Wireless Sensors Network Based Physiological Parameters Monitoring System

A Project Report
Submitted in partial fulfilment of the
requirements for the Degree of

Master of Engineering

In

ELECTRONICS AND TELECOMMUNICATION ENGINEERING

By

Karandeep Malhi

SCHOOL OF ENGINEERING AND ADVANCED TECHNOLOGY
MASSEY UNIVERSITY
PALMERSTON NORTH
NEW ZEALAND
JULY 2010

To my family

ABSTRACT

Continuous technological innovation in research and development in the last two decades has resulted in development of different smart systems for heath monitoring for individuals at their home with wireless technology. A wearable non-invasive device has been developed to monitor physiological parameters (such as body-temperature, heart rate, detection of fall) of a human subject. The system consists of an electronic device which is worn on the wrist and finger, by the person to be monitored. The system can be used by elderly or the person at risk or even by a normal person for the monitoring of physiological parameters. Using several sensors to measure different vital signs, the person can be wirelessly monitored within his own home, may be defined as a smart home. A heart-rate sensor has been developed to monitor the heart rate continuously. An accelerometer has been used to detect falls. The device has the capability to determine the stressed condition of the person and may be used to send an alarm signal to a receiver unit that is connected to a computer. This sets off an alarm which can go to a care-giver, allowing help to be provided to the person. Since no vision sensors (camera or infra-red) are used, the system is non-invasive, respects privacy and it is expected that it will find wide acceptance. The system can be used in combination with the bed sensor (part of the home monitoring system) to monitor the person during the night. The complete system will help to monitor the person during day and night and will be suitable to an elderly living alone at home.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisors Associate Professor Dr. Subhas Mukhopadhyay and Senior Lecturer Dr. Gourab Sen Gupta, who have given me encouragement and assistance to complete this work. I am deeply indebted to Associate Professor Dr. Subhas Mukhopadhyay, for his continuous support and supervision of my research work and providing me with valuable advice and expert guidance, and above all for his technical feedback. Without his help, this work would not have been possible.

I sincerely thank Senior Lecturer Dr. Gourab Sen Gupta since part of my research work presented in this thesis was done in close consultation with him. I would like to express thanks to him for his valuable advice for technical matters, guidance related to microcontroller programming and numerous fruitful suggestions.

I would like to acknowledge the efforts of Mr. Ken Mercer, Mr. Colin Plaw, Bruce Collins, Anthony Wade and Clive Bardell. I would like to thank the SEAT IT support staff for providing me with required IT tools to run software for my study.

I would also like to thank Mathias Haefke, summer exchange trainee from Rostock University, Germany, whose work on development of GUI, using ZigBee communication has been very useful for this project. Also I would like to thank Julia Schnepper (summer exchange trainee) from Rostock University, Germany, for her efforts on testing sensors used in this work.

Special thanks to my husband Satinder Gill, for his constructive advice throughout the year. I wish to acknowledge Alexander Keerl and Kathleen Lennie, for their careful reading of an early draft of project thesis.

Finally and most importantly I would like to thank my parents, for their unconditional love and support throughout my time at university. Thank you for all the sacrifices you have made to give me a better chance in life.

CONTENTS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

1.1 Introduction to Sensor

A sensor is a device that detects and measures a physical quantity. New sensing principles and techniques are investigated in scientific research and sensors are widely used in scientific research and are an integral part of automated systems.

A sensor is a device which provides a usable output in response to a specified measurand [1]. The *output* is defined as an electrical quantity which can be any type of signal, and *measurand* is defined as a physical quantity, property or a condition. Figure 1-1 illustrates the typical sensing process.

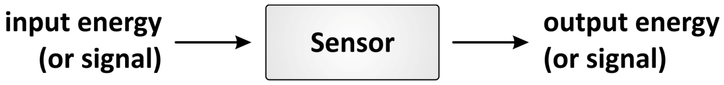


Figure 1-1: The Sensing Process: A measured input energy or signal is evaluated by a sensing device, giving an output energy or signal as a result of the sensing process.

.

Sensors can be classified into *passive* and *active* sensors [2]: An active or self-generating sensor is one that emits its own energy (or generates a signal) without the need for any external power supply. The energy travels from the device to the target to be studied. An example for an active sensor is the microwaves. Doppler Radar is an example of active remote sensing technology based on active sensors [3]. A Passive sensor is one that detects radiations emitted or reflected by the object or surrounding area under observation. Examples are thermocouple, film photography and radiometers.

The sensing method is based on the principle of type of energy involved. Table 1-1 shows different types of energy and measurands.

Table 1-1: **Energy types and corresponding Measurands [1]**

Type of Energy	**Measurands**
Mechanical	Length, area, volume, force, acceleration, torque, mass flow, acoustic intensity and so on.
Thermal	Temperature, heat flow, entropy, state of matter.
Electrical	Charge, current, voltage, resistance, inductance, capacitance, dielectric constant, polarization, frequency, electric field, dipole moment, and so forth.
Magnetic	Field intensity, flux density, permeability, magnetic moment and so forth.
Radiant	Intensity, phase, refractive index, reflectance, transmittance, absorbance, wavelength, polarization and so on.
Chemical	Concentration, composition, oxidation/reduction potential, reaction rate, pH, and the like.

To ensure the best possible measurement result (low noise level), sensors are normally calibrated for certain conditions. These comprise the normal environmental conditions from where the data is collected (temperature, air moisture, radiation,) as well as the dynamic range of the measurand. Many sensors need to be amplified to be useful [4].

Sensors can also be classified on the basis of signal transduction principles, input physical quantity, material, technology, application and property but not limited to these criteria [1]. Selection criteria depend on many factors, such as availability, cost, power consumption and environmental conditions Sensor classification can be done under schemes. Schemes can range from the simple to the complex [28]. White describes in [28] a comprehensive and flexible scheme to classify sensor characteristics. Classification of sensors based on applications can show segmentation in a very broad manner [1].

The feasibility of sensor technology is determined by sensor manufacturers by keeping in mind while designing, the various factors including cost reduction, reliability, system compatibility, safety in hazardous/hostile environments and non-invasive design. To make sure sensors perform at their best, selection of sensors depends on factors such as availability, power consumption, cost and environmental conditions [2, 4].

Figure 1-2 shows an application based classification of sensors.

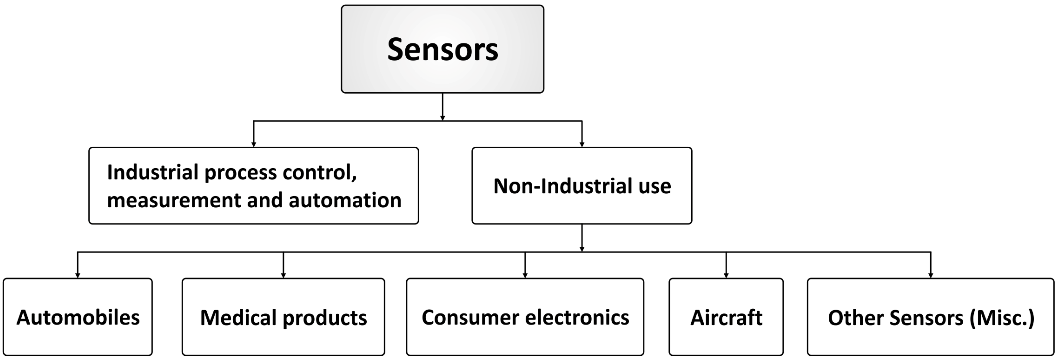


Figure 1-2: Classification of sensors-Applications [1]

In general, most sensors fall into two categories: Digital sensors and analog sensors.

Digital sensors represent quantities in discrete units. The output signal is a digital representation of the input signal and has discrete values measured at discrete times. Example of digital sensors includes switches and position encoders. These sensors produce pulse trains of transitions between two states of voltages, which are either low or high. On the other hand analog sensors produce output which is directly proportional to the input and is continuous in both magnitude and time. Examples include microphones or resistance temperature devices (RTD).The disadvantages of analog type sensors include crosstalk, system noise as well as performance reductions while transmitting the analog signals over long distances.

The common properties of an ideal sensor are:

- Optimal measurement accuracy
- Fast and continuous response (operation)
- Linearity
- Sensitivity: Sensitivity and good resolution
- Insensitivity to electrical and the other environmental interference
- Long term stability

- Low operation and maintenance cost
- Durability
- Meeting safety requirements
- Acceptance by users

Sensors play an important part in real-time systems and are critical to today's innovative society by providing the connection between the real world and the world of process control and computers. The accuracy of a sensor determines the overall accuracy and reliability of the control system. The new trend of miniaturising electronic devices leads to the necessity of small, relatively independent entities which can interact with each other and with other entities, forming collective and associative environments [29]. The emerging sensing technologies can be briefly classified and represented as shown in Table 1-2.

Table 1-2: Sensor Technologies [1]

Sensors	Technology	Applications
Image Sensors	CMOS Based	Traffic and security surveillance, blind spot detection as auto sensors (robots etc.), video conferencing, consumer electronics, Biometrics, PC Imaging
Motion Detectors	IR, Ultrasonic, Microwave/Radar	Obstruction detection (robots, auto), Security detection (intrusion), toilet activation, kiosks videograms and simulations, light activation
Biosensors	Electrochemical	Water testing, food testing (contamination detection), medical care device, biological warfare agent detection
Accelerometers	MEMS Based	Vehicle dynamic system (auto), Patient monitoring (including Pacemakers etc.)

1.2 Literature research

The rapid development of microelectronics, micromechanics, integrated optics and other related technologies has enabled us to develop various kinds of sensors, both wired and wireless, which enable us to sense and measure data more efficiently and accurately. Efficiency relates to the speed of measurement, energy consumed for the measurement and processing resources required. Sensors can be wired and wireless: A wireless sensor has various advantages over wired sensors. Wireless sensors are flexible and can be easily reconfigured. They can be used in places geographically far apart to monitor activities remotely. Wireless sensors are the most important technology identified in the twenty first century [24]. Wireless sensing units integrate wireless communications and mobile computing with transducers to deliver a sensor platform which is inexpensive to install in numerous applications. Indeed, co-locating computational power and radio frequency RF communication within the sensor unit itself is a distinct feature of wireless sensing.

Due to the miniaturization of active and passive components and easy availability of components, most new research on focused at improving quality of human life in terms of health [13] by designing and fabricating sensors which are either in direct contact with the human body (invasive) or indirectly (non-invasive). One of the reasons for more development in this area is the rise in global population and ageing population [11], one statistic provided by US department of health is that by 2050 over 20% of the world’s population will be above 65 years of age. This results in a requirement for medical care, which is expensive for long term monitoring, and long waiting lists for consultations with health professionals.

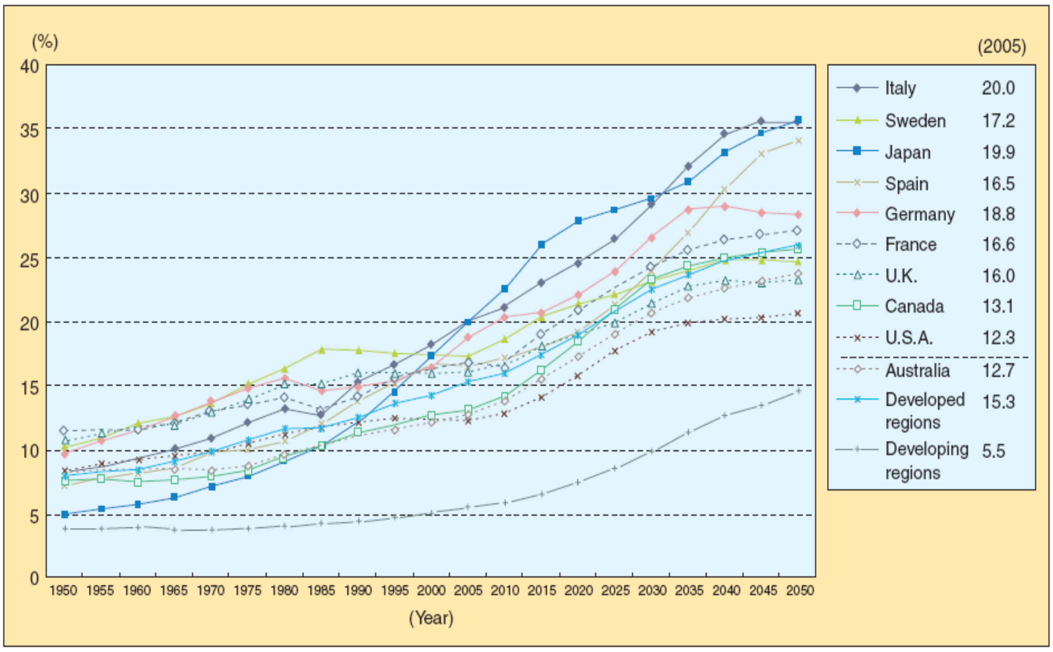


Figure 1-3: The world population projection

Figure 1-3 shows the world population projection from 1950 to 2050. The United Nations, revision for 2004 in world population prospects shows data for Japan based on

“ Population Census of Japan” (a Ministry of Internal Affairs and Communications) and also “Population projection for Japan” based on (National Institute of Population and Social Security Research), January, 2002.

These facts show an increasing demand for long term health monitoring which is affordable, continuous and unobtrusive [12], which will result in considerable impact on annual medical costs [13] and health management [14]. Wearable systems for continuous health monitoring are a key technology in helping the transition to more practical and affordable healthcare. It not only allows the user to closely monitor changes in her or his physiological parameters but also provides feedback to help maintain an optimal health status.

The health monitoring systems developed in the last decade due to the continuous technological innovation in research and development resulted in prototyping of various wearable physiological parameters monitoring systems [15]. The main features of these systems include low power consumption, being integrated, smart, compact in size, robust, and wireless. Wearable systems for health monitoring need to satisfy a great variety of criteria and constraints [16], including aforementioned features and also privacy of medical data. These factors need to be considered while designing such systems.

Chris et al. [30] describe that during the last few years there has been a significant increase in the number and variety of wearable health monitoring devices, ranging from simple pulse, activity monitors, and portable Holter monitors, to sophisticated and expensive implantable sensors. However, wider acceptance of the existing systems is still limited due to remote offline analysis and data processing. This makes such devices impractical for continual monitoring and early detection of medical disorders. In other words the existing systems are rarely made affordable.

Personal Health Monitoring Systems (PHMS) can provide the ways to balance prompt health care, early diagnosis and more cost effective patient monitoring. To meet these requirements, the European Health Commission has funded collaborative research projects since the mid-1990's as shown in Table 1-3.

Table 1-3: Various PHMS for Health care [19], [17]

	Technologies	**Measurement**	**Application**
AMON Project	Wrist-worn device with wireless connection to telemedicine center	Vital signs and physical activity	Monitoring high risk cardiac/respiratory patients
MOBIHEALTH Project	Body area network with wireless connection to medical services	Vital signs	Remote monitoring of chronically ill patients
WEALTHY and MERMOTH Projects	Biomedical clothes with textile electrodes and wireless connection to medical center	Vital signs	Monitoring of cardiac parameters and patients
MYHEART Project	Biomedical clothes with textile electrodes and electronic sensors. Wireless connection to medical services	Vital signs and physical activity	Prevention and monitoring of cardiovascular diseases
VTAMN Project	Biomedical clothes with integrated sensors and electronics	Vital signs	Disease monitoring
Mamagoose Pyjama, Verhaert	Biomedical pajama for infants	Vital signs	Detection of Sudden Infant Death Syndrome
LifeShirt, Vivometrics Inc.	Biomedical shirt with integrated sensors	Vital signs	Sleep diagnostics, disease monitoring
SmartShirt, Sensatex Inc.	Textile platform with embedded electronic sencors and a conductive fiber grid to transmit data	Vital signs	Health monitoring
GlucoWatch G2 Biographer, Cygnus Inc.	Wearable device (forearm)	Glucose level	Noninvasive glucose monitor for diabetics

A system developed at MIT known as LiveNet [20] is another early wearable system capable of capturing real-time data, context classification and streaming for long term behaviour modelling in specific health cases such as Parkinson's disease or epilepsy.

Another wearable PHMS constituted a wearable Smart Vest [21] in which Researchers used sensors integrated on the fabric, collecting bio signals in a non-invasive and unobtrusive manner. The AMON [18] reports a wrist-worn device capable of measuring bio-signals such as blood pressure, skin temperature, oxygen saturation, Electro Cardio Graph ECG and activity status of the user. This developed system is more specialized for high risk respiratory patients and classifies estimated health condition of patient on manually predefined limit values for each signal.

The European Union funded research projects such as MyHeart [22], WEALTHY [23], and MERMOTH [24] developed using smart clothing systems, which are wearable garments .This helps in early health risk detection and prevention by utilizing smart sensing fabrics and interactive textiles to measure multi-parameter physiological monitoring. The invention reported in [10] relates to a biomechanical monitoring apparatus to be worn by the user to provide information on physiological parameters over a specific period of time. The monitoring apparatus is a fusion of different sensors which are used to measure different physiological parameters of interest.

MEDIC [33] is a medical embedded device for individual care and has a proposed design framework which includes embedded artificial intelligence in the form of an inference engine, based on naïve Bayes classifiers for detection of patient conditions and managing system resources in a dynamic manner. It is based on a general architecture for wearable sensor systems which can be customised according to an individual's requirement. Also it can be reconfigured and remotely controlled wirelessly by commands sent to the system. HeartToGo [34] is a cell phone based wearable platform, which is capable of continuously monitoring the patient's ECG signal via a wireless ECG sensor. As this system is particularly based on cardiovascular monitoring, it specialises in providing more accurate results and classification of ECG patterns. It uses artificial neural network based machine learning schemes.

The Biosensor Net was developed at Imperial College London by a team from computing, biomedical engineering, electronic engineering, and medicine. The sensors developed can monitor patients as they go about their everyday activities. The system notifies healthcare workers of problems and compiles data for trend analysis and medical research. This new miniaturized wireless biosensor technology of self-managing, context-aware bio-sensing networks can improve patient care.

Market Research

There are several healthcare monitors available on the markets that monitor physiological parameters. They all have their advantages and disadvantages. The key advantage of our developed system over the ones that already exist in the market is that it sends out an alarm automatically without the patient/user having to push a panic button in an emergency.

Philips Lifeline medical alarm, St John's Lifelink, and ADT's NevaAlone medical alert system are some of the few medical alarms available in the market. They all have the same principle in that a wearable personal help button (that can be worn as a pendant or a wrist watch) is used. When the user needs medical attention, they press the alarm button and it activates the remote receiver unit or the telephone and dials the response centre for an ambulance.

ADT NevaAlone [5] and Vital Link [7] is an elderly healthcare monitoring system that can be worn as a pendant or a wrist band with a personal help button. In the event of an emergency, the user presses the alarm button and the signal is sent to the ADT monitoring specialist via telephone line and an ambulance is sent. Another similar product available is the St. John Lifelink™ Medical Alarm [8], the alarm is connected to the St. John ambulance communications centre. When the alarm button, worn as a pendant, is pressed, the communications centre will call back to check that the alarm was not set off by mistake. If no one answers the phone, an ambulance will be sent to the user's house immediately.

Smart Link's Medi-Call [9] is a system which has a base station placed in the house, a wrist watch and a wall mounted alarm button. In a case of emergency, the user can press any of these emergency buttons and the call for help will be sent via a telephone line and an ambulance will be sent. This system has an added advantage over the other systems as it has many posts where the user can press the alarm for help. The major drawback for all the above systems is that the user has to press the alarm button in an emergency. But in most cases the user may not be in a position to press the button in cases where they have become paralyzed or unconscious. Also, the health care monitors discussed are quite expensive. Along with the expense of buying the system, there is a monthly or fortnightly monitoring fee which varies from 70 to 100 dollars.

None of these products actually measure the physical parameters and send out an alarm when the patient needs medical attention. The alarm button has to be pressed in order for them to activate the alarm. This could be disadvantageous when the user is unconscious or not in a position to push the alarm button.

A&D Medical, a San Jose based company [31], and their Life Source products allow monitoring of health via wireless connections. Wireless Complete Health Monitoring System is a combo kit containing: a scale, a blood pressure cuff, and a pedometer/motion sensor for determining your daily activity which monitors activity, weight, and blood pressure. Although this product is under $200 it is not portable and unfortunately has some software/compatibility issues [32].

1.3 Project Overview and Scope

Figure 1-4 shows the block diagram of the physiological parameters measuring system developed. The physiological parameters measuring sensors are designed and the signals from the sensors are interfaced with a microcontroller for processing. The UART of microcontroller sends the sensor data to the ZigBee module through serial communication, from where the data is sent wirelessly to the ZigBee co-ordinator which then stores the data on computer in a database for reference.

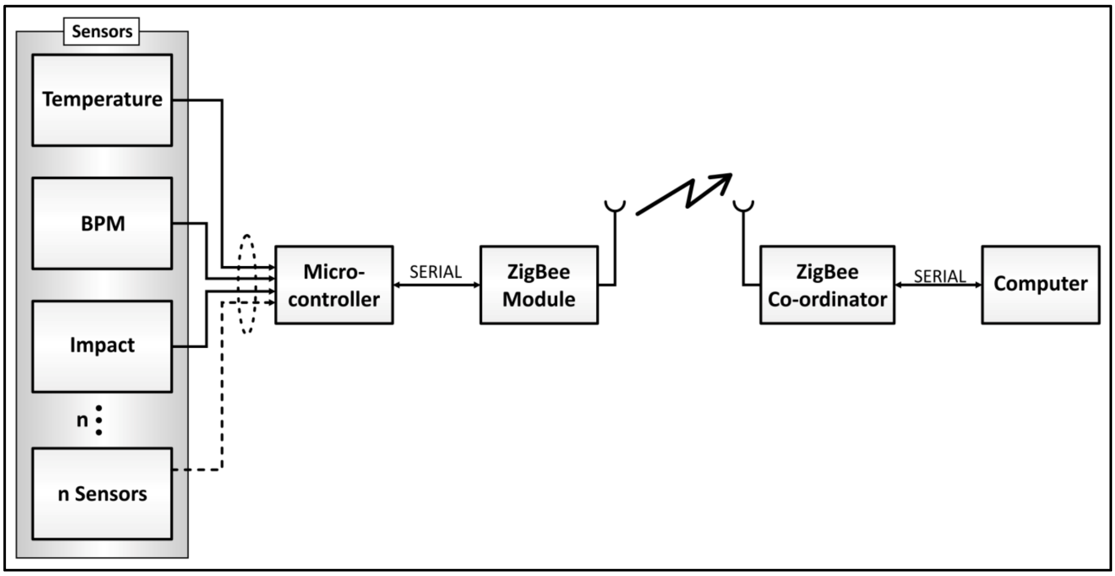


Figure 1-4: Block diagram of system

The project is based on health care monitoring for the elderly people living alone and at risk patients who require constant monitoring or care. Usually, it is not feasible to have a caregiver all the time. So, they need a device that can monitor their health and report an alarm to the caregiver when something goes wrong. This way, the patients keep their independence and eliminate the need for constant monitoring. The basic principle in this project is to develop a device that monitors the health of at risk patients through a non-invasive approach by researching the available technologies and methodologies. The product should be cheap and affordable for ordinary people. The aim of this project is to make the product compact and non-invasive in order for the clients to not feel uncomfortable wearing it all the time. The system consists of sensors incorporated into a wrist band that would measure the vital signs such as temperature, heart rate, and blood pressure. These measurements are wirelessly sent to a computer in the same house. At Massey University similar work has previously been done with different approaches. This project is further research and more accurate approaches towards the final results are utilized. The approaches and techniques used are detailed with each sensor design in this report. The inputs from the sensors are processed by a microcontroller. Then, the processed data is transmitted to a receiver unit that decodes the information and sets an alarm when any of the parameters go beyond a defined range of values. It will be a cost effective system, where no additional charges need to be paid for monitoring in contrast to most of the currently available systems which charge a monthly fee.

Since the alarm generated is visual, the caregiver is required to check the computer regularly. This could be risky if they are a bit late to check in any emergency situations. This can be changed to produce a sound alarm as soon as the receiver detects that the patient's health requires attention. The device constraints are:

- It should be compact and non-intrusive.
- It must be compact and convenient enough to be able to wear everyday and should not be a burden for the user. The monitoring of the patient's vital signs should be non-invasive in order for them to be able to wear it all the time and not feel uncomfortable.
- It must be robust and have a user friendly interface for the caregiver to communicate with it properly.

Test Plans/Objectives

The system should be tested at each of these stages-

The measurement of the physical parameters, then they are fed in to the microcontroller and the results are stored on the database. Since this project requires medical testing it is a requirement to measure the physical parameters from several people to make sure the system works reliably.

The device has sensors incorporated onto a wrist band to monitor the vital signs such as heart rate or blood pressure. These measurements, which are in analog and digital form, are sent to the Analog to Digital Converter (ADC) channel and digital port of the microcontroller and transmitted wirelessly over to a receiver (PC) that holds the data and generates an alarm when required.

1.4 Organization of thesis

This thesis is organized into eight chapters. Chapter one gives a general overview of sensors. It details the classification of sensors used in today’s innovative world of technology. The literature review of wearable heath monitoring systems developed or currently under development around the globe at research institutes is presented. Chapter two describes measurement, construction, technique and experimental results of the human body skin temperature sensor. Similarly chapter three, four and five describe the heart rate sensor, impact sensor and bed sensors respectively. The integration of all three sensors is explained in Chapter six. Chapter seven discusses the communication between sensors and microcontroller board through the ZigBee wireless communication standard and also software and algorithms are discussed. Chapter eight concludes the work, summarising the system developed, and future work is discussed briefly.

Chapter 2: Measurement of Human Body Temperature

2.1 Introduction

A patient’s temperature can be used to detect the symptoms of medical stress that might lead to various health conditions, including stroke, heart attacks and shock. Thus this variation in temperature is measured on the skin, as it is simple and can give an indication of what is happening with the patient’s body temperature.

Skin is used to regulate the body temperature. The normal temperature of skin is about 33 °C or 91 °F. The skin is the largest organ in the human body. It protects the body from the sun's rays. It also keeps body temperature normal 37 °C [35].

The purpose of the skin thermal sensors is to 'sense' any thermal threat from the environment. The surrounding temperature is sensed by tiny sensors embedded in our skin, which are extremely sensitive to changes in surrounding temperatures [35]. Skin temperature depends on air temperature and time spent in that environment. Such weather factors as wind chill and humidity cause changes in skin temperature. The flow of energy to and from the skin determines our sense of hot and cold see Figure 2.1. Heat flows from higher to lower temperature, so the human skin will not drop below that of surrounding air, regardless of wind, but by sweating the skin temperature can be lower than the surrounding air by cold caused by condensation [37].

The vascular apparatus of skin is a very important factor of human body local thermoregulation. In the cold (Figure 2-1, left part) skin blood vessels (mainly arterioles) are constricted. Additional blood is supplied to abdominal cavity vessels from dilated arteriovenous anastomoses (A small blood vessel with a relatively thick muscular coat that provides a direct connection between an artery and vein, thus bypassing capillaries [39]). High and low temperature induces dilation and constriction of human skin blood vessels, respectively. This reaction is mediated by the autonomous nervous system (parasympathetic and sympathetic nervous system, respectively). Subcutaneous cellular tissue also contributes to mechanisms of self regulation of human body temperature [38].

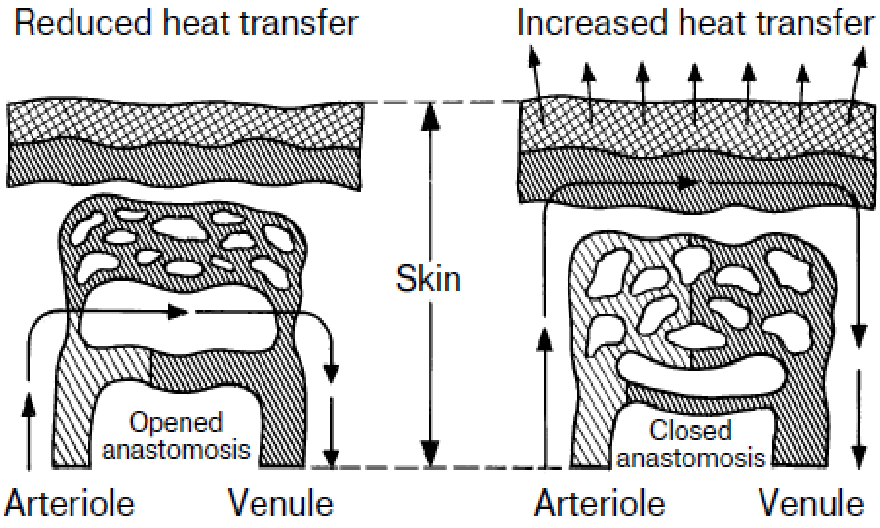


Figure 2-1: Vascular Apparatus of Skin (Thermoregulation) [38]

Skin temperature can simply be used to detect the symptoms (such as shivering, sweating, and fevers) of several medical conditions such as trauma, injury, heart attack, stroke, heat exhaustion or burns.

2.2 Measurement of Human Body Temperature

Temperature is the most often-measured environmental quantity. Body temperature is not fixed, but is responsive to cyclical changes. Body temperature for an average active human being, increases during the day reaching a maximum point by evening and then lowers to a minimum point in early morning [36]. Body temperature is an estimate of the average temperature of the core portions of the body as reflected by the temperature of the blood in the major vessels.

Traditionally, body temperature has been measured by contact thermometers in the oral, rectal or auxiliary sites. These sites are choices of convenience rather than correctness, because they often do not represent internal (core) body temperatures with the necessary accuracy.

The hypothalamus is the accurate site to measure body temperature. It is located in the base of the brain and acts as the body’s thermostat. It functions by monitoring heat sensors throughout the body and adjusting the temperature based on the body’s needs. The goal of the hypothalamus is to maintain the body’s core temperature between 36°C to 38°C (96.9°F to 100.4°F). But access to the hypothalamus is quite inconvenient.
The common sites for measuring Human Body Temperature are briefly discussed below:

Oral: The most commonly used site is the sublingual area. It is considered a fairly accurate site due to its close proximity to the lingual and external carotid arteries. However, on average it runs lower than core temperature by approximately 0.5°C (0.8°F). Correct placement of the oral thermometer is important for accuracy.

Tympanic: Ear thermometers measure eardrum temperature using infrared sensors. The blood supply to the tympanic membrane is shared with the brain. However, this method of measuring body temperature is not as accurate as rectal measurement and has a low sensitivity for fevers, missing three or four out of every ten fevers in children. Ear temperature measurement may be acceptable for observing trends in body temperature but is less useful in consistently identifying fevers.

Rectal: For many years rectal temperature measurement was considered the “gold standard” especially in pediatric patients.

Axilla and Groin: These two sites are popular with the lay public due to their non-invasiveness and accessibility. Glass-mercury, electronic or single-use-chemical thermometers can be used, but the clinical accuracy at these sites is suspect. With glass-mercury and single use-chemical thermometers the device must remain in position between 8 and 11minutes. These sites are not located near a major artery or thermo receptor and may not reflect temperature fluctuations. These factors may alter readings as much as 1.2°C (2.2°F) lower than actual core temperature. Lastly, if the patient is in shock the peripheral vaso-constriction will adversely affect the reading. Other core accessible sites are the oesophagus, tympanic membrane and urinary bladder. These sites are used when accurate knowledge of core temperature is critical.

2.3 Temperature Sensor

Internal body procedures like metabolism or immune reaction are always combined with a change of body temperature and so is the skin temperature. So this physical parameter is a good indicator for discerning critical changes. The temperature of the human skin is the resultant of several factors, as heat is supplied from the subcutaneous tissues and lost from the surface of the body by radiation, conduction, and the vaporization of water [40].

The measuring time of modern digital sensors available from leading manufacturers is a fraction of seconds, even with reduced dimensions and weight of sensors. This is due to the specificity of the human body as an object of temperature measurement [38].

In this project temperature measurement is done using a DS600 analog-Output Temperature Sensor [39]. It provides an accuracy of ±0.5 °C over its wide temperature range of -20°C to +100°C. This accuracy is valid over its entire operating range of 2.7V to 5.5V. The DS600 is available in 8-pin µSOP (Micro Small Outline Package) package with an exposed pad on the bottom of the chip for quick thermal response. It requires no external components and has an exposed pad which helps in making the maximum skin area in contact with the temperature sensor to get the best temperature reading.

