Design and implementation of a wearable healthcare monitoring system

Article *in* International Journal of Electronic Healthcare · February 2009

DOI: 10.1504/IJEH.2009.026273 · Source: PubMed

Assim Sagahyroon
American University of Sharjah

Ali Ghazy
American University of Sharjah

Design and implementation of a wearable healthcare monitoring system

Assim Sagahyroon*

Computer Engineering Department,
American University of Sharjah,
Sharjah, UAE
Email: asagahyroon@aus.edu
*Corresponding author

Hazem Raddy

CCC (Consolidated Contractors Company),
Dubai, UAE
Email: HRaddy@ccc.ae

Ali Ghazy

Institute of Applied Technology,
Dubai, UAE
Email: ali.ghazi@iat.ac.ae

Umair Suleman

Aim 168 Inc.,
Dubai, UAE
Email: admin@aim168.com

Abstract: A wearable healthcare monitoring unit that integrates various technologies was developed to provide patients with the option of leading a healthy and independent life without risks or confinement to medical facilities. The unit consists of various sensors integrated to a microcontroller and attached to the patient's body, reading vital signs and transmitting these readings via a Bluetooth link to the patient's mobile phone. Short-Messaging-Service (SMS) is incorporated in the design to alert a physician in emergency cases. Additionally, an application program running on the mobile phone uses the internet to update (at regular intervals) the patient records in a hospital database with the most recent readings. To reduce development costs, the components used were both off-the-shelf and affordable.

Keywords: electronic healthcare monitoring; wearable electronics; sensors; microcontroller; Bluetooth; sequence packet protocol; mobile application.

Reference to this paper should be made as follows: Sagahyroon, A., Raddy, H., Ghazy, A. and Suleman, U. (2009) 'Design and implementation of a wearable healthcare monitoring system', *Int. J. Electronic Healthcare*, Vol. 5, No. 1, pp.68–86.

Biographical notes: Assim Sagahyroon is an Associate Professor with the Computer Engineering Department at the American University of Sharjah. He earned his Masters degree form Northwestern University, and a PhD form University of Arizona. Before joining the American University of Sharjah, he was with the California State University. He worked in industry for IBM, Lucent and Zhone Technology. He worked as a Reviewer for the NSF, and various international conferences and journals. His current areas of research include engineering and FPGA-based applications, and power consumption in VLSI circuits.

Hazem Raddy has earned a BSc degree in Computer Engineering from the American University of Sharjah in 2006. He currently works as a System Engineer with the Consolidated Contractors Company (CCC) working in mobile technology and its applications in the construction field.

Ali Ghazy has earned a BSc degree in Computer Engineering from the American University of Sharjah in 2006. He is currently working on a Masters degree in Mechatronics while working as a Lab Engineer for the Institute of Applied Technology in Dubai.

Umair Suleman holds a BSc degree in Computer Engineering from the American University of Sharjah. Upon graduation he worked in the IT side of the banking industry. He recently founded his own company (www.aim68.com) working on the portal development segment of the technology.

1 Introduction

In recent years, health monitoring has emerged as a viable and practical application field for wearable electronics. A wearable system can be broadly defined as a mobile electronic device that can be unobtrusively embedded in the user's outfit as part of the clothing or as a wearable accessory (Lukowicz et al., 2002). These systems can be used for health monitoring of both the elderly and sick, and have a huge potential in supporting an independent life for the two groups. Also, advancements in sensor technology and mobile communications give opportunities to the design and implementation of tools which enable extended independent living at home and improvement of quality of life for individuals. Some of the benefits of these wearable health monitoring systems include (Jovanov et al., 2004) the following:

- Patients can benefit from continuous monitoring as a part of a diagnostic procedure, optimal maintenance of a chronic condition or during supervised recovery from an acute event or surgical procedure.
- Timely warnings can be issued to the patient, and a specialised medical response service can be activated in the event of medical emergencies.
- Continuous monitoring with early detection likely has the potential to provide patients with an increased level of confidence, which in turn may improve quality of life.

- Ambulatory monitoring will allow patients to engage in normal activities of daily life, rather than staying at home or close to specialised medical services. This would allow the patients to go ahead with normal lives and still be taken care of via the wearable health monitoring systems.
- Wearable computers or devices have the advantage of going where a laptop cannot.

While several 'intelligent homes' and architectures to support elder-care have been documented, many of these have focused on the support of leisure activities, home automation and support for communication. Very few systems that support the home life and healthcare of older and sick people have been developed to improve quality of life and alleviation of risk (Perry et al., 2004). The need for the application of the recent advances in technology to promote healthy ageing was argued by Intille (2004). The authors put out the argument that the rapid adoption of mobile computing and the emergence of context aware computing are two ubiquitous computing trends that can be utilised efficiently to deploy applications that are greatly needed in the field of preventive healthcare.

Some of the proposed approaches to the design of these wearable monitoring devices included the use of broadband communication networks in providing homecare services. Guillen et al. (2002) discuss the design and implementation of tele-home-care multimedia platform that runs over Integrated Services Digital Networks (ISDN) and internet protocol using videoconferencing standards H.320 and H.323, and standard TV sets for patient interaction. Their platform allows for online remote monitoring of heart sound and blood pressure. A wearable point of care system that uses the IEEE 1073 Medical Information Bus (MIB) standards is discussed by Yao et al. (2005). The paper addresses the usability issues where standards are applied for plug-and-play devices, and it describes a design approach to a three level point-of-care system that uses Bluetooth and MIB. A wearable multi-parameter physiological monitoring system for space and terrestrial applications was described by Mundt et al. (2005). The authors argue that despite recent advances in medical technology, none of the available devices provides a combination of wearability, size and functionality that satisfied NASA requirements. Fadlee et al. (2005) have discussed the design of a processor that samples signals from sensors on a patient and then transmits digital data over a Bluetooth link to a mobile telephone that uses GPRS (General Packet Radio Service). The essence of their research was to demonstrate the future potential of mobile communications in telemedicine. An architectural framework of a system that utilises mobile techniques to wirelessly monitor the ECG of cardiac patients was discussed by Sneha and Varshney (2007). Thulasi and Srivatsa (2007) described the design and implementation of a wearable cardiac telemedicine platform. Their system was intended for use by patients who survived myocardial infarction and needed post-surgery monitoring without hampering their daily life by being confined to medical institutions.

This paper is organised as follows: in Section 2, we provide the readers with an overall view of the proposed system, its design and the interaction between its various building blocks. The mobile application program developed for this project is discussed in Section 3. The paper is concluded in Section 4.

2 System design

The primary objective of this work is to develop a light weight, wearable healthcare monitoring system to assist patients in monitoring their own conditions and also to allow physicians to care for the well-being of their patients without restricting them to the inside of a medical institution or a healthcare facility. To reduce development costs, the components used were both off-the-shelf and affordable. This work also focuses on the advantages offered by integrating front-end physiological parameter extraction devices with mobile technology while making use of industry-standard networking technologies.

