International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 12 (2016) pp 7809-7814

Implementation of Heart Rate Variability Analysis Algorithm on FPGA Platform

Nur Sabrina Risman[a], Siti Norhayati MD Yassin[b], Nazrul Anuar Nayan[a], and Sia Chen Wei[c], Yuan Wen Hau[b]
[a]*Department of Electrical, Electronic and Systems Engineering, Universiti Kebangsaan Malaysia, Selangor, Malaysia.*
[b]*Biochip and Medical SoC Laboratory, Universiti Teknologi Malaysia, Johor, Malaysia.*
[c]*VeCAD Research Laboratory, Universiti Teknologi Malaysia, Johor, Malaysia.*

Abstract
Recently, many studies have attempted to develop portable cardiac monitoring systems that can be used outside of a hospital setting. To achieve this objective, in this paper, a field-programmable gate array (FPGA) design and the implementation of an embedded electrocardiography (ECG) system using System-on-Chip (ECG-SoC) technology are proposed. The proposed system performs ECG pre-processing and heart rate variability (HRV) feature extraction that are suitable for remote homecare monitoring and rural health care applications. The development of ECG-SoC consists of four phases: the development of the hardware system, the Nios II-Linux embedded operating system (OS), the ECG-SoC software, and followed by system integration. The results of this study show that the developed ECG-SoC system is capable of compiling a raw ECG dataset, detecting QRS, computing R-R intervals, and displaying the FFT output and power spectrum analysis. The proposed system has the potential to be used in the future as a portable stand alone medical device for heart disease detection and monitoring.

Keywords: Electrocardiography (ECG), field-programmable gate array (FPGA), heart rate variability (HRV), feature extraction, heart disease.

INTRODUCTION

The increasing number of elderly individuals has made health monitoring products indispensable. These devices play an important role in the early and rapid warning of heart and cardiovascular disease. Therefore, the development of ECG diagnostic equipment is a high priority. However, ECG devices in use today are bulky and unwieldy for patients to use for long-term health monitoring. In additional, these medical products can be too expensive for personal use (Tseng and Fang 2011) (Fang, Huang et al. 2013).

Today, advanced patient monitoring techniques have been developed for tracking patient conditions (Redondi, Chirico et al. 2013) but little clinical evidence has been collected to evaluate their efficacy (Nangalia, Prytherch et al. 2010). Therefore, despite recent progress in novel sensing devices, the standard of care for most physicians and healthcare systems remains the periodic observation of patients' vital signs (Tarasssenko and Clifton 2011). An ECG is a graphic tracing of the voltage generated by the cardiac or heart muscle during heartbeat activity (Clifford and Clifton 2012). ECG monitoring devices must be able to extract the features of an ECG signal in real time (Shyu and Hu 2007). Heart rate variability (HRV) is a normal physiological phenomenon where the interval between successive heart beats of an individual varies over time. HRV reliably reflects the many physiological factors modulating the normal rhythm of the heart (Hu, Lin et al. 2011). The timing, frequency, and nonlinearity of this signal are the major processing factors that are extracted from the HRV signal, and the resulting features are useful tools for diagnosing different problems related to heart function (Dabanloo, Moharreri et al. 2010, Monson, Wirthlin et al. 2012).

FPGA is a low-energy option (Rezaei, Moharreri et al. 2013, Garcia, Jara et al. 2014), particularly when comparing the power demands of a microprocessor and an FPGA during extensive data processing applications (Khatib, Poletti et al. 2006, Voykin, Bui et al. 2013). FPGAs are well suited for the hardware implementation of real-time signal processing (Frigo, Raby et al. 2010). The implementation of a System-on-Programmable-Chip (SoPC) using FPGA integrates the application functionalities of an end product into a single chip. SoPC designs have been used in telecommunications products and many other electronic devices (Hung, Zhang et al. 2004, Monson, Wirthlin et al. 2012).

