Body Sensor Networks in Fetal Monitoring with NFC Enabled Android Devices

Michael Korostelev
Temple University, College of Engineering
Philadelphia, PA, USA
mike.k@temple.edu

Li Bai
Temple University, College of Engineering
Philadelphia, PA, USA
lbai@temple.edu

Jie Wu
Temple University, Department of Computer and Information Sciences
Philadelphia, PA, USA
jiewu@temple.edu

Chiu Chiang Tan
Temple University, Department of Computer and Information Sciences
Philadelphia, PA, USA
cctan@temple.edu

Dimitrios Mastrogiannis
Temple University, School of Medicine
Philadelphia, PA, USA
dimitrios.mastrogiannis@temple.edu

ABSTRACT

As many areas of the medical field are becoming more and more interested in telemedicine and outpatient care, we focused on obstetrics and the monitoring of fetal health as an area of medicine that could benefit greatly from *body sensor network* (BSN) technology. In this paper, we explore some of the hardware sensor interface challenges involved with condensing the fetal monitoring procedure into a portable system, specifically for a system integration with a smartphone. There are two contributions of our hardware interface with an Android-based smartphone: i) using IOIO hardware board to prototype an universal interface to *commercial-off-the-shelf* (COTS) hospital-graded medical devices, ii) using Google's *Near Field Communication* (NFC) Android devices for initiating our monitoring mobile application and secure interface hardware devices and a backend *hospital information system* (HIS). We demonstrated this prototype that could integrate with current existing medical monitoring devices and capture BSN information in a seamless manner without cumbersome initialization processes. Also, it reduced the risk that applications can accidentally be initialized and corrupted HIS unintentionally. As a result, the system can be potentially used by a remote monitoring setup without a presence of a medical profession, and even streamline in-patient monitoring process with actual precision for the medical measurements.

Keywords

Android, NFC, Body Sensor Networks, Obstetrics, Gynecology, Fetal Monitoring, Tococardiography

1. INTRODUCTION

Rapid growth of the *body sensor network* (BSN) research area has recently coincided with rapid growth in portable computing in the form of smart phones, mobile devices and tablets. We want to harness the capabilities that are offered with these common *commercial-off-the-shelf* (COTS) devices and apply them for developing wireless BSN's. Android and iOS devices are two major mobile platforms to offer significant computing power, wireless communication, on-board sensors and a platform for user interface. The devices make excellent hosts for telemedicine and mobile health.

Fetal health monitoring (FHM) makes use of cardiotocography (CTG) to record the fetal heart beat as well as uterine contractions with a machine called the *electronic fetal monitor* (EFM)as shown in Figure 1.

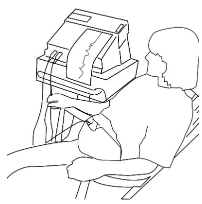

Figure 1: A typical electronic fetal monitoring equipment in a hospital

This equipment is cumbersome, expensive and can only service a single patient at a time. Patients, pregnant mothers with reduced mobility, living in remote areas face hardships with travel and cumbersome equipment is combined with the need for frequent monitoring which can occur on a daily or weekly basis.

To address actual patient monitoring and automatic diagnostic issues, cardiotocography is investigated greatly in the past ten years. Many studies [4] explore ways to better understand monitoring results with machine learning techniques or improving on sensitivity, not many, however, look into novel ways of administering this monitoring. According

BODYNETS 2012, September 24-26, Oslo, Norway

DOI 10.4108/icst.bodynets.2012.249947

to the American College of Obstetricians and Gynecologists (ACOG), electronic fetal heart rate monitoring is increasing in use, *e.g.* 62% of pregnant women in 1988, 74% in 1992, and 85% in 2002 [1]. Even though these numbers look high, the procedure is still largely reserved for higher risk pregnancies, with regular patients only receiving monitoring a few times throughout the pregnancy. Some may even forgo monitoring all together due to limited access to facilities in remote areas.

In recent years, some portable monitoring systems have emerged. Apart from extending the reach of EFM to patients in remote areas and making it more accessible to low risk patients, implementation in mobile platforms allows integration into social networks. Currently social networks [5] largely depend on humans to update and share information but sensor networks can deliver the desired event data in an ambient intelligent fashion.