2.1 Overall system architecture

A high-level illustration of the system is depicted in Figure 1. A microcontroller-based unit collects and processes data from the sensors that are attached to the patient's body to measure vital signs such as the patient's heartbeat and temperature. Using a Bluetooth connection between the microcontroller and a PDA, this data is then transmitted and stored in the memory of a PDA carried by the patient. After the PDA unit receives the data, two things take place: first, the data (readings) are displayed on the PDA screen in a format that is understandable by the patient, and secondly, an application program running in the PDA analyses the readings to assess if any of the different values read is above a certain threshold value (this value is set by the patient's physician). If this value is exceeded, an alert SMS is immediately sent via the GSM (global system for mobile communication) network to a physician's mobile informing him or her of the patient's current critical readings. The doctor can then call the patient and instruct him on what are the immediate and necessary actions to be taken till help arrives. The PDA unit also updates (at regular intervals) the patient's record in a hospital database. This is accomplished using an internet connection via the GPRS network. A protocol was developed to provide for secure communication amongst the units and to allow for secured and confidential communication of patients' data.

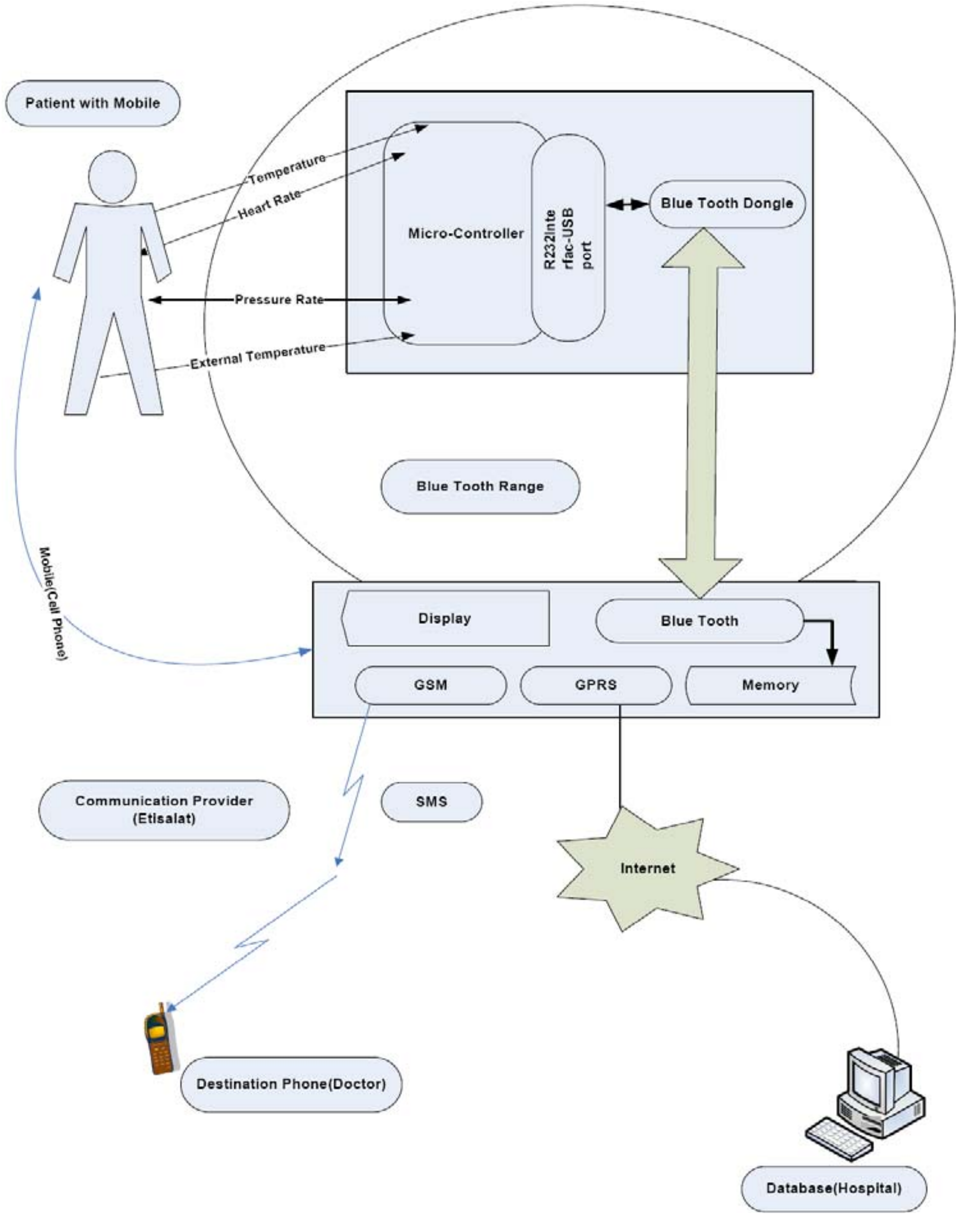
2.2 Microcontroller – Bluetooth – PDA connectivity

2.2.1 Bluetooth connection

In the implementation phase, we selected the MiniDragon + microcontroller platform due to its compact size (6 × 8 cm), reduced cost, reduced power consumption, as well as other various desirable features. It uses the 9S12DP256 compact microcontroller designed using a 16-bit RISC-based architecture (Freescale Semiconductors, 2008). It has 256 Kb of Flash EEPROM, 12.0 Kb of RAM, 4.0 Kb of EEPROM, 2 asynchronous Serial Communications Interfaces (SCI), three Serial Peripheral Interfaces (SPI), an 8 channel IC/OC enhanced capture timer, two 8-channel, 10-bit Analog-To-Digital Converters (ADC), in addition to some other salient features that makes it quite suitable for this project.

Bluetooth was the technology of choice due to the fact that it satisfies critical objectives in the intended application: (i) it is ideal for small distance data links, (ii) it is increasingly becoming a common feature in most mobile units and (iii) the security features it offers are an added advantage to our system since it allows for the encryption of the data transferred between the controller and the mobile device.