Previous work on ECG analysis can be categorized into four types of solutions: (i) hybrid solutions (Shyu and Hu 2007), (ii) System-on-Chip (SoC) solutions (Van Helleputte, Tomasik et al. 2008, Nambiar, K-H. et al. 2012), (iii) handheld device solutions (Khatib, Poletti et al. 2006, Tan, Chang et al. 2011), and (iv) ASIC solutions (Desel, Reichel et al. 1996, Van Helleputte, Tomasik et al. 2008). Classical solutions allow neither patient mobility nor remote analyses, as the devices are plugged in to a wall outlet, therefore requiring an excessive amount of beds in a healthcare center. Hybrid systems apply artificial neural networks (ANN) and wavelet neural network for decision-making. The SoC solution runs 12-lead analyses in a single chip to provide reliable ECG analysis. The commercial solution (Hung, Zhang et al. 2004, Sanchez, Alvarado– Nava et al. 2012) takes 8-input sensor lines, calculates lead signals, and analyzes them in one digital signal processing (DSP) step, but this process is time-consuming. A new solution (Jun and Hong-Hai 2004) uses a flexible SoC that combines the acquisition of multiple physiological signals, such as ECG, electroencephalography (EEG), and respiration signals, with on-chip digital signal processing. Whereas handheld solutions (Tan, Chang et al. 2011) only read and transmit data. This novel solution (Desel, Reichel et al. 1996) uses a portable Linux-based ECG measurement and monitoring system that supports 12-lead ECG data acquisition and remote diagnosis via the Internet. The ASIC solution (Yazicioglu, Kim et al. 2010, Sanchez, Alvarado– Nava et al. 2012) is only used for data acquisition before transmission.

In this paper, a field-programmable gate array (FPGA) design and the implementation of an embedded electrocardiography (ECG) system using System-on-Chip (ECG-SoC) technology are proposed.

PROPOSED SYSTEM

The ECG-SoC Cyclone II FPGA architecture design shown in Figure 1 is designed to enable a noble technology using an Nios II processor, Avalon on-chip communication bus, and Nios II–Linux as the embedded operating system.

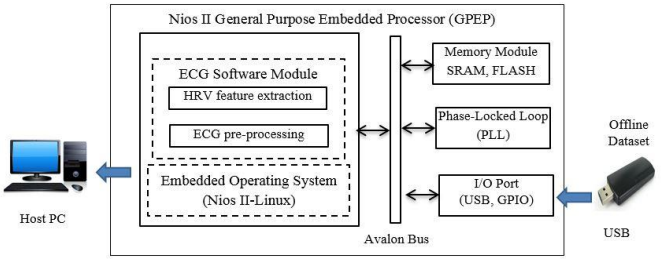

Figure 1: ECG-SoC Cyclone II FPGA.

The ECG-SoC is designed for ECG pre-processing and HRV feature extraction. As a result, rather than using an online data acquisition module, the ECG-SoC will perform computations based on an offline dataset. The ECG-SoC architecture consists of both hardware (HW) and software (SW) partitions. The HW partition consists of memory modules, a phase-locked loop (PLL), and input/output (I/O) ports. The functions of these components, respectively, are to store the image of an embedded operating system that consists of programs, the dataset, and other important files; to control the clock signal of the host computer, the system, and the targeted board; to communicate and transmit data externally; and to accelerate time-consuming operations in the system. Alongside these components, a system bus provides a communication channel according to certain specifications among these components. In this paper we present the development of ECG-SOC system which consists of four phases, which are the development of the hardware system, the Nios II–Linux embedded operating system (OS), ECG-SoC software, and system integration.

Hardware System Development

There are two stages in the development of the HW system. In the first stage, the Nios II GPEP, memory, and other standard peripherals are defined to form a Nios II system module using SOPC Builder (System-on-Programmable-Chip). The configuration includes the reset vector and the exception vector of the Nios II processor, the base address and the interrupt request (IRQ) assignment of every peripheral, and their clock signal source. The SOPC Builder generates process-initiated multiple Verilog HDL (.v) files and the system configuration file (.ptf), which will be used to configure and compile the software module. In stage two, the Quartus II creates a top-level file of the Nios II system module as targeted development board, device, and pin options. The compilation process then begins to perform synthesis, fitting, and timing analysis processes to generate the netlist and HW programming file (.sof/.pof), which is downloaded to the Cyclone II DE2 FPGA board during final system integration.

