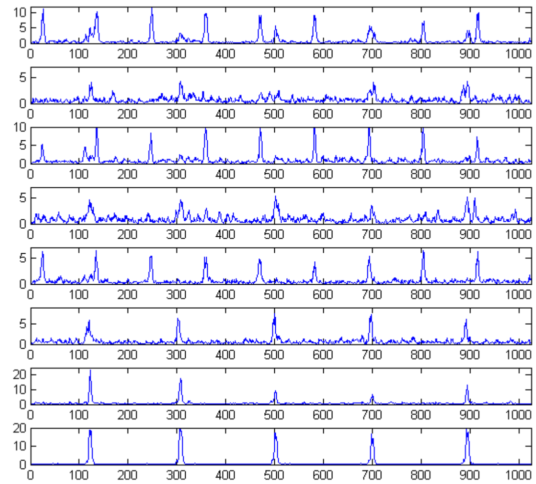

Figure 6 - Feature signals. On the x axis there is the sample number, on the y axis there is the amplitude, measured in arbitrary units.

B. Noise channels identification

The feature signals obtained using the filtering stage are the inputs of the noise channel identification stage. This phase is needed for sending a limited number of channels to the clustering stage, accelerating the classification phase, to avoid working on signals with no clinical interest.

For each channel, a suitable threshold is computed as function of the maximum amplitude, to exclude uninteresting signals. This is done to remove the baseline noise.

In our implementation, this threshold has been set to 30% of the maximum value, so we can remove the baseline noise without removing plausible QRS peaks.

After that another first order derivative filter, like the one described in the previous section, is applied, for highlighting the transactions from positive to negative derivative of each channel.

Those transactions are defined as the *turning points* (TP), related to the number of QRS events of each signal.

The number of the detected turning points is used to evaluate if a channel should be sent to the clustering stage or not.

Our architecture must work with ECG records of about 4 seconds, as they come from the wearable acquisition system designed by the Polytechnic of Milan [23]. This system has been developed using a STR711RF2 microcontroller, which is equipped with a 16-bit A/D converter. The number of turning points can be used for calculating the cardiac frequency. We choose to send to the clustering phase all those channels that contain a number of turning points belonging to the range [43-190] bpm, so those channels may contain mother and fetal signals.

We also generalized this threshold for working with different record lengths, implementing the empirical formula:

$$3 * \frac{nums}{1024} \leq TP \leq 13 * \frac{nums}{1024} \qquad (16)$$

where $nums$ is the number of samples to be processed and 1024 is the number of samples processed by default in our implementation (250 Hz sampling frequency with a 1 KB buffer). In this phase, beyond the number of turning points, the average distance between two adjacent turning points and the standard deviation of the distances is elaborated. Those three values are needed by the clustering phase.

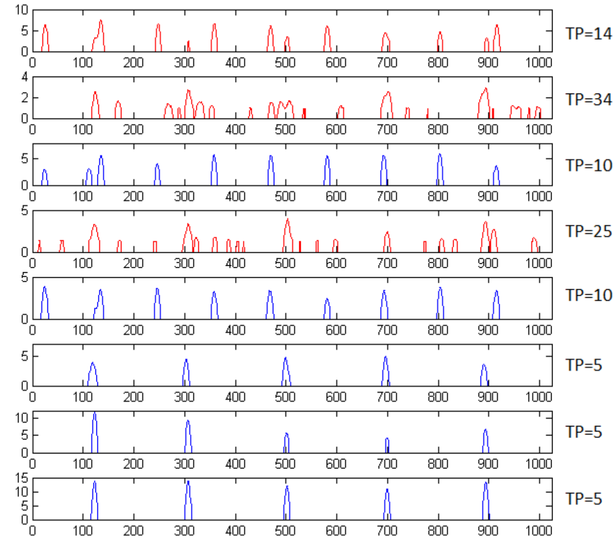

Figure 7 - Noise channel identification outputs. On the x axis there is the sample number, while on the y axis there is the amplitude, measured in arbitrary units.

Figure 7 shows the output of the noise channel identification phase, assuming as input the signals shown in figure 6. The red ones are the channel labelled as noise, while the blue ones are the signals used as input for the clustering stage.