Figure 1 System overview (see online version for colours)



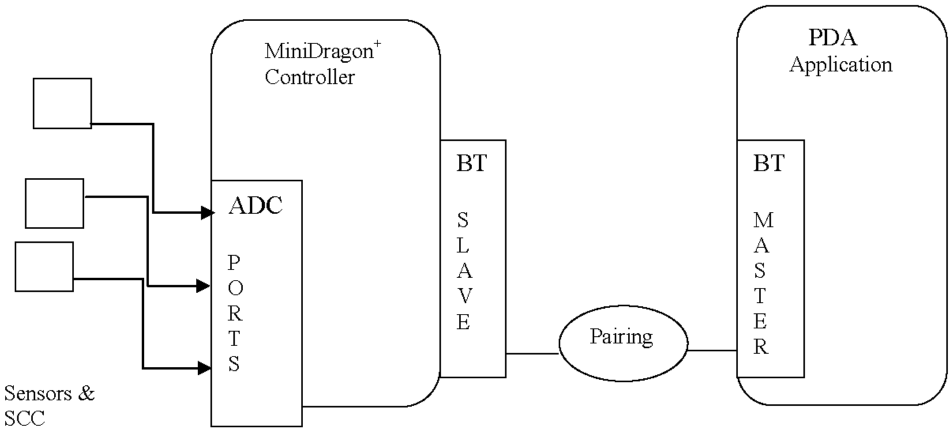
Since the microcontroller unit *does not* have a built-in Bluetooth interface, a Blueport device acting as a Bluetooth Serial Adapter (Grid Connect Inc., 2008) was used to allow for the needed communication between the microcontroller and the patient's PDA. By connecting this device to one of the microcontroller serial ports and properly configuring it, the microcontroller will be able to transmit and receive data via a Bluetooth connection as required by the proposed design approach in this work.

The Blueport device used comes with five modes of operation (Grid Connect Inc., 2008; SBS, 2001) slave, master, auto-connect, auto-discovery and command mode. In this implementation, we use two modes, namely, command and slave modes. The following instructions are used to configure the Blueport device while on command mode:

SU,9600	*(sets UART Baudrate to 9600)*
SN,myname	*(sets Bluetooth name to 'myname')*
SA,1	*(enables secure authentication and encryption)*
SP,secret	*(sets security pincode to 'secret')*
SF,1	*(restores all values to factory defaults)*

A block diagram depicting the connectivity between the different components is shown in Figure 2.

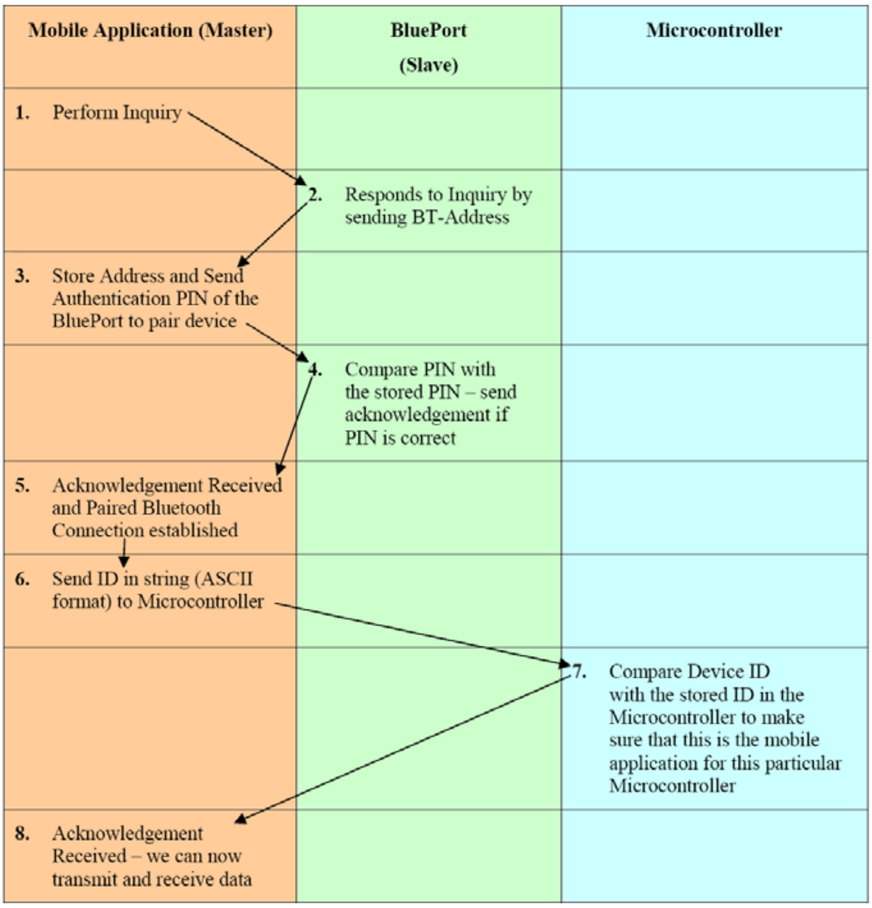
Figure 2 Connectivity layout



Data from the sensors and Signal Conditioning Circuitry (SCC) is converted to its digital equivalent using ADC converters, and then transmitted to the PDA via a Bluetooth connection. Prior to this, the Bluetooth Master and Slave must bond or pair. After setting the Blueport device in the 'command mode' for configuration purposes, it is next set to operate in the 'slave' mode with the mobile application running on the PDA acting as the master station. Communication between the two (master and slave) follows a protocol developed for this project. The protocol follows the sequence shown in Figure 3.

In the proposed protocol sequence, the mobile application running on the master (PDA) side activates and instructs the Bluetooth manager to search for slaves. The master keeps comparing addresses of discovered devices till it finds the address of the BT device connected to the microcontroller. When the address is received, the application then requests connectivity using an agreed upon authentication 'pin code'. The pin code is used for security purposes and to ensure connectivity to the right slave. Following authentication, a paired Bluetooth connection will be established between the Blueport device and the PDA. The above sequence consists of Steps 1 to 5 of the protocol as illustrated in Figure 3. After the pairing is complete, a handshaking procedure (Steps 6, 7 and 8) to initiate the actual data transfer takes place between the application running on the PDA and the microcontroller. These last three steps will be discussed in a later section.

Figure 3 Master-Slave BT connection sequence (see online version for colours)



2.2.2 *Software implementation*

After the pairing between the Bluetooth components as described in the previous section; the application program running in the PDA sends and receives data from the microcontroller through virtual serial ports created using SPP (Sequence Packet Protocol).

Microsoft Visual Basic.net Compact Framework was used to develop the application program running on the PDA. The PDA used in this work is the i-mate PDA2k (imate, 2006). It is based around the Intel XScale PXA263 processor running at 400 MHz and Windows Mobile 2003 Second Edition for Pocket PC Phone Edition. It has 128 MB of RAM, and has a 64 MB ROM. To have full access and control of the Bluetooth features available on the PDA, we needed to import an external library for visual basic. The library components are compatible with the Microsoft Bluetooth Stack (WindComm). The Bluetooth stack is basically a reserved area in memory used to keep track and carryout the internal operations of a Bluetooth connection which includes supporting different protocols and providing services for each protocol. The imported library functions allowed us to access services offered by the SPP within the Bluetooth stack and hence the ability to discover devices and connect to other devices within our application. In essence, SPP is used to achieve a cable replacement solution.