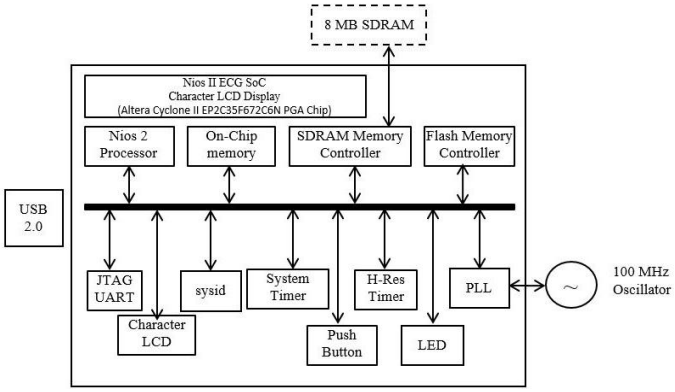

Figure 2: ECG-SoC hardware system configuration

Figure 2 illustrates the configuration of the ECG-SoC system HW architecture. This architecture is used for the entire system of the Cyclone II DE2 FPGA board. The labeled and shaded block with many modules inside depicts the inner side of the chip. These modules are added to the system using SOPC Builder.

On-chip memory, the SDRAM memory controller, and the flash memory controller are the three types of memory modules used. The PLL used in this system design operates using a 100 MHz clock signal, and a high-resolution timer (Hi-Res. Timer) and a system timer are also used for better timing control and to simplify timing issues and the overall board layout. The *sysid*, or system ID, acquires an address specifically for this system design. The I/O port involved in this project is a JTAG UART, and its external peripheral is USB 2. 0.

Nios II-Linux Embedded OS Development

Nios II-Linux is a terminal-based Linux embedded operating system using Debian GNU/Linux 2. 6. 28-rc6, which can run and compile Linux-based programs and libraries. These software tools include a distributed versioning system, relational databases, debuggers, and cross compilers. The cross-compilation process compresses information into one file, called the Linux kernel image and usually named an auto extractable image (*zImage*). The Cyclone II development board is programmed with the ECG-SoC embedded system, and the programming file (.sof) must conform to the same hardware system architecture. To compile the Nios II-Linux kernel of the ECG-SoC, the system configuration file (.ptf) of the targeted hardware system is required to configure the Linux kernel.

ECG-SoC Software Development

The development of the SW partition of the ECG-SoC system involves ECG pre-processing and ECG feature extraction, which are shown in Figure 3. Pre-processing, which aims to improve the signal-to-noise ratio and enhance the accuracy of the analysis and measurement, includes the removal of baseline wander, high-frequency noise, and high-frequency

random noise caused by power line interference (50 Hz, 60 Hz).