C. Clustering

The authors of [34] proposed an agglomerative clustering based on the Pearson correlation index for FECG identification. However, we chose to test different clustering techniques in order to improve the performance.

Instead of using the Pearson correlation index for evaluating the distance between two clusters, we used a single linking cluster working on the number of turning points.

We also adopted a K-means clustering method, testing two different ways of partitioning data. The first one uses only the number of turning points as parameter to perform the partitioning, while the second one uses also the standard deviation between all the turning points distance and the average between the distances. Those values are computed by the previous noise channel identification stage.

We tested four clustering algorithms (agglomerative clustering from [31], single linking clustering and the two K-means clustering described before) with our dataset and we found that the best accuracy is obtained using the K-means method based only on the number of the turning points (accuracy of 100%). The agglomerative clustering proposed in [31] shows an accuracy of 80%, while the single linking clustering an accuracy of 95% and, finally, the K-means based on two parameters shows an accuracy of 60%.

After those experiments, we chose to work with the K-means clustering based only on turning point numbers. This algorithm automatically classifies data into two distinct groups, creating a fetal cluster and a maternal cluster using as information only the number of turning points calculated by the previous phase.

K-means clustering results depend on the initialization of the algorithm and it is not guaranteed that the obtained result is the optimal one, but it can be a sub-optimal solution. This is due to a non linear objective function that is minimized in the clustering stage, that ends when a minimum is reached. However it is not assured that it is a global minimum.

To avoid this problem we chose to run five different instances of the K-means clustering algorithm with different initializations. When all the five algorithms ended, the values of the objective functions are compared and the minimum one is chosen as optimal one. The corresponding partition is selected as output of the clustering stage.

In the K-means algorithm, the number of groups to be produced as output must be set. We set this value to two, because we foresee to have only maternal and fetal signals as input of this stage. Experiments have pointed out that some noise channels can pass the previous filtering stages and enter in the clustering stage. Those channels are typically classified in the fetal cluster due to the number of their turning points. To avoid this problem, when in the fetal cluster there is more than one signal, the one is selected whose distances between the turning points feature the lower standard deviation. This choice has been made because a cardiac signal is always more regular than noise.

Moreover, our experiments showed that the K-means algorithm performs better than agglomerative clustering proposed in [34] (see section VII).

VI. System Implementation on FPGA

The whole FECG extraction and identification system has been implemented on an Altera Stratix V 5SGXEA7N2F45C2 FPGA. We chose this device because Stratix is the high-end FPGA developed by Altera.

Since floating point calculations are expensive in terms of resource usage and execution time, we used a fixed point data representation. Each sample is represented with a 32 bit word, where the 12 most significant bits are used for storing the integer part, while the remaining 20 are used for the decimal one.

First of all, we developed a fixed point library for performing addition, subtraction, multiplication and division. The blocks have been connected using the pipeline philosophy, to maximize the working frequency of the architecture.

The division block requires a latency and a number of resources greater than the others, so we used it only when strictly required. For all those operations where the divisor is a constant value, we obtain the result using a series of shifts and additions, with a consequent resource saving.

For what concerns Infomax, the architecture can be divided into three main parts. The first one computes the product:

$$U = W_{old}\, x \qquad (17)$$

where W_{old} is the matrix estimated at the previous iteration. After that we compute the learning rule, according to (7), given by:

$$\Delta W_k = L\left(BI + \left(1 - 2\left(\frac{1}{1+e^{-U}}\right)\right) U^T \right) W_k \qquad (18)$$

where L is a suitable constant, B is the number of samples acquired for each channel and k is the considered iteration. In our case, B is equal to 1024; this value is related to the records length (about 4 seconds) and the sampling frequency (250 Hz). Equation (18) is used for computing the new separation matrix:

$$W_{k+1} = W_k + \Delta W_k \qquad (19)$$

The third stage of the Infomax architecture computes the termination criteria given by:

$$change = \sum_{i=1}^{N} \sum_{j=1}^{M} \left(W_k^{i,j} - W_{k-1}^{i,j} \right)^2 \qquad (20)$$

where the threshold is chosen for stopping the iterative process when the mean squared error is less than 2.5%. The most complex stage of the architecture is the computation of equation (18), which is made up of four main blocks. The first one computes the C matrix, given by:

$$C = 1 - 2\left(\frac{1}{1+e^{-u}}\right) \qquad (21)$$

where the exponential function is implemented using the corresponding Taylor series truncated at the fifth term. Moreover, the inversion is computed using Newton's method, which is an iterative method. We experimentally found that seven iterations are sufficient to reach a suitable precision for our architecture. After that, the architecture computes the D matrix, given by:

$$D = L(BI + Cu^T) \qquad (22)$$

The next two steps are a multiplication and a sum for computing the new separation matrix. It is important to emphasize that all matrix multiplications have been performed in parallel for achieving better performance. While the new separation matrix is computed, we also calculate equation (20) to establish if the algorithm converges. If this criteria is not fulfilled, the control logic saves the new separation matrix into a suitable RAM and, subsequently, starts another iteration of Infomax, otherwise the separated samples are computed and saved in suitable FIFO memories, which are the input of the identification and classification phases.

The schema of this elaboration chain is depicted in figure 8.

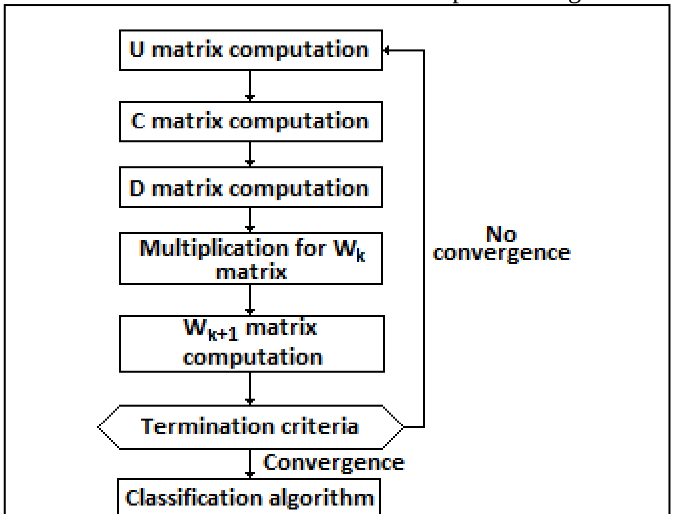

Figure 8 - Schema of the Infomax implementation

The first operation of the classification algorithm is *feature signals creation*. As shown by Figure 5, the first order derivative filter and the series of second order derivative filters, the moving average filter and the MOBD filter can be performed in parallel.

We instantiated eight different blocks for feature signals creation that are able to elaborate in parallel all the Infomax output channels.
The second phase, the *noise channel identification*, requires a buffer for storing the samples during the threshold evaluation. This buffer is implemented through a simple 1 KB FIFO memory. When the input samples are produced by the previous phase, the maximum value is calculated while storing the data inside the FIFO. After the maximum value has been computed, the threshold is calculated and the FIFO is read. The output sample is set to zero if the read value is below the threshold otherwise it is kept as it has been acquired.
While transferring the correct output, this phase also calculates the number of turning points (i.e. the average of the distance). The standard deviation is not directly computed, because it requires to perform a square root operation, which is heavy in terms of FPGA usage. For this reason, we chose to compute the cheaper variance.
For each feature signal, one of those blocks is created. Those eight blocks are connected to another block used for establishing if each feature signal is noise or not.
After that, the selected turning points numbers of the feature signals, the average of distance and the standard deviations are stored in separated FIFOs.
Those data are the inputs of K-means block.
As before mentioned, K-means requires an initialization, that must be a random value for the best performance. We implemented a Galois linear feedback shift register [36], for producing a pseudorandom 32 bit value.
This circuit also needs a seed, which can be given using an input pin, so at each execution the initialization can be changed.
We created five K-means blocks, that work in parallel on the same data; each K-mean block has a different initialization value. Each one computes the *objective* function given by:

$$SSE = \sum_{i=1}^{2} \sum_{x \in Cl_i} \left(m_i - x \right)^2 \qquad (23)$$

where x is an element of the cluster Cl_i and m_i is the average value of the same cluster.
When all the K-means blocks end their elaboration, all the SSEs are processed and the results related to the minimum are selected as output.
If the selected cluster contains more than a channel, the output is chosen by comparing the average distance of the turning points and the standard deviation of the distance as described in the previous section.
The framework of the implementation is shown in figure 9. Please note that each channel is elaborated in parallel for what concerns the feature signal creation and the turning points count. Also the *noise identification* phase is performed on each channel in parallel, but at the end of this block there is a synchronization logic that waits the end of operations for all the channels. This is done because the clustering phase needs to work on the full dataset, which is ready after the end of the noise identification on all the channels. The output of this phase is the identifier of the channel containing the fetal signal (indicated as fetal channel in figure 9).

VII. Experimental results and discussion

First of all, we tested the proposed elaboration chain using the well-known Physionet database [37]. The tests show that our algorithm is capable of correctly separate and identify the fetal signal.
We tested the proposed architecture both with synthetic data and with real data. The first dataset is the well-known one by De Moor et al. [35], while the latter is a dataset of 343 tracks acquired from pregnant volunteers at the Polytechnic of Milan and at the Naples hospital “Vincenzo Cardarelli”. The volunteers were informed on the nature of the research and about the use of the acquired data and expressed their consent to the purpose of their use. Those real tracks have been acquired using the wearable prototype described in [38]. We stored the input samples, both the synthetic and the real, on a suitable on-chip RAM.
The proposed system correctly separates and classifies both the simulated and the real tracks.
After separation of each track into eight channels, there is always at least one channel containing the fetal signal. In all the considered cases the fetal track is correctly distinguished by the architecture, so it is possible to say that, for the considered dataset, there are no false negatives.

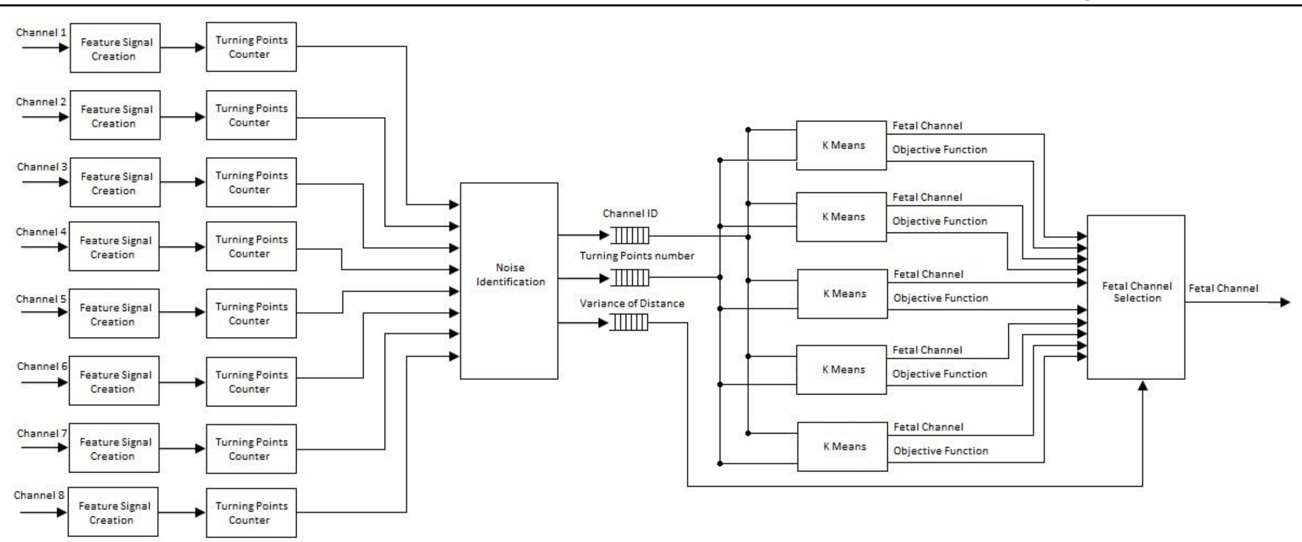

Figure 9 - Schema of the classification algorithm implementation

TABLE I – Comparisons between the proposed work and the state of the art

	Device	Technique	Working frequency [MHz]	Time [ms]	Power consumption [W]
Ortega et al. [30]	dsPIC30F6014A	LMS (No BSS)	30	0.55	0.45-1
Pani et al. [34]	OMAP L-137 DSP	OL-JADE	300	120	1
Hasan et al. [25]	FPGA	Neural Network	N. A.	N. A.	N. A.
Morales et al. [40]	FPGA	Adaptive filtering (NO BSS)	50	600	N. A.
Proposed work	FPGA	Infomax + K_Means	95	1.8-27	1
Proposed work	FPGA	Infomax + K_Means	50	3.4-54	0.55