The Sequence Packet Protocol provides reliable, flow-controlled, two-way transmission of data. It is a byte-stream protocol used to support streaming data. Sockets using the SPP protocol are either 'active' or 'passive'. Active sockets initiate connections to passive sockets. Passive sockets may 'underspecify' their location to match incoming connection requests from multiple networks. This allows a single server to provide service to clients on multiple networks. The SPP port may still be specified at this time; if the port is not specified the system will assign one. Once the port is assigned it can be used to send and receive data between the connected devices (SPP, 2008).

The *SPPConnect* command was used to connect to the devices found during the discovery mode. This command is used to initiate a connection with the specified device. It follows the syntax below:

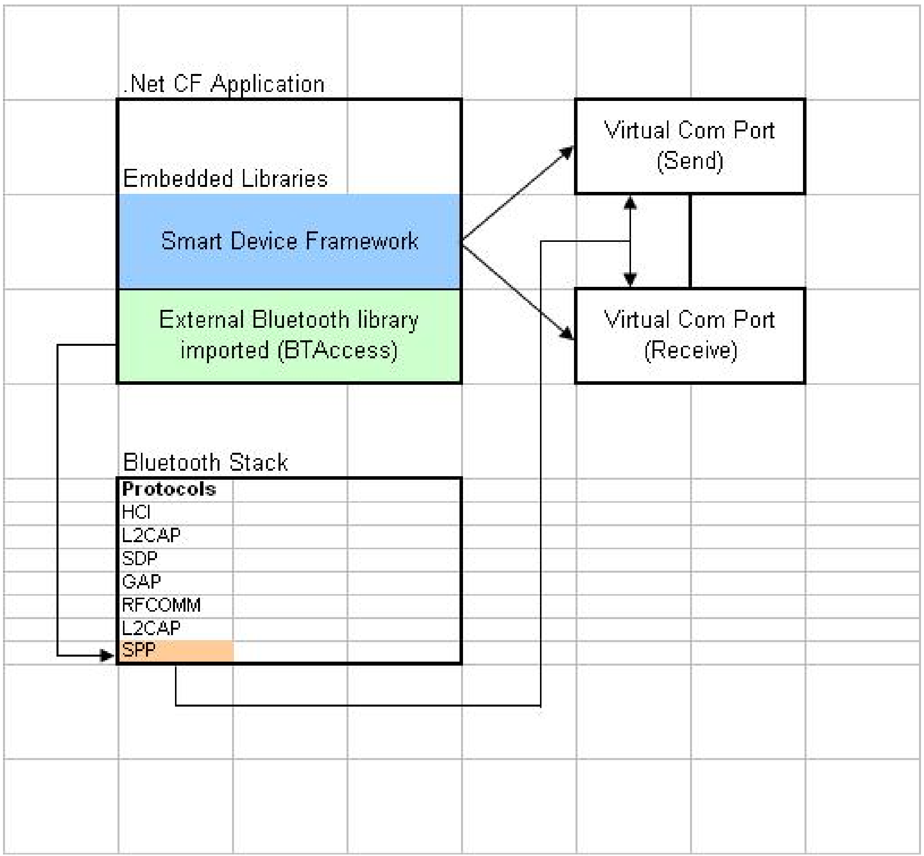
SPPConnect [BD Address] [Service]

The remote BD address must be specified. The remote service is optional. If not specified, the first registered SPP service will be used by default.

After using the above-mentioned function to connect to a selected device, the SPP library then creates a communication port for the PDA which allows the mobile application to send and receive data through it. It basically creates a virtual port as if it was a physical port and hence allows the PDA application to communicate with external devices wirelessly via a Bluetooth-based connection.

Following the creation of the virtual port, we now need to access this port from the PDA application. At the time of implementation of this project, there was no library within the Microsoft.NET Compact Framework to facilitate this access and hence, we had to use an external library as explained earlier. This external library is part of the Smart Device Framework which we used as well in building other parts of the application. Once a link is established with the Virtual Com Port, the process becomes straight forward. Data is transmitted and received between the Mobile Unit and the controller via Bluetooth. Figure 4 illustrates the relationships and interactions between the various modules and components.

Figure 4 Mobile unit preparation for bluetooth connectivity (see online version for colours)



3 The mobile application

The application that runs on the mobile unit controls and supervises the operation of the system. As indicated earlier, Microsoft Visual Basic .net (VB.net) Compact Framework was used in the development of this application. VB.net Compact Framework allows developers to create applications using components and class libraries from the .net framework. Most of the components required to develop the application program for this work were available from the standard .net Compact Framework library, except for the components and libraries needed for Bluetooth interface and connectivity, etc., which were imported. We also made use of the Smart Device Framework to enrich and extend the .net CF capabilities. In particular, we used the Smart Device Framework to implement the following functions:

1 access of the virtual serial port
2 entry of a text file that contains patients records into a registry
3 running of the application in the background.

In general, the services provided by the application running on the PDA can be divided into three categories: administrative, patient related and communicative. These services include the following:

- initial application parameters settings
- password control
- graphing capabilities
- sending and receiving sensor data from Microcontroller via Bluetooth
- storing all the values read until they are uploaded to the internet – in a text file stored in the System Registry (implemented using Smart Device Framework)
- running of the application throughout in the background (implemented using SmartDevice Framework)
- sending of an SMS to physician in case of an emergency (implemented using SmartDevice Framework)
- uploading of data at the end of a specified time period (implemented using SystemNet component using HTTP Web Request and HTTP Web Response)
- modification of threshold values, SMS contact number and frequency of online updates
- access of database
- Active Server Pages (ASP) query strings used for uploading data to a server.

A brief description of the aforementioned tasks and features is given hereafter.

3.1 Initial settings (entered by the administrator)

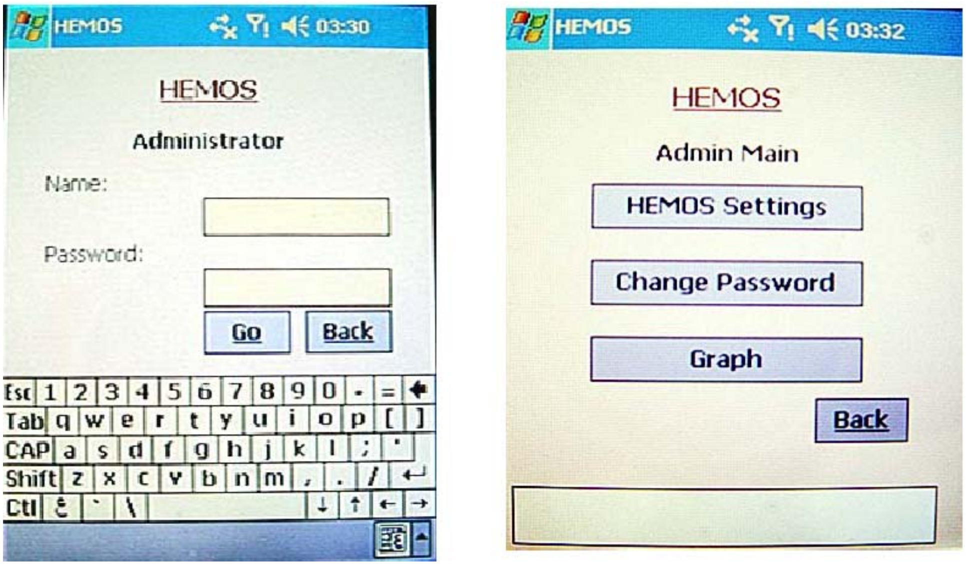
Initially, the administrator will have to enter some parameters that depend on the patient's condition, physician recommendation, etc. These settings include the following:

- Patient Information, used by the application to format the SMS message to be sent and to identify the patient when updating the database.
- SMS number of doctor and nurse; used by the application to send an SMS message if a critical situation is reached.
- Threshold (Limit) values, used by the application to compare with the values retrieved from the microcontrollers (sensors). This helps the application decide if the situation is critical and warrants the transmission of an SMS message to the physician.
- Frequency of online updating, used by the application to set a timer of the specified frequency or time so that when it triggers, it activates or fires an updating process of all of the values being read from the microcontroller up to that point in time.
- Frequency of reading from sensor, used by the application to set a timer of the specified frequency or time so that when it triggers, it activates or fires a process of reading or requesting values from the microcontroller.

3.2 Password control

This feature allows the administrator to change his or her own password if he or she chooses to. The password is stored in a registry. The application retrieves it when checking if the password the administrator had entered is correct, or not. It displays an ‘accepted’ message for a match with the stored password, and a ‘denial of access’ message if different. Figure 5 shows PDA screen captures reflecting an administrative activity.

Figure 5 Administrative-related screens (see online version for colours)

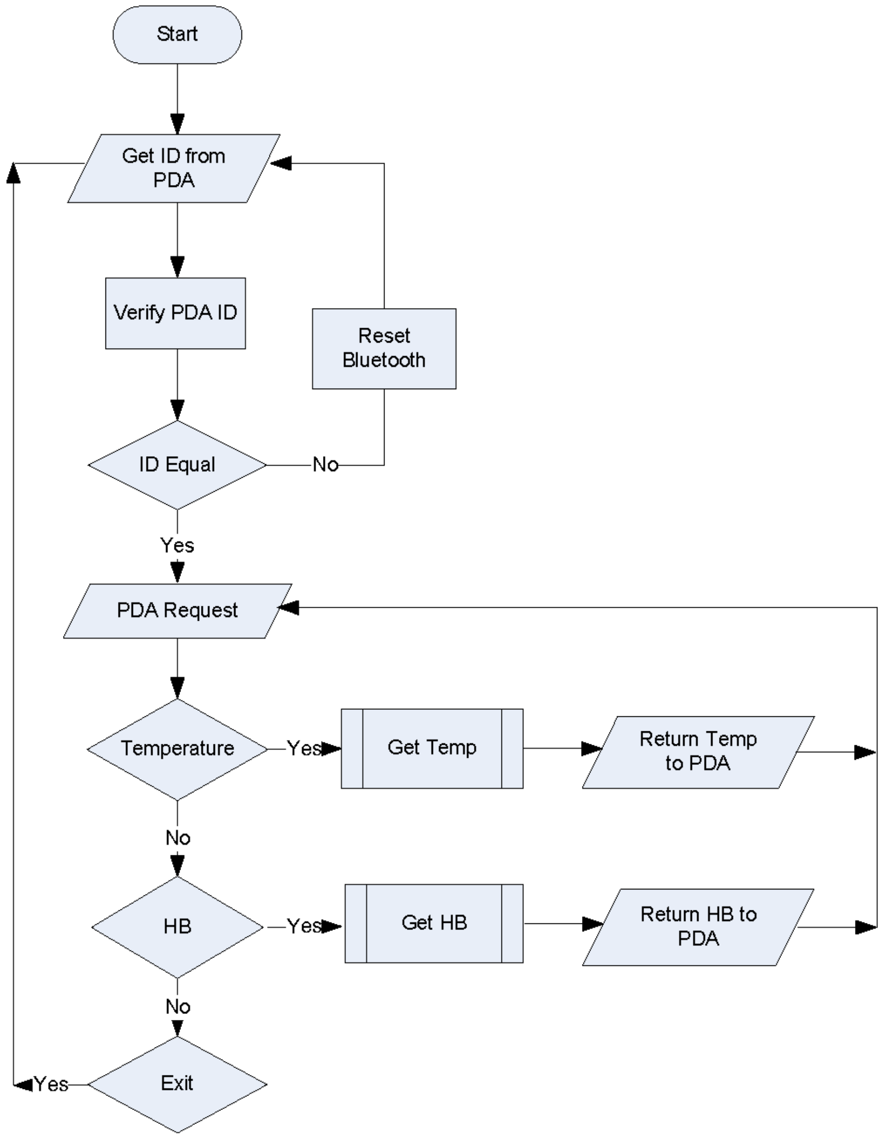


3.3 Transmission/reception of sensor data

When a reading timer is triggered, it fires or activates a reading process for obtaining and recording the readings from the sensors via the microcontroller. The connectivity between the application and microcontroller uses the Bluetooth protocol. Therefore, there has to be a master and a slave. In this implementation, the master is the PDA, and the slave is the microcontroller side. During the reading process, the application activates and instructs the Bluetooth manager residing in the PDA to search for slaves. By address and name search, the application keeps on comparing the found devices until it finds the right one. When the Bluetooth device is found, the application issues a connectivity request. The entities then cycle through Steps 1 to 5 of the protocol sequence described earlier (Figure 3) until a bonding connection is established. After the pairing between the Bluetooth devices is complete, a handshake procedure follows. First, the application sends its own ID, which is an alphanumeric number to the microcontroller (Step 6 of the protocol). The microcontroller checks if the ID is the one that it should connects with. If true, then the microcontroller sends back its own ID, which is also alphanumeric number, to the PDA (Step 7). This ID is received by the application on the PDA that in turn checks it to see if it is the right device. If it is, the application requests and initiates the reading process (Step 8), else the application terminates the connection.

Figure 6 shows the steps that take place during a read sequence. The application sends its identification to the microcontroller. The controller compares the received identification against a pre-stored ID to confirm that this is the ID of the application that it should communicate with and sends back and acknowledge with its own ID. If all is correct, the microcontroller will wait for the PDA to make a request. The PDA application can then either request the patient's temperature or heartbeat, for example, or exit, and terminate the connection. If the PDA is requesting a temperature read, the controller will then initialise the appropriate ADC, read the temperature value and sends it to the PDA. On the other hand, if the PDA requests to end the session, the microcontroller will reset the Bluetooth connection waiting for the start of a new session.