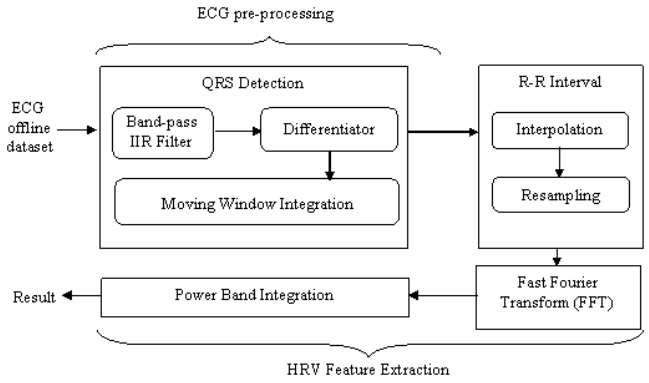

Figure 3: Block diagram of ECG pre-processing and HRV feature extraction

In the pre-processing stage, the QRS detection algorithm is adapted to detect the QRS complex of ECG data to perform HRV analysis. This detection module consists of an infinite impulse response (IIR) filter with a band-pass frequency of 5 to 15 Hz. The filter output is differentiated to provide complex slope information for peak detection. The squaring function is replaced by the absolute function to accelerate computation and toggles all negative peaks. Moving-window integration is then applied to smooth out the data. After QRS complex detection, the ECG signal is transformed into a valley-like signal that has two peaks of significantly different heights. To detect a heartbeat from the ECG signal, the R peaks of the QRS complex are used. In the R-R interval module, the peak of the ECG signal is detected by comparing the current value with the next eight values. Values higher than the next eight values and higher than a set threshold level are treated as the QRS peak. In the next block unit, interpolation is performed between two R-R intervals to obtain continuous R-R interval results from the calculation of the detected peaks. To enable the fast Fourier transform (FFT) process, the QRS peak signals are regularly resampled at 4 Hz by employing linear interpolation.

The next step is to obtain the power spectrum of the R-R intervals. The use of spectral methods in the analysis of the ECG promotes an understanding of the implied behavior of time domain models, especially when linear transformations are involved (Murthy, Haywood et al. 1971). The FFT is performed for both real and imaginary values, all of which are essential for power spectral analysis, as shown by the following Eq. (1).

$$PS = \frac{\sqrt{r^2 + i^2}}{N} \tag{1}$$

Where *PS* is the power spectrum, *r is the* real, *i* is the imaginary, and *N* is for total number of R-R intervals or heart rate (HR) data being fast Fourier transformed.

Based on the spectral diagram of the heart rate, three types of power bands can be derived from the different bands of frequencies. These power bands are very low frequency (VLF: 0.008-0. 04 Hz), low frequency (LF: 0.04-0.15 Hz), and high frequency (HF: 0.15-0.5 Hz). To obtain the HRV of the HF band, the power spectrum is integrated from 0.15 Hz to 0.4 Hz. The following equations (2) (3) (4) are then applied, where HR (*f*) is the frequency spectrum of the heart rate (HR).

$$HRV_{VLF} = \sum_{f=0.008}^{0.04} \frac{|HR(f)|^2}{T} \tag{2}$$

$$HRV_{LF} = \sum_{f=0.04}^{0.15} \frac{|HR(f)|^2}{T} \tag{3}$$

$$HRV_{HF} = \sum_{f=0.15}^{0.4} \frac{|HR(f)|^2}{T} \tag{4}$$

Where *HRV* is the heart rate variability, *VLF* is the very low frequency, *LF* is the low frequency, *HF* is the high frequency, *HR* is the heart rate, *f* is the frequency, and *T* is the time.

The ECG system software is constructed based on appropriate specifications for an ECG-SoC embedded system to provide good ECG pre-processing and HRV feature extraction functionality. The sampling frequency used is 200 Hz with a threshold value of 8 kHz. In this development we use 12 000 ECG data, the detected FFT count of 1024 with 240s FFT window size.

System Integration

The FPGA development board is programmed with the ECG-SoC embedded system, utilizing the programming file (.sof).

RESULTS AND DISCUSSION

During system functionality verification, an offline dataset is stored in a USB portable device and plugged into ECG-SoC application. Figure 4 and Figure 5 show the results of the execution of ECG-SoC in Nios II-Linux.