2.4 Construction

The temperature sensor used for measuring temperature is the DS600 from Analog Devices [39]. The advantage of considering DS600 over other temperature sensors available is that it requires no external components. The pin configuration of DS600 is shown in Figure 2-2

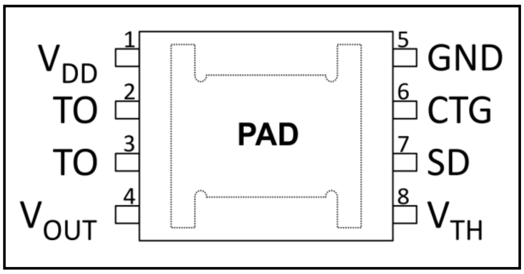


Figure 2-2: Top view of DS600 Pin Configuration

A typical application circuit is shown in Figure 2-3. The sensor is powered with 3.3V from the controller board. Pin no. 4 gives output Voltage, which is proportional to the die temperature in degrees centigrade and Pin no. 8 is ground. There is provision for thermostat application in this chip, though this is not required in this part of the project.

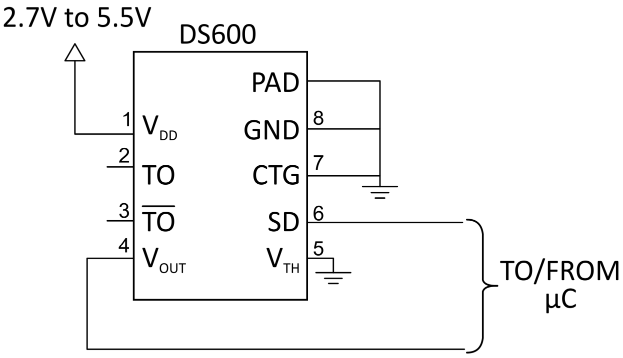


Figure 2-3: Circuit application

DS600 measures its own temperature and provides these measurements in the form of analog output Voltage i.e. proportional to degrees centigrade. It is mentioned in the data sheet that output voltage characteristics is factory calibrated for a typical output gain of $(\Delta V/\Delta T)$ of +6.45mV/°C and a DC offset (VOS) of 509mV.

Over the required range the sensor has an accuracy of ±0.5°C. But to improve a better signal-to-noise ratio the microcontroller sums up 50 values and makes two averages of them. The second one is made after eliminating all the values which are far away from the first made average.

To calibrate the sensor, the DC signal was measured and the corresponding ADC value the on microcontroller is noted at room temperature (in laboratory), at corridor and outside temperature. Figure 2-4 shows the transfer function of in laboratory experimental readings using this sensor of human skin temperature against output voltage.

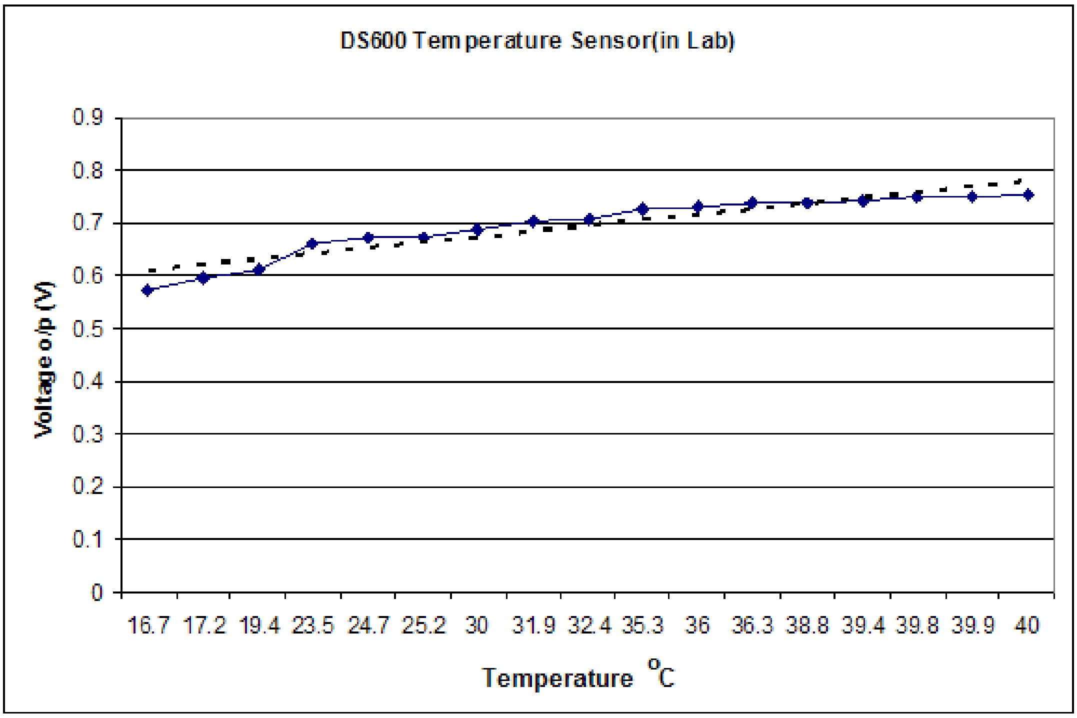


Figure 2-4: Voltage output characteristics

This is compared with original DS600 Transfer function as shown in Figure 2-5

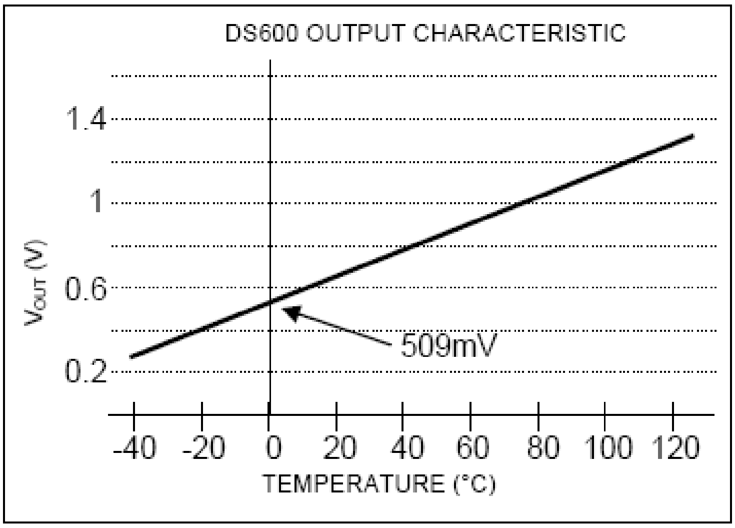


Figure 2-5: Transfer function of DS600

The typical output voltage transfer characteristic equation is:

V_{out} = (Device Temperature (°C) * ($\Delta V/\Delta T$) + V_{os})

The operating circuit of DS600 is shown in Figure2.3, where V_{out} from pin no. 4 is connected to Silabs C8051f020 microcontroller ADC port 0.3

2.5 Temperature sensing Technique

Several temperature sensing techniques are currently in widespread usage. The right one for the individual application depends on the required temperature range, linearity, accuracy, cost, features, and ease of designing the necessary support circuitry.

The sensor is mounted within the wrist strap, positioned in such a way that it is in direct contact with skin, allowing it to measure the external temperature of the body as shown in Figure2.5 (put wrist strap photo in/ out).

To get quick thermal response from the sensor, the exposed pad is placed on a PCB with precision cut PCB hole. The IC is then mounted and soldered on the PCB, such that bottom part of chip is exposed to get maximum exposure to the skin surface under temperature measurement. The PCB layout of the IC is shown below in

Figure 2-6.and Figure 2.7 displays IC on PCB.

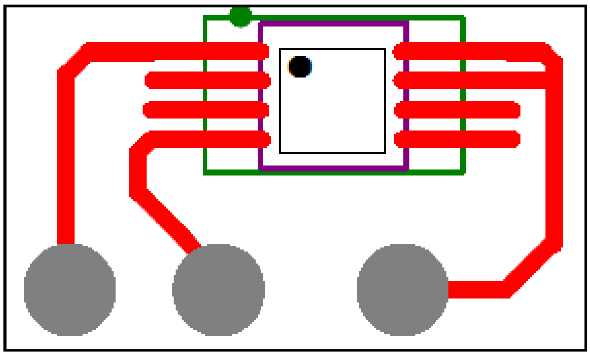


Figure 2-6: DS600 IC PCB layout in Altium Protel software

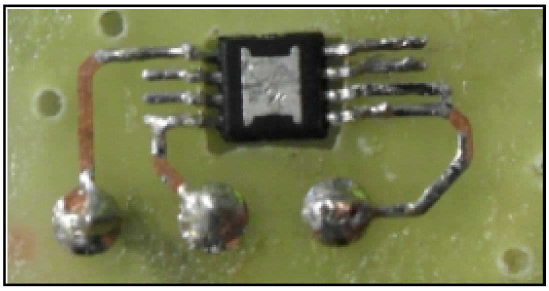


Figure 2-7: DS600 IC on Printed Circuit Board

2.6 Experimental Results

For collecting experimental data each sensor was tested separately. The necessary programs were written in the programming language C with the provided Silicon Labs© software. Though the temperature measurements were taken at several, different parts of the body, the place that was chosen for later "wearing" the monitoring unit is the wrist. This seemed to be the most comfortable and least annoying position for it because people are used to wear accessories at this part of the body i.e. a wristwatch or a bracelet.

The temperature sensor check was first done with the room temperature (22°C) amongst other things under lab conditions. In a second test, after passing the general check, the skin temperature data was collected from various volunteers at certain relevant parts of the body.

First quite a few results are obtained by just using a PCB mounted only with a temperature sensor and is placed on human body in three different locations, which are wrist, upper arm and neck. For comparison thermometer is used as reference.

On the following pages the experimental results of the DS600 sensor are outlined.

Figure 2-8 shows the sensitivity of DS600 temperature sensor under laboratory conditions. The variation in rise of temperature shows a quick response when it is touched by a finger and at the same time quickly falls after touch is removed.

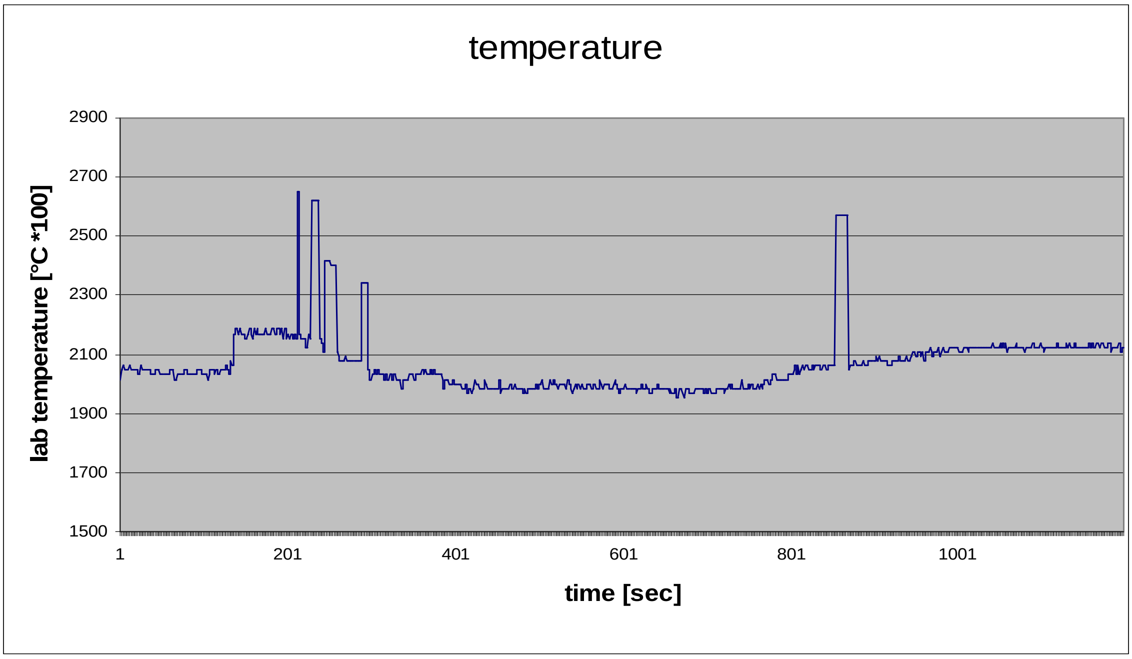


Figure 2-8: Response time of the DS600 temperature sensor

The sensor PCB is placed on the wrist, then on the upper forearm and on neck. The readings are obtained at different time intervals. At initial stages as discussed above, the skin temperature was taken at three different positions on males and females. Due to software issues the results displayed by DS600 on the microcontroller were not very accurate, but after software changes the results could be significantly improved. The Figure 2-9 shows the comparison of temperatures measured by the developed temperature sensor to a thermometer (reference temperature). The temperature at three different positions (wrist, neck and upper arm) was measured for three male and three female persons. The temperatures were measured at different times with varying ambient conditions. Figure 2-10 shows the variations in temperature measured with respect to different positions.

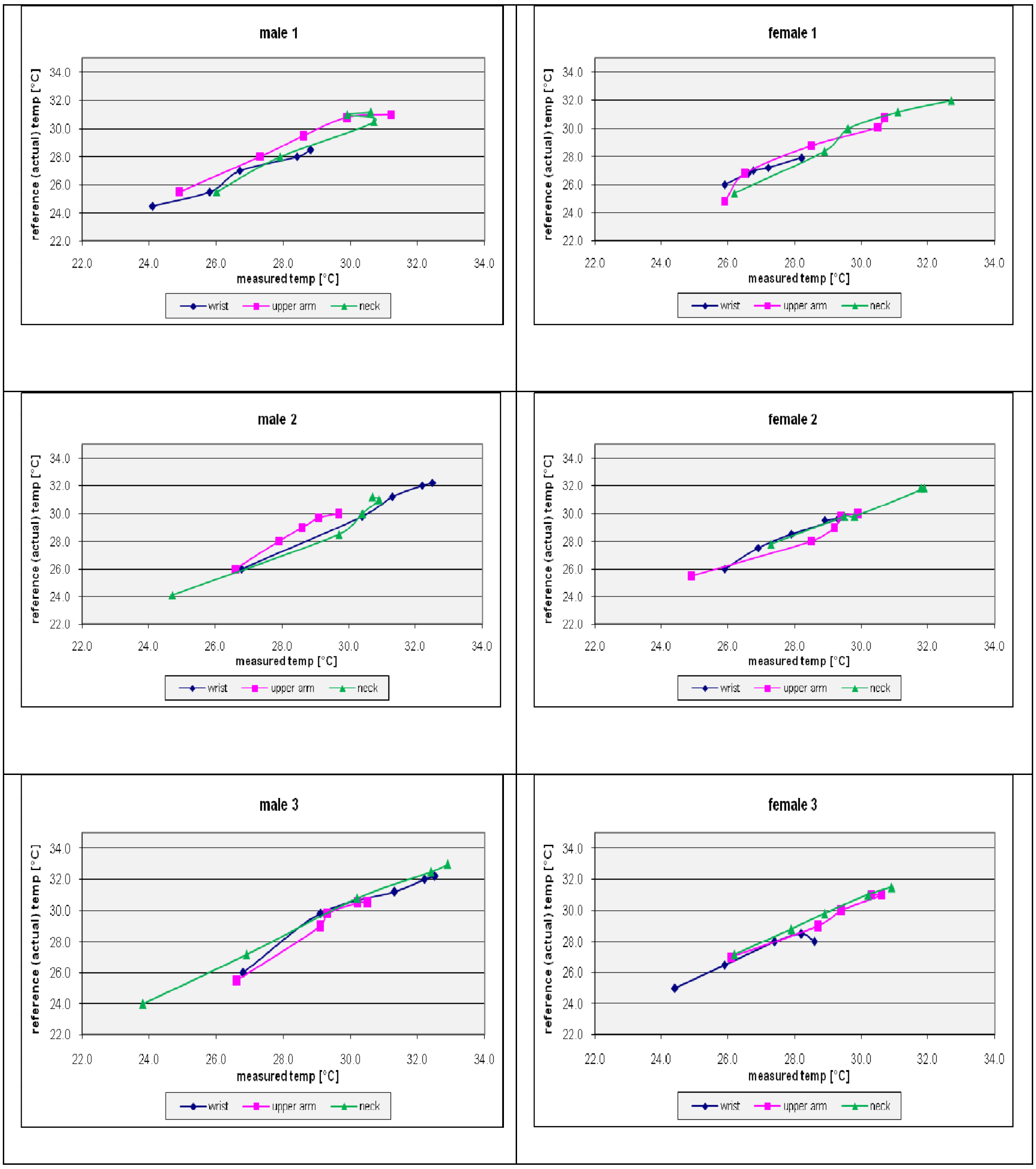


Figure 2-9: Test results of Temperature sensor on 3 male and 3 female subjects

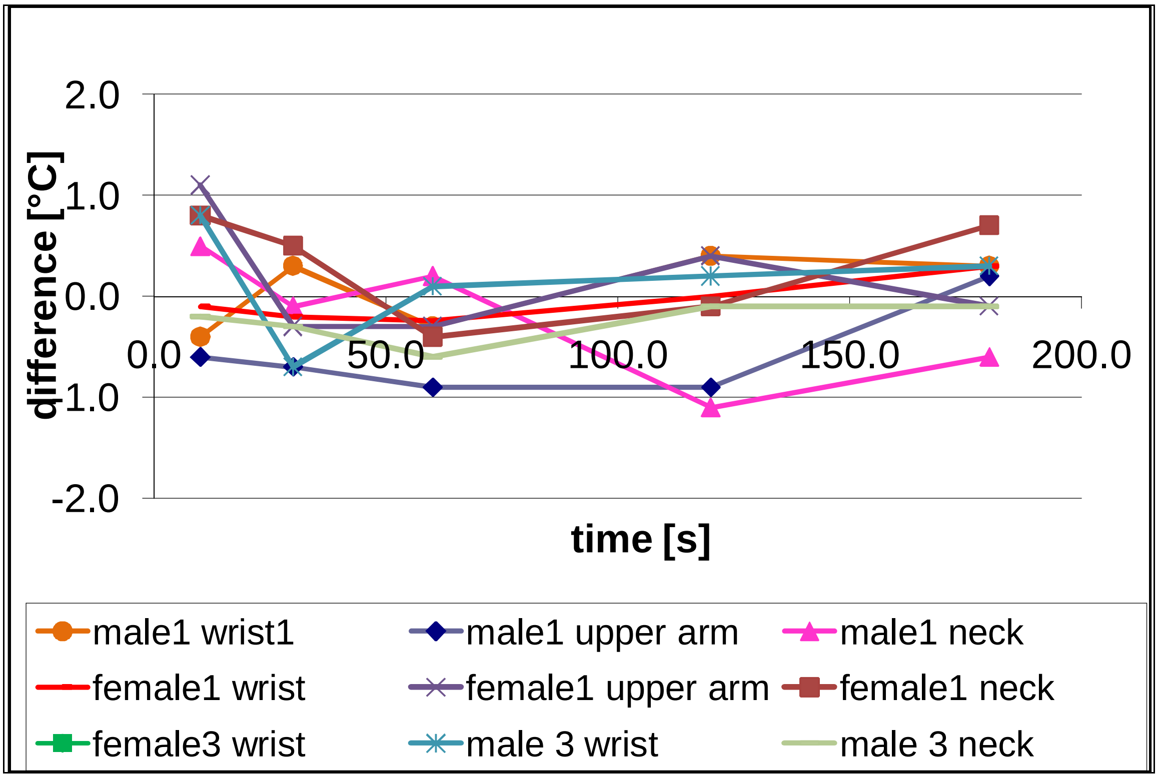


Figure 2-10: Temperature difference from reference temperature

After further calibration and software improvements, more accurate test results are obtained and the measurement error lies between ±0.5°C. Table 2-1 shows the measurement errors for two subjects.

Table 2-1: Measurement Errors

Time (Seconds) After...	Temperature (°C)		Reference (°C)		Error (°C)	
	Male	Female	Male	Female	Male	Female
10	22	22	22	22	0	0
30	25	24	24.5	24.2	+0.5	-0.2
60	26.5	25.8	26.8	26	-0.3	-0.2
90	30	28	30.5	28.5	-0.5	-0.5
120	35	32	35.6	32.4	-0.6	-0.4
180	35.5	32.9	36	33	-0.5	-0.1

After measuring temperature at neck, upper arm and wrist, the wrist position was considered to be the best place to measure skin temperature with the developed sensor. Figure 2.11 shows a quick adaption of the sensor to a change from a female to a male subject for measuring temperature at wrist.

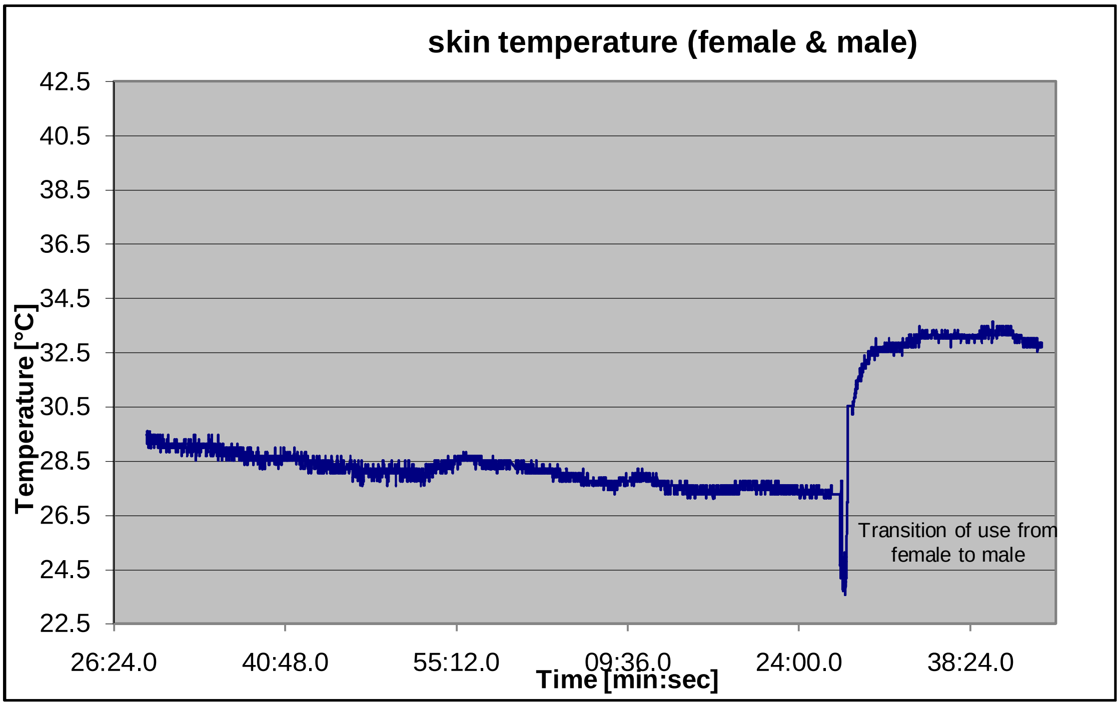


Figure 2-11: Skin temperature of female & male subjects

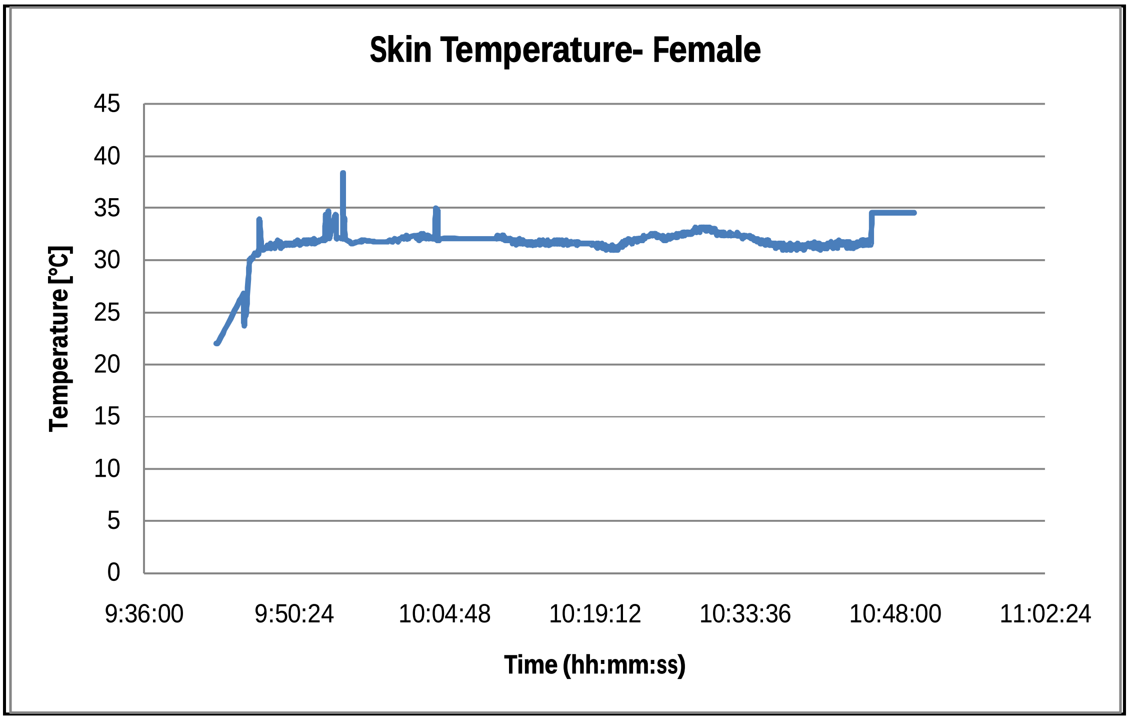
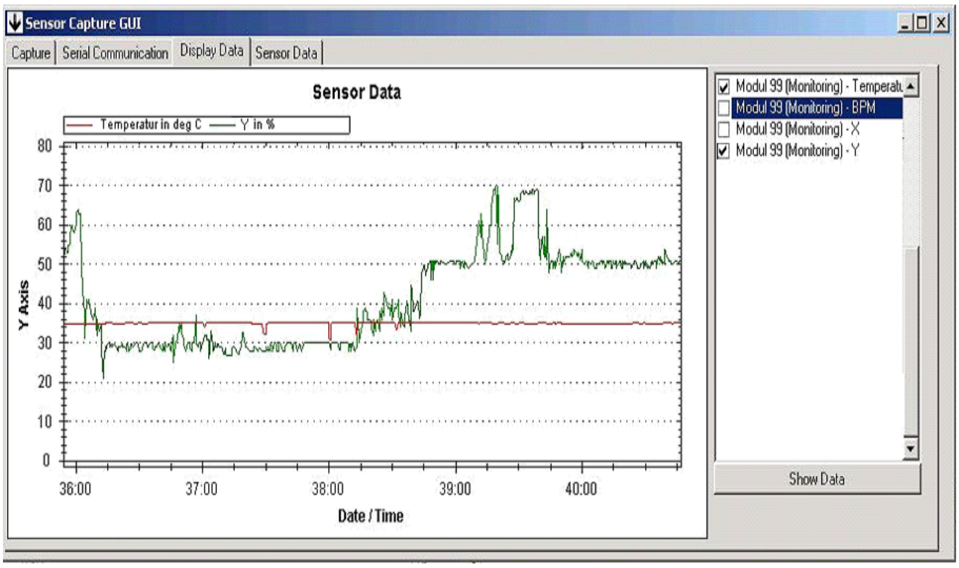


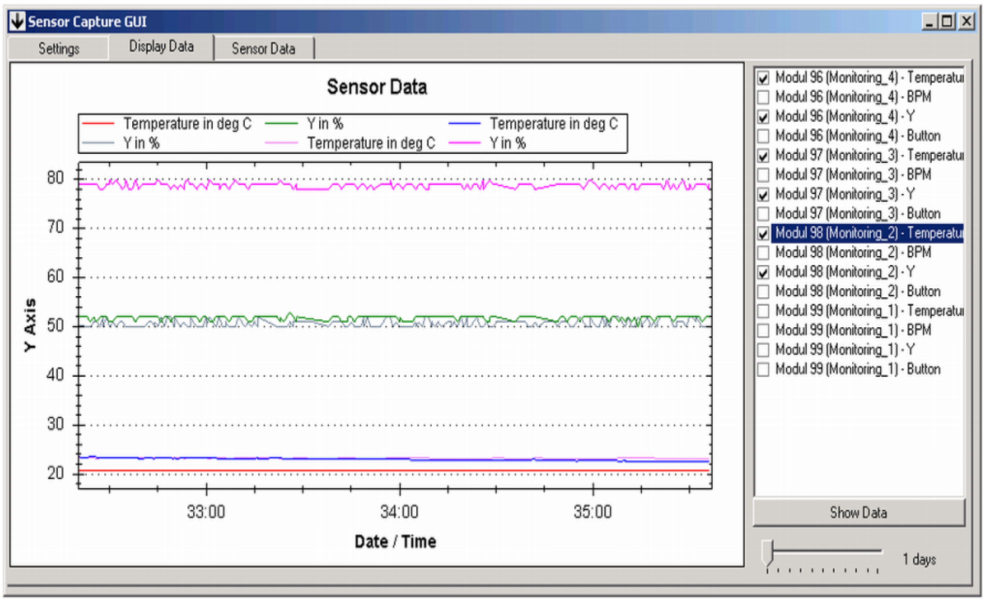
Figure 2-12: Skin temperature of female while and after running

After integrating two sensors together in one unit, the results were obtained from the wrist mounted device in Figure 2-13.



☑ Modul 99 (Monitoring) - Temperatu
☐ Modul 99 (Monitoring) - BPM
☐ Modul 99 (Monitoring) - X
☑ Modul 99 (Monitoring) - Y

Figure 2-13: Temperature & Impact sensor of subject



☑ Modul 96 (Monitoring_4) - Temperatu
☐ Modul 96 (Monitoring_4) - BPM
☑ Modul 96 (Monitoring_4) - Y
☐ Modul 96 (Monitoring_4) - Button
☑ Modul 97 (Monitoring_3) - Temperatu
☐ Modul 97 (Monitoring_3) - BPM
☑ Modul 97 (Monitoring_3) - Y
☐ Modul 97 (Monitoring_3) - Button
☑ Modul 98 (Monitoring_2) - Temperatu
☐ Modul 98 (Monitoring_2) - BPM
☑ Modul 98 (Monitoring_2) - Y
☐ Modul 98 (Monitoring_2) - Button
☐ Modul 99 (Monitoring_1) - Temperatu
☐ Modul 99 (Monitoring_1) - BPM
☐ Modul 99 (Monitoring_1) - Y
☐ Modul 99 (Monitoring_1) - Button

Figure 2-14: Temperature & Impact sensor of subject in GUI

Discussion: The results shown in Figure 2-11, Figure 2-12, 2.13 and Figure 2-14 depict the information received wirelessly from sensor units, however these results can be improved by removing the temperature outliers when sensor is worn/unworn. Moreover by improving the algorithm for the database noise/unrelated signals can be filtered, thus leaving more memory for the correct collated data received from sensor units. By having a flexible PCB, not only the efficiency of the temperature sensor will improve but also the mechanical strain on IC pins will be minimized. This will result in less noise/unrelated signals. With the microcontroller on a small PCB will result in fitting with sensor PCB enclosure, thus making it one complete unit and then it can be used to take more temperature results in different environments/ situations.

Chapter 3: Heart Rate measurement

3.1 Introduction

Heart rate is defined as the number of times the heart beats in one minute. Heart rate is a precisely regulated variable, which plays a critical role in health and disease [41]. Heart beat is formed as a result of contraction of ventricles (the lower chambers of the heart). Usually it is measured by stethoscope. The pulse is the bulge of an artery from the wave of blood coursing through the blood vessel as a result of the heart beat. The pulse is often taken to determine the heart rate, at the wrist in most cases [43].

However for at risk persons or in critical case a pulse can be taken at the neck, on the left side of the chest (near the heart) for continuous monitoring by using electrocardiogram in form of electrocardiograph (ECG) waveform. The electocardiogram (EKG) is simply a voltmeter that uses up to 12 different leads (electrodes) placed on designated areas of the body. Figure 3.1 shows a typical Electrocardiograph (ECG) waveform.