Moreover, there are also no false positives, since the clustering stage works considering two attributes: number of turning points and variance of distance between two consecutives turning points. Theoretically, a false positive, is a noisy channel classified as fetal signal. For being classified as fetal, a noisy channel must exhibit a correct number of turning points with regular distances. This is an improbable situation and is not present in our datasets. Finally, if a fetal channel reaches the clustering phase, it is always correctly classified, so the recall is 100%.

We have also estimated the 95% confidence interval using a binomial test (*binofit* Matlab function), obtaining as result the interval (98.9%-100%).

It should be emphasized, moreover, that a careful filtering has to be done to obtain accurate results. In our case, we filtered the signals using an high pass FIR with a cut-off frequency of 5 Hz, for removing slow signals such as movements and breath.

We implemented this system on an Altera Stratix V 5SGXEA7N2F45C2 FPGA, equipped with 622k logic elements (LEs), 512 18x18 bit multipliers, 256 27x27 bit multipliers, 6.25 MB of memory and 1064 pins.

The resource usage is less than 15% for what concerns Logic Elements (29780 LEs), less than 3% of total memory (1668714 blocks) and about 44% of embedded multipliers (122 DSPs). The working frequency is 95 MHz. Power consumption has been estimated using Altera Quartus II Power Play Analyzer and is less than 1 W, so it is compatible with the constraints given by a portable system.

The elaboration time depends on the number of Infomax iterations, on the number of turning points and on the iteration needed on the clustering phase.

We tested the architecture using Altera DSP builder, which gives a suitable interface between the targeted FPGA and the PC. The results provided by the FPGA are the same shown in figures 3, 6 and 7. In general, the results provided by the FPGA are the same of the software simulation.

De Moor signals required only 2 Infomax iterations and a single clustering iteration, while, real signals required 30 Infomax iterations and 5 clustering iterations in the worst case. At the working frequency of 95 MHz, this corresponds to an elaboration time from 1.8 ms to 27 ms. The real-time constrain is satisfied, since the used tracks are of about 4 s.

The work proposed in [34], implementing OL-JADE on an OMAP L-137 DSP took about 120 ms to separate sources, so our implementation is faster than this one. The power consumption of the solution described in [34] is about 1 W, which is comparable with our power consumption. We also performed tests halving the working frequency of our architecture; the power consumption decreases to the value of about 550 mW, with a maximum elaboration time of about 54 ms, values that outperform the solution proposed in [34] while remaining still real-time compliant.

For what concerns the implementation carried out in [30] elaboration times are faster, but the adopted technique is not a BSS approach, so the separation quality is lower. Moreover, the power consumption is 1 W, since authors of [30] claimed a current absorption of 200 mA and the component is supplied with 5 V.

FPGA has been exploited in [11] but authors give no results about elaboration time and working frequency, so it is not possible to carry out a comparison. Another work based on FPGA is [39], which is real time compliant, but validation have been conducted only using synthetic tracks. Moreover, the adopted technique is based only on filtering and maternal ECG elimination, which is less reliable than modern BSS algorithms.

Finally, FPGA has been also exploited in [40], but the adopted technique is based on adaptive filtering, so it does not provide a good separation since it is not a BSS method. Elaboration time is about 600 ms working at 50 MHz, so the system is slower than our proposal. About power consumption, it is not possible to make comparisons since authors of [40] did not provide this information.

Table I gives a summary of these comparisons.

VIII. Conclusions

In this paper, we present a novel architecture for FECG extraction and identification. A suitable dataset, made up of both synthetic and in vivo signals, has been correctly classified by the system. Estimated power consumption is compatible with the constraints given by a portable device, overall target of this project.