Figure 6 Data reading protocol (see online version for colours)

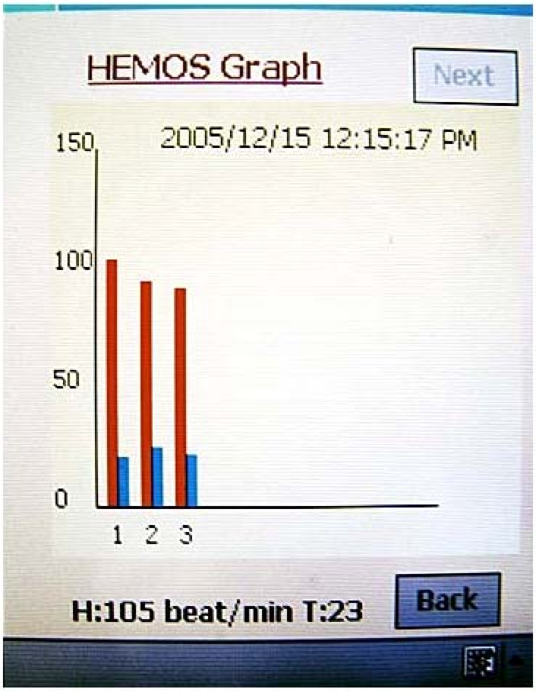


3.4 Patient-related data

This feature provides the patient or user with ability to request and read the sensor readings on his/her PDA in an understandable format. It is similar to the reading process that is programmed or set on a scheduled basis set by the administrator. The only difference is that this is an immediate reading requested by the user or the patient.

Additionally, there is a *graphing* feature that allows the patient or user to view the values recorded from the sensor via the microcontroller as a graph of blocks. On the y-axis, the range is from 0 to 150 (heart beat rate axis in) and on the x-axis, order of the recording event [first recorded reading (1), second recorded reading (2)…etc.]. The patient or user can choose any of the graphed readings and expand it by clicking once on it to display on the top and lower part of the graph: time this reading was recorded and the values being read respectively. A sample graph is shown in Figure 7.

Figure 7 HEMOS sample graph (see online version for colours)



3.5 Readings storage

Each time a reading is retrieved, it is stored in a text file in the windows directory.

Each reading has a format similar to the one below:

'Heart Rate: 105 Temperature: 24 Recorded Time: 2005/12/09 05:15:35 AM'

The application keeps on writing and saving to this text file until an update timer triggers, signalling the start of an update process. In this process, the application reads the records from the text file one-by-one, uploading it to the database on a web server using ASP.

When the process is complete, the application program recreates a new text file in preparation the next new readings. In addition to this text file, the application creates an accompanying text file which acts as a system log. Activities taking place within the application such as reading, checking, sending and updating events are recorded in this system log text file with success and failure rating.

3.6 Data upload

The sensor readings are stored in a text file that resides in the registry of the PDA. When an updating timer triggers the uploading of this data commences. The steps that take place thereafter are summarised below:

- The application on the PDA uses a built in property called 'HTTPWebRequest' to find any possible connection to the internet via GPRS or a wireless network
- The recorded readings are saved in a text file in the window directory; next, the application opens and reads this text file retrieving the readings one at time. Each time it formats a string in an ASP format link as shown below:

  'www.bikezunlimited.com/check.asp?PID=6666&HR=80&T=24&Time=2005/12/09 04:40:20 AM'
- The application then uses the established internet connection to update a database on the web server. As soon this link is opened or sent to the server, the ASP, which is in this case 'check.asp', in return takes this link and parses it into a meaningful format. For example, the link shown above when parsed read as follows:

  Patient ID=6666 from (PID=6666), Heart Rate = 80 beat/minutes from HR=80.

Figure 8 illustrates the steps and the components of the upload process, and Figure 9 is a screen capture of an instance from the database.

3.7 Transmission of SMS emergency messages

After the application captures the sensor readings from the microcontroller, it performs a comparison between these readings and pre-specified threshold limits (entered by administrator and specified by physician) to check if the patient or the user is in a critical condition or not. If the situation is a critical, the application immediately sends an SMS to the physician using the stored settings (SMS number, patient's name and ID) along with the values of the critical readings. The SMS message has the following format:

'HEMOS Notification:

Patient ID = 6666

Name = Hazem Raddy

Heart Rate = 80

Temperature = 24

Time = 2005/12/09 04:40:20 AM'

Figure 8 Upload process (see online version for colours)

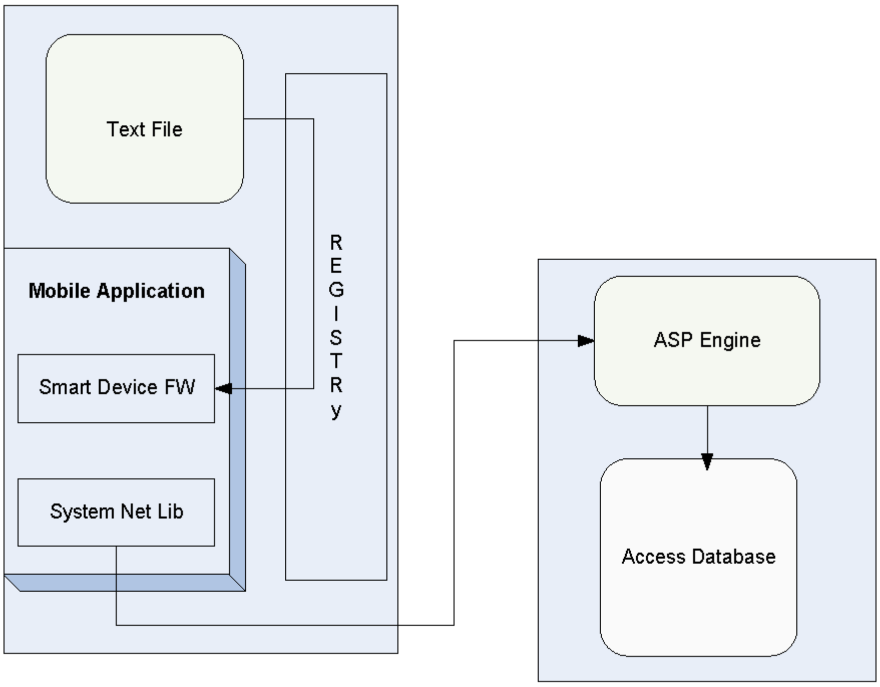


Figure 9 A patient database record (see online version for colours)