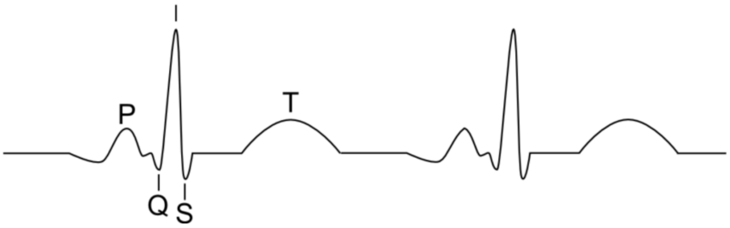


Figure 3-1: ECG waveform showing PQRST phases of the pulse

The analysis of heart rate and blood pressure variability has yielded additional information on the autonomic circulation control, which has proven to have diagnostic and prognostic implications in a number of clinically relevant conditions such as hypertension, acute myocardial infarction, heart failure, and predisposition to sudden cardiac death [44].

Numerous disorders can be detected by the means of analysing electrocardiograms (ECG) and changes in the heart rate. Therefore, it is very important to continuously monitor the heart rate during the patient's daily life and activities. Several sensors have been developed in the past in order to monitor the heart rate of a patient through ECG non-invasively. These include electric methods that use metal contacts on either sides of the patient's heart. Pressure sensors can be used to measure the pulsatile arterial blood flow. Piezoelectric sensors have been used in some applications that have been investigated in this project. Another method researched used high resolution temperature measurements to determine the heart rate.

"A high resolution temperature measurement system able to measure the temperature fluctuations down to 0.1mC based on thermistors which are fed with an AC signal to ensure a good signal to noise ratio. Temperature fluctuations were measured on the skin, in the proximity of different arteries, from where heart rate was determined."[42]
However this method is not convenient for ambulatory or long-term measurements due to the inconvenience to the patients caused by the electrodes.

3.2 The Heart rate Sensor

Three main approaches to detecting a patient's heart rate were researched. These were using electrical signals from the heart to determine the heart rate, using pressure sensors on the patient's wrist to detect the pulsation of blood in the veins, and using **n**ear-**i**nfra**r**ed **s**pectroscopy (NIRS) to detect the change in volume of blood in a patch of the patient's skin. Electrical methods require metallic contacts on either side of the patient's heart (i.e. one on each arm), while pressure based approaches are too prone to noise introduced by movement. Because these two methods were unfeasible for the project discussed in this thesis, it was decided to investigate NIR-sensors. NIR sensors are based on the following principle:

An infra-red light source is directed at the patient’s skin, and the amount of light that is reflected is measured by a photo-diode. Infrared light is absorbed by haemoglobin (metalloprotein in the red blood cells) but not by most other tissue types present in the human body. Hence, the amount of light detected varies with the pulse of the patient (as the amount of blood increases and decreases as a function of heart pulse). This can be measured to determine the heart rate of a patient.

Figure 3-2 shows the application of NIRS [45]. For exact measurements the heart rate sensor needs a part of the body where veins are very close to the skin’s surface. Suitable positions are the wrist, the earlobe and either the proximal phalanx of the finger or the fingertip.

A custom heart rate sensor was designed to read the patient’s heart beats per minute (bpm). The designed sensor is very small and inexpensive. Near-infrared spectroscopy is a non-invasive approach to determine the heart rate. The near-infrared spectroscopy method is a non-invasive variant more useful than others.

The NIR principle is implemented in the heart rate sensor. It consists of a Gallium Arsenide (GaAs) infrared LED with a wavelength of 940nm. It transmits the signal through the tissue and is received by an infra red sensor-Phototransistor (SDP8406).The time between the impulses are measured and counted so that frequency is obtained. This frequency is transformed with a calculation (explained in detail in chapter 6) into a value in BPM (beats per minute).

$$\text{BPM} = (\text{Frequency}*6)/10$$

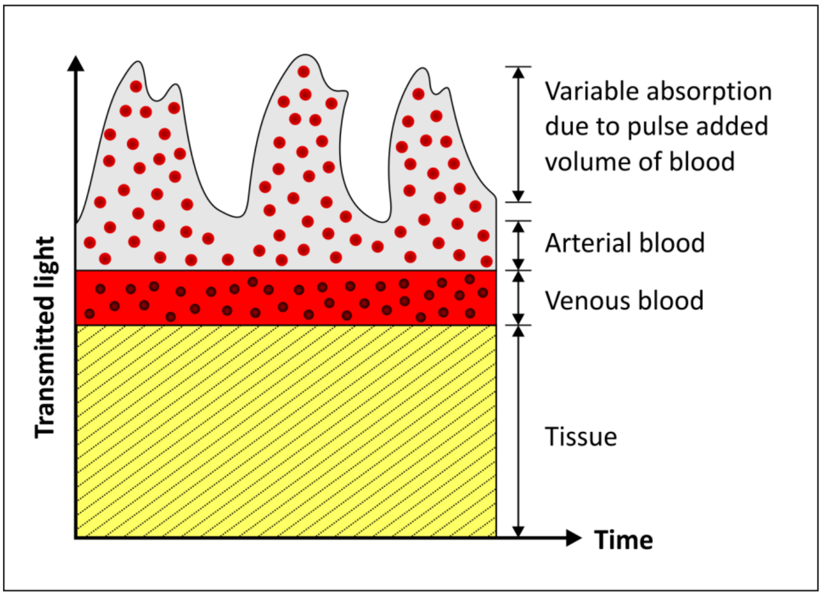


Figure 3-2: Application of NIR light in a NIRS [45]

This approach is applicable in this project as we require the sensors to be non-invasive and also be compact in order for the user to be able to wear them during their everyday activities. The block diagram is shown in figure 3.3.

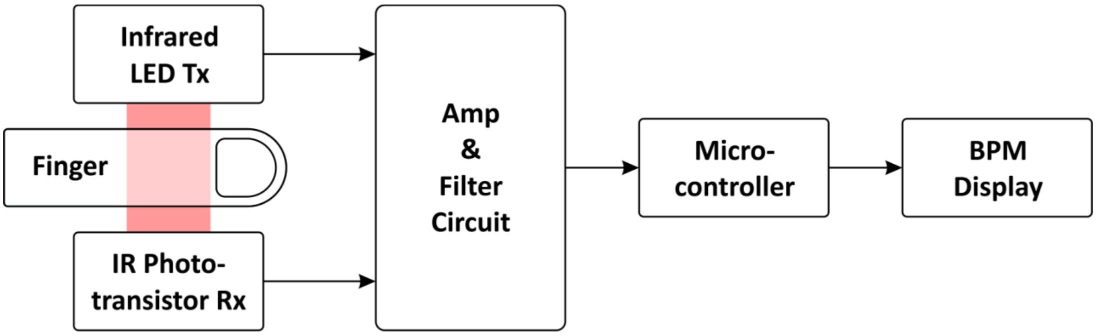


Figure 3-3: Block Diagram of Heart Rate Sensor

3.3 Construction

A silicon phototransistor (SDP8406), moulded into a flat side-facing package and a GaAs Infrared Emitting Diode were used in the sensor. For testing purposes first 5mm

GaAs infrared LED was used, which later on was replaced by 1.9mm axial LED, shown in Figure 3-5. The axial LED was mounted on a PCB for better hold on the fabric, used for holding and positioning senor around the finger (see Figure 6-17). The technique used to measure the heart rate is based on near-infrared spectroscopy (NIRS). This allowed for designing a non-invasive and low cost method of measuring the pulse. The amount of light that was detected by the phototransistor varied with the patient's pulse, as the amount of absorbed IR light changed with the flow of blood, which is directly linked to the heart rate. These sensors were chosen as they are very compact and inexpensive. Also, they are less susceptible to noise than pressure sensors. The near-infrared spectroscopy technique used is non-invasive and helps for the continuous monitoring of the physical parameters without causing much discomfort to the user.

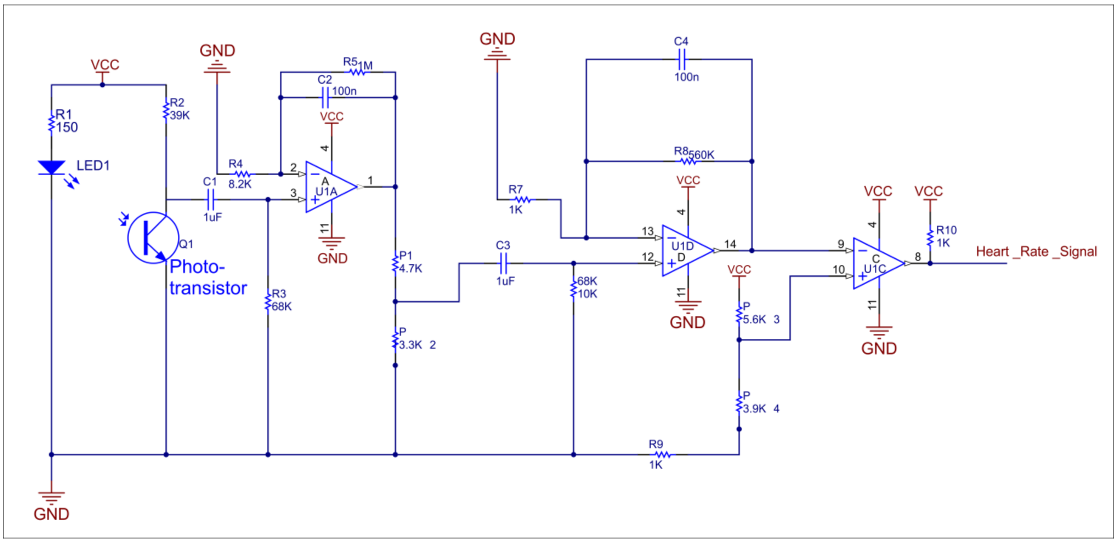
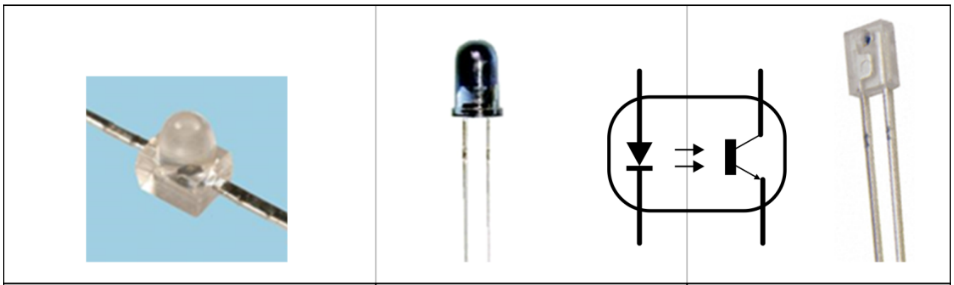


Figure 3-4: Circuit Diagram of Heart Rate Sensor

The circuit diagram of the heart rate sensor shown in figure 3.4. The circuit consists of 2 operational amplifiers, a comparator, a microcontroller, and an LCD. The operational amplifier used is the Low Power Quad Operational Amplifier LM324 by National Semiconductor [47]. Two amplifiers are used to amplify and a third one is used as a comparator to get amplified pulse.



OPTEK - OP270 - IR, 1.9mm, DOMED, AXIAL LED	**GaAs Infrared LED**	**Phototransistor (SPD8406)**

Figure 3-5: Types of LED's, phototransistor used

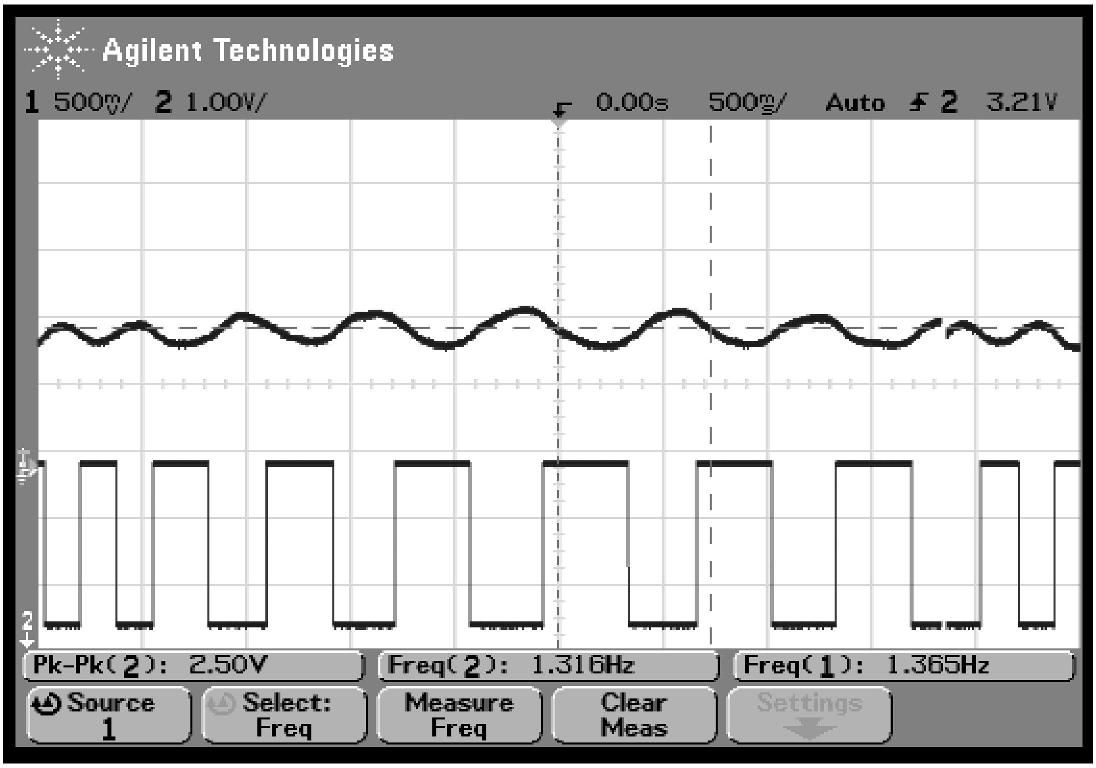
3.4 Technique

The basic principle of Near-infrared spectroscopy is that the amount of light absorbed by the photodetector varies with the patient’s pulse (pulsatile blood flow) as this light is only absorbed by the haemoglobin in the blood. In other words, the amount of light that is detected by the phototransistor varies with the patient’s heart pulse, as the amount of absorbed IR light changed with the flow of blood, which is directly linked to the heart rate.

An infra-red light source is directed at the individual’s skin, and the amount of light that is reflected is measured by a phototransistor. Infrared light is absorbed by haemoglobin but not by most other tissue present in the human body. Because of this, the amount of light detected varied with the pulse of the patient (as the amount of blood increases and decreases as the heart pulses). This can be measured to determine the heart rate of a patient.

3.5 Experimental Results

To get the best and most accurate results with the heart rate sensor we chose to measure the pulse at the finger tip like commercial devices do. Nevertheless, it was checked for working on the wrist and the finger, too. The signal (analog) originally was too small to detect, and without amplification proved to be too noisy to extract the heart rate. Because of this, operational amplifiers were used to extract the heart rate signal. After amplifying, the signal was fed to comparator, resulting in output in the form of pulses. The signal in the form of pulses is interfaced with a microcontroller through its digital port for further processing. The waveforms at the collector of the photo-transistor and at the final point (Heart_rate_signal) are shown in Figure 3-6.



1 500mV/ 2 1.00V/ 0.00s 500ms/ Auto 2 3.21V

Pk-Pk(2): 2.50V

Freq(2): 1.316Hz

Freq(1): 1.365Hz

Figure 3-6: The heart rate signal at the collector of the photo-transistor and at the output (Heart_Rate_signal)

A sport watch (WR30M) with a similar sensor was used as reference. Heart rate measurement data were collected simultaneously. The sensor after filtering provided a clean wave that when observed on an oscilloscope confirmed that the sensor was correctly measuring the patients pulse.

Table 3-1: Experimental Results

Subject	Heart Rate (BPM)		Reference: Pulse watch (BPM)		Error (BPM)	
	Male	**Female**	**Male**	**Female**	**Male**	**Female**
1	58	55	57	54	+1	+1
2	78	59	77	57	+1	+2
3	54	58	55	58	-1	0
4	72	68	71	71	+1	-3
5	68	65	67	64	+1	+1
6	75	58	75	59	0	-1
7	69	64	69	64	0	0
8	85	87	86	86	-1	+1
9	79	77	81	79	-2	-2
10	70	61	69	62	+1	-1
11	75	92	77	93	+2	-1
12	65	58	64	58	+1	0

Figure 3-7 shows the heart rate in beats per minute displayed from data collected in database of GUI.

After filtering unwanted signal values, the correct heart rate values in beats per minute (BPM) are shown from results taken from a user between 9:40 and 10:50 am from the database in Figure 3-8

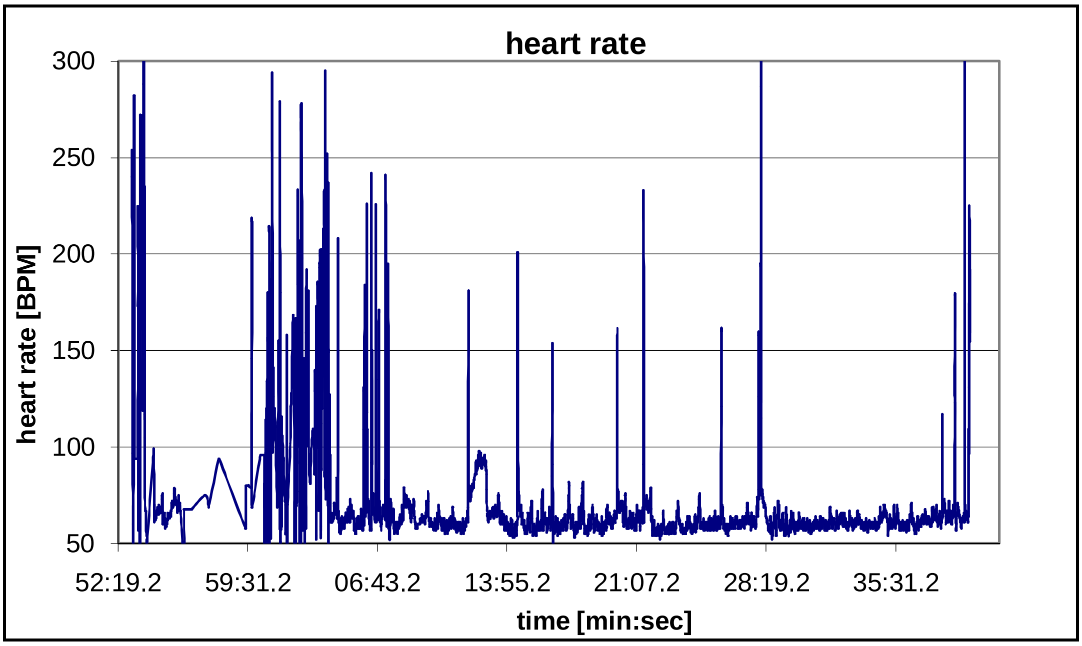


Figure 3-7: Heart rate signal before filtering

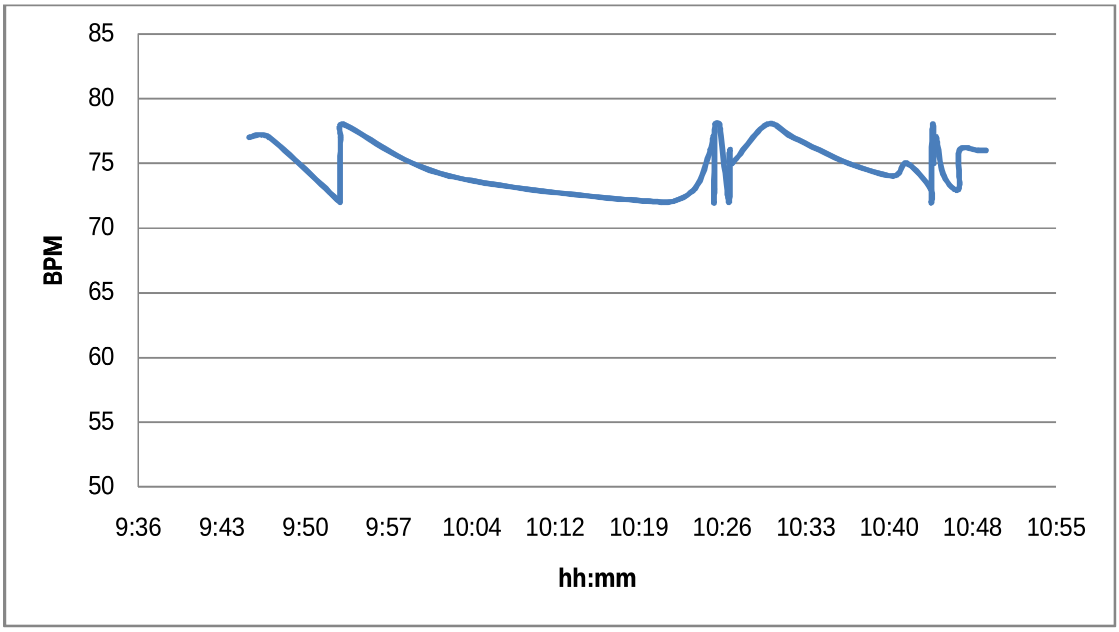
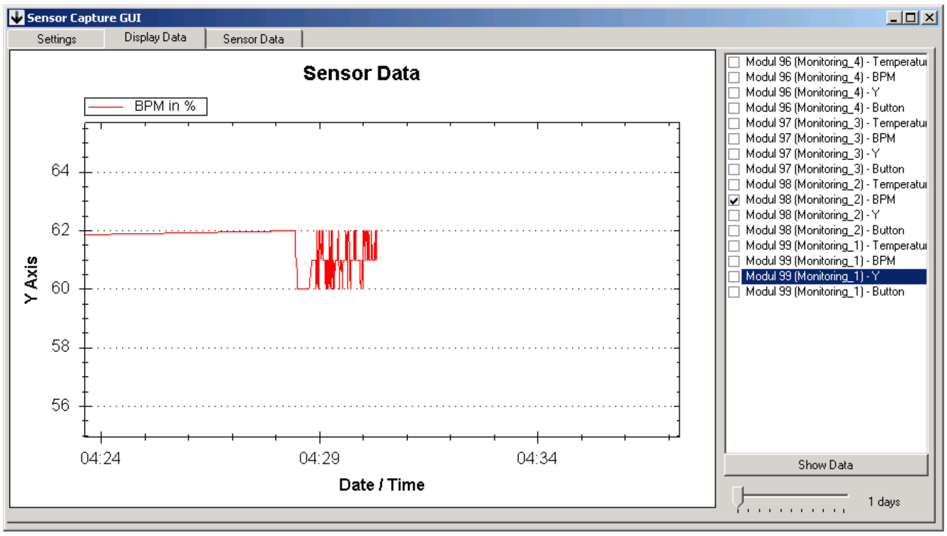


Figure 3-8: BPM after filtering

From display in GUI it can be seen that the signal is much stronger and steadier. The signal was also readable during movement. When the sensor in the form of ring around finger the sensor did not move, which resulted in a signal which was much stronger and steadier (steady signal response is greatly dependant on maintaining a constant distance from the sensor to the skin).



☐ Modul 96 (Monitoring_4) - Temperatu
☐ Modul 96 (Monitoring_4) - BPM
☐ Modul 96 (Monitoring_4) - Y
☐ Modul 96 (Monitoring_4) - Button
☐ Modul 97 (Monitoring_3) - Temperatu
☐ Modul 97 (Monitoring_3) - BPM
☐ Modul 97 (Monitoring_3) - Y
☐ Modul 97 (Monitoring_3) - Button
☐ Modul 98 (Monitoring_2) - Temperatu
☑ Modul 98 (Monitoring_2) - BPM
☐ Modul 98 (Monitoring_2) - Y
☐ Modul 98 (Monitoring_2) - Button
☐ Modul 99 (Monitoring_1) - Temperatu
☐ Modul 99 (Monitoring_1) - BPM
☐ Modul 99 (Monitoring_1) - Y
☐ Modul 99 (Monitoring_1) - Button

Figure 3-9: BPM result

It was observed that while measuring heart rate with the sensor developed, it showed more accuracy (see Figure 3-10) in terms of continuous monitoring in comparison to the finger pulse oximeter [66] used as reference.

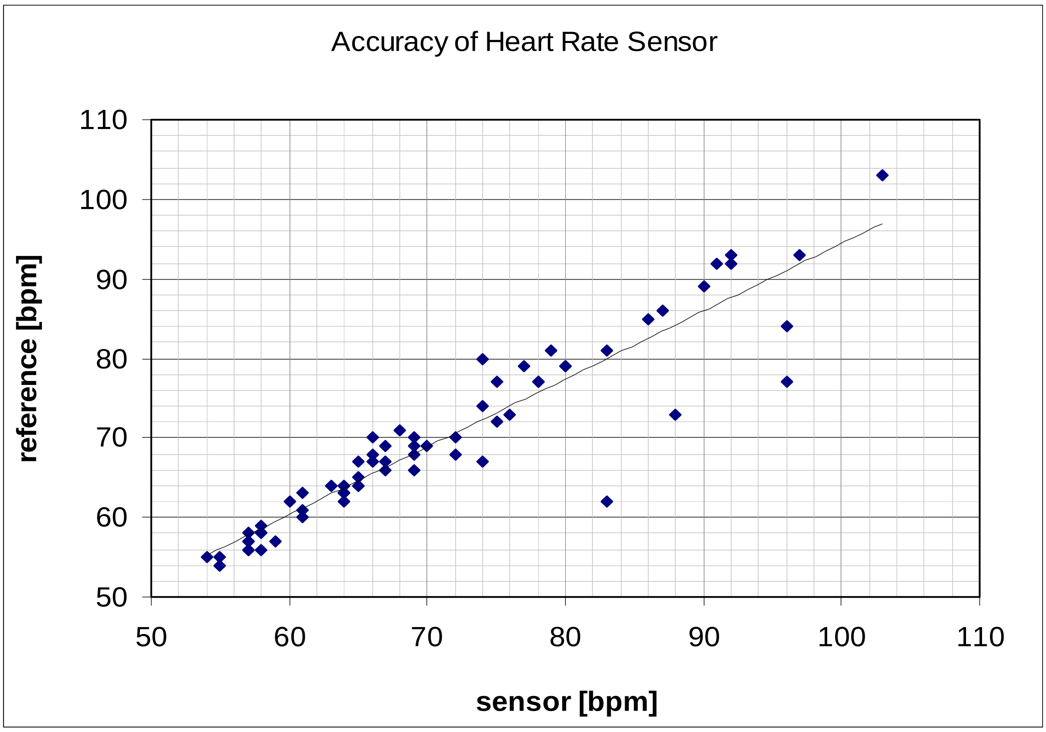


Figure 3-10: Accuracy of Heart Rate Sensor: A heart rate change can be detected very quickly using the developed sensor system.

Chapter 4: Impact sensor

4.1 Introduction

Impact or fall detection sensors are ideal for sensing whether or not a person has fallen over, indicating serious injuries. The ideal way to implement this sensor was to use an accelerometer.

Injuries sustained from falls can be fatal resulting in serious physical and physiological consequences. Falls are a common issue, but they are difficult to define austerely. Injuries may result in broken bones or superficial cuts to the skin and may include damage to underlying bone tissue [48, 49, and 50]. In most countries worldwide, falls are the leading cause of death or injury-related hospitalisation among people of 65 years and older in society [51]. It would seem that falls are just another thing in day to day life but research in Ireland in 2005 showed that falls in older adults accounted for 83% of all fatal falls [52] and on estimation annually cost 10.8 million Euro for one Irish hospital [53]. Fall related injuries resulting in admissions of older adults are a significant financial burden to health services worldwide.

Recent advances in microelectronics, miniaturisation of electronic components and at the same time a reduction in overall cost of technology have resulted in more affordable and wearable health monitoring systems. According to [54], the fundamental approach of using accelerometers to detect falls was first published by [55], [56]. In this approach, a change in body orientation from upright to lying down that occurs immediately after a large negative acceleration indicates a fall. These two conditions have been incorporated into fall detection algorithms using accelerometers [57]. The authors in [68] discussed the feasibility of using a wireless sensor network to detect fall events by observing different activities unique acceleration profiles.

4.2 Motion Detection Sensor (Impact Sensor)

After researching various ways of fall detection, the ideal way to implement a fall detecting sensor in this project was to use an accelerometer. In the prototype the ADXL213 from Analog Devices [59] was used, capable of measuring dynamic and static changes in acceleration in the horizontal and vertical axis. This sensor operates by using small plates suspended by springs, which deflect when subjected to acceleration/gravity. The ADXL213 was chosen as it was more readily available and provided static force measurement.

Figure 4-1:ADXL213 accelerometer

In the prototype an Analog Device ADXL213 [12] was used, capable of measuring dynamic and static changes in acceleration in the horizontal and vertical axis. This sensor operates by using small plates suspended by springs, which deflect when subjected to acceleration/gravity. The outputs are digital signals of which the duty cycles (ratio of pulse width to period) are proportional to acceleration (30%/g). The duty cycle outputs can be directly measured by a microcontroller without an A/D converter or glue logic.

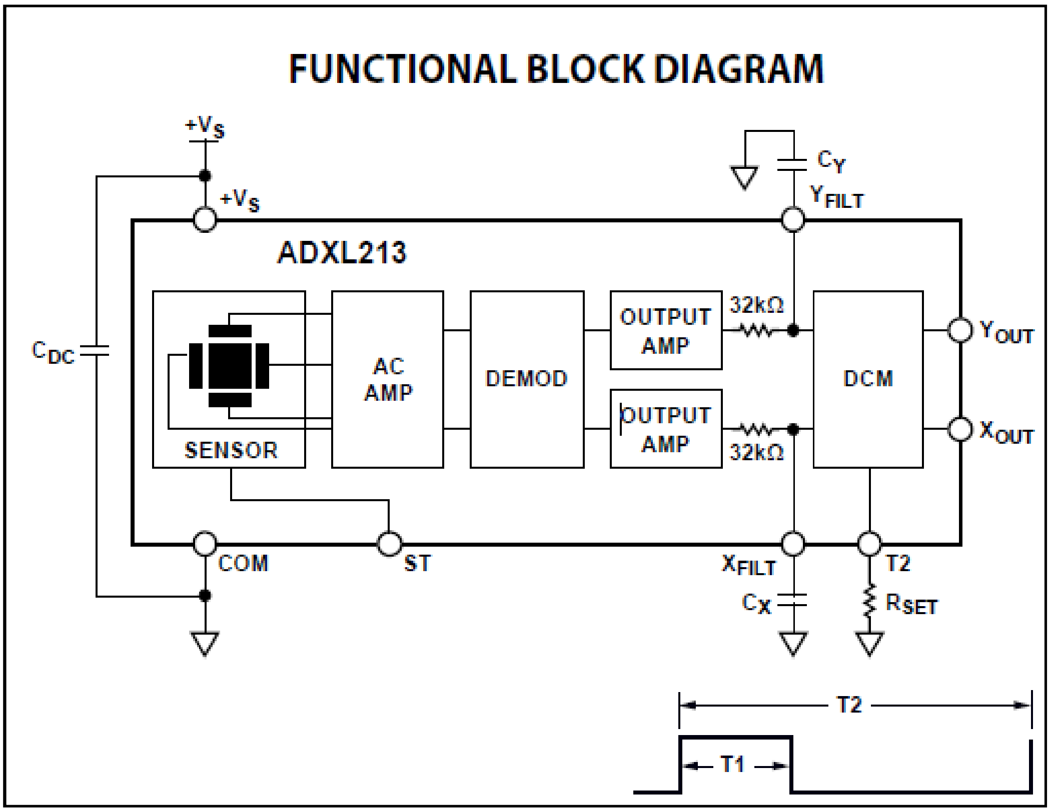


Figure 4-2: Functional Block Diagram of ADXL213

4.3 Construction

The ADXL213 is a complete dual axis acceleration measurement system on a single monolithic IC, containing a polysilicon surface-micromachined sensor and signal conditioning circuitry to implement open-loop acceleration measurement architecture. The ADXL213 is capable of measuring both positive and negative accelerations with a dynamic range of ±1.2 g.

The acceleration can be determined by measuring the length of the positive pulse width (t1) and the period (t2). The nominal transfer function of the ADXL213 is:

Sensitivity = Minimum magnitude of input signal required to produce a specified output signal having a specified signal-to-noise ratio, or other specified criteria.