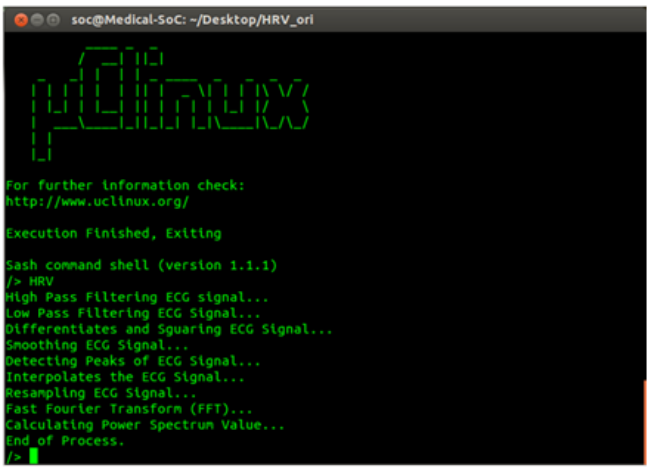

Figure 4: ECG-SoC execution in uCLinux

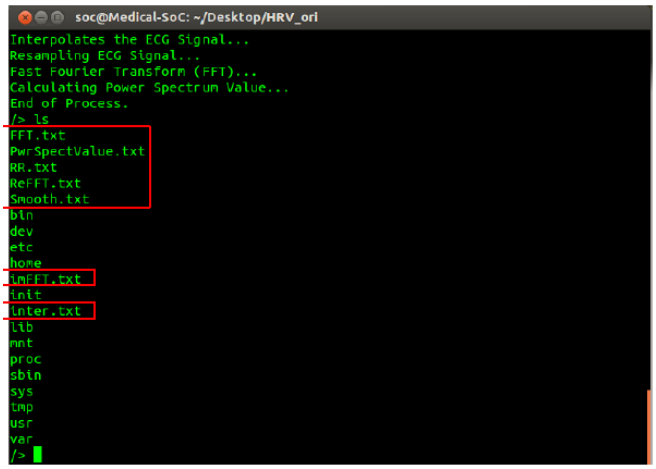

Figure 5: Results file generated from the execution

At this stage, the execution directories show the lists of their contents and will generate results. As the HW accelerator is not included in this project, the software requires a lengthy execution time, which affects the compilation time and Nios II terminal performance. This subsection generates results from ECG pre-processing and HRV feature extraction software. Figure 6 shows the differentiated ECG signals that are obtained after the compilation of the ECG software. Note that each data point is converted to its absolute value to cover all of the negative peaks of the ECG signal. Figure 7 demonstrates the output where squaring operation has been performed immediately before the smoothing operation by moving window integration. The resulting values are larger after the squaring operation and appear as shown above after being integrated to obtain a smooth ECG signal. Figure 8 shows the QRS peak detected by our developed algorithm. In Figure 9, linear interpolation has been applied to the detected QRS peaks for the resampling at 4 Hz. Figure 10 depicts the FFT output consist of both real and imaginary parts. Note that the negative values result when an operation is performed prior to performing the FFT. This first operation includes the subtraction of larger values, resulting in negative values for some numbers. After the generation of the FFT outputs, power spectrum analysis is performed using Eq. (1). Maximum frequency content within 0.01 – 0.1 Hz is measured by analyzing the FFT power spectrum. The inputs used consist of real and imaginary values. The graph produced from the equation is shown in Figure 11. The graph is plotted as the power spectrum (W) versus the frequency (Hz) up to 1 Hz. The red dotted line identifies the maximum range of VLF, the black dotted line shows the maximum limit of LF, and the limit of HF is shown by a green dotted line. From this power analysis graph, we understand that the maximum frequency is 0.03 Hz.