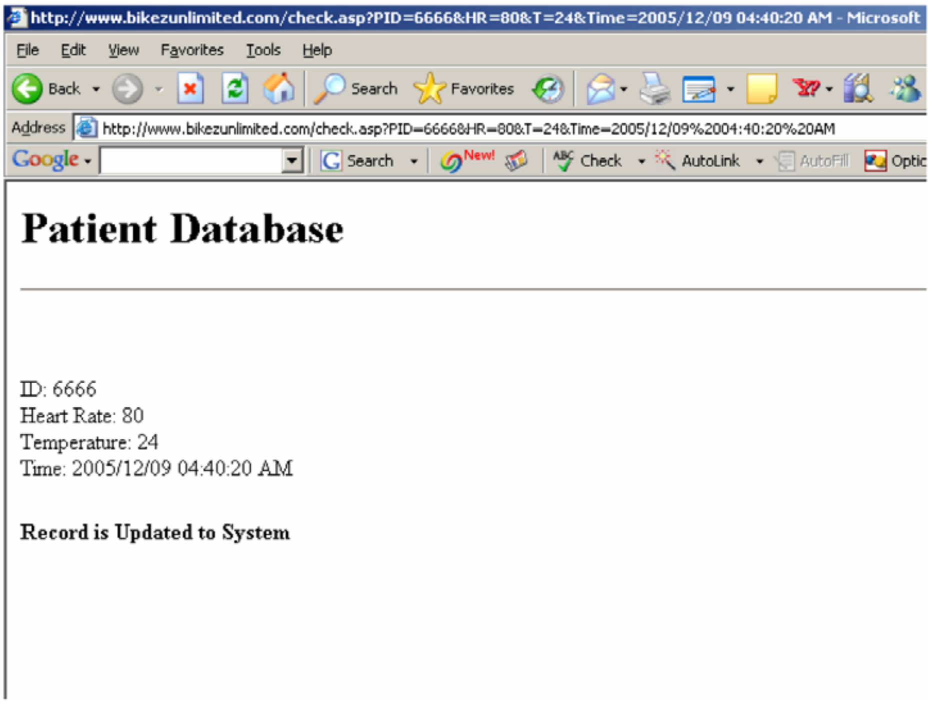
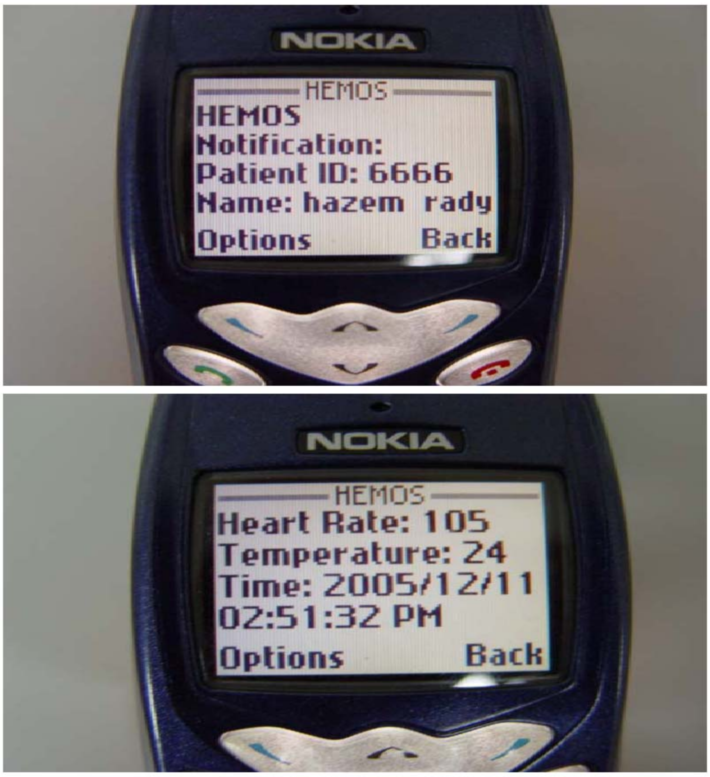


Figure 10 is a screen capture of a physician's mobile after receiving an emergency message. The HEMOS acronym stands for '**He**althcare **Mo**nitoring **S**ystem'.

Figure 10 A HEMOS SMS message (see online version for colours)

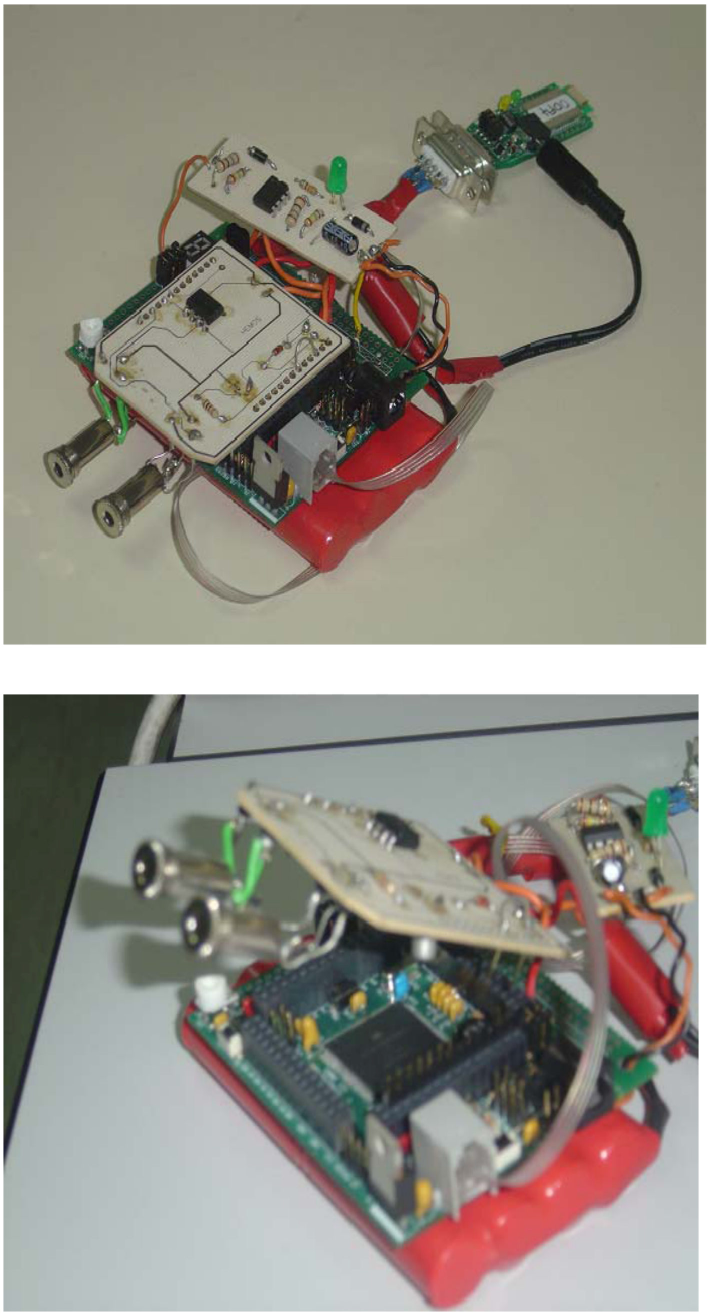


4 Discussion and conclusions

Pervasive healthcare has emerged as a viable option in improving the productivity of healthcare practitioners and in facilitating the delivery of a wide range of medical services, and in countering the increasing costs of the traditional healthcare delivery options. In the coming years, we will see an increase in the adoption of mobile technology as a conduit for facilitating communication between patients and their physicians, and for improving the quality of life of outside patients (Varshney, 2003; Shieh et al., 2008; Standing and Standing, 2008). In this work, we discussed the design and implementation of a wearable platform that builds on the recent advances in wireless technology and the ever-increasing computing power of mobile devices.