However, for more practical medical monitoring, earlier systems presented in 2006 [3] only focused on measuring heart rates and developed as supplementary tools rather than substitutions to existing sensing devices. To help maintain the accuracy and reliability of equipment used by obstetricians, the portable system we propose should be used to provide a hardware wireless interface to those existing hospital devices. We can significantly reduce the cost of new devices and minimize "reinventing the wheel" practices to modify existing diagnostic technologies by using these new medical monitoring devices. To be mindful of cost and encourage adoption of the system, we want to utilize as many current hospital-grade sensors as possible and provide means of their integration into a wireless monitoring system.

We focus on the Android platform as the host for the interface, data collection and gateway to the care provider or patient health record system. As a result of using off-the-shelf components, there are security risks. We are challenged with maintaining privacy of the patient while still allowing access to trusted parties such as care providers. We propose to use *radio-frequency-identification* (RFID) integrated with a smart phone to initiate our mobile monitoring application and all secure protocols to the back-end *hospital information system* (HIS).

In this research, we explore challenges involved with scaling down this system with little modification to currently existing medical devices. We propose some solutions to the variety of issues in sensing technology including interoperability, security and user experience that arise with this proposed approach.

The rest of the paper is organized as follows: section 2 presents overall system architecture. The proposed design is given in section 3, then section 4 presents the conclusion and some ideas for future work.

2. OVERALL SYSTEM ARCHITECTURE

Since we are primarily concerned with using readily available hardware, our task is to find components that best accommodate integration into a modular system. The modules we partition the BSN system are as follows: *Base Station* (BS), *Sensor Interface* (SI), and *Sensor Nodes* (SN). A Base Station communicates with the Sensor Interface which in turn communicates with Sensor Nodes. Each communication channel is bidirectional. We consider two models for this system: a) one centralized model in Figure 2, and b) one distributed model.

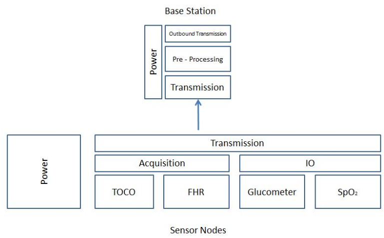

Figure 2: The centralized model for portable EFM integrates individual sensors into the sensor interface device.

2.1 Centralized and Distributed Models

Both configurations have their advantages. A centralized model relieves some of the concerns with power and communication since sensors would be physically tethered to the sensor interface. The transmission of data will be single channel and once a data packet is assembled, no scheduling is required. In order to accommodate scalability and reconfigurability, the sensor interface has to have a number of universal ports that can accept inputs from a verity of sensors. It must then be able to determine the type of sensor that has been plugged in and handle it's data stream appropriately. This is a similar idea to the USB interface in personal computers. The distributed model puts power and transmission components on each distributed sensor node. This method best supports the integration of additional sensors as the care provider may require and allows the system to re-configurable based on patient needs and risk level. The base station device in this case has to be able to properly handle scalablity. It has to detect when a new type of sensor node is used and accept and format inputs from a variety of sensor nodes. The interaction between the base station and nodes follows the message-passing distributed computing paradigm implemented in [6].

Apart from its obvious advantages, the distributed model will create complications in scheduling transmissions and increase cost of individual sensor nodes. On the other hand, the centralized model does not have these issues, but physically limits the number of sensors that can be used.

For our initial studies and to better develop the base station application, we chose the centralized model. At the present stage, the number of sensors is limited to just two, Oxford Instruments TOCO pressure sensor and a 2.0Mhz ultrasound transducer. Both of these are common and in use at hospitals. The following sections will outline system components and describe our integration strategies.

3. PROPOSED DESIGNS AND INTERFACES

There are two contributions of our proposed design and interface with an Android-based smartphone: i) using IOIO hardware board to prototype a universal interface to COTS hospital-graded medical devices, ii) using Google's *Near Field*

Communication (NFC) Android devices for initiating our mobile monitoring application with physical security and secure interface hardware devices and a back-end HIS.

3.1 Android IOIO

In order to conduct sensor readings, I/O capability is required, this is not available on consumer devices like smart phones and tablets. For this an interface for the sensors must be created. Different sensors will have a number of output protocols, even though usually we just need to sense analog voltages, the sensor interface must be able to support a number of methods like *Universal Asynchronous Receiver-Transmitter* (UART), *Pulse Width Modulation* (PWM) and *Serial Peripheral Interface* (SPI). There needs to be an A/D with a high resolution and general purpose I/O for interaction with arbitrary digital input. The IOIO API provides capability to do most of these things, however, we are still challenged with designing analog circuitry to filter and amplify the signals from the ultrasonic transducer.