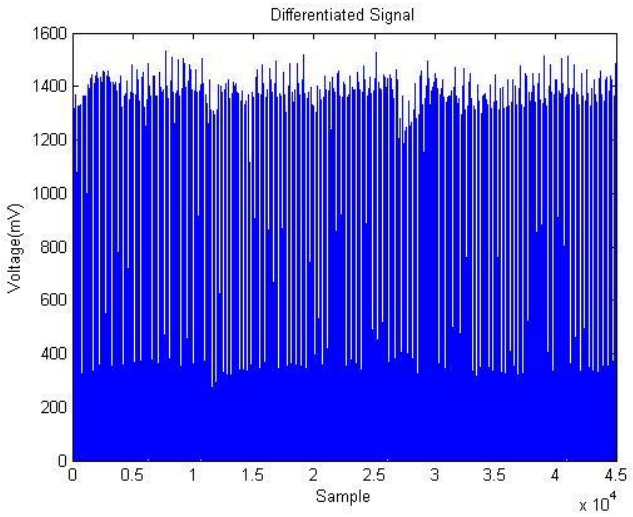

Figure 6: Intermediate results for differential signals

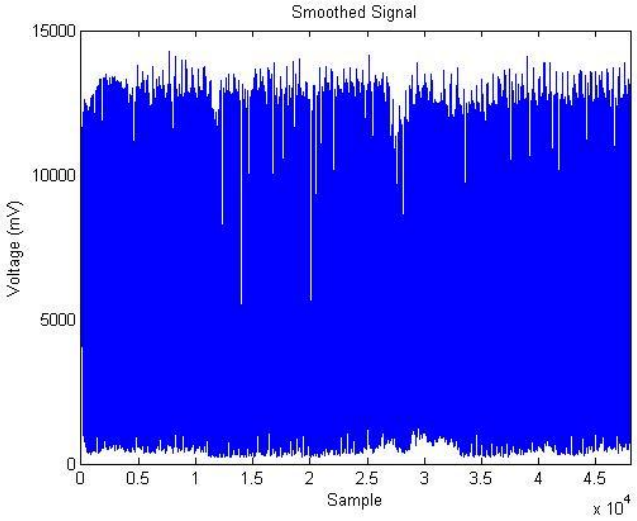

Figure 7: Intermediate results for smoothed signals

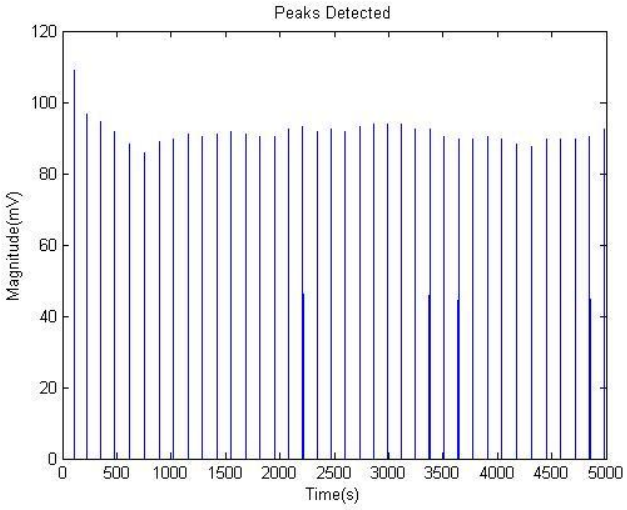

Figure 8: Peak Detection

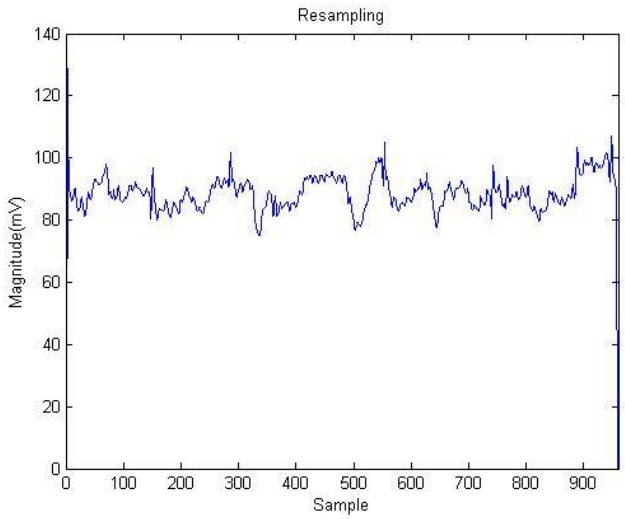

Figure 9: Resampling at 4 Hz by employing linear interpolation

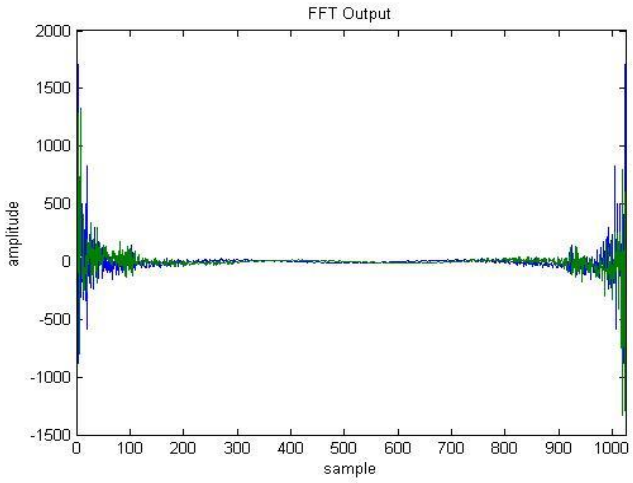

Figure 10: Fast Fourier Transform

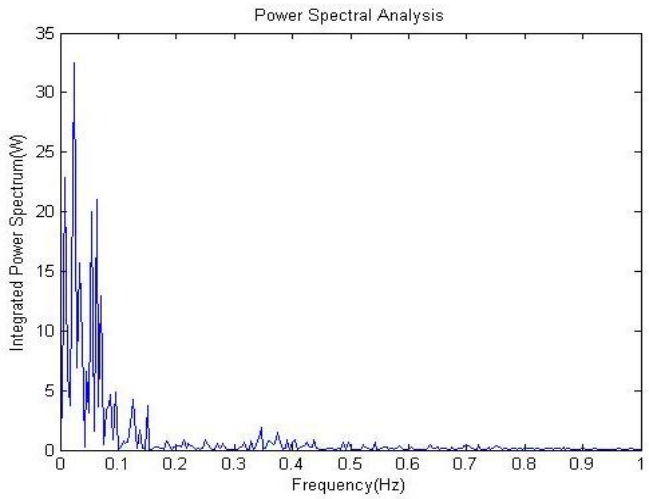

Figure 11: Integrated power spectrum versus frequency

CONCLUSION

This paper presents a SoC based on an embedded ECG system (ECG-SoC) using hardware/software co-design techniques and Altera technology to perform ECG pre-processing and HRV feature extraction based on an offline dataset. The execution of these software functions shows that the ECG-SoC system integrates the functions of ECG dataset acquisition, storage, and processing with integrated hardware and software. The proposed system has the advantage of portability and some software modifications and improvements. The proposed system is able to produce appropriate ECG data analyses in an ECG-SoC design that can be applied to a wide range of cardiac monitoring applications.

ACKNOWLEDGEMENT

The study was supported by National Heart Institute Q. JI30000. 2445. 00G77, and Universiti Kebangsaan Malaysia ETP 2013-078.

REFERENCES