The proposed platform and its implementation were assessed using temperature and heartbeat as the vital medical parameters to be monitored. However, the design is scalable, and additional sensors can be integrated by making use of the available ADC channels in the microcontroller. The circuitry of the wearable kit is shown in Figure 11. It weighs approximately 250 grams. The temperature sensor used is an LM 35. It is an integrated circuit sensor that can be used to measure temperature with an electrical output proportional to the temperature (in °C). A heart rate monitor receiver (upper circuit board of Figure 11) was built to facilitate communication with a wireless heart rate measurement transmitter that the patient will be wearing.

Figure 11 Wearable kit circuitry (see online version for colours)



To prove the feasibility of the proposed concept, and with the help of few volunteers, the complete system was evaluated targeting critical indicators such as mobility, accuracy, data transmission capability and power consumption or battery life.

The mobility of the prototype was considered with factors such as wiring, weight and size in mind. In general, the volunteers felt that wearing the monitoring kit did not slow them in regard to their daily activities; however, encasing the prototype kit in a rugged box or container will be of benefit. Additionally, users indicated that even though they had minimal training (in using the device) they were still able to operate it without difficulty.

To assess the accuracy of transmitted data, store and forward integrity tests were performed using different time windows. For example, readings were captured and stored locally in the PDA for approximately 10 minutes and then transmitted via the internet to the remote site. Bluetooth comes with adequate buffering capabilities and hence no data loss was detected locally. There were not any noticeable transmission errors on the other end; however, results indicated that the accuracy of the HRM (Heart Rate per Minute) readings needs to be improved. This can be accomplished via improvements in surrounding conditions (to minimise noise effects), and perhaps better algorithms for heartbeat calculations. The SMS alerting mechanism was also tested and the average delay was around 8 seconds.

Power consumption has two folds here, PDA consumption and the monitoring kit prototype power consumption. It is observed that the consumption of the PDA deteriorates when the frequency of uploads is increased. On the other hand, the kit was run continuously acquiring data for about 9 hours using 9.6 V, 700 mAh batteries.

Finally, and in conclusion, this paper discussed a framework and a methodology for the implementation of a wearable healthcare monitoring prototype system based on existing standards, and that uses off-the-shelf components. Custom software has been developed, and successfully utilised. The concept had been adequately proven, and the critical components that constitute or make the infrastructure of such an application were presented and discussed.

References

Fadlee, M., Rasid, A. and Bryan, W. (2005) ‘Bluetooth telemedicine processor for multichannel biomedical signal transmission via mobile cellular networks’, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 9, No. 1, pp.35–43.

Freescale Semiconductors (2008) *Minidragon Microcontroller’s Manual*. Available online at: www.freescale.com

Grid Connect Inc. (2008) *Blueport Device Manual*. Available online at: www.gridconnect.com

Guillen, S., Arredondo, M., Traver, V., Garcia, J. and Frenandez, C. (2002) ‘Multimedia telehomecare system using standard TV set’, *IEEE Transactions on Biomedical Engineering*, Vol. 49, No. 12, pp.1431–1437.

Imate (2006) *Imate Manual*. Available online at: www.imate.com

Intille, S. (2004) ‘A new research challenge: persuasive technology to motivate healthy aging’, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 8, No. 3, pp.235–237.

Jovanov, E., Milenković, A., Basham, S., Clark, D. and Kelley, D. (2004) ‘Reconfigurable intelligent sensors for health monitoring: a case study of pulse oximeter sensor’, *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, pp.4759–4762.

Lukowicz, P., Anliker, U., Ward, J., Troster, G., Hirt, E. and Neufelt, C. (2002) 'AMON: a wearable medical computer for high risk patients', *Proceedings of the 6th International Symposium on Wearable Computers*, pp.133–134.

Mundt, C., Montogomery, K., Udoh, U., Barker, V.N., Thonier, G.C., Tellier, A.M., Ricks, R.D., Darling, R.B., Cagle, Y.D., Cabrol, N.A., Ruoss, S.J., Swain, J.L., Hines, J.W. and Kovacs, G.T.A. (2005) 'A multiparameter wearable physiological monitoring system for space and terrestrial applications', *IEEE Transactions on Information Technology in Biomedicine*, Vol. 9, No. 3, pp.382–391.

Perry, M., Dowdall, A., Lines, L. and Hone, K. (2004) 'Multimodal and ubiquitous computing systems: supporting independent-living older users', *IEEE Transactions on Information Technology in Biomedicine*, Vol. 8, No. 3, pp.258–270.

SBS (2001) Specification of the Bluetooth system 1.1. Available online at: www.bluetooth.com

Shieh, Y., Tsai, Y., Anavim, A., Shieh, M. and Lin, M. (2008) 'Mobile healthcare: the opportunities and challenges', *International Journal of Electronic Healthcare*, Vol. 4, No. 2, pp.208–219.

Sneha, S. and Varshney, U. (2007) 'A wireless ECG monitoring system for pervasive healthcare', *International Journal of Electronic Healthcare*, Vol. 3, No. 1, pp.32–50.

SPP (2008) Sequence Packet Protocol. Available online at: http://www.protocols.com/pbook/xns.htm#SPP

Standing, S. and Standing, C. (2008) 'Mobile technology and healthcare: the adoption issues and systemic problems', *International Journal of Electronic Healthcare*, Vol. 4, Nos. 3/4, pp.221–235.

Thulasi, V., Bai, V. and Srivatsa, S. (2007) 'Design of a wearable telemedicine system', *International Journal of Electronic Healthcare*, Vol. 3, No. 3, pp.303–316.

Varshney, U. (2003) 'Pervasive healthcare', *IEEE Computer*, pp.138–140.

Yao, J., Schmitz, R. and Warren, S. (2005) 'A wearable point-of-care system for home use that incorporates plug-and-play and wireless standards', *IEEE Transactions on Information Technology in Biomedicine*, Vol. 9, No. 3, pp.363–371.