The proposed architecture outperforms the elaboration times of the other works in literature who implement similar algorithms for successfully separating the tracks.

Moreover, the resource usage is compliant with the implementation of future algorithms on the same FPGA. For example, it is possible to improve the system by adding diagnostic functions, such as morphological analysis of the fetal track, which can also be performed in real-time. Another possibility is the development of a module for storing the separated tracks on an SD/SDHC card, making possible an off-line analysis of daily-acquired data.

It is also possible to integrate a Wi-Fi module for sending the fetal tracks to a dock-unit, which is responsible of data storage and eventually transmit them to a medical center. In particular, the Texas Instruments' WL1807MOD Wi-Fi module has a power consumption during transmission of 2.8 W, assuming a bit rate of 54 Mbps. In our application, this bit rate is too high, so it is reasonable to adopt a slower transmission, with the aim of reducing power consumption.

If we assume to power the whole system (the acquisition board, our FPGA and the Wi-Fi module) using a single TR1865 lithium battery, this will guarantee about 3 hours and 30 minutes of autonomy working at 95 MHz.

At last, it is possible to add a cryptography function for protecting data before transmission, since those data are strictly personal.

Another possibility is to use a smaller FPGA, such as an Altera Cyclone V, which is equipped with less logic resources. This device has a lower power consumption than the one considered by us. This choice will further reduce the power consumption of our system. However, this device is suitable for housing only our system, without the possibility to expand its functions with the features described above.

Moreover, we will explore solutions based on ARM Cortex M3 and M4 MCU which are less performant devices than our FPGA, but will hopefully guarantee a lower power consumption.

IX. Conflicts of Interest Disclosure

The authors (who participated in the research and in the article preparation jointly and on an equal basis) hereby certify that no personal or financial relationships exist with people or any other organization that could inappropriately influence the work.

X. Acknowledgements

The authors gratefully thank prof. Andrea Fanelli, prof. Danilo Pani and prof. Giovanni Magenes for their useful advices.

This work was funded by the Italian Ministry for the Health (PRIN 2009).

References

[1] M. Ferrario, M. G. Signorini, G. Magenes, "*New indexes from Fetal Heart Rate Analysis for the identification of severe intra uterine growth restricted fetuses*", Engineering in Medicine and Biology Society, 2006. EMBS '06, 28th Annual Conference of the IEEE, 30 August-3 September 2006, pp. 1458-1461.

[2] M. Di Rienzo, P. Meriggi, F. Rizzo, E. Vaini, A. Faini, G. Merati, G. Parati, P. Castiglioni, "*A wearable system for the seismocardiogram assessment in daily life conditions*", Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE, 30 August-3 September 2011, pp. 4263-4266.

[3] P. K. Jain, A. K. Tiwari "*Heart monitoring systems – A review*", Comp. in Biology and Medicine, vol. 54, no. 1, August 2014, pp.1-13.

[4] H. Han, J. Kim, "*Artifacts in wearable photoplethysmographs during daily life motions and their reduction with least mean square based active noise concellation method*", Comp. in Biology and Medicine, vol. 42, no. 4, April 2012, pp.387-393.

[5] D. Bansal, M. Khan, A. K. Salhan "*A computer based wireless system for online acquisition, monitoring and processing of ECG waveforms*", Comp. in Biology and Medicine, vol. 34, no. 4, April 2009, pp. 361-367.

[6] S. M. M. Martens, C. Rabotti, M. Mischi, R. J. Sluijter, "*A robust fetal ECG detection method for abdominal recordings*", Physiological Measurement, vol. 28, no. 4, March 2007, pp. 373-388.

[7] E. C. Karvounis, M. G. Tsipouras, D. I. Fotiadis, K. K. Naka "*An Automated Methodology for Fetal Heart Rate Extraction From the Abdominal Electrocardiogram*", IEEE Transactions on Information Technology in Biomedicine, vol. 11, no. 6, November 2007, pp. 628–638.

[8] M. Ming, W. Ning, L. San-Ya, "*Extraction of FECG Based on Time Frequency Blind Source Separation and Wavelet Denoising*", Bioinformatics and Biomedical Engineering, ICBBE 2009, 3rd International Conference on, Beijing, 11-13 June 2009, pp. 1-3.