[1] Clifford, G. D. and D. Clifton (2012). "Wireless technology in disease management and medicine." Annu Rev Med **63**: 479-492.

[2] Dabanloo, N. J., S. Moharreri, S. Parvaneh and A. M. Nasrabadi (2010). "Application of Novel Mapping for Heart Rate Phase Space and Its Role in Cardiac Arrhythmia Diagnosis." Computing in Cardiology **37**: 209-212.

[3] Desel, T., T. Reichel, S. Rudischhauser and H. A. Hauer (1996). A CMOS Nine Channel ECG Measurement IC. 2nd International Conference on ASIC.

[4] Fang, W.-C., H.-C. Huang and S.-Y. Tseng (2013). "Design of heart rate variability processor for portable 3-lead ECG monitoring system-on-chip." Expert Systems with Applications **40**(5): 1491-1504.

[5] Frigo, J. R., E. Y. Raby, S. M. Brennan, C. Wolinski, C. Wagner, F. Charot, E. Rosten and K. V. K. (2010). Energy efficient sensor node implementation. 18th annual ACM/SIGDA international symposium on Field Programmable Gate Arrays.

[6] Garcia, G. J., C. A. Jara, J. Pomares, A. Alabdo, L. M. Poggi and F. Torres (2014). "A survey on FPGA-based sensor systems: towards intelligent and reconfigurable low-power sensors for computer vision, control and signal processing." Sensors (Basel) **14**(4): 6247-6278.

[7] Hu, W., C. C. Lin and L. U. Shyu (2011). "An Implementation of a Real-Time and Parallel Processing ECG Features Extraction Algorithm in a Field Programmable Gate Array (FPGA)." Computing in Cardiology **38**: 801-804.