Acceleration = ((t1/t2) – Zero g Bias)/Sensitivity

$$Acceleration = \frac{\frac{t_1}{t_2} - ZeroGBias}{Sensitivity}$$

Where in the case of the ADXL213

Zero g Bias = 50% nominal

Sensitivity = 30%/g nominal

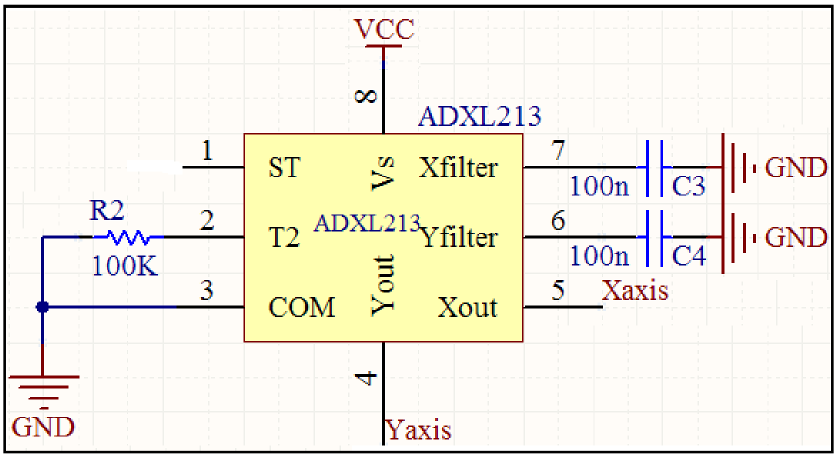


Figure 4-3: Circuit Diagram of motion detection sensor

4.4 Technique

As mentioned above, the sensor operates by using small plates suspended by springs, which deflect when subjected to acceleration/gravity. The outputs are digital signals of which the duty cycles (ratio of pulse width to period) are proportional to acceleration. This duty cycle can be directly measured using microcontroller.

Due to the surface-micromachined polysilicon structure of the sensor built on top of the silicon wafer, polysilicon springs suspend the structure over the surface of the wafer and provide a resistance against acceleration forces. Deflection of the structure is measured using a differential capacitor that consists of independent fixed plates and plates attached to the moving mass. The fixed plates are driven by 180° out-of-phase square waves.

Acceleration deflects the beam and unbalances the differential capacitor, resulting in an output square wave with amplitude proportional to acceleration. Phase sensitive demodulation techniques are then used to rectify the signal and determine the direction of acceleration.

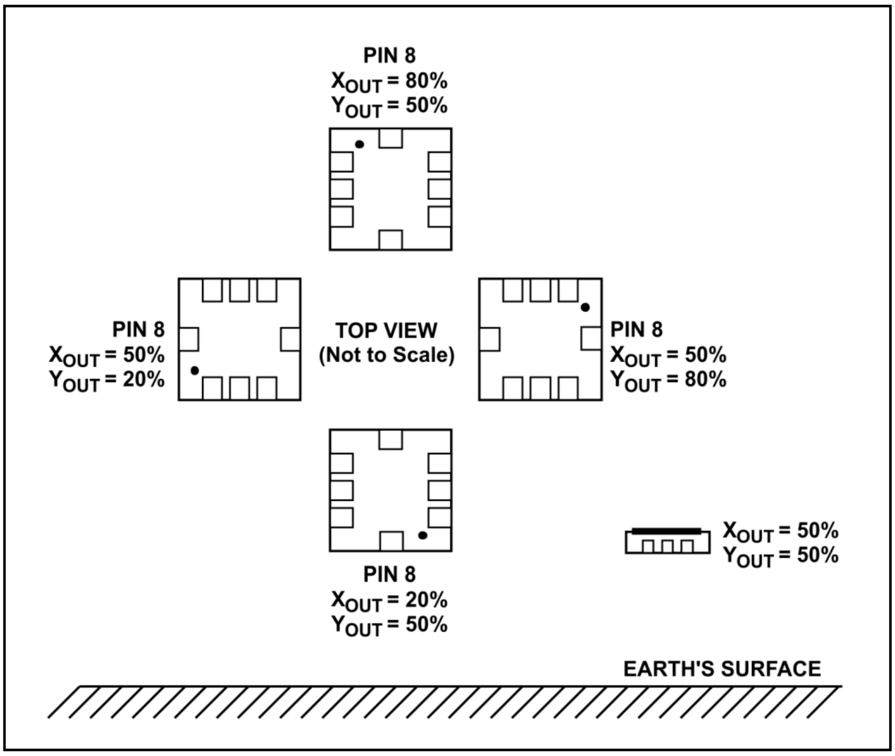


Figure 4-4: Theory of operation

4.5 Experimental Results

The output of the accelerometer was tested with walking and simulated falling. Figure 4.5 displays the movement of the user in respect to duty cycle (%) and Figure 4-5 shows the various movements of the user in the given time period

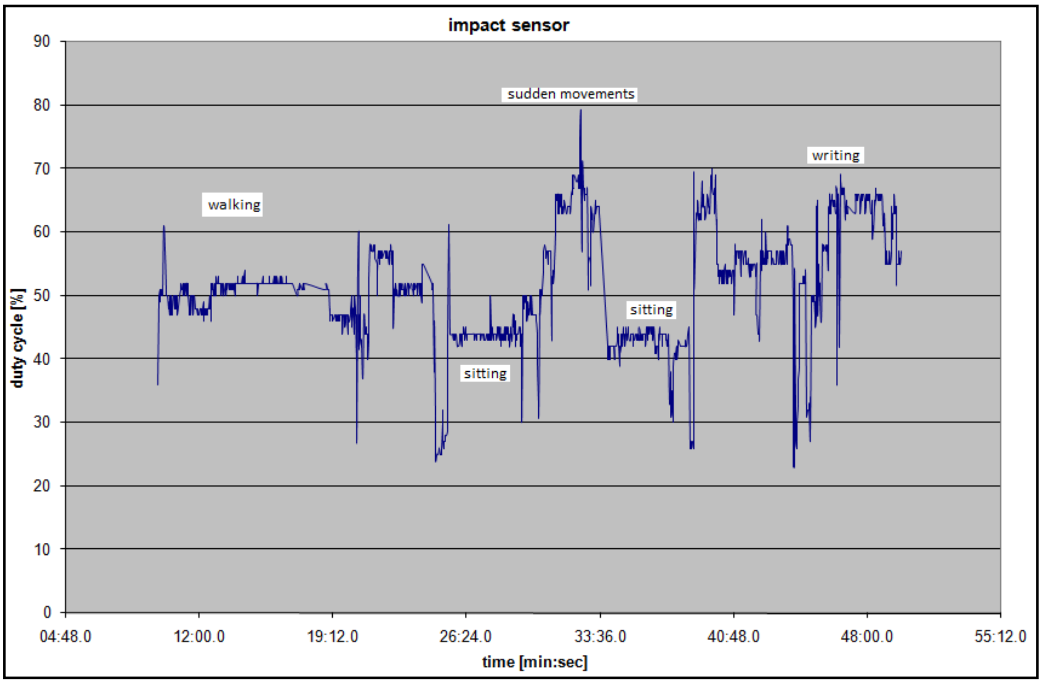
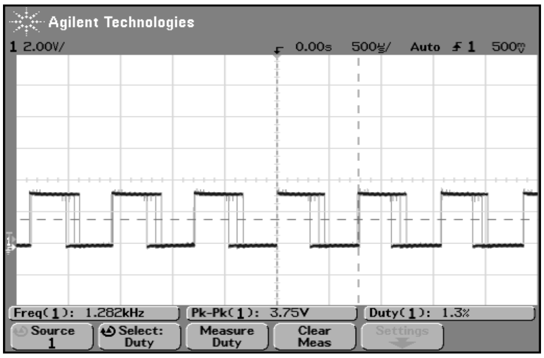
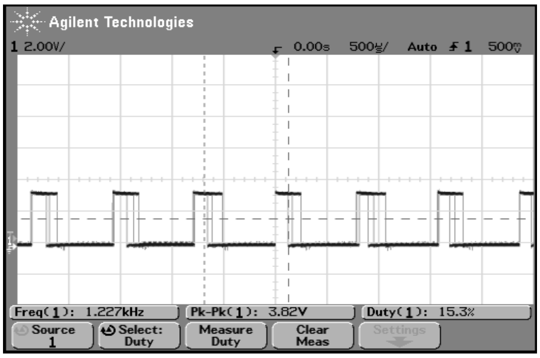


Figure 4-5: Various positions of user

The output of the accelerometer in the form of duty cycle percentage is displayed on oscilloscope with duty cycle of 1.3%, 15.3%, 19.7%, 25.5%, 50% and 72%. These different duty cycle values indicate the various positions of the person wearing the impact sensor. When the user is standing in normal position, value of duty cycle is around 50%.





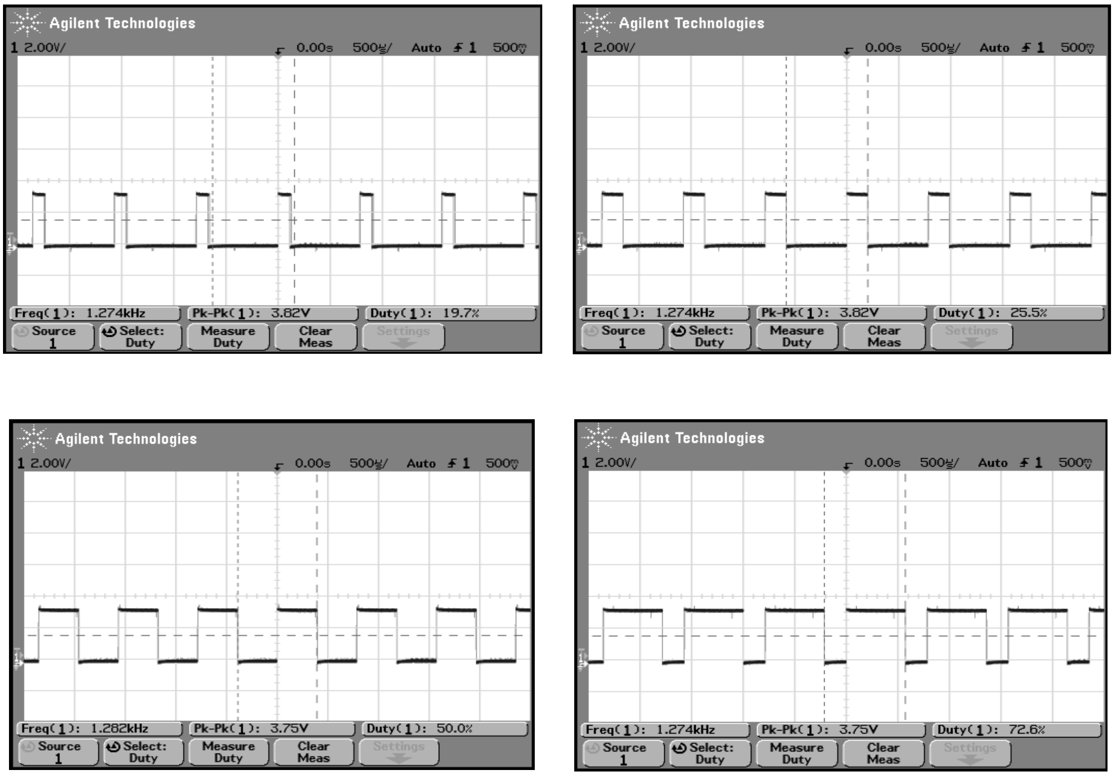
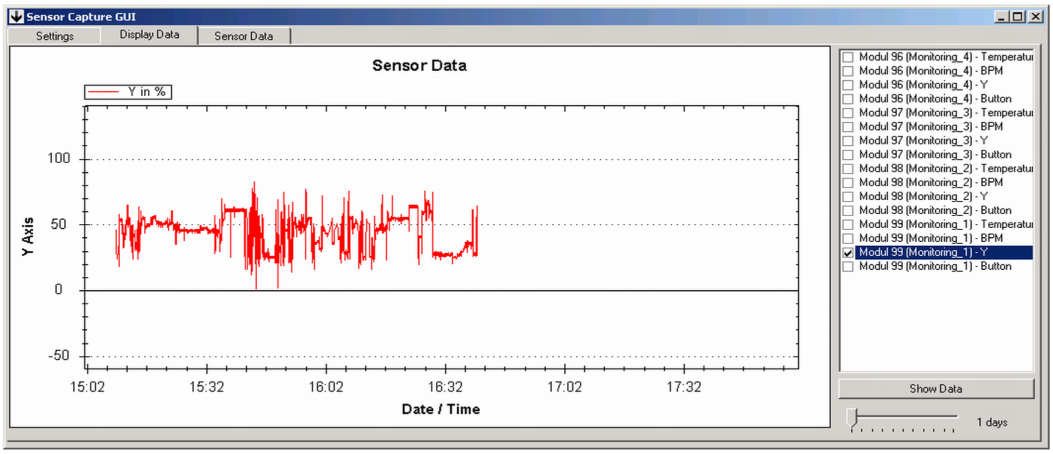


Figure 4-6: Duty cycle values of accelerometer in oscilloscope

The output of the accelerometer was tested with walking, everyday movements like sitting, standing, writing etc, and simulated falls. The results showed the difference was simple to detect and proved the accuracy of the algorithm. Figure 4-7 shows the impact sensor output.



☐ Modul 96 (Monitoring_4) - Temperatui
☐ Modul 96 (Monitoring_4) - BPM
☐ Modul 96 (Monitoring_4) - Y
☐ Modul 96 (Monitoring_4) - Button
☐ Modul 97 (Monitoring_3) - Temperatui
☐ Modul 97 (Monitoring_3) - BPM
☐ Modul 97 (Monitoring_3) - Y
☐ Modul 97 (Monitoring_3) - Button
☐ Modul 98 (Monitoring_2) - Temperatui
☐ Modul 98 (Monitoring_2) - BPM
☐ Modul 98 (Monitoring_2) - Y
☐ Modul 98 (Monitoring_2) - Button
☐ Modul 99 (Monitoring_1) - Temperatui
☐ Modul 99 (Monitoring_1) - BPM
☑ Modul 99 (Monitoring_1) - Y
☐ Modul 99 (Monitoring_1) - Button

Figure 4-7: Impact sensor output in GUI

It can be observed in figure 4.7 that standing and walking positions are noticeable. After some time the user is writing and two falls (under controlled conditions) detected (see Figure 4-8).

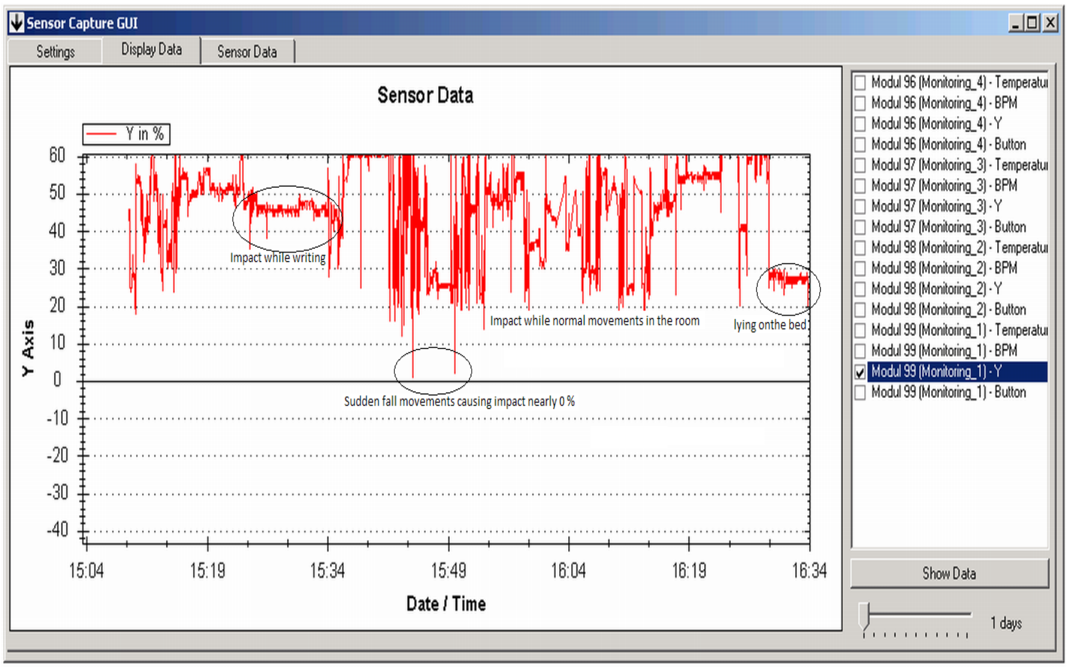


Figure 4-8: Duty cycle variation under dynamic transitions

On Analysis of the impact sensor outputs, it shows the difference in duty cycle during falls (under controlled conditions). The impact sensor value on a fall on the floor is nearly 0%, while a fall on mattress is between 3-5%, refer Figure 4-9.

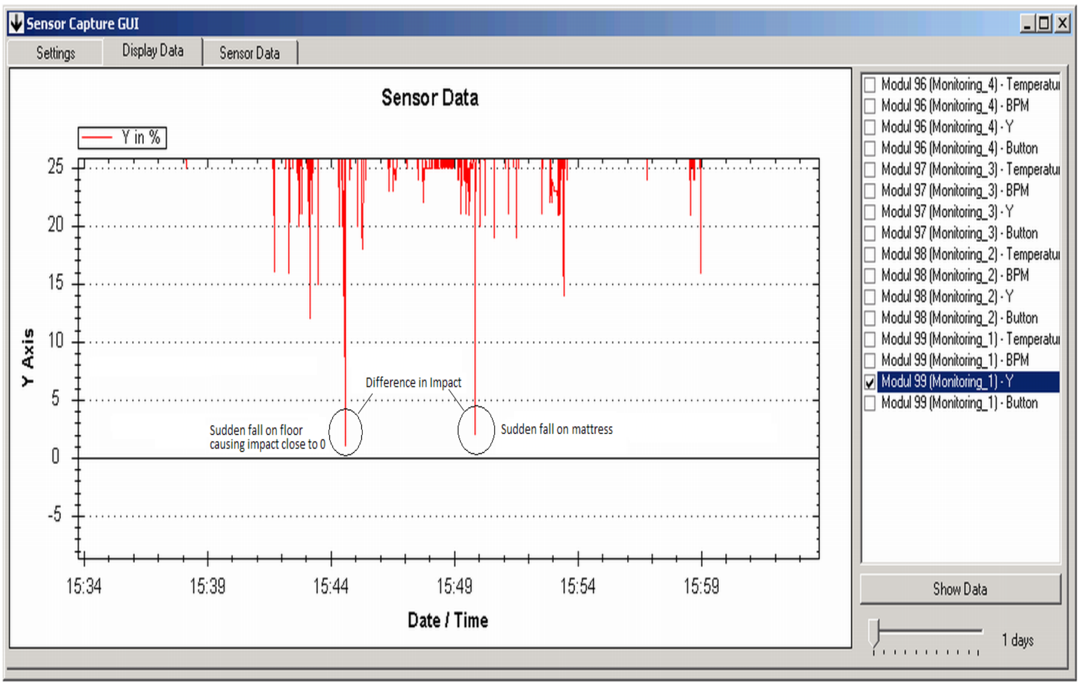


Figure 4-9: Difference in duty cycle on falls

It is mentioned in the code that if the value of the duty cycle of impact sensor lies below or above a given range an alarm is sent to notify the situation. But this can also generate false alarms.

As mentioned earlier, the ADXL213 is dual axis accelerometer having the capacity to measure both dynamic and static accelerations. The sensitivity of accelerometer (ADXL213) is very high in that it detects even slight variations in the posture, resulting in many false positives from fall-like activities such as sitting down quickly and jumping, thus generating a false alarm. To avoid/decrease false alarms, complex inference techniques are required to improve activity recognition accuracy of the sensor.

After researching work related to fall detection, falls are the most widespread domestic accidents among the elderly. Furthermore, it frequently happens that elderly people who have previously experienced a fall fear a new fall and sink gradually into inactivity. Less mobility leads progressively to an increase in the risk of a fall [67]. Literature review reveals that reliable fall detection based raw sensor data is much discussed in literature and requires algorithm development of wide scope based on deeper knowledge of specific [69]application principles as outlined in [68] to monitor a range of human movement.

Chapter 5: Bed sensors

5.1 Introduction

In the absence of helpers at night for a person living alone a preventive measure is required to avoid sudden accidents in the home environment. A person can stay in physiologically stable condition on a bed. The main aim of this kind of sensor system is to monitor the sleeping pattern of a person, who is living all alone in a house and needs immediate medical attention in case of an emergency or if an abnormal situation arises. This kind of intelligent sensor can perform various data collection and processing functions, including data interpretation, storing data on a database. While several sensors are readily available off the shelf, making them “intelligent” in the context of a specific application (such as monitoring of the elderly) is always a challenging task. In case of an intelligent bed-sensor, it not only detects the usage pattern of the bed but also has capabilities to collate the data and flag out abnormalities.

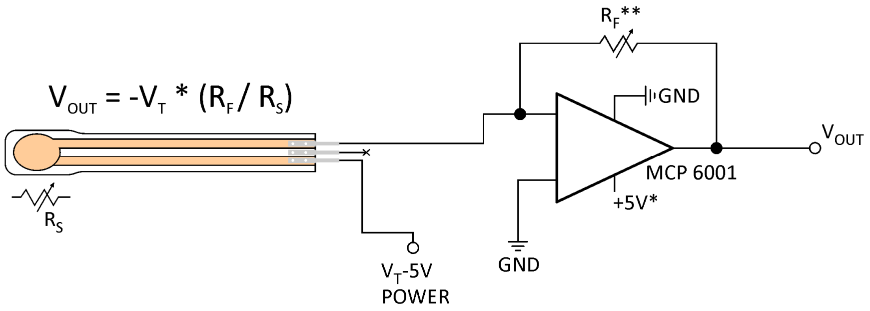
By monitoring some of these physiological quantities and creating a database of collected data over several months or years could be helpful for many purposes such as retrospective studies of the patients or a data base for public health studies.

Bed Monitoring Wireless Sensor monitors the abnormalities in the person’s daily sleep window period and may act as a lifesaver. Let us consider three kinds of common situations, which a person can experience when living alone. Suppose a person who has a habit of waking up every morning at a particular time slot, but on a particular day, tends to be lying on the bed till late in the morning. Suppose a person who has a habit of going to use the bed at a particular time slot every night, but on a particular night if he/she is not on the bed even after the given time slot. This situation may indicate that the person might have had a serious health issue or fallen down somewhere in the house, which made him/her incapacitated to reach for medical assistance.

In the third scenario, if the person is lying on the bed and there is constant movement on the bed, which may indicate the person may not be feeling well, maybe needing some assistance. Many types of bed monitoring devices have been developed in the past, but these systems are mainly used to monitor/detect fall risk-patients and residents in a hospital scenario. A prototype of bed monitoring device has been developed, which has demonstrated its feasibility for monitoring the activity of a subject in bed. Being different from the previous developed methods, this device monitors the subject 24-hours 365 days with a low cost, less complexity and good accuracy.

5.2 Bed Sensors

The bed sensor system consists of four sensor units with a driving circuit and a microcontroller. The recommended driving circuit [60] is shown in Figure 5-1, it is driven by a -5 V DC excitation voltage. This circuit uses an inverting operational amplifier arrangement to produce an analog output based on the sensor resistance and a fixed reference resistance (R_F). The output of the inverting amplifier is connected to an analog-to-digital converter provide in the microcontroller.



- * Supply Voltages should be constant
- ** Reference Resistance R_F is 1kΩ to 100kΩ
- Sensor Resistance R_S at no load is > 5MΩ
- Max recommended current is 2.5mA

Figure 5-1: Sensor driving circuit

The ADC is used to change this voltage to a digital output. In this circuit, the sensitivity of the sensor could be adjusted by changing the reference resistance (R_F) and/or drive voltage (V_T); a lower reference resistance and/or drive voltage will make the sensor less sensitive, and increase its active force range. In the circuit shown, the dynamic force range of the sensor can be adjusted by changing the reference resistor (R_F) or by changing the Drive Voltage (V_O).

5.3 Construction

Figure 5-2, shows the actual circuit with four force sensors, the outputs of the inverting amplifier S1, S2, S3, S4 are connected to the four channels of analog to digital converter. Since Silabs 8051F020 is used in this project, the analog reference voltage of ADC converter is 2.4 V. Therefore the maximum output response of the sensor is made to be less than/equal to 2.4V. This is done by adjusting the R_F potentiometer.

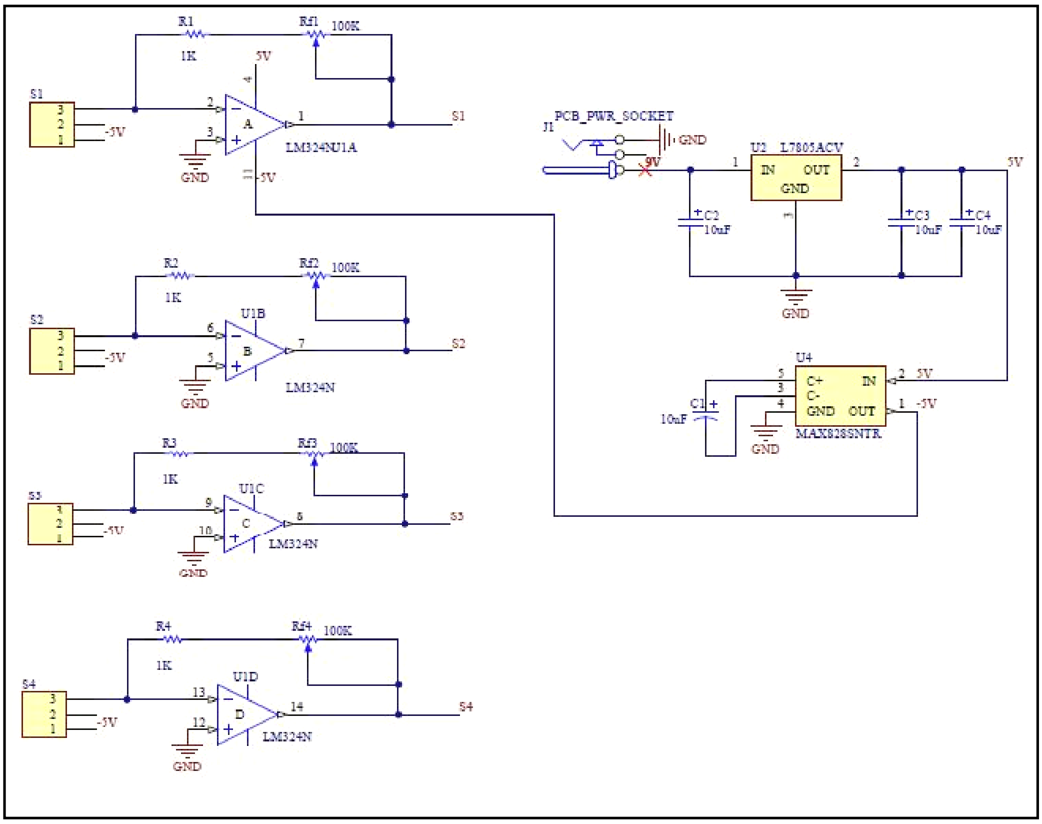


Figure 5-2: Bed Sensor Circuit Diagram

Since the driving circuit needs two supply voltages +5V to power up the inverting amplifier and -5V to drive sensors. LM7805, a 3-terminal positive voltage regulator is used to produce a constant +5V supply. MAX828 chip is used to produce -5V. The MAX828 are ultra-small monolithic, CMOS charge-pump inverters. It is used to generate a -5V supply from a +5V logic supply to power analog circuitry. This IC comes in a 5-pin SOT23-5 package and can deliver 25mA with a voltage drop of 500mV. Since MAX828 is a Switched-Capacitor Voltage Inverters it caused a 12 KHz noise at the output. This noise can considerably affect the output of the sensor, resulting in unstable output. To eliminate this noise a simple low-pass filter is employed at the output terminal of each sensor.

Using 0-25 lbs Flexiforce sensor, the force to voltage characteristics are noted and shown in the Figure 5-3,

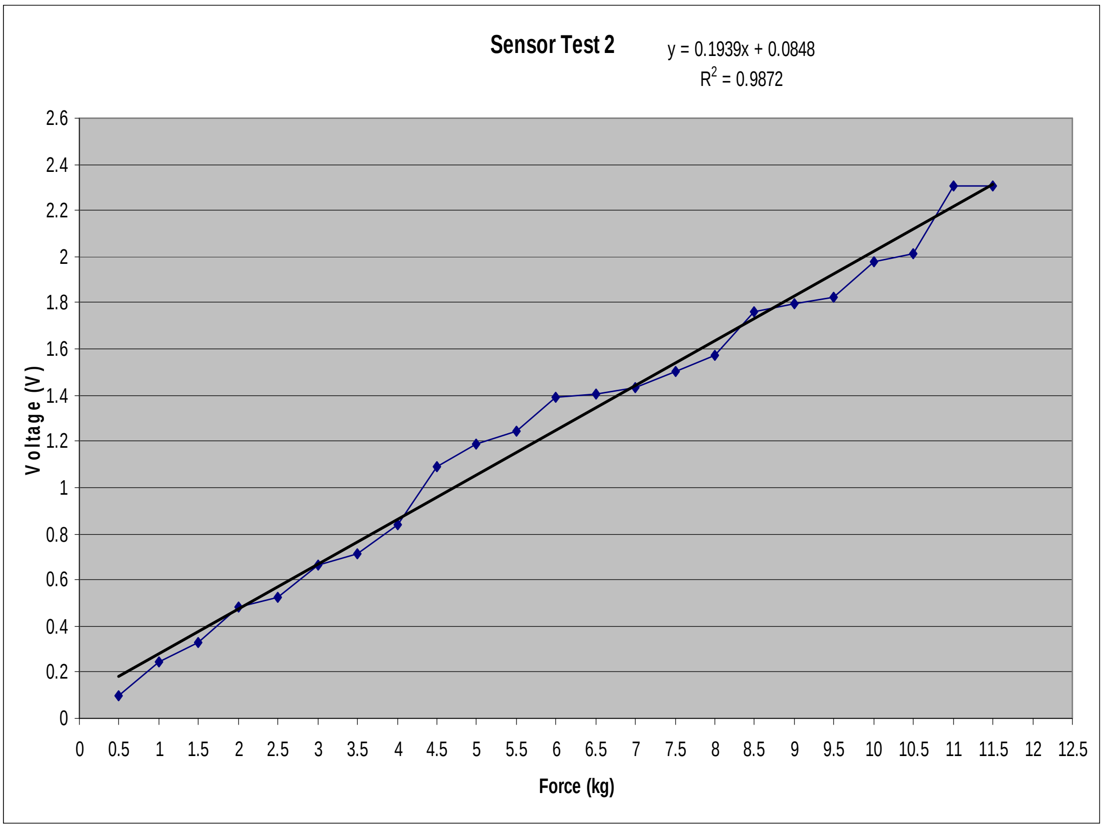


Figure 5-3: Force to Voltage characteristics

The sensing part of the sensor is sandwiched (taped) between two small metal discs (having a diameter almost the same as the active sensing area of the sensor) for equal weight distribution. The experiment was started off with 500 g of weight. Reading was taken with increment of 500 g weight until 11.5 Kg's. Three consecutive reading were taken for every additional weight added to the sensor to check its repeatability. After 11.5 Kg's the response of the sensor was saturated.

5.4 Technique

Tekscan's FlexiForce® sensor was incorporated into the bed check monitoring portion of the system. FlexiForce® sensor, shown in Figure 5-4, was chosen because of its flexibility, cost-effectiveness, and ease of integration. FlexiForce force sensors are ultra-thin (0.008"), flexible, non-obtrusive printed circuits.