[9] C. Kezi Selva Vijila, P. Kanagasabapathy, S. Johnson, "*Adaptive Neuro Fuzzy Inference System for Extraction of fECG*", IEEE Indicon 2005 Conference, 11-13 December 2015, pp. 224–227.

[10] Y. Zeng, S. Liu, J. Zhang, "*Extraction of Fetal ECG Signal via Adaptive Noise Cancellation Approach*", Bioinformatics and Biomedical Engineering, ICBBE 2008. The 2nd International Conference on, 11-16 May 2008, pp. 2270 - 2273.

[11] M. A. Hasan, M. B. I. Reaz, M. I. Ibrahimy, "*Fetal electrocardiogram extraction and R-peak detection for fetal heart rate monitoring using artificial neural network and correlation*", Neural Networks (IJCNN), The 2011 Conference on, 31 July-5 August 2011, pp. 15-20.

[12] Z. Shi, C. Zhang, "*Blind Source Extraction Using Generalized Autocorrelations*", IEEE Transactions on Neural Networks, vol. 18, no. 5, September 2007, pp. 1516–1524.

[13] Y. Yang, C. Guo, "*Blind Extraction for Noisy Fetal Electrocardiogram by Using Gaussian Moments*", 4th International Conference on Natural Computation, 18-20 October 2008, pp. 44–47.

[14] Y. Ye, J. Zeng, K. Lu, "*An Extraction Algorithm Based on Second-order Statistics for Noisy Mixtures*", Fuzzy Systems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conference on, 18-20 October 2008, pp. 281-285.

[15] J. Bell, T. J. Sejnowski, "*An information maximization approach to blind separation and blind deconvolution*", Neural Computation, vol. 7, no. 6, November 1995, pp. 1129–1159.

[16] J. F. Cardoso, A. Souloumiac, "*Blind beamforming for non-Gaussian signals*", Radar and Signal Processing, IEE Proceedings-F, vol. 140, no. 6, December 1993, pp. 362-370.

[17] S. Suppapola, Y. Sun, "*Nonlinear Transforms of ECG Signals for Digital*", IEEE Transactions on Biomedical Engineering, vol. 41, n. 4, April 1994, pp. 397 - 400.

[18] R. Poli, S. Cagnoni, G. Valli, "*Genetic design of optimum linear and nonlinear QRS detectors*", IEEE Transactions on Biomedical Engineering, vol. 42, no. 11, November 1995, pp.1137-1141.

[19] S. Kadambe, R. Murray, G. Faye Boudreaux-Bartels, "*Wavelet transform-based QRS complex detector*", IEEE Transactions on Biomedical Engineering, vol. 46, no. 7, July 1999, pp. 838-848.

[20] V. X. Afonso, W. J. Tompkins, T. Q. Nguyen, S. Luo, "*ECG beat detection using filter banks*", IEEE Transactions on Biomedical Engineering, vol. 46, no. 2, February 1999, pp. 192-202.

[21] H. P. Van Geijn, "*Cardiotocograpy. Textbook of Perinatal Medicine*"; Parthenon Publishing, vol. 2, 2006, pp. 1424-1428.

[22] M. A. Hasan, M. B. I. Reaz, M. Ibrahimy, M. S. Hussain, J. Uddin, "*Detection and Processing Techniques of FECG Signal for Fetal Monitoring*", Biological Procedures Online, vol. 11, no. 1, March 2009, pp. 263-295.

[23] A. Fanelli, M. G. Signorini, M. Ferrario, P. Perego, L. Piccini, G. Andreoni, G. Magenes, "*Telefetalcare: A first prototype of a wearable fetal electrocardiograph*", Engineering in Medicine and Biology Society, EMBC 2011, 2011 Annual International Conference of the IEEE, 30 August-3 September 2011, pp. 6899-6902.

[24] E. C. Karvounis, M. G. Tsipouras, D. I. Fotiadis, "*Detection of Fetal Heart Rate Through 3-D Phase Space Analysis From Multivariate Abdominal Recordings*", IEEE Transactions on Biomedical Engineering, vol. 56, no. 5, May 2009, pp. 1394-1406.