Android itself supports USB peripherals and accessories. There are two possible modes, *USB accessory* and *USB host*. In USB accessory mode, the external USB hardware acts as the USB host, which gives Android-powered devices[9] the ability to interact with USB hardware. In USB host mode, the Android device acts as the host. The IOIO works in USB accessory mode.

Android IOIO is an I/O prototyping board designed for Android device (OS versions 1.5 and greater). The IOIO board contains a single PIC24 microcontroller that acts as a USB host and interprets commands from an Android app. In addition, the IOIO can interact with peripheral devices in the same way as most MCUs. Digital Input/Output, PWM, Analog Input, I2C, SPI, and UART control can all be used with the IOIO. Code to control these interfaces is written in the same way as standard Android applications with the help of a simple to use app-level library.

The board provides a connection to an Android device via a USB or Bluetooth connection. The IOIO device is able to communicate with the BS either via standard socket through addon modules or Bluetooth by simply replacing the USB cable with a COTS Bluetooth 2.0 module. This is fast enough to support the required sampling rates for obstetrics. (Once every three seconds for heart rate and TOCO pressure) The sensor interface is shown in Figure 3 and includes both sensors, TOCO and ultrasonic, Bluetooth module, IOIO interface board with integration board.

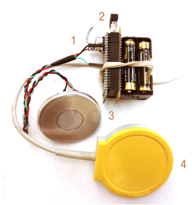

Figure 3: The IOIO (1) coupled with a COTS Bluetooth 2.0 module (2) samples the TOCO sensor (3) and through integration hardware, the ultrasonic transducer(4).

3.2 Android NFC

In most medical applications, system verification is common practice in hospitals and are important as physicians and nurses need to authenticate themselves with identification tags before data collections can be initiated and stored in HIS. For our proposed system, there is a potential drawback for this security concern. Also, the challenge of keeping information secure exists when a password is easily distributed by word of mouth and electronic means. Furthermore, it is unfeasible to require patients to remember a password if we want to extend the system to some patients with possible memory loss, Alzheimer's or other disabling conditions. The monitoring problem can occur if the software application is initiated accidentally or system can lock out patient monitoring when multiple passwords were entered incorrectly. Without proper verification procedure, the back-end HIS can be easily and unintentionally corrupted with unreliable information. To address this problem, we propose the use Android NFC technologies to reduce the risk.

For our previous work in a mobile transit payment project [2], we used mobile devices and RFID to verify passenger information. As NFC becomes a popular choice for mobile transit payment technology, we could use NFC as a verification system to initiate our mobile monitoring software application and determine the process the flow of the monitoring procedures.

We can store an encryption key on the card to verify the patient's identify. When a NFC enabled smartphone is in proximity of an authenticated patient's contactless smartcard, our mobile monitoring application is initiated. Otherwise, the application cannot be started. After the application starts, the monitoring process can be text-synthesized and direct the patient to start a monitoring process.

Since each patient may have different monitoring needs, information can be encrypted on the card and direct the patient with proper monitoring process. Also, a number of medical devices and their calibration information can be stored on the card since some of that information can be patient specific (e.g. ultrasonic levels for obese patients). In addition, the smart-card can initialize all connections to secured HIS for proper logging information.

The system can also provide flexibility according to patient needs. For example, one patient may require 120-minutes of monitoring, and another patient may just do a 20-minute check up. The contactless smart-card can reduce the cumbersome processes, and automate redundant tasks through the information on the RFID card. After completion, the smartphone will terminate the process and halt until data is securely transferred into HIS and acknowledged by the HIS.

In order to best evaluate the methods to for implementation of both the IOIO accessory board and RFID security, we must consider a potential use case scenario for the portable EFM system, as shown in Figure 4. Knowing how an application is to be used allows us to combine the technologies into a functioning application.

In the next section, we discuss Android programming and the structure for an application that follows this use case.

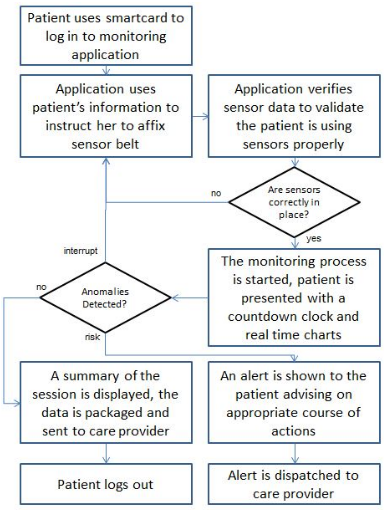

Figure 4: A use case for a portable EFM system

3.3 Android Development and Implementation

The Android platform is at the center of our portable EFM. In order to develop software applications for the mobile *operating system* (OS), we need to understand how applications must be structured. With an Android [7] system, applications are built composed of *activities*. They are components of the complete application and provide processes for user interaction. Most applications can consist of a number of activities. These activities are in turn composed of smaller building blocks called *views*. Views host components for user interfaces such as buttons, text fields, and graphics.