Figure 5-4: FlexiForce® sensor

Their construction allows for easy manoeuvring and an exclusive fit, even during the most delicate of procedures. The FlexiForce sensor [59] uses a resistive-based technology. By application of an external force to the "active sensing area" of the sensor results in a change in the resistance of the sensing element which is in inverse proportion to the force applied on the sensor. These sensors are constructed of two layers of substrate composed of polyester film. On each layer, silver is applied for electrical contact which is then followed by a layer of pressure sensitive ink. Adhesive is then used to laminate the two layers of substrate together to form the sensor. The silver circle on top of the pressure-sensitive ink defines the "active sensing area." Silver extends from the sensing area to the connectors at the other end of the sensor forming the conductive leads. FlexiForce sensors are strategically placed underneath the bed legs to determine if a force is being exerted on the bed (confirming whether someone is lying on it). The voltage output varies linearly with the exerted force. The pressure sensor can measure up to 444 N of force (100 lbs). To differentiate between inanimate objects and human beings, we need to calibrate the sensor unit so that the force applied by each extremity should produce a significant voltage output. The pressure sensor basically acts like a means to detect force, as somebody lies on the bed, this will be accomplished by setting up a force-to-voltage circuit. In addition, the sensors will need to be conditioned. Once this is accomplished, the sensor's output is repeatable within 2.5%

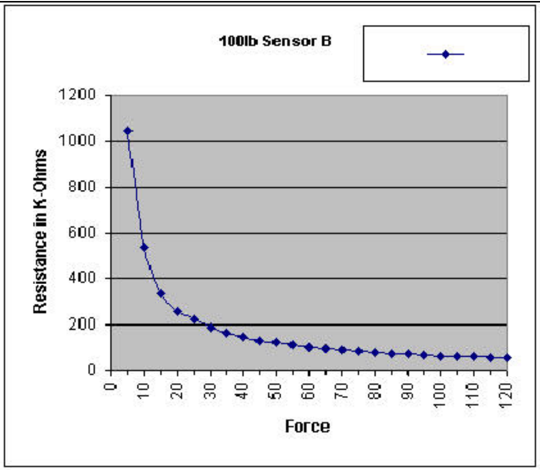


Figure 5-5: Resistance Curve

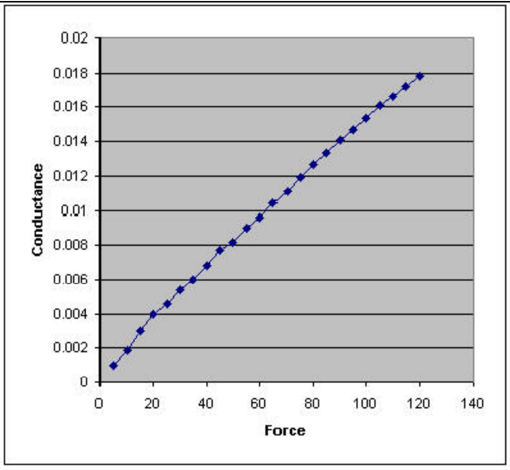


Figure 5-6: Conductance Curve

Some of the physical parameters of flexi force sensor are:

- The operating range of temperature is from -9°C to 60°C,
- Repeatability is +/- 2.5% of full scale (conditioned sensor when 80% force applied),
- Linearity is <+/- 5%,
- Hysteresis is <4.5% of full scale

According to the Tekscan's user manual, to calibrate, apply a known force to the sensor, and equate the sensor resistance output to this force. Repeat this step with a number of known forces that approximate the load range to be used in testing. A Force versus Conductance (1/R) plot can be drawn as shown in Figure 5.5 and Figure 5.6. A linear interpolation can then be done between zero load and the known calibration loads, to determine the actual force range that matches the sensor output range.

5.5 Experimental Results

In experimentation we used a table to test sensors. The table used was an evenly balanced table with equal amount of force being exerted on each of the legs. The four sensors were connected to the bottom of the four legs of the table as shown in Figure 5-7.

Figure 5-7: Experimental Setup

The sensors were sandwiched between flat metal plates with almost the same diameter as the sensing part of the sensor. These plates were used to ensure that equal amount of force was distributed to every point in the sensing head of the sensor. It was also made sure that the table is being tested on an even floor so that equal amount of force is applied on each of the legs.

The potentiometers (R_f) on the driving circuit were also adjusted to show the same output value for all the sensors in idle condition. This was also useful as the force was applied on the sensors, as all the values increased with the same rate and showed the same value when weight was applied at the centre point of the table.

The sensors were connected to the driving circuit which gives the analog output voltage as the force is applied on a sensor. This analog input was then sent to the microcontroller in which the ADC converts it to a digital output and displays it on the LCD as shown in Figure 5-8.

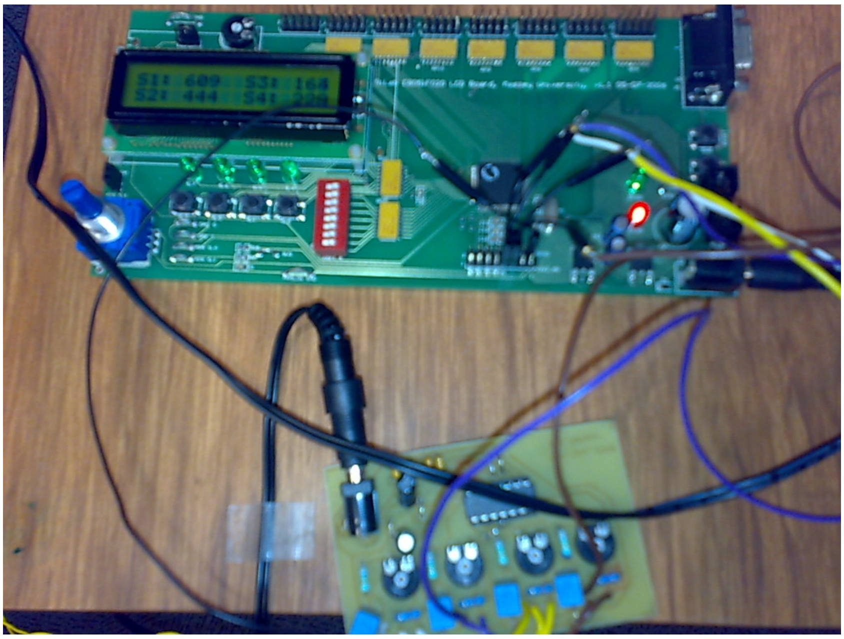


Figure 5-8: Sensor response displayed on LCD in grams

In this lab setup, further each of these zones was further divided into cells along horizontal x axis and vertical y axis as shown in Figure 5-9.

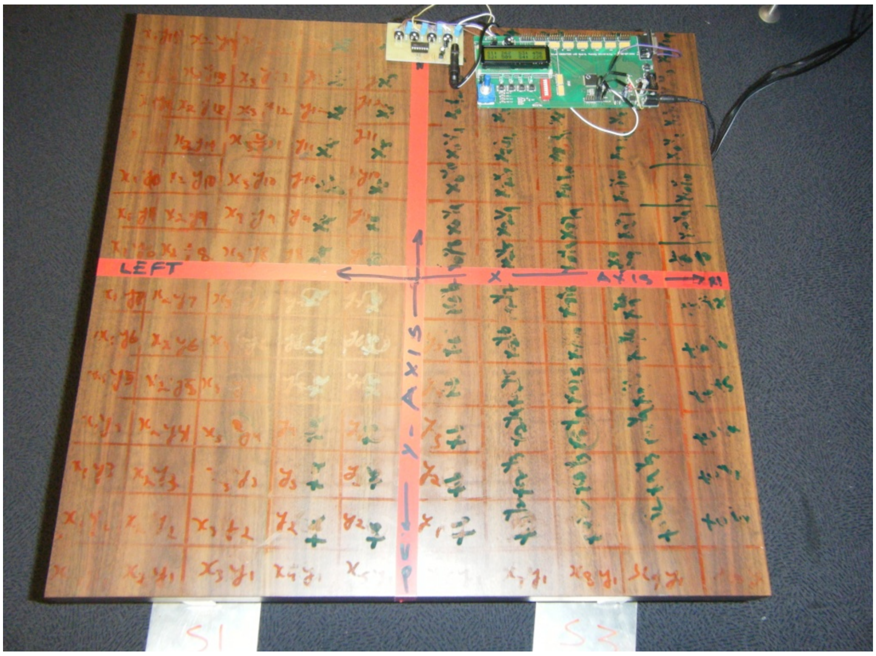


Figure 5-9: Cells along the X and Y axis

The cells were 10x14 that is 10 cells along the X axis and 14 cells along the Y axis as shown in Figure 5-9 and Figure 5-10 explains zone allocation in two dimensional drawing. The table was calibrated in this way to get the exact reaction of the sensors and to see how force being applied in each of the cells affected the output. The force verses voltage results were taken by placing equal amount of weight in each of the cells and recording the output from the microcontroller. This gave us a good idea of how the output reacts to force applied at every point on the table. These results determined the reaction of our sensors when applied with force.

A fixed amount of weight (6kg) was used to test the sensors. We used a thin solid cylindrical base at the bottom of the weight to make sure that the whole force was acting directly on the particular cell on the table.

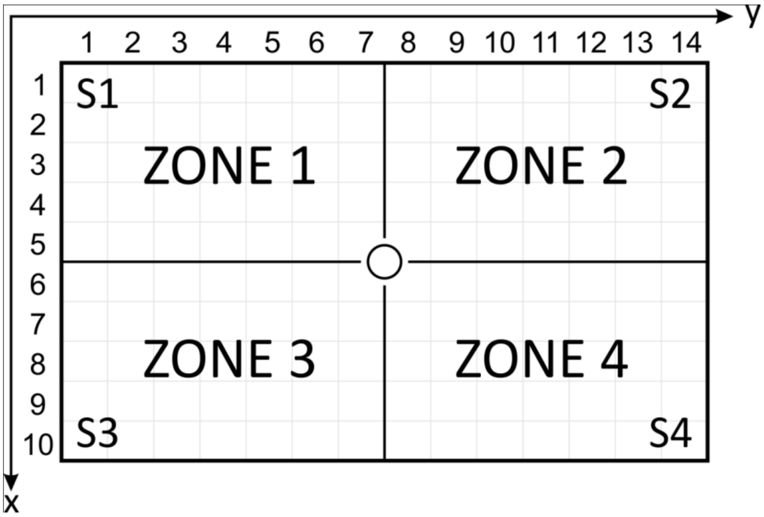


Figure 5-10: Zone allocation with respect to X-Y axis

After the set-up was complete for the experiments results were taken. The fixed weight of 6kg was used to apply force at each of the cells and the corresponding output from the microcontroller was taken.

The following surface plots show the result of our experiments done on the table. The plots were plotted as the reaction force output which results as the 6kg weight is applied on each of the cell. The plots show exactly where the maximum force is being applied. The force is plotted on the vertical Z axis and the horizontal Y and X axes represent the cell where the force is being applied. The surface colour distribution represents the amount of weight that is being applied on the sensors.

From this first plot, shown in Figure 5-11, we can see the response of sensor 1. The highest value for this sensor is 884 which is equal to 8.84 Kg's. This value is at the cell X_1Y_1 which is the position right above the leg of the table where sensor1 is attached. The minimum value for this sensor is at cell $X_{10}Y_{14}$ which is above the leg where sensor4 is attached. This result is true with actual fact that the sensor will show the highest value when the weight is directly above it and show the minimum value when the weight is at the farthest point from the sensor.

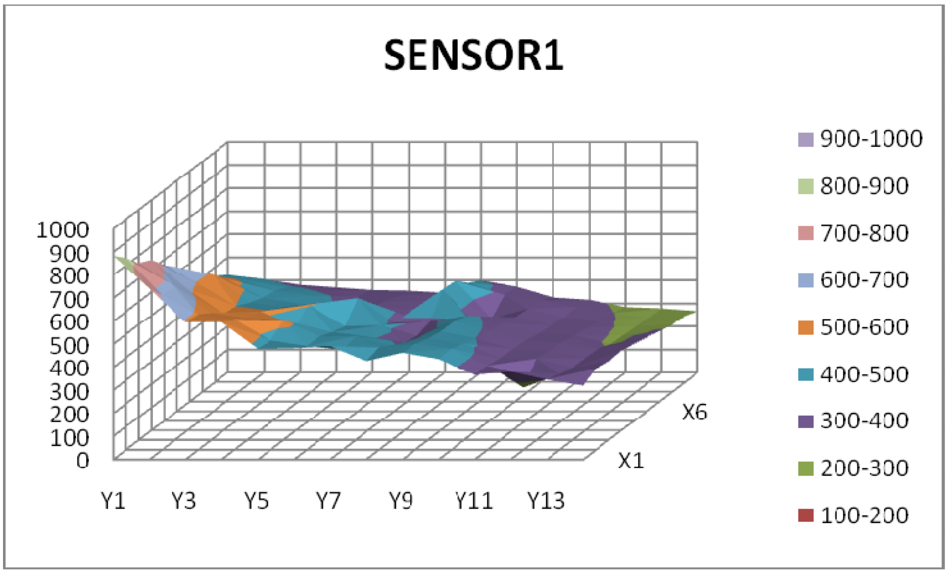


Figure 5-11: Sensor 1 response

The following plot, shown in Figure 5-12, is the result for Sensor2. From this plot we can see that the highest value is around 700, which is at the cell X_1Y_{14} which is located right above the leg where sensor2 is connected. The minimum value is 225 at cell $X_{10}Y_1$ which is the farthest part from sensor2.

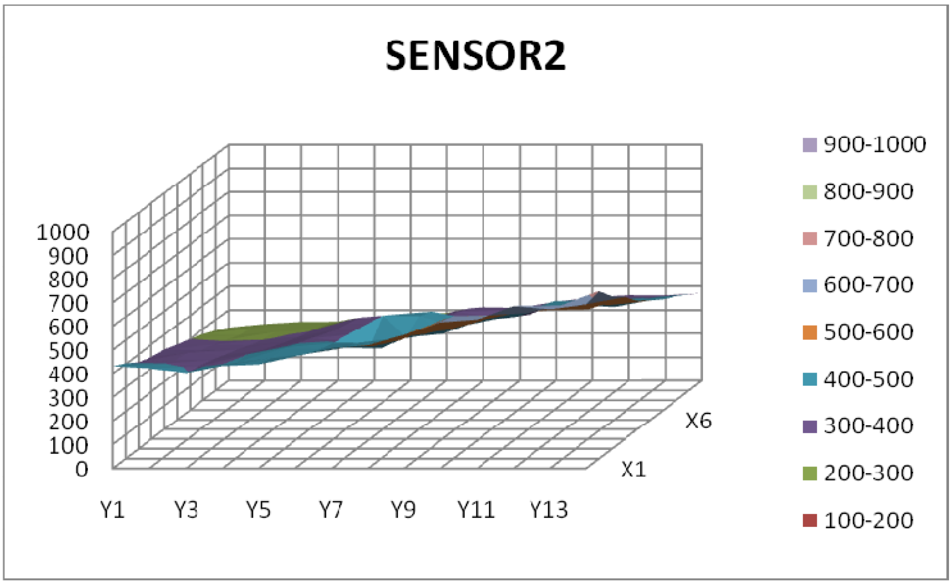


Figure 5-12: Sensor 2 response

Similarly it can be seen from the surface plot for sensor3, shown in Figure 5-13 that the maximum value is at cell $X_{10}Y_1$ which is 845. This cell is located on top of the leg of the table where sensor3 is placed. The minimum value for this sensor will be at the cell located above the sensor2.

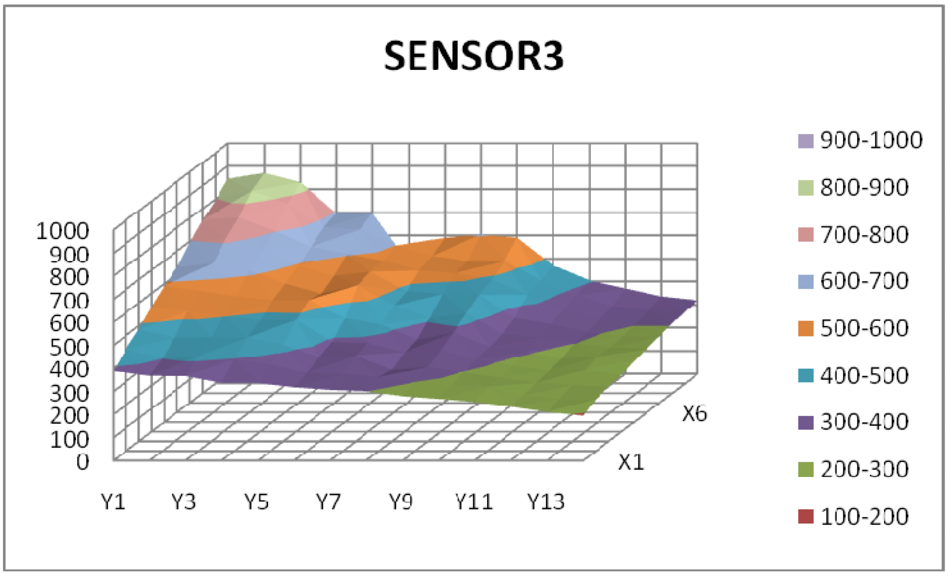


Figure 5-13: Sensor 3 response

Sensor4, shown in Figure 5-14, also shows similar response with the highest value 888 being at the cell $X_{10}Y_{14}$ which is above the leg connected to sensor4. The minimum value for this sensor is at the cell located above the sensor1.

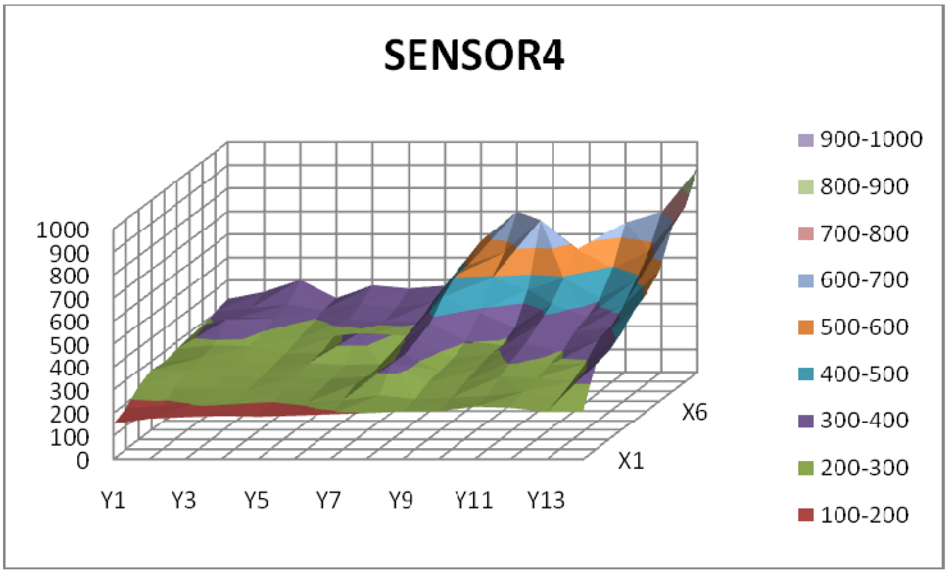


Figure 5-14: Sensor 4 response

The following plot, shown in Figure 5-15, shows the combined results of all the four sensor outputs. It can be seen from the U-shaped curve that more force is being applied to the corners. These are the four corners of the table under which the sensors are placed. This is true with the fact that sensors will experience the maximum force when the weight is placed directly on top of the sensor.

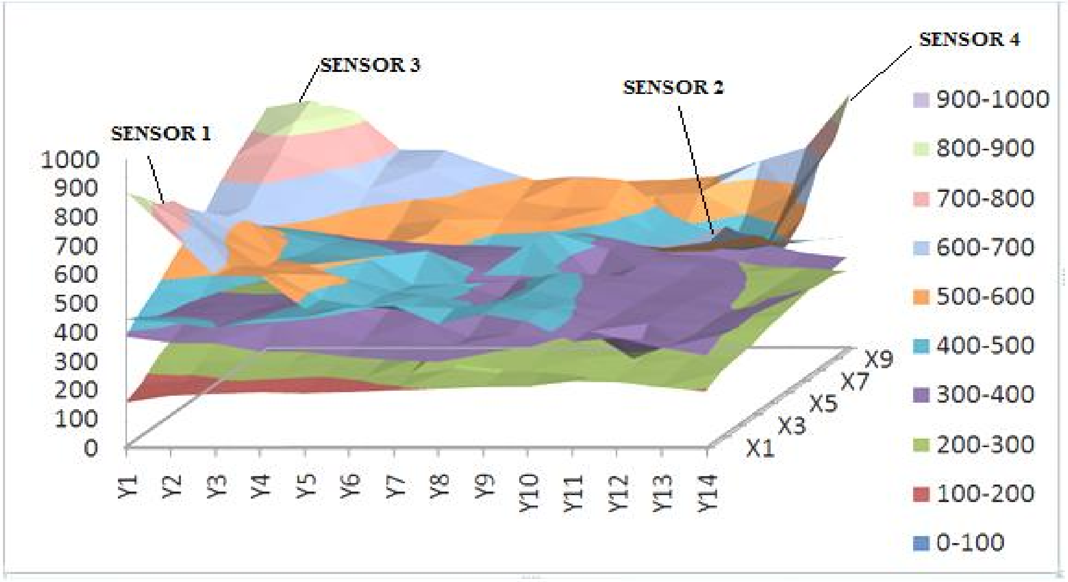


Figure 5-15: Combined results of all the four sensor outputs

The table was calibrated and divided into four zones known as Zone1, Zone2, Zone3 and Zone4 as shown in Figure 5-16.

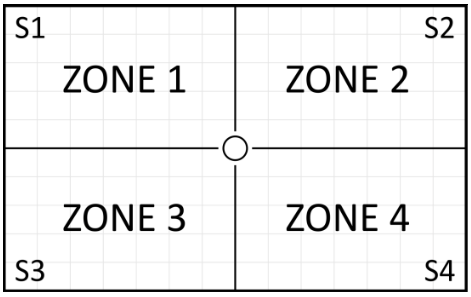


Figure 5-16: Zone Allocation

Ideally if the weight is in the centre point of the table, the response of sensors,

$S1 = S2 = S3 = S4 = S_{AVG}$; where $S_{AVG} = 1/4(S1+S2+S3+S4)$.

In real time situation this may not be possible as there is an error of the sensor's response.

To detect the position of the weight with respect to the zone on the table, the microcontroller program is made to continuously monitor the ADC inputs from the four sensors. And the weight position is calculated by the following equations:

If (S1+S2) > (S3+S4), then the weight is positioned at upper half (UH),
If (S3 +S4) > (S1+S2), then the weight is positioned at lower half (LoH),
If (S1+S3) > (S2+S4), then the weight is positioned at left half (LeH),
If (S2+S4) > (S1+S3), then the weight is positioned at right half (RH).
Based on the above observation, it can be concluded that
ZONE 1 : If UH and LeH both are true,
ZONE 2 : If UH and RH both are true,
ZONE 3 : If LeH and LoH both are true,
ZONE 4 : If LoH and RH both are true.
Table 5-1 shows the experimental observation.

Table 5-1: Experimental observation of locating the weight

Position	Sensors 1	Sensors 2	Sensor 3	Sensor 4	Observation
ZONE 1	890	441	498	370	ZONE 1
ZONE 2	470	791	385	485	ZONE 2
ZONE 3	480	405	815	460	ZONE 3
ZONE 4	385	452	472	857	ZONE 4

Once the above observations are done further investigations are carried out to calculate/estimate, at which location the weight is predominant. The weight is positioned at various points along the line joining the middle point and the edge of the location of sensors 1. The maximum value of the sensor 1 signal corresponding to the location of the weight on top of leg 1 is assumed as 1000. By using the following interpolation technique the position of the weight is calculated and is shown in Table 5.2. Max weight: Maximum signal of the sensor1 corresponding to the weight located on leg1

$S_{AVG:}$ Average signal of the sensors

S1: Signal of sensor 1

Distance: Distance between the centres of the table to the edge corresponding to the sensor 1

Position (p) = $(S1 - S_{AVG})$* Distance(d) /$(S1 - S_{AVG})$

The Figure 5-17 shows how position is calculated by observing the distance (d) of weight.

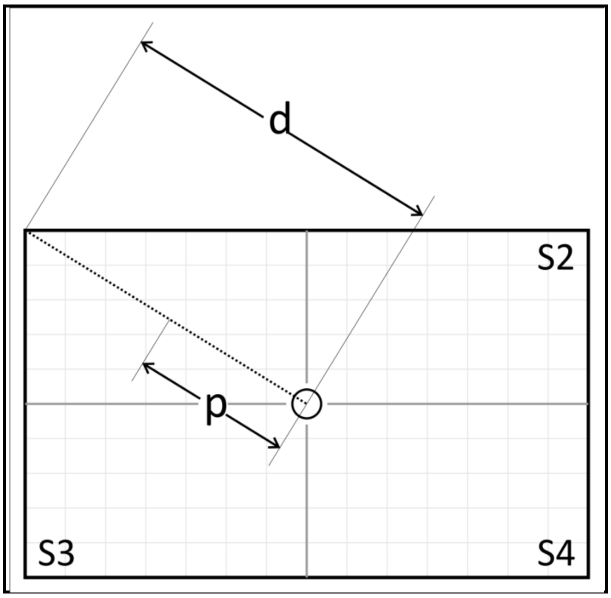


Figure 5-17: Calculation of position (p)

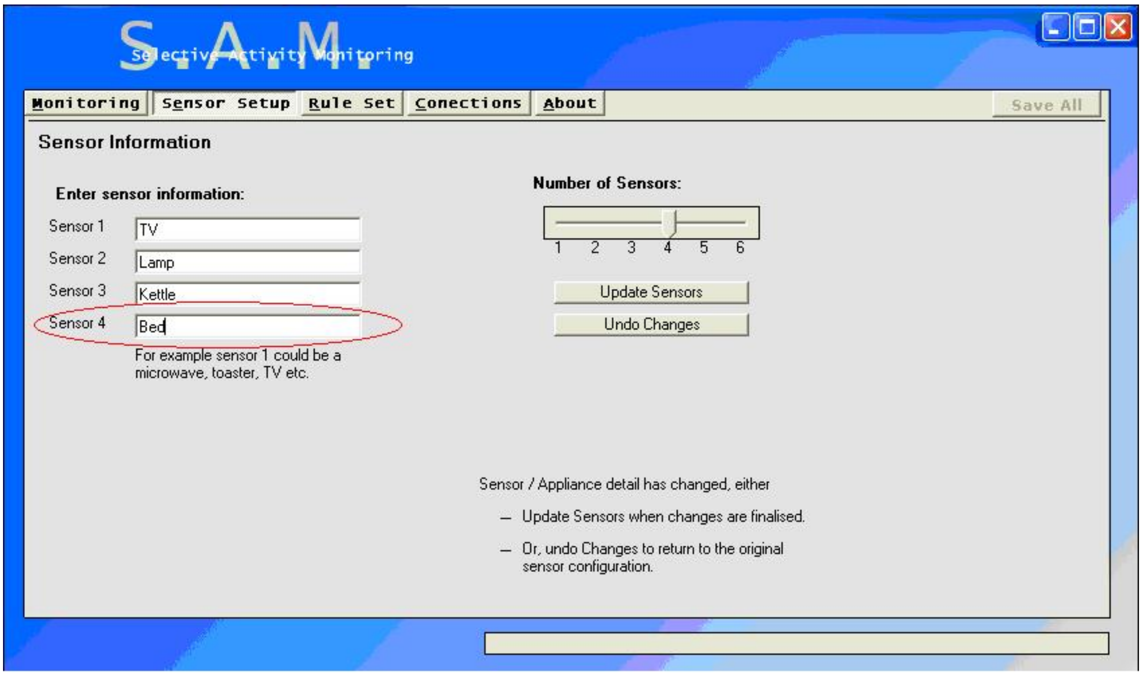
From the Table 5.2 it is seen that the system can be used to estimate the location of the weight. The error can be reduced further if the signals of the other sensors are also used in the interpolation technique.

Table 5-2: Location of the weight

Actual position of the weight from the centre (cm)	Calculated position (cm)	Error (%)
0	4	-
10	9	10
20	17	15
30	27	10
40	39	2.5
50	49	2

Under No-load condition the output from the voltage comparator stays at high state at +5V. But when a load is applied, a change in the resistance in the FlexiForce sensor causes the voltage to reduce dramatically to a order of milli volts, experimental results shows that the voltage drops from +5V to +0.1178V. When the load is applied on the sensor the output from the voltage comparator goes to a low state. This output signal is then fed as an interrupt to a micro controller trough the port pin available on the Silabs® C8051F020 microcontroller development board. The microcontroller polls for the interrupt signal, which is generated when there is a transitions from low state signal to high state signal and vice versa. Thus makes the bed sensor to monitor bed activity continuously.

In this process of monitoring, there arises the case of temporary loading/unloading of the sensor unit. This may occur when somebody uses the bed, which is being monitored, for a certain period. This causes the sensor unit to send unnecessary data points to the controlling unit, causing a huge amount of data build up in its data base. This can worsen the SAM system decision making capabilities by reducing its sensitivity. To avoid this, a rigid rule set need to be set in the supervisory program which runs on the PC, to which the whole unit is connected. Thus by monitoring the person's daily activity, data is collated by Personal Computer which saves all data for processing as well as future use in GUI software shown in Figure 5-18. The habits or the life-style of the person under care is stored in the system.



Sensor Information

Enter sensor information:

Sensor 1 TV
Sensor 2 Lamp
Sensor 3 Kettle
Sensor 4 Bed

For example sensor 1 could be a microwave, toaster, TV etc.

Sensor / Appliance detail has changed, either

- Update Sensors when changes are finalised.
- Or, undo Changes to return to the original sensor configuration.

Figure 5-18: GUI for activity monitoring

The collected data points are compared with the stored data points and depending on the situation the actions are defined as unusual or abnormal. If the system detects any unusual activity a warning or alarm message is issued to the concerned person.

Wireless communication with controller

The Bed monitoring sensor system consists of a RF module for wireless communication with the SAM Controller, and a 10 pin port for communication with the microcontroller and a Force sensor driving circuit. The necessary electronic circuit is shown in Figure 5-19. For the Bed Monitoring the sensor unit consists of a Radiometrix BiM-418-40chip for RF communication, a force sensor FlexiForce Sensor fitted to a driving circuit and a Silab C8051F020 Microcontroller development board.

The Radiometrix BiM-418-40 is a Miniature PCB Mounting module which operated at UHF 418Mhz and capable of half duplex data transmission at speeds up to 40 Kbit/s over distances of 30 metres "in-building" and 120 metres open ground [9].

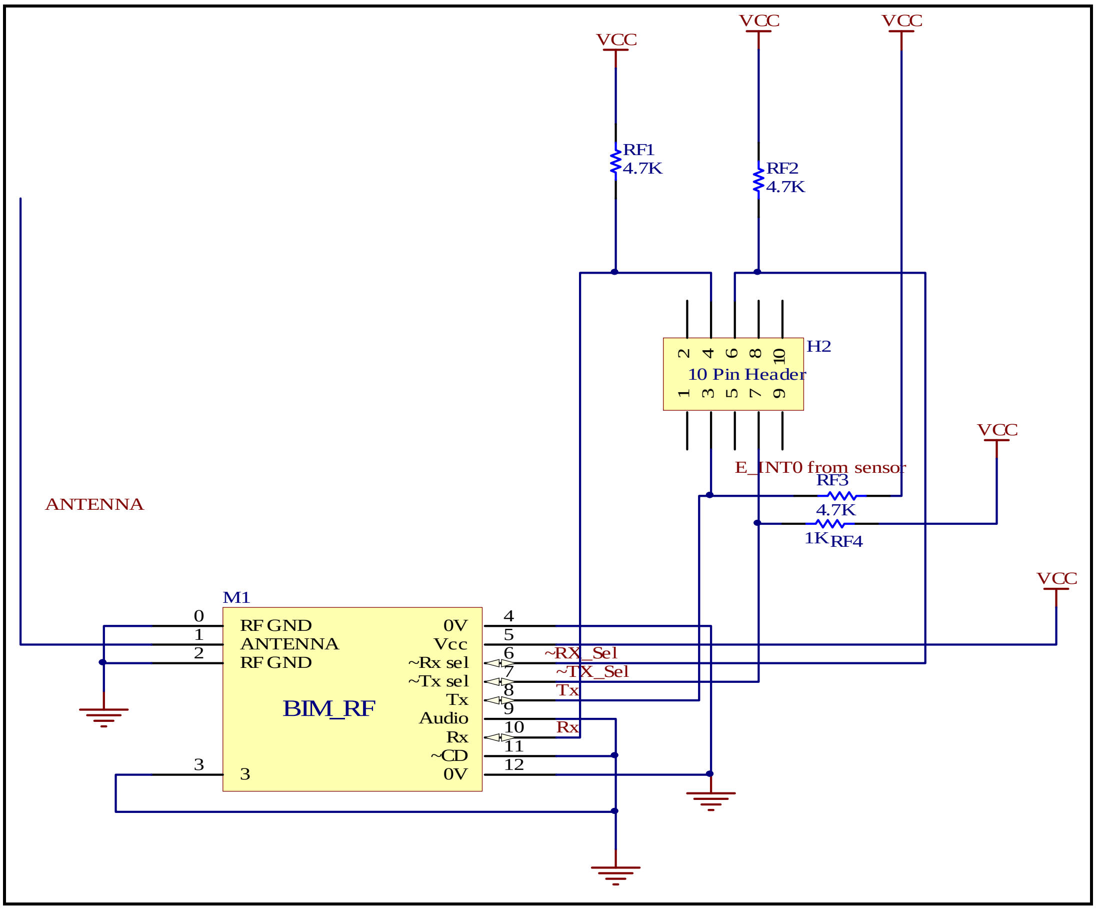


Figure 5-19: RF Module/ Port Pins

The Bed Monitoring Wireless Device communicates wirelessly with a controller unit. This controller unit is connected to a Personal Computer loaded with a supervisory program through RS232 interface.