[25] M. A. Hasan, M. I. Ibrahimy, M. Reaz, J. Uddin, M. S. Hussain, "*VHDL modeling of FECG extraction from the composite abdominal ECG using Atificial Intelligence*", Industrial Technology, ICIT 2009, IEEE International Conference on, 10-13 February 2009, pp. 1-5.

[26] M. I. Ibrahimy, K. A. S. Al-Khateeb, M. A. Hasan, S. M. A. Motakabber, "*An Emergency Medical Care Network System for Fetal ECG Monitoring*", Electronic Design, Test and Application (DELTA), 2011 Sixth IEEE Int. Symposium on, 17-19 January 2011, pp. 176-179.

[27] S. Mucelli, D. Pani, L. Raffo, "*Real-time fetal ECG extraction with JADE on floating point DSP*", Electronics Letters, vol. 43, no. 18, August 2008, pp. 963-965.

[28] D. Pani, S. Argiolas, L. Raffo, "*A DSP algorithm and system for real-time ECG extraction*", 35th annual conference on Computers in Cardiology, 14-17 September 2008, pp. 1065-1068.

[29] L. Xuan-Ang, M. Xiao-Min, D. Ying-Qiang, Q. Zhi-Hui, C. En-Qing, "Research on a Fetal Electrocardiogram Separation System Based on DSP", International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS 2010, 6-8 December 2010, pp. 1-4.

[30] R. Arias-Ortega, M. J. Gaitan-Gonzalez, O. Yanez-Suarez, "*Implementation of a real-time algorithm for maternal and fetal heart rate monitoring in a digital signal controller platform*", 32nd Annual International Conference of the IEEE, EMBS 2010, 31 August-4 September 2010, pp. 2354-2358.

[31] J. Bell, T. J. Sejnowski, "*The "independent components" of natural scenes are edge filters*", Vision Research, vol. 37, no. 23, December 1997, pp. 3327–3338.

[32] B. U. Köhler, C. Hennig, R. Orglmeister, "*The Principles of Software QRS Detection*", IEEE Engineering in medicine and biology,vol. 21, no. 1, February 2002, pp. 42-57.

[33] J. Pan, W. J. Tompkins, "*A Real-Time QRS Detection Algorithm*", IEEE Transactions on Biomedical Engineering, vol. 32, n. 3, March 1985, pp. 230-236.

[34] D. Pani, A. Dessi, B. Cabras, L. Raffo, "*A Real-Time Algorithm for Tracking of Foetal ECG Sources Obtained by Block-on-Line BSS Techniques*", Computing in Cardiology, 9-12 September 2012, p. 65−68.

[35] B. De Moor, L. De Lathauwer, J. Vandewalle, "*Fetal Electrocardiogram Extraction by Blind Source Subspace Separation*", IEEE Transactions on Biomedical Engineering, vol. 47, no. 5, May 2000, pp. 567–572.

[36] A. K. Panda, P. Rajput, B. Shukla, "*FPGA Implementation of 8, 16 and 32 Bit LFSR with Maximum Length Feedback Polynomial using VHDL*", Communication Systems and Network Technologies (CSNT), 2012 International Conference on, 11-13 May 2012, pp. 769-773.

[37] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, H. E. Stanley, "*PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals*". [http://circ.ahajournals.org/cgi/content/full/101/23/e215], 2000 (June 13).

[38] A. Fanelli, M. G. Signorini, P. Perego, G. Magenes, G. Andreoni, "*A Smart Wearable Prototype for Fetal Monitoring*", Advances in Human Aspects of Healthcare, CRC Press. 2012, pp. 33-39.

[39] I. Hatai, I. Chakrabarti, S. Banerjee, "*FPGA Implementation of a Fetal Heart Rate Measuring System*", Advances in Electrical Engineering (ICAEE), 2013 International Conference on, 19-21 December 2013, pp. 160-164.

[40] D. P. Morales, A. Garcia, E. Castillo, M. A. Carvajal, L. Parrilla, A. J. Palma, "*An application of reconfigurable technologies for non-invasive fetal heart rate extraction*", Medical Engineering and Physics, vol. 35, no. 7, July 2013, pp. 1005-1014.