Through collaboration with medical professionals in obstetrics and gynecology, we have developed a list of requirements for a base station application and user interface.

A snapshot of the application is shown in Figure 5.

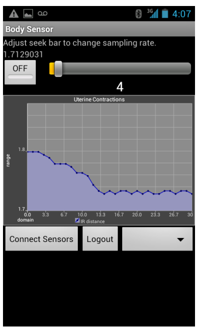

Figure 5: The EFM Monitoring Activity interface

4. CONCLUSION

The ultimate goal of this project is to provide obstetricians and gynecologists with a new portable wireless fetal heart monitoring system that will improve accurate, reliable and secure data collection. The proposed system will provide medical professions with up-to-the-minute information and aggregates data with existing diagnostic systems to notify physicians and patients with rapid feedback if abnormal activities were discovered. As mentioned earlier, our design goal was to develop a system that will enable patients with an easy monitoring in the comfort in their homes efficiently. The proposed system, if fully utilized, should be able to replicate the data collection as the same in the hospitals, with the exception that this system will be more efficient and provide valuable logistical monitoring and care to our patients. In the future, we hope to incorporate with a distributed model so that we can eliminate all hardware interfaces required to the IOIO device. As a result, it reduces the mistakes to connect a medical monitoring device to a wrong interface which can potentially cause damages to the device.

5. REFERENCES

[1] L. Coughlin, A. Huntzinger, "Practice Guidelines Briefs", *American Academy of Family Physicians*, 2005 Aug 1;72(3):527.

[2] Li Bai; Kane, G.; Lyons, P.; , "Open architecture for contactless smartcard-based portable electronic payment systems," Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on , vol., no., pp.715-719, 23-26 Aug. 2008

[3] Jianfeng Chen; Koksoon Phua; Ying Song; Louis Shue; , "A portable phonocardiographic fetal heart rate monitor," *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on* , vol., no., pp.4 pp., 21-24 May 2006 doi: 10.1109/ISCAS.2006.1693041

[4] R. Sameni and G.D. Clifford, âĂIJA review of Fetal ECG Signal Processing: Issues and Promising DirectionsâĂİ, *The Open Pacing,Electrophysiology & Therapy Journal*, vol. 3, pp. 4-20, 2010.

[5] Rahman, M.A.; El Saddik, A. ; Gueaieb, W., "Data visualization: From body sensor network to social networks", *Robotic and Sensors Environments, 2009. ROSE 2009. IEEE International Workshop on*, pp: 157 - 162 , 6-7 Nov. 2009

[6] Mi Zhang, and Alexander A. Sawchuk, "A Customizable Framework of Body Area Sensor Network for Rehabilitation", *Applied Sciences in Biomedical and Communication Technologies, 2009. ISABEL 2009. 2nd International Symposium on* Page(s): 1 - 6, 2009

[7] Z. Mednieks, âĂIJProgramming Android,âĂİ OâĂŹReilly Media, ISBN: 1449389697, 2011.

[8] S. van Delden and A. Whigham. "A Bluetooth-based Architecture for Android Communication with an Articulated Robot." *International Conference on Collaboration Technologies and Systems - Collaborative Robotics and Human Robot Interaction.* Denver, Colorado. May 21-25, 2012.

[9] Hui-Kang Teng; Cheng-Min Lin; Kuo-Chen Lang; Yi-Lun Chu; Shou-Jie Chen; , "An Implementation for Connecting Android Powered Device to External World," *Innovations in Bio-inspired Computing and Applications (IBICA), 2011 Second International Conference on* , vol., no., pp.303-305, 16-18 Dec. 2011 doi: 10.1109/IBICA.2011.80

[10] Ytai Ben-Tsvi, "IOIO Documentation", available at https://github.com/ytai/ioio/wiki/IOIOLib-Basics

[11] "MF1ICS50 Functional Specification, Product Data Sheet" NXP Inc., January 2008