[8] Hung, K., Y. T. Zhang and B. Tai (2004). Wearable Medical Devices for Tele-Home Healthcare. 26th Annual International Conference of the IEEE EMBS San Francisco, CA, USA.

[9] Jun, D. and Z. Hong-Hai (2004). Mobile ECG detector through GPRS/Internet. Proceedings of the 17th IEEE Symposium on Computer-Based Medical Systems (CBMS'04).

[10] Khatib, I. A., F. Poletti, D. Bertozzi, L. Benini, M. Bechara, H. Khalifeh, A. Jantsch and R. Nabiev (2006). A multiprocessor system-on-chip for real-time biomedical monitoring and analysis: architectural design space exploration. Proceedings of the 43rd annual Design Automation Conference (DAC), San Francisco, California, USA.

[11] Monson, J. S., M. Wirthlin and B. Hutchings (2012). "A Fault Injection Analysis of Linux Operating on an FPGA-Embedded Platform." International Journal of Reconfigurable Computing **2012**: 1-11.

[12] Murthy, V. K., L. J. Haywood, J. Richardson, R. Kalaba, S. Saltzberg, G. Harvey and D. Vereeke (1971). "Analysis of Power Spectral Densities of Electrocardiograms." Mathematical Biosciences **12**: 41-51.

[13] Nambiar, V. P., M. K-H. and S. C.W. (2012). Evolvable block-based neural networks for classification of driver drowsiness based on heart rate variability. IEEE Int Conf on Circuits and Systems (ICCAS).

[14] Nangalia, V., D. R. Prytherch and G. B. Smith (2010). "Health technology assessment review: Remote monitoring of vital signs - current status and future challenges." Critical Care **14**(233): 1-8.

[15] Redondi, A., M. Chirico, L. Borsani, M. Cesana and M. Tagliasacchi (2013). "An integrated system based on wireless sensor networks for patient monitoring, localization and tracking." Ad Hoc Networks **11**(1): 39-53.

[16] Rezaei, S., S. Moharreri, H. Ajorloo and S. Salavatian (2013). "Implementation of Heart Rate Variability Signal Processing into FPGA: System on-Chip Design." Computing in Cardiology **40**: 397-400.

[17] Sanchez, A. R., O. Alvarado– Nava and F. J. Z. Martinez (2012). Network monitoring system based on an FPGA with Linux. IEEE Technologies Applied to Electronics Teaching (TAEE).

[18] Shyu, L.-Y. and W. Hu (2007). "Intelligent Hybrid Methods for ECG Classification-A Review." Journal of Medical and Biological Engineering **28**(1): 1-10.

[19] Tan, T. H., C. S. Chang, Y. F. Huang, Y. F. Chen and C. Lee (2011). "Development of a portable Linux-based ECG measurement and monitoring system." J Med Syst **35**(4): 559-569.

[20] Tarasssenko, L. and D. A. Clifton (2011). "Semiconductor wireless technology for chronic disease management." Electronics Letters(Special Supplement: Semiconductor in Personalised Medicine): S30-S32.

[21] Tseng, S. Y. and W. C. Fang (2011). An EKG System-on-Chip for Portable Time-Frequency HRV Analysis IEEE International Conference on Consumer Electronics (ICCE).

[22] Van Helleputte, N., J. M. Tomasik, W. Galjan, A. Mora-Sanchez, D. Schroeder, W. H. Krautschneider and R. Puers (2008). "A flexible system-on-chip (SoC) for biomedical signal acquisition and processing." Sensors and Actuators A: Physical **142**(1): 361-368.

[23] Voykin, A. D., F. M. Bui and R. J. Bolton (2013). FPGA based reconfigurable body area network using Nios II and uClinux. 26th IEEE Canadian Conference of Electrical And Computer Engineering (CCECE), Regina, Saskatchewan.

[24] Yazicioglu, R. F., S. Kim, T. Torfs, P. Merken and C. Van Hoof (2010). A 30uW Analog Signal Processor ASIC for Biomedical Signal Monitoring. IEEE International Solid-State Circuits Conference (ISSCC), , San Francisco, CA.