Chapter 6: Integration of Sensors

Once again here is project overview of the unit/system developed in Figure 6-1. The sensors on a PCB are mounted in the enclosure, which is connected to microcontroller C8051F020 via ribbon cable and an xbee module is connected to UART of port 0 of microcontroller to Tx and Rx at pins 0& 1 respectively. On the receiver end/unit the xbee co-ordinator receives data from various systems/units operating at the time and stores the information on a database on the computer.

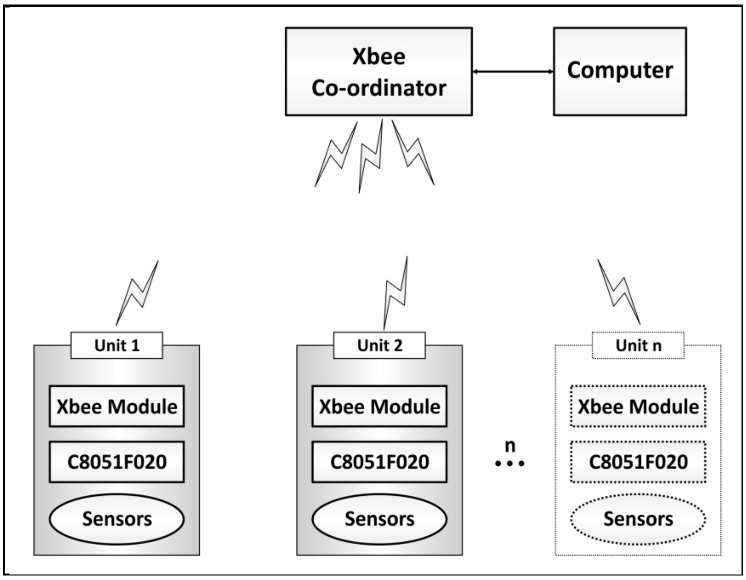


Figure 6-1: Sensor unit/system overview

The three **p**hysiological **p**arameter **m**easuring **s**ensors (PPMS) were first designed and printed on separate printed circuit boards (PCBs) for laboratory tests. After testing and reviewing, a new prototype was designed with all three sensors on one PCB. The PPMS-unit PCB was interfaced with a microcontroller board via cable through a header socket.

6.1 Prototype hardware Design

6.1.1 Sensor Unit

The sensor unit consists of a temperature sensor, a heart rate sensor and an impact sensor for measuring physiological parameters of the human body. The design of this sensor unit is done in Altium Designer summer 09 (Protel) software. The schematics of three sensors are first drawn in a schematic document and then the PCB is designed in Altium Protel software. The sensor PCB connects each sensor to power rails and to a header pin to connect the signals to the microcontroller board. It contains a one switch that allows the user to manually turn the alarm on and off. Apart from this it contains temperature sensor, accelerometer and heart rate circuitry i.e. infra red (IR) emitter LED as transmitter and phototransistor as receiver. Figure 6-2 shows the circuit diagram for the heart rate sensor, calculating beats per minute (BPM).

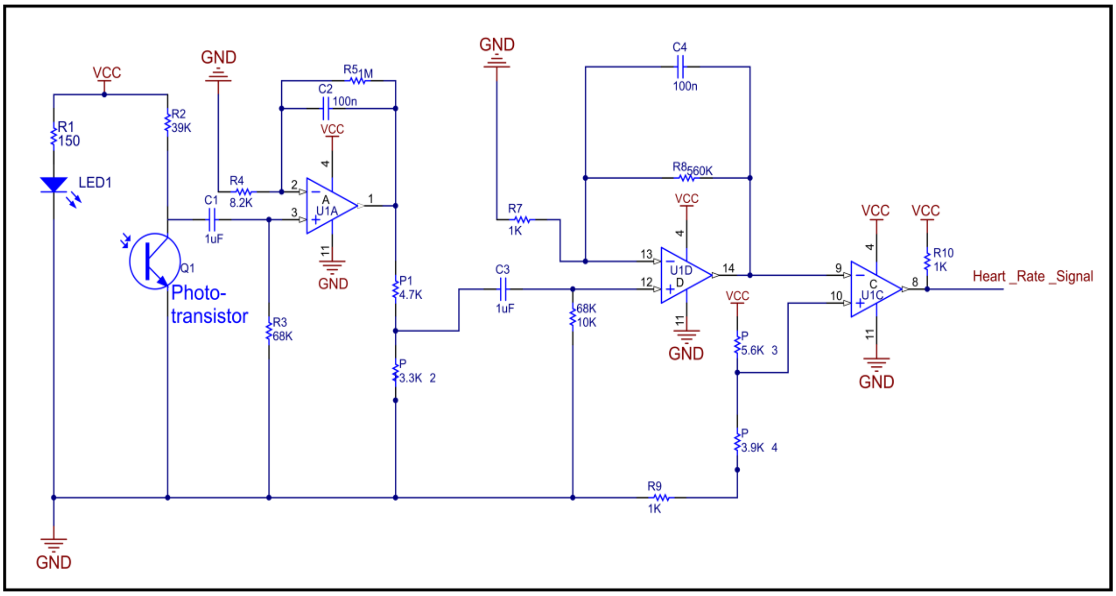


Figure 6-2: Cicuit diagram of Heart rate sensor

The IR LED and phototransistor are powered by +3.3V and the signals from here are then fed to the operational amplifier (LM324). In this quad op-amp the first two op-amps are used to amplify the signal which is between 1-3 Hz and then a third op-amp is used as a comparator. The phototransistor signal is received by pin 2 of LM324 and AC coupled by ’C2’ to remove the DC component of the signal, which is then amplified and fed to the second op-amp at pin 12 and again amplified and then the signal is fed to pin 9 op amp used as a comparator. The signal coming out of pin 8 is then connected to the header pin and fed to the digital port of microcontroller at pin2 of port 0. Similarly schematics are drawn for the impact sensor using accelerometer ADXL213 and temperature IC DS600. The schematics are shown in Figure 6-3.

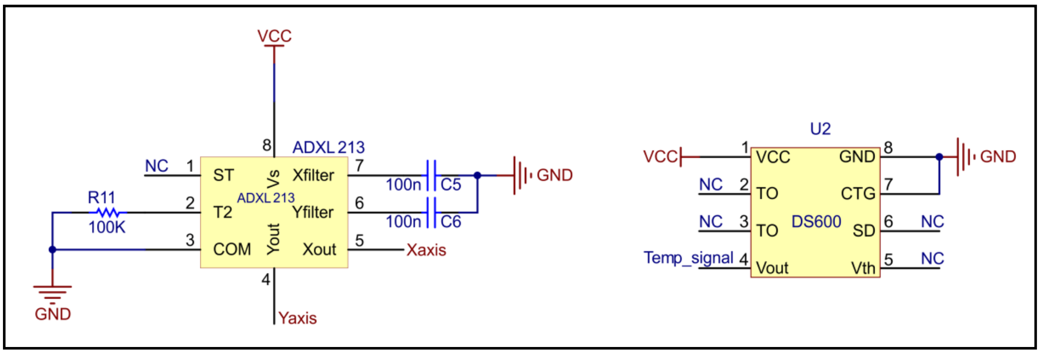


Figure 6-3: Schematics for Impact Sensor and Temperature Sensor

Figure 6-4 shows the PCB layout of the heart rate and impact sensors and input for the temperature sensor, which is on a separate PCB so that it is mounted on a wrist strap, the DS600 IC PCB layout is shown in Figure 6-7. The temperature sensor is mounted on a PCB, which is inside the wrist strap, so its PCB is separate. Figure 6-5 shows the actual Altium Protel file of the PCB design showing the top layer. Similarly Figure 6-6 displays the bottom layer of the PCB in Altium Protel software.

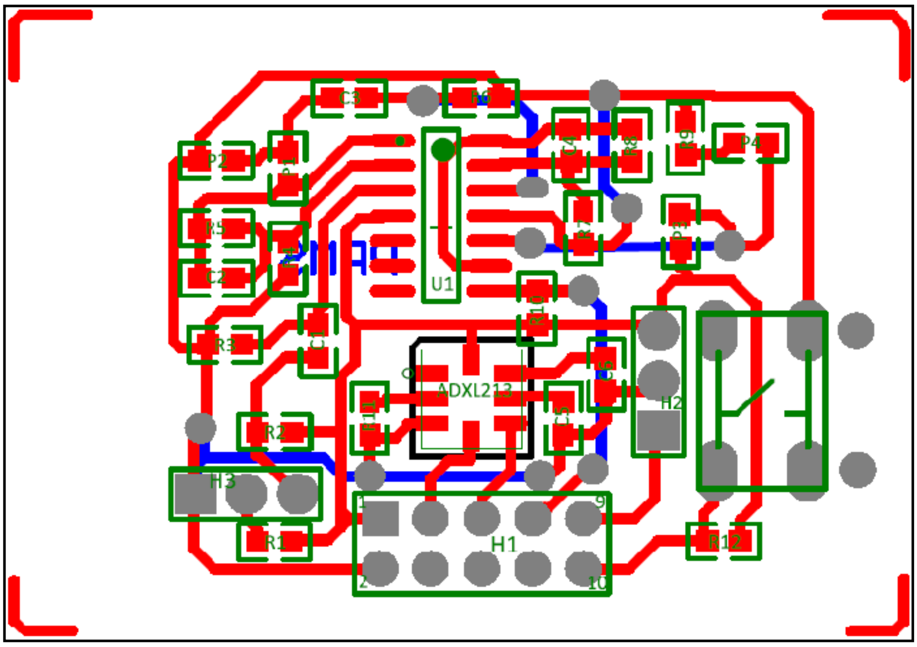


Figure 6-4: PCB Layout of sensor unit (top and bottom layers)

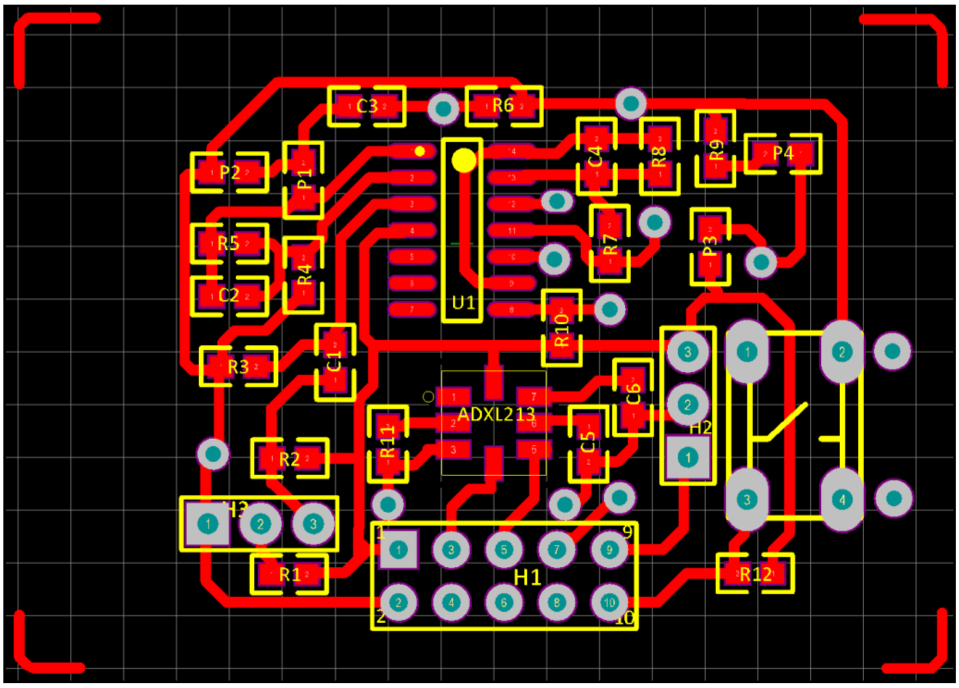


Figure 6-5: Top Layer of the PCB (sensor unit)

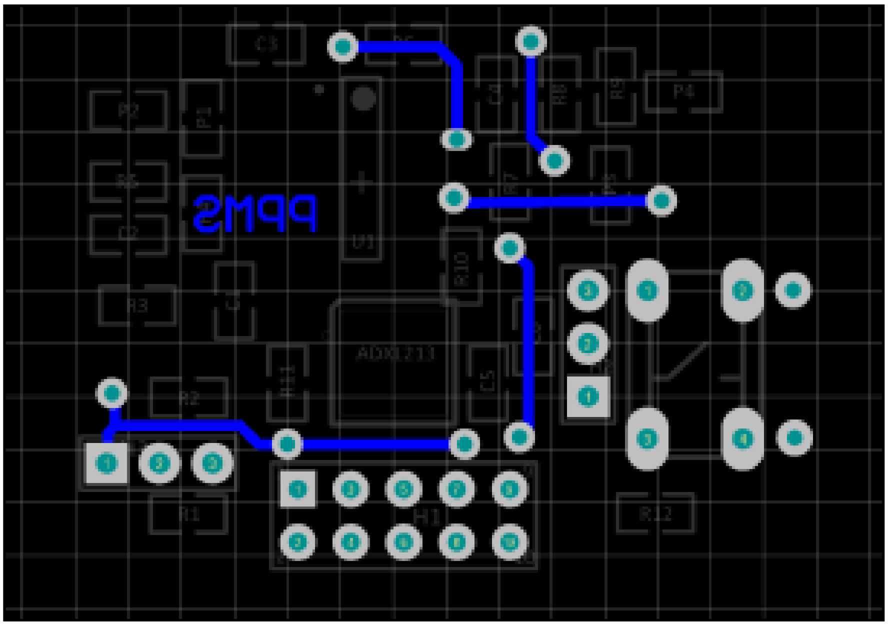


Figure 6-6: Bottom Layer of the PCB (sensor unit)

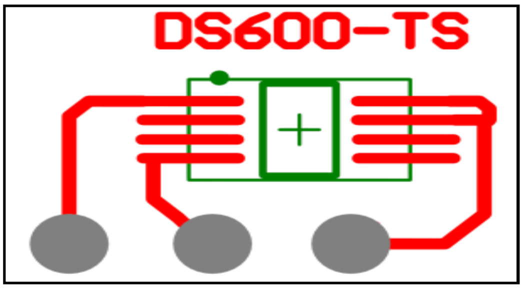


Figure 6-7: PCB layout of Sensor (Temperature) unit

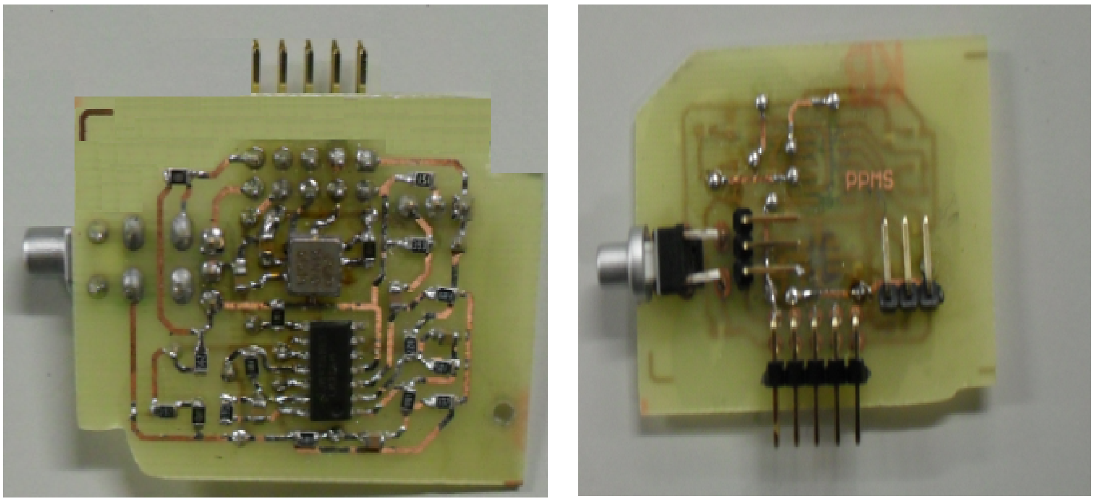


Figure 6-8: Sensor unit PCB

Figure 6-8 shows the final PCB of the sensor unit which includes the circuitry all three sensors, the emergency button and header connected via ribbon cable to the microcontroller for processing. The microcontroller used is discussed in the next section.

6.1.2 Microcontroller

The C8051F020 LCD board microcontroller is used to integrate the signals and to send them wirelessly using UART. The C8051F020 microcontroller [61] is a fully integrated mixed-signal System-on-a-Chip MCU available in a 100 pin TQFP package (Figure 6-9). This microcontroller was chosen for control signals for several reasons including its ease of programming, reliability, robustness and simple handling. It can be powered with 9V DC and operates internally with 3.3V. Here it is seen that the C8051F020 has 8 digital input/output ports and 8 channels of 12 bit ADC, which is sufficient for the 3 sensors (4 inputs) needed for the design, as well as allowing the addition of extra sensors at a later time. A total of 64 general purpose ports Input / Output pins. Lower ports (0, 1, 2, and 3) are bit and byte addressable. Upper ports (4, 5, 6, and 7) are byte-addressable only. Access to the ports is possible through reading and writing the corresponding Port Data Registers. Push-Pull or Open-Drain output modes and weak pull-ups. All analog and digital peripherals are enabled/disabled and configured by user software.

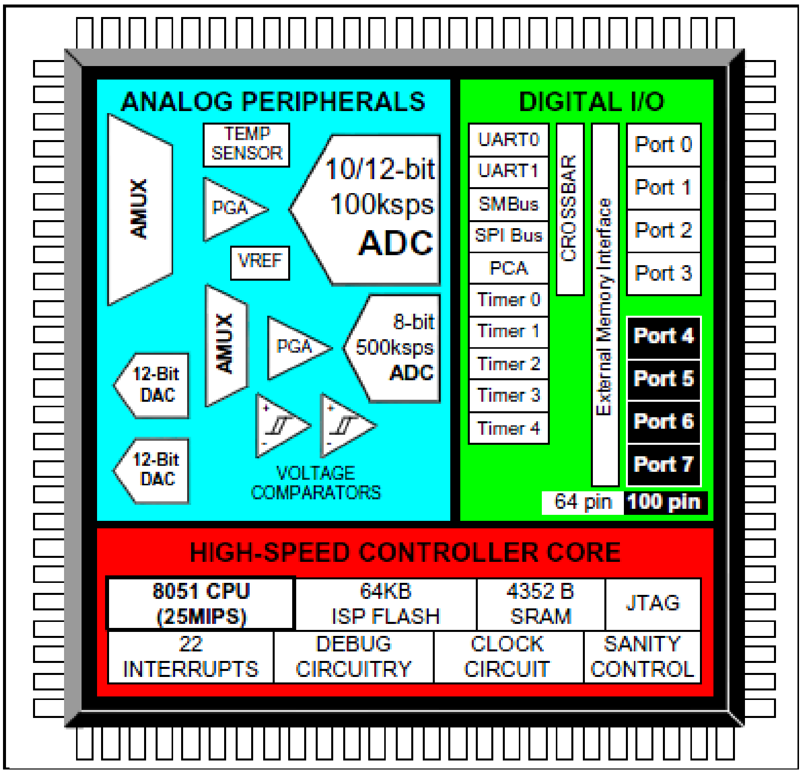


Figure 6-9: C8051F020 System Overview [62]

Its main features are shown in Table 6-1.

Table 6-1: C8051F020 Features

Peak Throughput	25 MIPS
FLASH Program Memory	64K
On-chip Data RAM	4352 bytes
Full-dupley UARTS	x 2
16-bit Timers	x 5
Digital I/O Ports	4 pins
12-bit 100ksps ADC	8 channels
8-bit 500ksps ADC	8 channels
DAC Resolution	12 bits
DAC Outputs	x 2
Analog Comparators	x 2
Interrupts	2 Levels
PCA (Programmable Counter Arrays)	

Also the block diagram is shown in Figure 6-10.

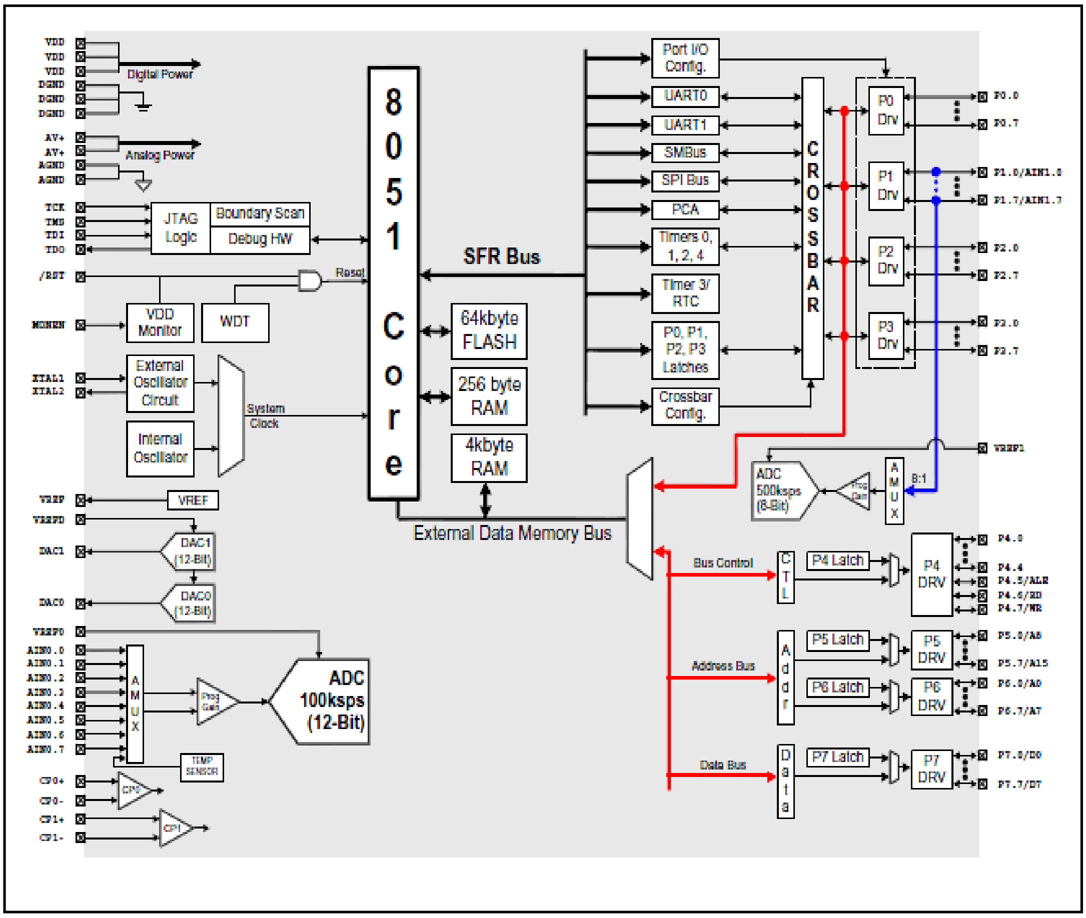


Figure 6-10: Block Diagram of C8051F020 [62]

Some features of C8051F020 microcontroller:

- High-Speed pipelined 8051-compatible microcontroller core (up to 25 MIPS)
- 12-bit 100 ksps 8-channel ADC
- 8-bit ADC 500 ksps 8-channel ADC
- Two 12-bit DACs with programmable update scheduling
- 64k bytes of FLASH memory (program memory)
- 4352 (4096 + 256) bytes of on-chip RAM (data memory)
- External Data Memory Interface with 64k byte address space
- SPI, SMBus/I2C, and (2) UART serial interfaces implemented in hardware
- Five general purpose 16-bit Timers
- Programmable Counter/Timer Array with five capture/compare modules

- On-chip Watchdog Timer, VDD Monitor, and Temperature Sensor

Input from the temperature sensor is connected to an ADC channel on the microcontroller and is time-multiplexed. The heart rate signal and impact sensor signals are interrupt driven. They use interrupt_0 and interrupt_1 respectively. Figure 6-11 shows the (development board) microcontroller board and a ZigBee module.

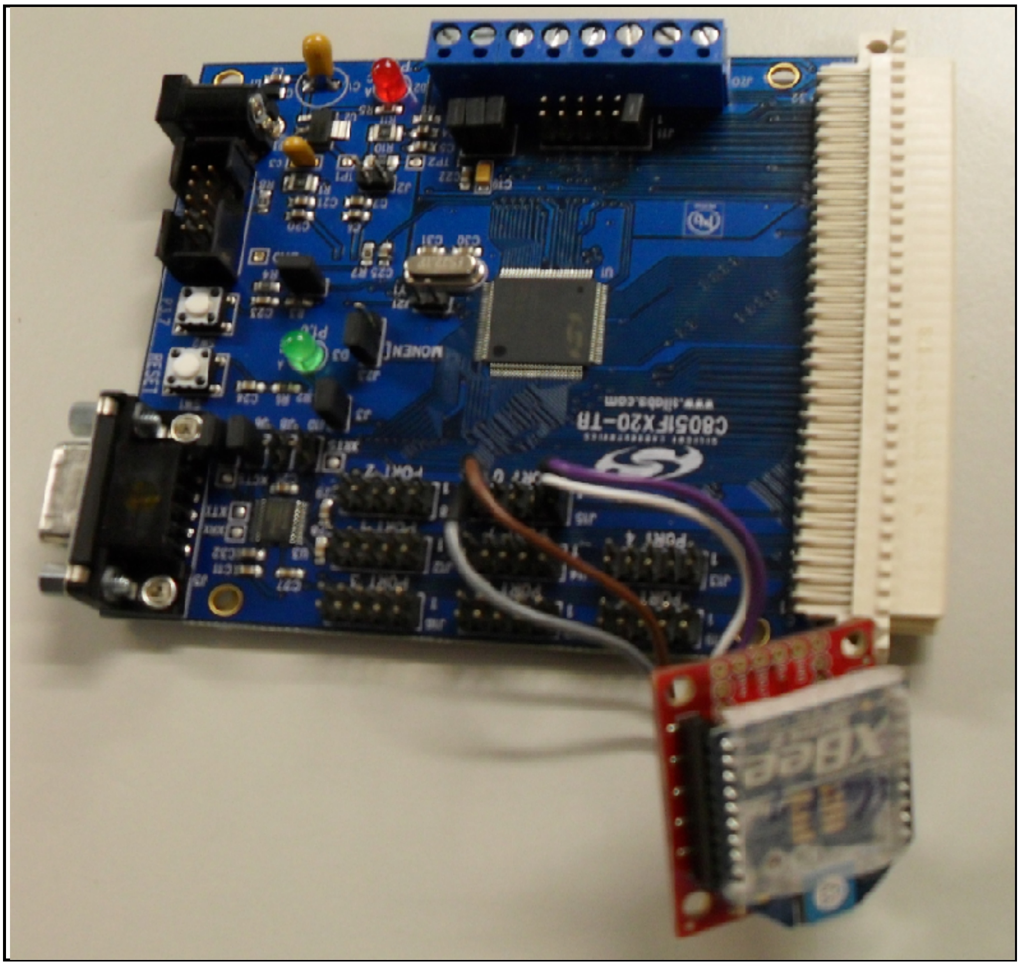


Figure 6-11: C8051F020 board and XBee module

6.2 Receiver Unit

The receiver unit for developed sensor unit consists of the ZigBee co-ordinator connected to the computer via USB cable. The co-ordinator receives information wirelessly from the wrist band on the health signs of the patient (determined by the sensors). This data is then decoded and saved in a database on the computer. The computer is running software (explained in detail in the next section) that takes this

information and displays it in a user-friendly manner. Devices that have a UART interface can connect directly to the pins of the RF module, which is possible with C8051F020 microcontroller. System Data flow is shown in
Figure **6-12**.

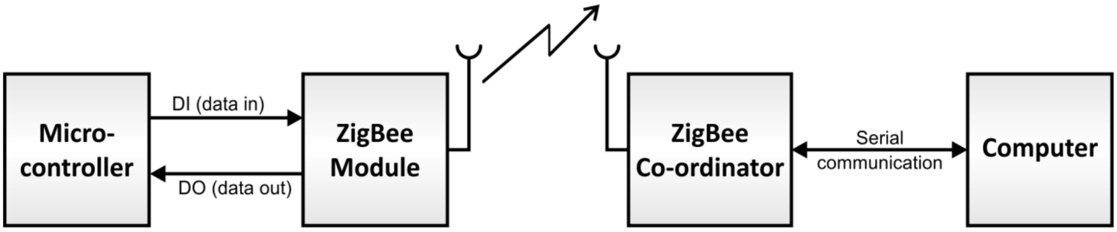


Figure 6-12: System Data Flow Diagram

Each data byte consists of a start bit (low), 8 data bits (least significant bit first) and a stop bit (high). The following figure illustrates the serial bit pattern of data passing through the module

6.2.1 ZigBee Modules

To save the data measured by the sensors it was necessary to build a network between the sensors and to set up a computer receiving and storing the values. For the communication ZigBee modules, (Figure 6-13) are used. These provide a wide range and a couple of low power modes, which could be used to reduce the current consumption of the circuit. The XBee modules support both transparent and API (Application Programming Interface) serial interfaces.

Figure 6-13: XBee Module

6.2.2 XBee Module

These Modules provide a possibility to build an easy to configure Network, with a high data rate up to 230400 Baud/s. The XBee/XBee-PRO ZB RF Modules are designed to operate within the ZigBee protocol and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between remote devices. They come in a preconfigured mode and establish the communication automatically. In addition they are powered by 2.7 to 3.3V and can be connected to the C8051F020 without any additional power-supply circuit. The modules operate within the ISM 2.4 GHz frequency band and are compatible with XBee RS-232 Adapter. The XBee modules do not specifically require any external circuitry or specific connections for proper operation.

Some Features of XBee modules [64] are listed below:

1. **High Performance, Low Cost XBee**
    - Indoor/Urban: up to 133’ (40 m)
    - Outdoor line-of-sight: up to 400’ (120 m)
    - Transmit Power: 2 mW (3 dBm)
    - Receiver Sensitivity: -96 dBm
2. **Advanced Networking & Security**
    - Retries and Acknowledgements DSSS (Direct Sequence Spread Spectrum)
    - Each direct sequence channel has over 65,000 unique network addresses available
    - Point-to-point, point-to-multipoint and peer-to-peer topologies supported
    - Self-routing, self-healing and fault-tolerant mesh networking
3. **Low Power XBee**
    - TX Peak Current: 40 mA (@3.3 V)
    - RX Current: 40 mA (@3.3 V)
    - Power-down Current: < 1
4. **Easy-to-Use**
    - No configuration necessary for out-of box RF communications
    - AT and API Command Modes for □configuring module parameters

- Small form factor
- Extensive command set
- Free X-CTU Software (Testing and configuration software) and updates available on XBee website.

Electrical Connection

To connect the XBee module to the Microcontroller only 4 wires are necessary. The Power-Supply (3.3V), Ground and TX and RX of the Microcontroller are connected to VCC, GND, DIN and DOUT of the XBee module. Figure 6-14 shows pin configuration of XBee module.

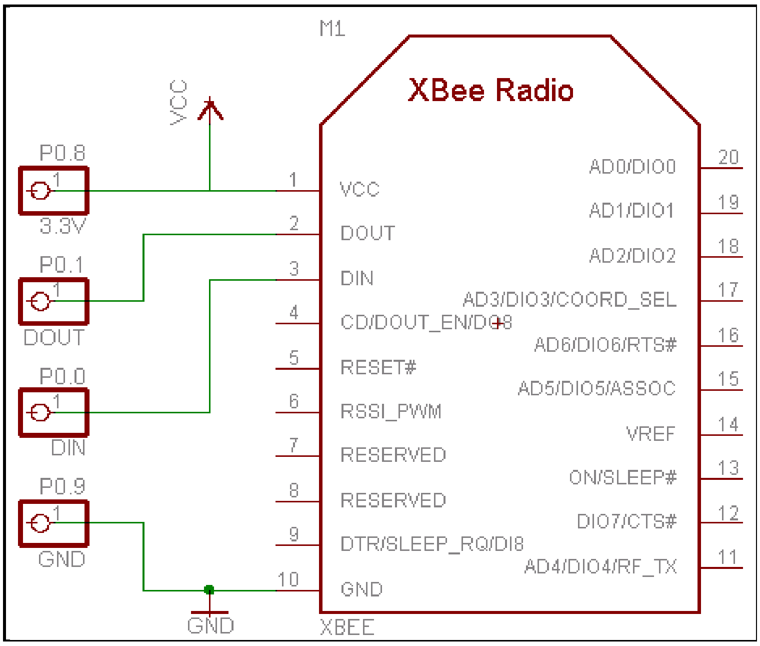


Figure 6-14: XBee Electrical connection layout

Configuration and set up of XBee modules is discussed in next chapter: communication.

6.3 Final Prototype

The final prototype is shown in Figure 6-16, and on a user in Figure 6-15. The sensor's PCB are fitted into the enclosure, which is mounted on a wrist band. The enclosure has holes on one side through which inputs of heart rate sensor and temperature sensor enter into the enclosure. The header connector inside the enclosure is connected to ribbon cable which then takes all the sensor signals to the C8051F020 microcontroller development board ,which was not placed into the wrist strap (although with extra time this would be made into a separate PCB and also fitted into enclosure). The PCB is powered by a 3.3 V power rail of the microcontroller board, connected through ribbon cable.

A 9V battery was connected to microcontroller board. This provided power for the sensor unit and ZigBee module through 3.3V power rail on microcontroller analog and digital ports

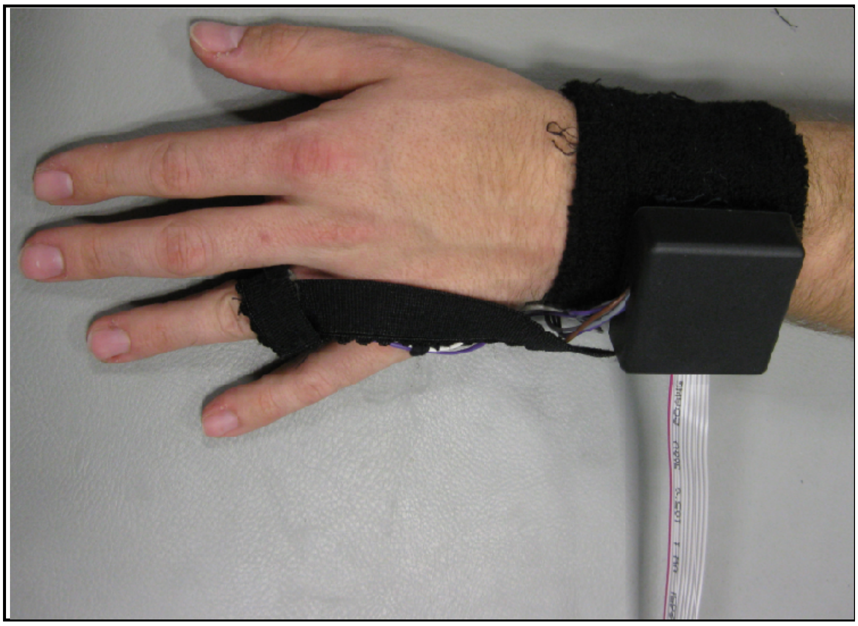


Figure 6-15: Final Prototype unit on a user

Figure 6-18 shows the temperature sensor and emergency button on wrist band and enclosure respectively.

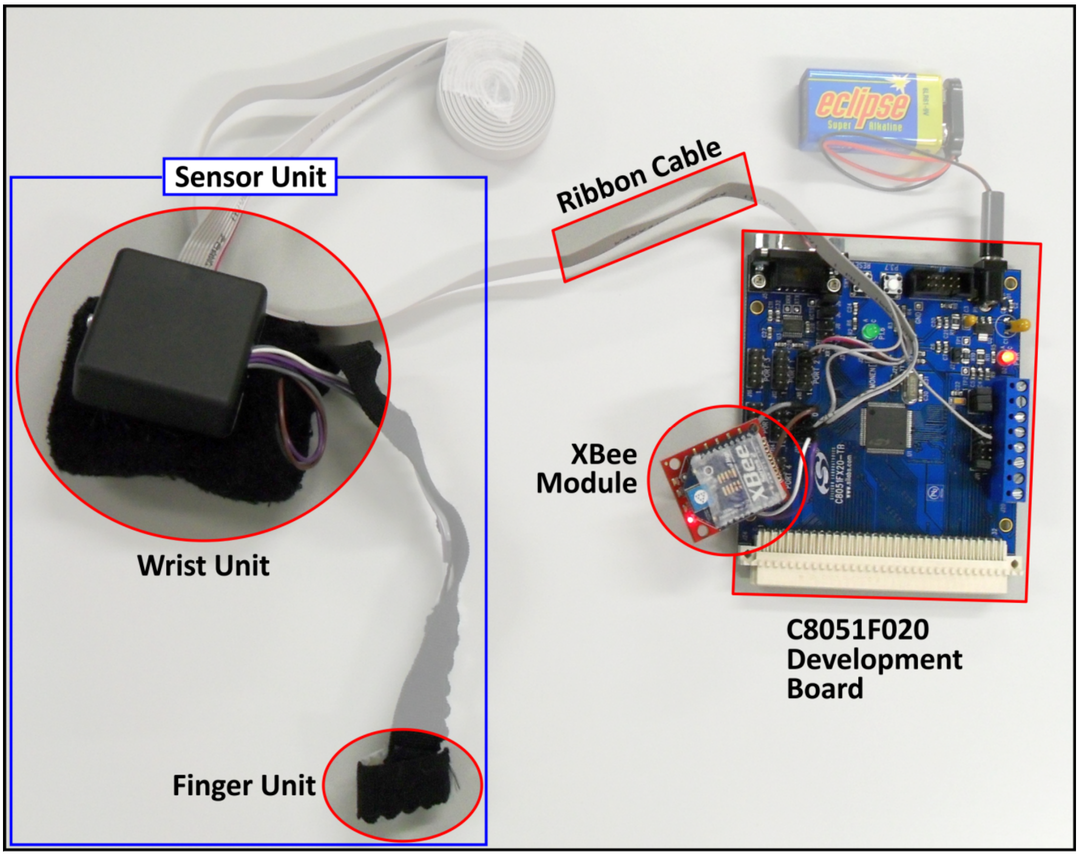


Figure 6-16: A sensor unit

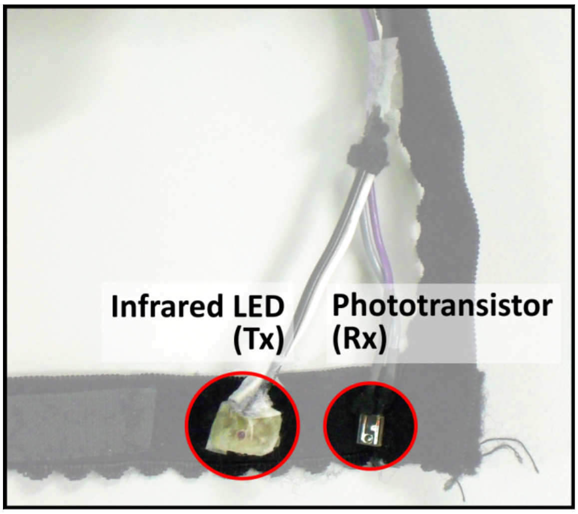


Figure 6-17: Heart rate sensor (finger unit)

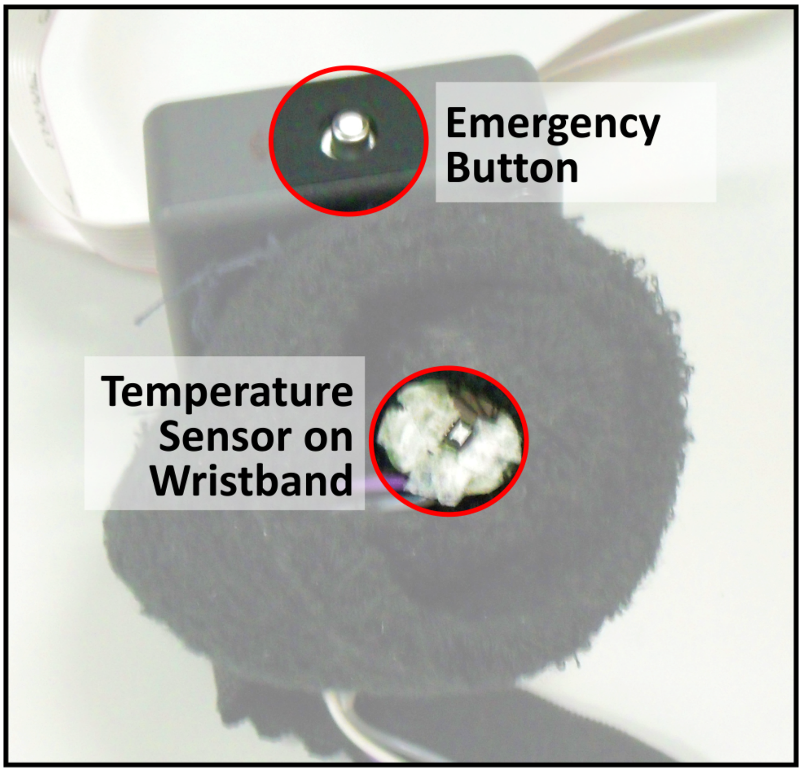
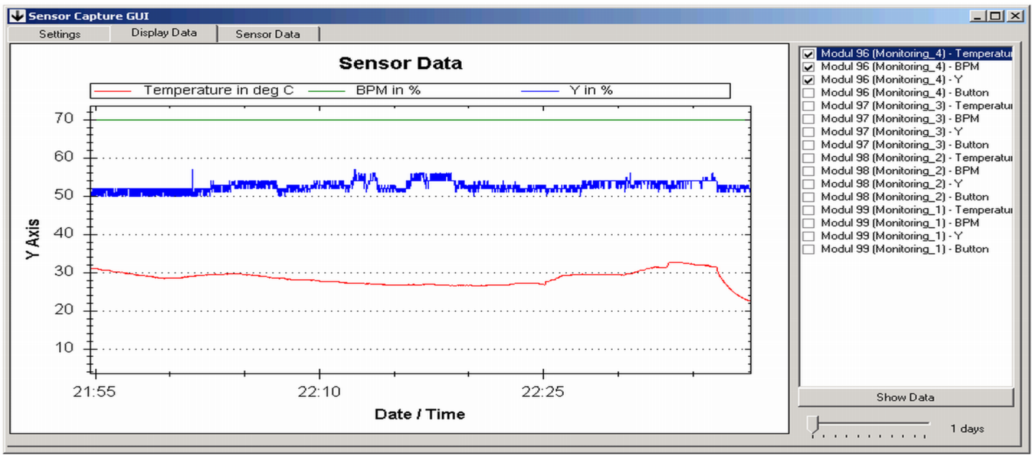


Figure 6-18: Temperature sensor and Emergency button (wrist unit)

Figure 6-19 shows output from one developed prototype unit/system on a user.



☑ Modul 96 (Monitoring_4) - Temperatu
☑ Modul 96 (Monitoring_4) - BPM
☑ Modul 96 (Monitoring_4) - Y
☐ Modul 96 (Monitoring_4) - Button
☐ Modul 97 (Monitoring_3) - Temperatu
☐ Modul 97 (Monitoring_3) - BPM
☐ Modul 97 (Monitoring_3) - Y
☐ Modul 97 (Monitoring_3) - Button
☐ Modul 98 (Monitoring_2) - Temperatu
☐ Modul 98 (Monitoring_2) - BPM
☐ Modul 98 (Monitoring_2) - Y
☐ Modul 98 (Monitoring_2) - Button
☐ Modul 99 (Monitoring_1) - Temperatu
☐ Modul 99 (Monitoring_1) - BPM
☐ Modul 99 (Monitoring_1) - Y
☐ Modul 99 (Monitoring_1) - Button

Figure 6-19: SensorData displayed from one unit/system in GUI

Power consumption

When in operation, the wrist unit consumes 20mA of current at 3.3V power supply, supplied from pins of a port of microcontroller. It was also recorded off the DC power supply equipment display. The microcontroller is powered by 9V battery.

The XBee module connected to the microcontroller consumes 40 mA during transmission. However Xbee modules have the option of going in sleep mode while not transmitting. In sleep modes, XBee modules poll the XBee co-ordinator (their parent) every 100ms while they are awake to retrieve buffered data. Pin sleep of XBee allows external microcontroller to determine when the XBee should sleep and when it should wake by controlling the Sleep_RQ pin. It saves power when no data is transmitted. By using several power-down modes that could be used to reduce consumption during times when the wrist band is not transmitting, alternatively, the architecture could be altered so that packets are only sent when a value goes outside a pre-set range. This was noted for future developments.

Chapter 7: Communication

To save the data measured by the sensors it was necessary to build a network between the sensors and to set up a computer receiving and storing the values. For the communication, ZigBee modules were used.

7.1 Communication between Sensor Unit and Micro-controller

Communication between the wrist unit and the receiver unit is wireless, transmitted in the unlicensed 2.4GHz frequency band. Information is gathered every two seconds from the sensors and then encoded into a packet. This packet is then sent to the radio buffer on the microcontroller, and then transmitted.

The chart in

Figure **7-1** displays how the communication takes place between sensor unit, microcontroller and ZigBee

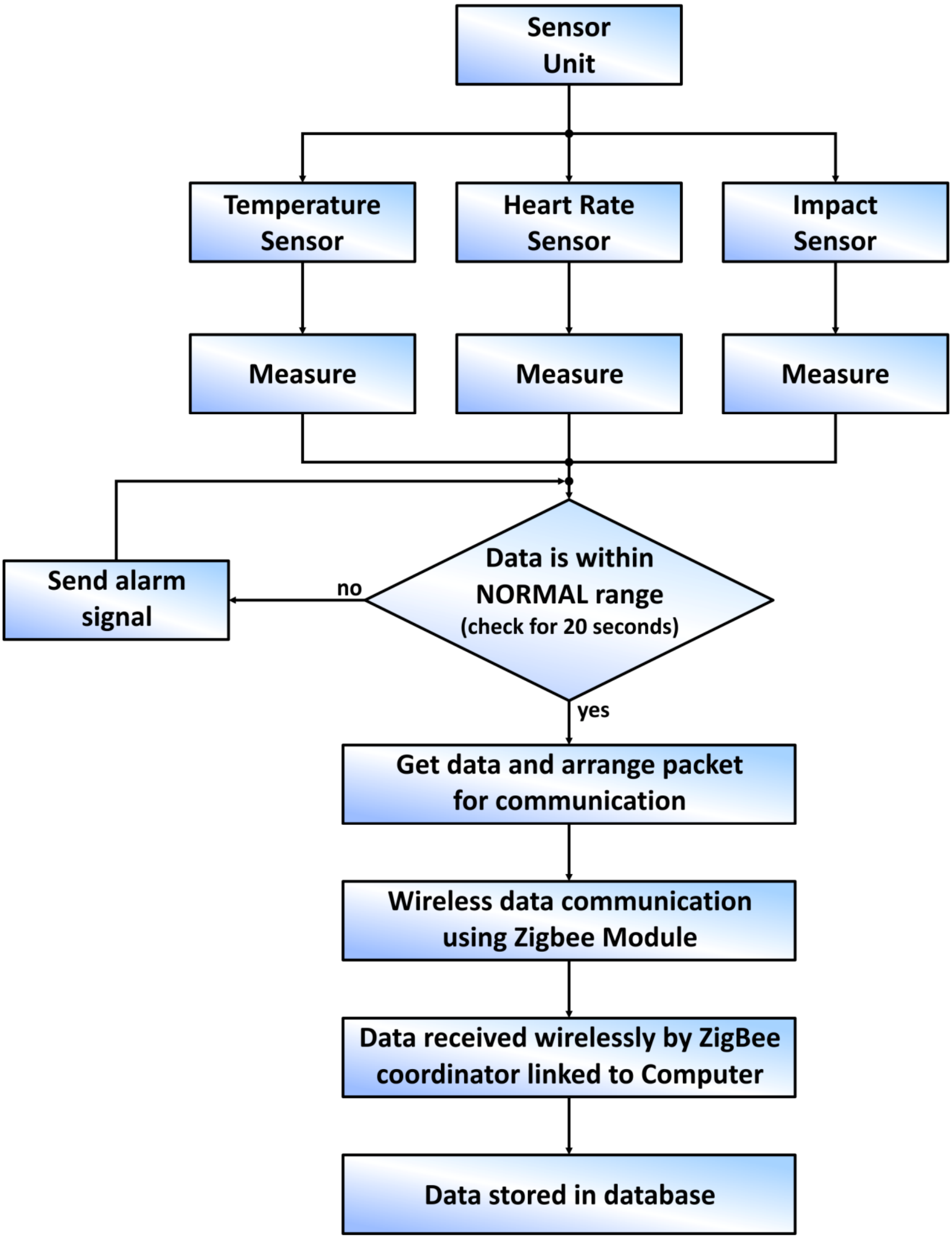


Figure 7-1: Communication algorithm

7.2 Software and Algorithms

Several algorithms were written for the microcontrollers used, as well as software for the computer the user is connected to. This section provides in detail the design of these.

Wrist Unit Microcontroller: The software for the C8051F020 was written in an evaluation version of SDCC compiler, which allows 64k bytes of object code.

The developed system software is based on three sensors which are designed to measure the physiological parameters of the body, the data is transmitted through ZigBee.

The code is designed on an interrupt driven model. All interrupts are triggered using the timers in auto-reload mode, triggering an interrupt every time the timer reaches its maximum value. At certain pre-defined intervals algorithms are run to gather information from the sensors, perform some processing and assemble the information into a packet.

7.2.1 Temperature Sensor Algorithm

The first algorithm that is running on the microcontroller simply generates the modulation signal for the temperature sensor. This is an interrupt set to run every 6.6667e-4 seconds. Analog signal from temperature sensor is connected to ADC0 of C8051F020 microcontroller, at pin #1. ADC0 conversion is started when Timer3 overflows. The Timer3 overflows when Timer3 interrupt is enabled. The ADC conversion is based on the principle of Successive Approximation Register (SAR). When ADC completes the conversion, it reads “end of conversion interrupt”. ADC values are taken in the form of arrays. 50 sample values are taken and measured, out of which threshold filtering is done. The data values above and below this threshold value are discarded to remove spurious inputs. Average of the ‘good’ data is calculated using the equation

Device Temperature (°C) = $(ADC_{out} - 509)*10000/645$

Where ADC_{out} (mV) = $(Result/16)*2430/4095$, Result is the 12-bit digital output of the ADC0.

7.2.2 Impact Sensor Algorithm

Figure 7-2 illustrates how the interfacing of impact sensor takes place with microcontroller.

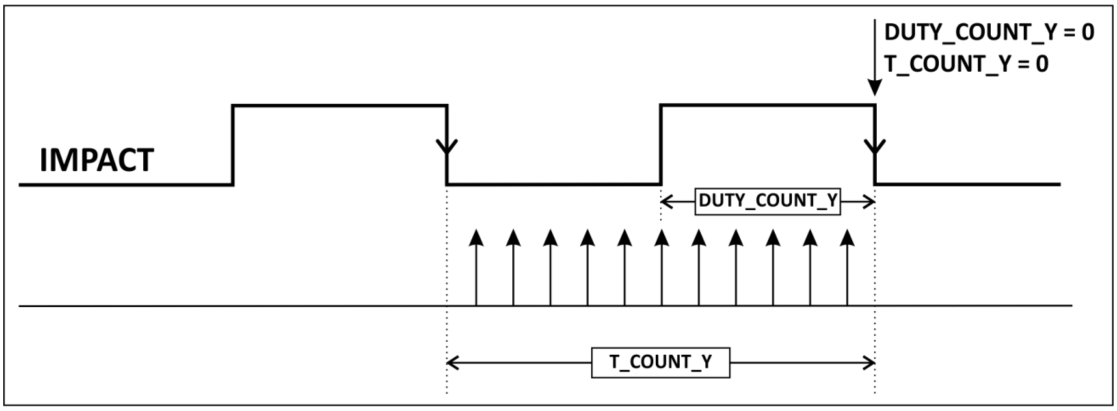


Figure 7-2: Impact sensor interfacing

Sensor signal (PWM) is fed to pin #3 of port0 (external hardware interrupt INT1) of the microcontroller. Timer0 generates “*software ticks*” every 10µs. Timer0 ISR is used to: increment the T_COUNT_Y and also increment DUTY_COUNT_Y only if the sensor signal is high. The width of the pulse (DUTY_COUNT_Y) and time period (T_COUNT_Y) are calculated.

And the duty cycle is given by:

Duty Cycle = (DUTY_COUNT_Y *100)/T_COUNT_Y or this can be written as

Dutycycle = dutycount * 100/Tcount value

7.2.3 Heart rate sensor Algorithm

Figure 7-3 shows sensor interfacing for BPM

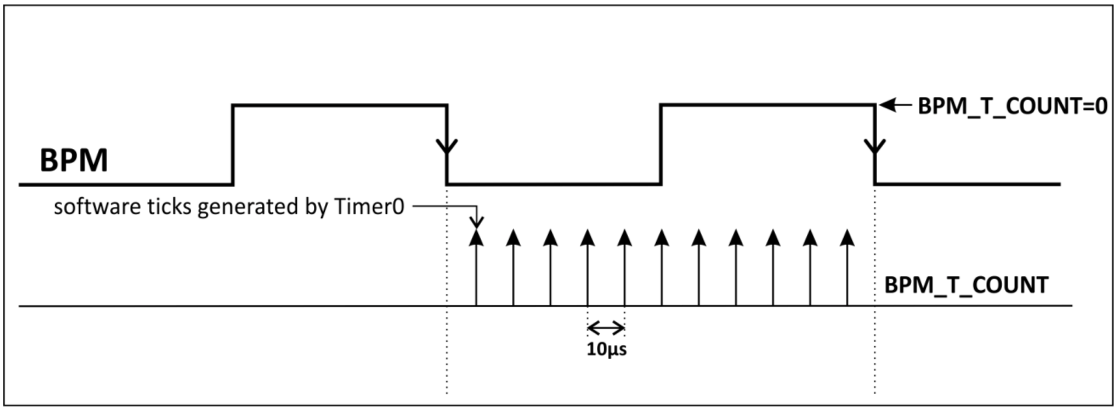


Figure 7-3: Heart rate (BPM) sensor interface

The sensor signal (pulses) is fed to pin #2 of port0, using external hardware interrupt INT0 oof the microcontroller. Due to which Timer0 generates “software ticks” every 10 µseconds. Timer0 is intialised using Timer 0 Interrupt Service Routine (ISR), which keeps incrementing the value of BPM_T_count value. INT0 ISR counts the number of software ticks between two negative edges of the incoming sensor signal. Frequency is counted using the equation:

Frequency = 10000000/BPM_T_count_value and to get the heart rate per minute, following equation is used:

BPM = Frequency*6/10

7.3 Receiver Algorithms

The receiver unit is much simpler than the wrist unit. It is in a constant waiting state for information, forwarding wirelessly to the serial port. The ZigBee co-ordinator is connected to a computer through a USB cable. This is illustrated in

Figure 7-4

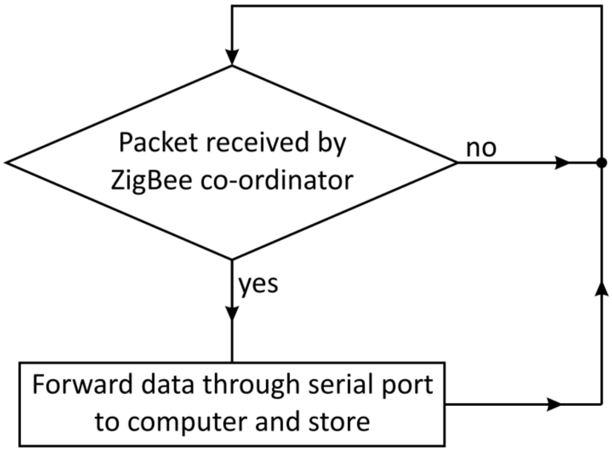


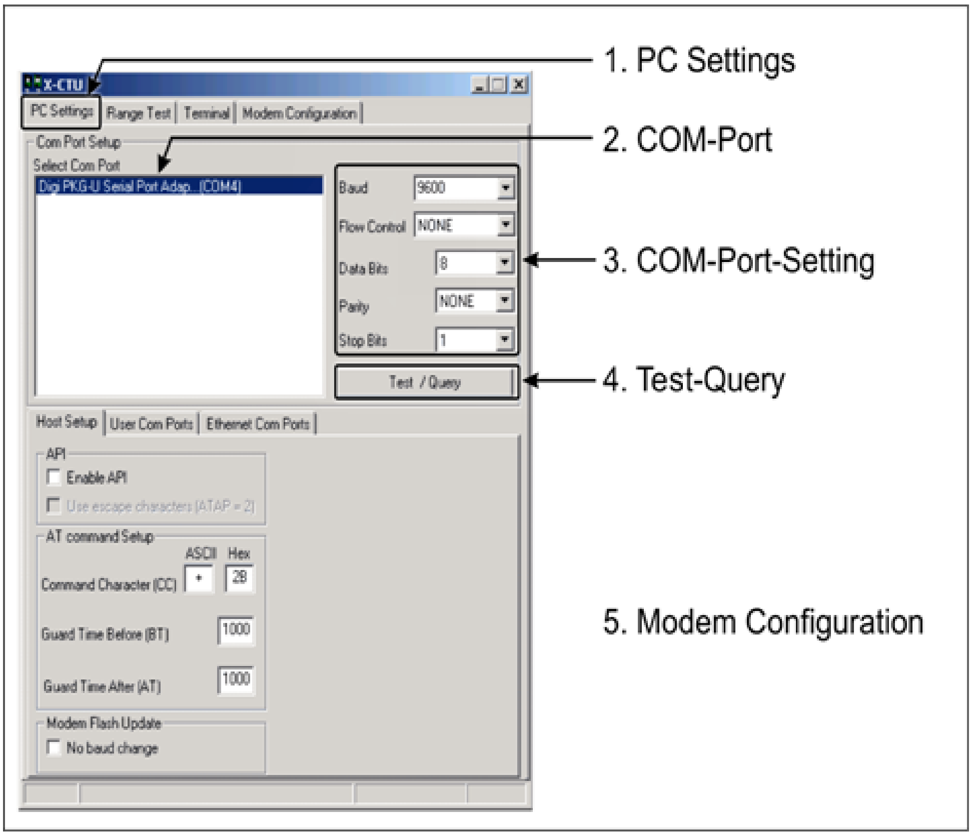
Figure 7-4: Receiver unit process

The program is a user interface, allowing a report on the current status of the individual. Once the user has connected to the receiver unit, data is automatically updated on the screen. BPM, temperature and impact (in both axes) is given on the display. The data is also plotted on a time graph which can be customized to show data received from any of the sensors.

7.3.1 XBee Configuration with X-CTU and settings

To configure the XBee Modules, the provided software X-CTU is used. X-CTU is a Windows-based application provided by Digi [65]. This program is designed to interact with the firmware files found on Digi's RF products and to provide a simple-to-use graphical user interface to the device. Figure 7-5 shows the main tabs of X-CTU window and each tab is detailed below.

- **PC Settings**: Allows a user to select the desired COM port and configure that port to fit the radios settings.
- **Range Test**: Allows a user to perform a range test between two radios.
- **Terminal**: Allows access to the computers COM port with a terminal emulation program. This tab also allows the ability to access the radios' firmware using AT commands. Also complete listing of AT commands is available on website[65].
- **Modem Configuration**: Allows the ability to program the radios' firmware settings via a graphical user interface. This tab also allows customers the ability to change firmware versions



Select Com Port: Digi PKG-U Serial Port Adap...(COM4)

Baud: 9600
Flow Control: NONE
Data Bits: 8
Parity: NONE
Stop Bits: 1

☐ Enable API
☐ Use escape characters (ATAP = 2)

Command Character (CC): ASCII + Hex 2B
Guard Time Before (BT): 1000
Guard Time After (AT): 1000

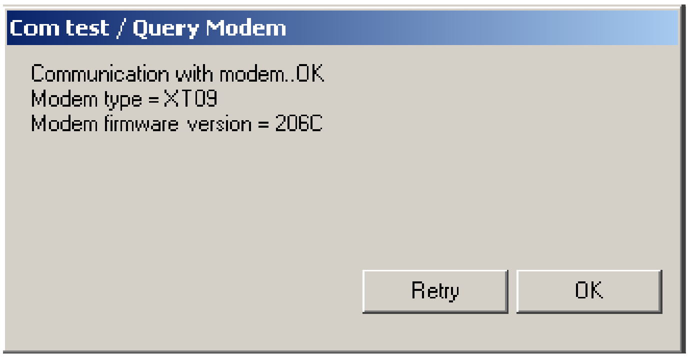
☐ No baud change

Figure 7-5: XBee Configuration with X-CTU

To set up a network the following conditions have to be fulfilled:

- Each network needs one Co-ordinator and several End-Devices
- All modules have to have the same firmware and PAN-ID

  The Test / Query button is used to test the selected COM port and PC settings. If the settings and COM port are correct, you will receive a response similar to the one depicted in Figure 7-6.



Communication with modem..OK
Modem type = XT09
Modem firmware version = 206C

Figure 7-6: Test/Query window

Modem Configuration tab

The Modem configuration tab has four basic functions:

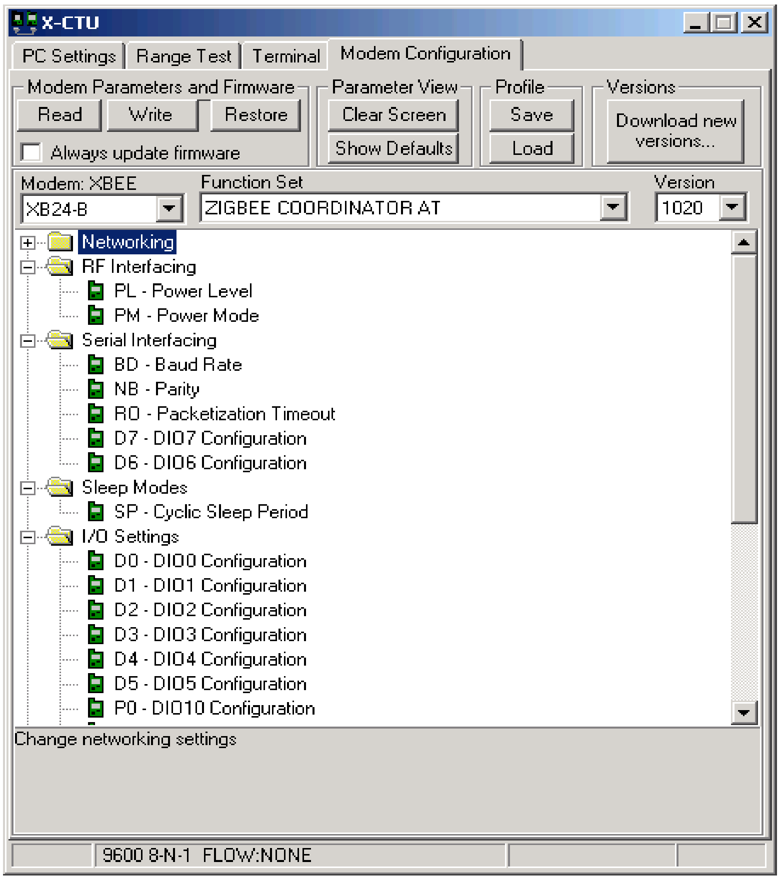
1. Provide a Graphical User Interface with a radio’s firmware
2. Read and Write firmware to the radio’s microcontroller
3. Download updated firmware files from either the web or from a compressed file
4. Saving or loading a modem profile

Reading a radio’s firmware

To read a radio’s firmware, follow the steps outlined below:

1. Connect the radio module to the interface board and connect this assembly or a packaged radio (PKG) to the PC’s corresponding port (IE: USB, RS232, Ethernet etc.).
2. Set the PC Settings tab (see Figure 3) to the radio’s default settings.

3.On the Modem Configuration tab, select “Read” from the Modem Parameters and Firmware section (see Figure 7-7).



Modem: XBEE XB24-B

Function Set ZIGBEE COORDINATOR AT

Version 1020

Change networking settings

9600 8-N-1 FLOW:NONE

Figure 7-7: Modem Configuration tab

If everything is set up correct, the co-ordinator establishes a connection to the End-Devices automatically. The Co-ordinator sends Broadcast Commands, and the End-Devices can send to Co-ordinator only. Although a Mesh-Network is possible, where the End-Devices act like a Router and forward data from devices which are out of range from the Co-ordinator.

7.3.2 XBee Communication Protocol

The ZigBee protocol is designed to provide a secure and reliable wireless data solution. The ZigBee platform has been designed to maintain a strong RF communication through hostile environments that are usually common in commercial and industrial applications. ZigBee data packets can be sent as either unicast or broadcast transmissions. Unicast transmissions route data from one source device to one destination device, whereas broadcast transmissions are sent to many or all devices in the network.

The transmission of the XBee Modules doesn’t provide a checksum or any other possibility to verify the correctness of the received data. To avoid corrupted data and to see which unit was sending the data, an own communication protocol is needed. The Microcontroller sends a String to the Computer. This String contains 27 characters (Figure 7-8). The first three chars are the name of the user, and then each is divided by a space, which is shown by symbol “minus” sensor data. A string consists of numbers in the following order:

099-3215-0068-0056-0001, in which,

- ➔ 32.15 Degrees is skin temperature (Scale Factor has to be set to 100)
- ➔ 68 is Beats per minute (BPM)
- ➔ 56.00% duty cycle of Impact sensor (Scale Factor again 100)

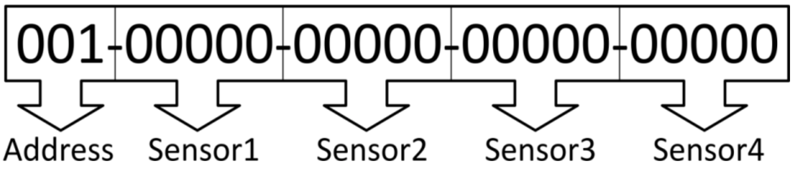


Figure 7-8: Data Packet composition

Each unit sends its data every 2 seconds to the co-ordinator. There the data has to be collected and tested for correctness.

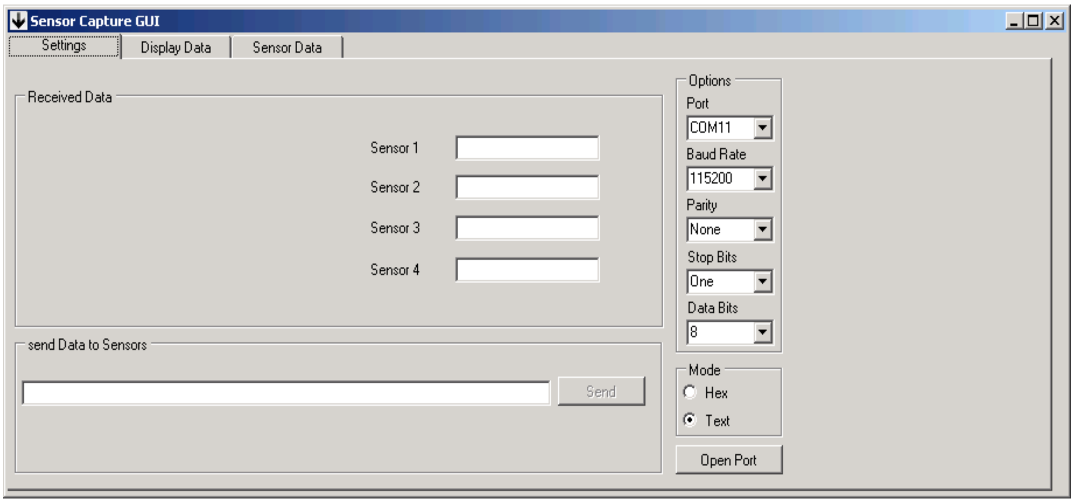
7.3.3 GUI of the Serial Port Communication Program

The Graphical User Interface (GUI) used for this project was written in C# language and captures the serial communication. It was developed carefully to serve the purpose the project was designed for. It provides the user with a friendly interface in which all sensor data can be retrieved easily with the aid of graphical representations. Providing a GUI, that enables the user to navigate easily through it and find all necessary information being displayed in a simple manner, was of high priority to the designer of the system.

Copyright of the original code

The original code of the GUI is a copyright of its first developer, Richard L. McCutchen, who created it on October 20, 2007. That code was used and modified to fit into the purpose of this project. The “Communication Manager” class was used, in particular, to host most of the active part of the code and that can be found under a region called “comPort_DataReceived”.

If everything is right you see the following:

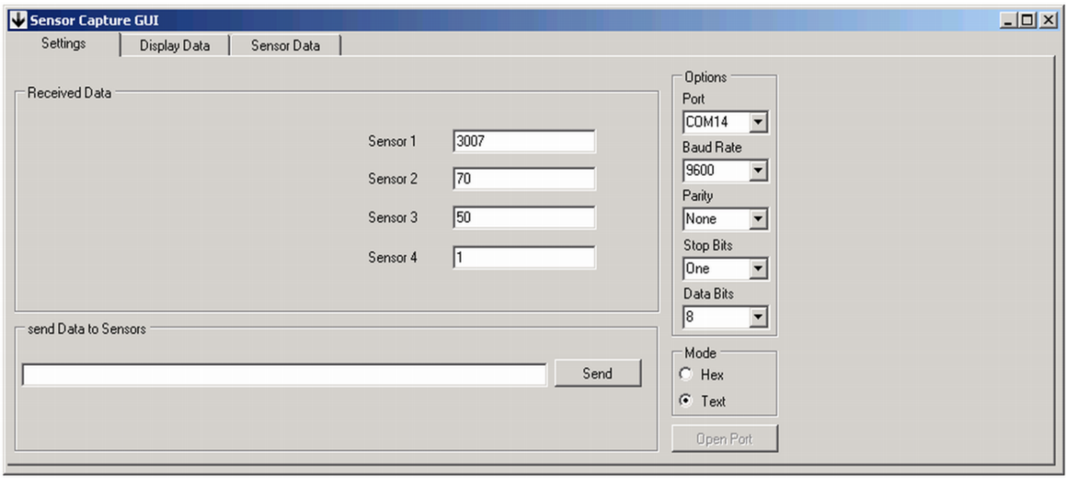


Sensor 1
Sensor 2
Sensor 3
Sensor 4

Port: COM11
Baud Rate: 115200
Parity: None
Stop Bits: One
Data Bits: 8
Mode: ○ Hex ● Text

Figure 7-9: GUI of the Serial Port Communication program

Select the correct Connection-Options (on the right) and click “Open Port”. Now Data should be received and in the text-fields “Sensor 1-4” values should occur.



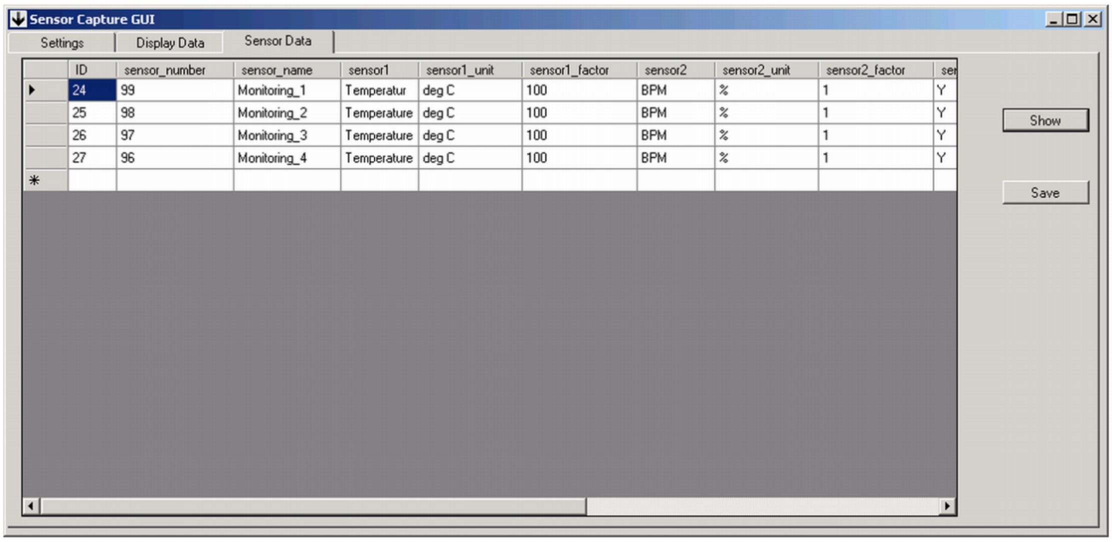
Sensor 1: 3007
Sensor 2: 70
Sensor 3: 50
Sensor 4: 1

Port: COM14
Baud Rate: 9600
Parity: None
Stop Bits: One
Data Bits: 8
Mode: ○ Hex ● Text

Figure 7-10: Displaying data in the GUI of the Serial Port Communication program

Now go to “Sensor-Data” and click on “Show”.

You should see different Sensor Settings and how they are stored in the Database.



ID	sensor_number	sensor_name	sensor1	sensor1_unit	sensor1_factor	sensor2	sensor2_unit	sensor2_factor	ser
24	99	Monitoring_1	Temperatur	deg C	100	BPM	%	1	Y
25	98	Monitoring_2	Temperature	deg C	100	BPM	%	1	Y
26	97	Monitoring_3	Temperature	deg C	100	BPM	%	1	Y
27	96	Monitoring_4	Temperature	deg C	100	BPM	%	1	Y

Figure 7-11: Sensor settings

Table 7-1 describes all the fields used in the Database to store different sensors data.

Table 7-1: Database field Description

Field	Meaning
ID	ID in Database, only necessary for storage and identification, does not have to be changed
sensor_number	“name” of the Sensor. This is the ID of the Sensor Unit, which is sent in the String (see Communication Protocol)
sensor_name	Name of the Sensor to make identification for User easier (like “Garden”, ”Room Name”)
sensorX	Value of sensor “X” that is measured
sensorX_unit	Unit of the measured Value
sensorX_factor	Scale-factor to get the right value. Must be 100 to get for example 20.00°C if 2000 is received

To edit Sensor-Data change the data in the columns and click on “save”.

To add a Sensor simply write in the last and empty line new values (ID should be left empty) and then click on “Save”.To show the received data go to “Display Data”.

Then you see Figure 7-12.

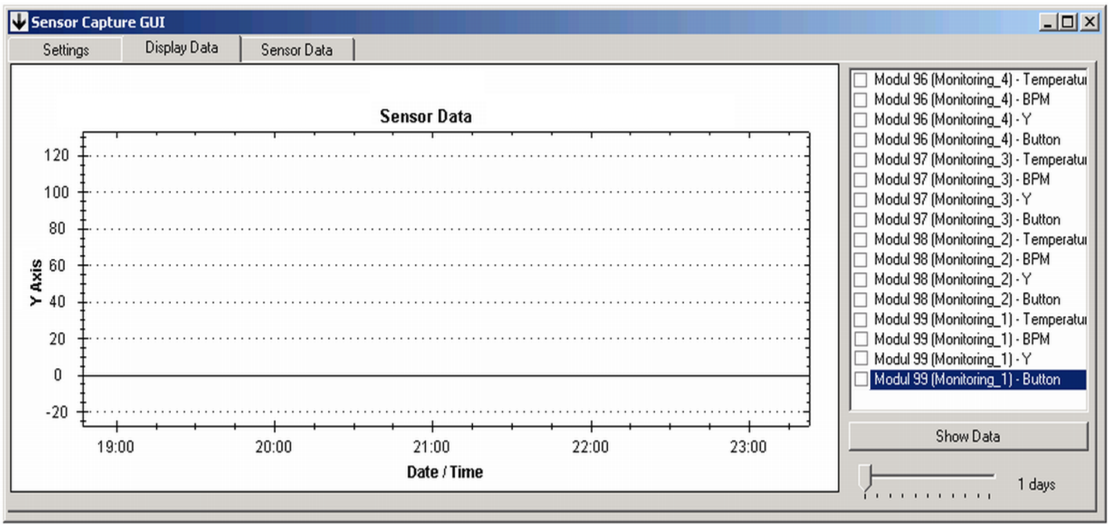


Figure 7-12: Sensor data can be retrieved easily in the GUI of the Serial Port Communication Program

Select the Sensors edited before and which you want to display by simply checking the boxes in front of the sensor. By changing the Track Bar you can choose how much data is to be displayed in the graph (this is necessary to reduce the display time). By clicking on “Show Data” the Values from the Database are displayed in the graph. You can now zoom in and out by using the wheel of the mouse or clicking in the graph.

On the first Tab (Settings) you can send Data to the Microcontrollers. Yet there is no command implemented but the Microcontrollers can receive the sent data, so only handling of data is required.

7.3.4 Database

The Database contains two Tables. One (t_save) contains the received Data. There are 7 different columns as described in Table 7-2.

Table 7-2: t_save description

Column	Contains
ID	Number of the entrance
sensor	Contains the sensor_number, which is the number of the Sensor unit
times	Contains the UNIX-Timestamp when the data was received
data1...data4	Contains the Value of the Sensor number X

When the data is received the values are stored in the direction string (Communication Protocol).

7.3.5 Algorithm of the code

The GUI provides the user with the option to choose the desired mode; text (i.e. string) or hex (i.e. binary). The following two sub-sections describe the algorithm followed in each mode which can be found in the “Communication Manager” class. Switch command was used with two cases as follows:

```
//determine the mode the user selected (binary/string)
switch (CurrentTransmissionType)
      {

      // String Mode
      case TransmissionType.Text:
            // Algorithm here
      break;
      // Binary Mode
      case TransmissionType.Hex:
            // Algorithm here
      break;
      }
```

7.3.6 String Mode

The string is the default mode for writing into the serial port. The algorithm used in this code can be summarized as follows:

- Read the data waiting in the existing buffer.
- Assign the current data as a string called “msg”.
- Create a buffer for the name, and a buffer for each sensor; humidity, temperature, pressure and light.
- Check the length of the msg buffer (it should be 27) since each sensor is represented by 5 characters, whereas 3 characters for the name. i.e. The format of the string looks like this:
  Name- temperature -BPM-Impact-panicbutton
  i.e. 000-00000-00000-00000-00000
  which is equal to 27 characters including the dashes in between.

- Create a connection to the data base using OleDbConnection and try-catch open it.

```
OleDbConnection con = new
OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data
Source="+VarClass.path);
```

- In order to get the desired string format, split the words in the serial string by (-).
- For each word in the string, remove all zeros to the left and consider the remaining length of the buffer.
- The name, denoted by the int k in the code, cannot be 0 or any 3-digit. Therefore, if it is a valid name, then insert the data into the Data Base using the following code:

```
string strSQL = "INSERT INTO t_save (sensor,times,data1,data2,data3,data4)
VALUES (" + k + "," + conv_Date2Timestam() + "," + Sensor 1 + "," + Sensor 2
+ "," + Sensor 3 + "," + Sensor 4 + ")";

OleDbCommand cmd = new OleDbCommand(strSQL, con);

cmd.ExecuteNonQuery();
con.Close();
```

- Add condition to the check that the length of each sensor buffer does not exceed 5 characters, and if so, then display them in the corresponding text boxes in the GUI.

7.3.7 Binary Mode

The algorithm used in this code can be summarised by the following steps:

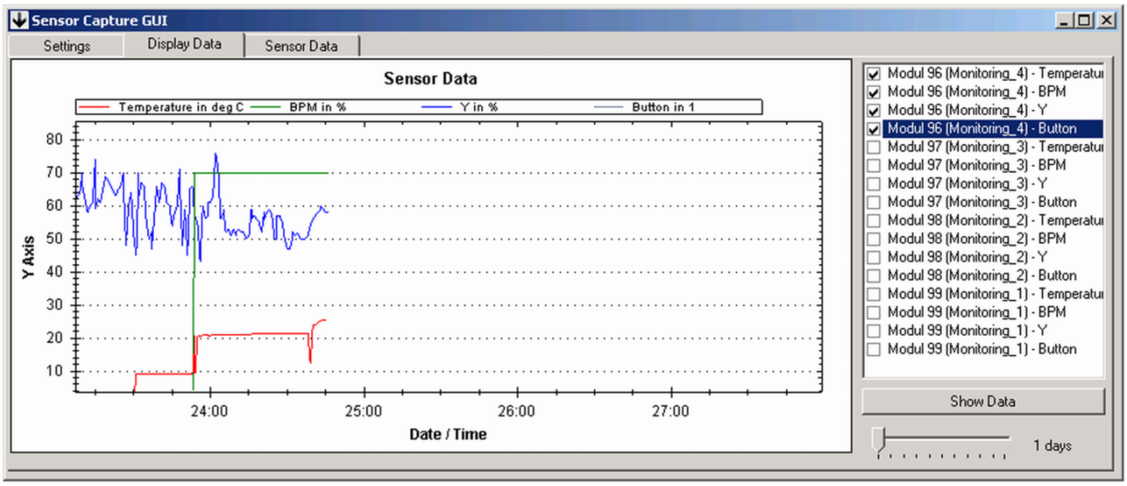
- Retrieve number of bytes in the buffer
- Create a byte array to hold the waiting data.

```
byte[] comBuffer = new byte[bytes];
```

- Read the data and store it.
- Display the data to the user.

The string, received as serial Data, is split into 5 parts (the address and sensors) and saved in an Access Database. In this stage the GUI also tests the data for correctness.

The Database contains a Table in which the different Settings for the sensors are stored. That makes it possible to attach different sensors and to change the position of the sensor data in the send string (section 7.3.2 XBee Communication Protocol). It is also possible to make a calibration for each sensor, for example to get the dot in the temperature. To get the right value from the sensor the sent data must be divided by 100 but in reality this value may be fine tuned to get the correct value. In the GUI it is possible to display one or multiple sensors (see figure 7.13) at the same time to give the user the chance to show different graphs and to compare different units.



☑ Modul 96 (Monitoring_4) - Temperatu
☑ Modul 96 (Monitoring_4) - BPM
☑ Modul 96 (Monitoring_4) - Y
☑ Modul 96 (Monitoring_4) - Button
☐ Modul 97 (Monitoring_3) - Temperatu
☐ Modul 97 (Monitoring_3) - BPM
☐ Modul 97 (Monitoring_3) - Y
☐ Modul 97 (Monitoring_3) - Button
☐ Modul 98 (Monitoring_2) - Temperatu
☐ Modul 98 (Monitoring_2) - BPM
☐ Modul 98 (Monitoring_2) - Y
☐ Modul 98 (Monitoring_2) - Button
☐ Modul 99 (Monitoring_1) - Temperatu
☐ Modul 99 (Monitoring_1) - BPM
☐ Modul 99 (Monitoring_1) - Y
☐ Modul 99 (Monitoring_1) - Button

Figure 7-13: Data display window showing three sensors data

Chapter 8: Conclusions

This report represents research, of an applied nature, done to monitor physiological parameters such as skin temperature, heart rate and body impact. A prototype of an integrated system with three sensors was successfully developed and tested to establish the proof of concept. The algorithms were tested and found to be accurate and reliable for the developed system. The novel aspect of the design is its low cost and detection of medical distress which does not necessitate pressing any panic button and last but not least ease of use. Individual sensors have been tested and results compared with references. The integrated system was tested with ZigBee wireless unit, and data was captured and displayed on GUI for sensor setup.

An important aspect of the design was miniaturisation, so that the system was as non-intrusive as possible to the wearer. This was achieved by the use of surface-mounted devices on the PCBs designed. Low power operational amplifier IC's were used to minimize battery consumption. The major costs come from the use of precision components, accelerometer and temperature sensor. The key advantage of our developed system over the ones that already exist in the market is that it sends out an alarm automatically without the patient/user having to push a panic button in an emergency. The developed system in comparision to already developed or available systems on the market is cheaper and smaller in size. Philips Lifeline medical alarm, St John's Lifelink, and ADT's NevaAlone medical alert system are some of the few medical alarms available in the market. They all have the same principle in that a wearable personal help button (that can be worn as a pendant or a wrist watch) is used. When the user needs medical attention, they press the alarm button and it activates the remote receiver unit or the telephone and dials the response centre for an ambulance. ADT NevaAlone [5] and Vital Link [7] is an elderly healthcare monitoring system that can be worn as a pendant or a wrist band with a personal help button. In the event of an emergency, the user presses the alarm button and the signal is sent to the ADT monitoring specialist via telephone line and an ambulance is sent. But in most cases the user may not be in a position to press the button in cases where they have become paralyzed or unconscious. Also, the health care monitors discussed are quite expensive. Along with the expense of buying the system, there is a monthly or fortnightly monitoring fee which varies from 70 to 100 dollars.

This report is documented extensively and the communication part discusses in step by step explanation how to use wireless devices, which can be useful for someone who wants to work on different wireless communication methodologies and their connection with Silabs microcontroller family. On the whole thesis can be used as a good referral source for further integrating/developing the system or for similar work/projects.

Also a bed monitoring system has been developed, which is part of an ongoing project of a smart home monitoring system.

8.1 Future Developments

An integrated system with three sensors developed. Although it was planned to add more physiological parameter measuring sensors, due to time constraints it was not done. There are key features of the design that could be improved given the time, which are detailed below. Developing several of these would make the device extendable to situations such as third world hospitals where cheap portable medical care is required, at risk inmates in prisons or health monitoring for athletes during exercise.

Blood Pressure

An Addition of a blood pressure sensor using sensing techniques like optical sensors could be used to continuously measure the blood pressure non-invasively in steady conditions. It would be possible to adapt heart rate sensor developed and its algorithm to also detect changes in blood pressure by measuring the speed of the pulse detected. This would provide a quick non-invasive alternative to the methods currently used (e.g. inflatable cuff band).

Blood Oxygen Count

The addition of extra infra-red sensors and algorithms would give the device the means to measure blood oxygen levels. This is very important as blood oxygen levels can give early warning of strokes, heart attacks, stress and shock.

All on Wrist (no finger unit)

The introduction of a finger strap was required to achieve a reliable heart rate sensor response, but further research into the techniques and components used could make it feasible to move the heart rate sensor back onto the wrist unit. This would make the device much more comfortable and more aesthetically pleasing to wear. Or even with the finger unit it can be designed in the shape of ring with more accurate hardware development tools (software)

Impact senor algorithm

A multi-stage fall detection algorithm with complex inference techniques would be required to improve the activity recognition accuracy of the sensor. This will improve overall accuracy and behavioral activity recognition

Flexible PCB

The final prototype was created on stiff PCB. Ideally, this would be implemented on flexible PCB, making the wrist band much more comfortable to wear. This has not been implemented because of time and resource constraints.

Water Proofing

Several issues were raised during user testing and evaluation, one of which was the ability to wear the wrist band in the shower (as this is a common place for an elderly person to slip, fall and injure themselves. This would require a waterproof material for the wrist band as well as sealing any open gaps.

Code to Reduce Transmission Time and Errors

Reducing power consumption was an important consideration. Further work in this field would involve reducing the amount of code that is transmitted to the receiver unit, possibly by implementing a form of compression in the packet. By sending less data, it is possible to also reduce the amount of time the microcontroller must transmit data, which draws more power than any other operation. Implementation of hamming codes could also allow errors in packet transmission to be detected and corrected.

Bed Sensors

An Intelligent Bed Sensor System is being developed based on a force sensor placed under the four legs of a bed. Currently the system can estimate the position of the actual loading point. The current sensors used for the laboratory study are not of higher capacity and cannot be used for the actual bed. The main objective of the work is to identify the sleeping pattern of the elderly person living alone. The response of the sensors can be used to investigate the normal sleep, sitting on the bed at night time etc. and the system is used as a part of the wireless sensors system based home monitoring system. The higher range sensors will be supplicated in an actual bed and the experimental results will be reported in the final version.

Chapter 9: References

[1] D. Patranabis, “Sensors and Transducers” (PHI Learning Pvt. Ltd), pp. 2, 3, 4

[2] J. Fraden, “Handbook of Modern Sensors: Physics, Designs, and Applications (American Institute of Physical Process)”, Second Edition, 1997, pp. 1, 5, 28, 33

[3] http://chesapeake.towson.edu/data/all_sensors.asp

[4] W. Gopel and K.D. Schierbaum, “Sensors: A Comprehensive Survey (City Publishers)”, vol. 2, pp. 5, 9, 1991

[5]ADT Security, “Neva Alone”, http://www.adtsecurity.co.nz/home_security/nevaalone

[6] LifeLink, “Medical Alert Alarm System”, http://www.callforassistance.com/

[7] Vital Link, “Emergency Response Systems”, http://www.vital-linkinc.com/

[8] St John, “Lifelink”, http://www.stjohn.org.nz/products/lifelink.aspx

[9] Smart-Link, “Medi-Call Dialler Unit”, http://www.smartlink.com.au/products/medi-call/index.htm

[10] J. R. Davis-Havill and J. L. Walley, “Biomechanical monitoring apparatus”. World Intellectual Property Organization, WO05120348A1, 22nd December 2005.

[11] Hao, Y., Foster. Wireless sensor networks for health monitoring applications. Physiological Measurement 29(11), R27-R56 (2008)

[12] M. Scholtz, “Addressing the global demands for improved healthcare,” in Proc. Telemedicine 21st Century, Opportunities Citizens, Society, Industry, pp.11-18 1999.

[13] J.M. Wilkinson, “Medical market for Microsystems,” Int. Newsletter Microsyst. MEMS, no. 4/02, p. 37, Sept. 2002.

[14] A. Lymberis, “Smart wearable systems for personalised health management: Current R&D and future challenges,” in Proc. 25th Ann. Int. Conf. IEEE EMBS, vol. 4, pp. 3716–3719 Sept. 2003.

[15] P. Bonato, “Wearable sensors/systems and their impact on biomedical engineering,” IEEE Eng. Med. Biol. Mag., vol. 22, no. 3, pp. 18–20, May/June 2003.

[16] A survey on wearable biosensor systems for health monitoring Pantelopoulos, Alexandros; Bourbakis, Nikolaos; Engineering in Medicine and Biology Society, 2008. 30th Annual International Conference of the IEEE Publication Year: 2008, Page(s): 4887 – 4890, EMBS 2008.

[17] Subhas Chandra Mukhopadhyay, Aime Lay-Ekuakille Eds., “Advances in Biomedical Sensing, Measurements, Instrumentation and Systems (Springer Berlin Heidelberg-Publishers) Lecture notes in Electrical Engineering Volume 55) pp.29-42, 2010.

[18] U. Anliker, J. A. Ward, P. Lukowicz, G. Tröster, F. Dolveck, M. Baer, F. Keita, E. B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Scmid, M. Vuskovic, “AMON: A Wearable Multiparameter Medical Monitoring and Alert System”, IEEE Transactions on Information Technology in Biomedicine, Vol.8, No. 4, Dec. 2004

[19] L. Gatzoulis, I. Iakovidis, “Wearable and Portable eHealth Systems, Technological Issues and Opportunities for Personalized Care”, IEEE Engineering in Medicine and Biology Magazine, September/October 2007

[20] M. Sung, C. Marci, A. Pentland, “Wearable feedback systems for rehabilitation”, Journal of NeuroEngineering and Rehabilitation, 29June, 2005

[21] Pandian, P.S., Mohanavelu, K., Safeer, K.P>, Kotresh, T.M, Shakunthala, D.T., Gopal, P., Padaki, V.C.: SMART Vest: Wearable multi-parameter remote physiological monitoring system. Medical Engineering & physics 30, 466-477 ,2008.

[22] J. Habetha, “The MyHeart Project – Fighting Cardiovascular Diseases by Prevention and Early Diagnosis”, 28th Annual International Conference, IEEE Engineering in Medicine and Biology Society (EMBS), NY City, USA, August 30 – Sept. 3, 2006.

[23] M.Pacelli, G.Loriga., N.Taccini, R.Paradiso, “Sensing Fabrics for Monitoring Physiological and Biomechanical Variables: E-textile solutions”, Proceedings of the 3rd IEEE-EMBS International Summer School and Symposium on Medical Devices and Biosensors, MIT, Boston, USA, Sept.4-6, 2006

[24] J.L. Weber, F. Porotte, “Medical Remote Monitoring with clothes”, PHealth, Luzerne, 31 Jan, 2006

[25] Sensatex, Inc., “Development of the Sensatex SmartShirt”, pHealth, 2006.

[26] Vivago at www.istsec.fi/vivago-pam/, accessed March 2008

[27] Chong, C. and Kumar, S. P.; “Sensor networks: Evolution, opportunities, and challenges,” Proceedings of the IEEE, vol. 91, no. 8, pp. 1247--1256, Aug. 2003.

[28] IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, (http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01539893) VOL. UFFC-34, NO. 2, A Sensor Classification Scheme Richard M. White, FELLOW, IEEE, March 1987.

[29] Razvan, Mihai Micea, Bogdan Ciubotaru, Dan Chiciudean, Vladimir Cretu, “eBML: A Formal Language for Behavior Modeling and Application Development in Robotic Collectives” IEEE International Workshop on Robotic and Sensors Environments, Ottawa, Canada, Proc. Of the ROSE 2007.

[30] Chris Otto, Aleksandar Milenković, Corey Sanders, Emil Jovanov, University of Alabama in Huntsville , System Architecture of a Wireless Body Area Sensor Network for Ubiquitous Health Monitoring, Journal of Mobile Multimedia, Vol. 1, No.4 pp. 307-326, 2006.

[31] http://www.andmedical.com/and_med.nsf/html/CP-1THW

[32] http://singularityhub.com/wireless-health-monitoring-system-for-under-200/-2009/08/25

[33] Wu, W.H., Bui, A.A.T., M.A., Au, L.K., Binney, J.D., Kaiser, W.J.:MEDIC: medical embedded device for individual care. Artificial intelligence in medicine42, 137-152, 2008.

[34] Jin, Z., Oresko,J.,Huang, S.,Cheng, A.C.: heartToGo: A Personalised Medicine Technology for Cardiovascular Disease Prevention ands Detection. In: IEEE/NIH Life Science Systems and Applications Workshop (LiSSA2009), pp. 80-83, 2009.

[35] http://ukhealthcare.uky.edu/content/content.asp?pageid=P00866

[36] http://en.wikipedia.org/wiki/Normal_human_body_temperature

[37] http://hypertextbook.com/facts/2001/AbantyFarzana.shtml

[38] Problems in the Measurement of Human Body Temperature- Biomedical Engineering, Vol. 42, No. 1, 2008, pp. 1_3. Translated from Meditsinskaya ekhnika, Vol.42, No. 1, pp. 3-6, 2008.

[38] http://www.answers.com/library/Sports+Science+and+Medicine-cid-2878778

[39] http://datasheets.maxim-ic.com/en/ds/DS600.pdf

[40] http://www.pnas.org/content/5/6/218.full.pdf

[41] Heart Rhythm, Volume 6, Issue 11, Supplement 1, , Pages S68-S75 Richard L. Verrier, Alex Tan, November 2009.

[42]N.Townsend and M.Term, "*Pulse Oximetry*", Medical Electronics, 2001.

[43] www.medterms.com/script/main/art.asp?articlekey=3674

[44] Progress in Cardiovascular Diseases, Volume 52, Issue 1, Pages 11-19, July-August 2009.

[45] Kremin, V. & Matviyenko, S., "*Pulse-Sensing Optical Mouse*", Circuit Cellular, vol. 194, pp. 12, 2006

[46] http://www.info-ab.uclm.es/labelec/solar/otros/infrarrojos/Datasheet/fotoTrans7.pdf

[47] http://www.national.com/mpf/LM/LM324.html#Overview

[48] Salva A, Bolibar I, Pera G, Arias C. Incidence and consequences of falls among elderly people living in the community. Med Clin (Barc);122(5):172–6, 2004.

[49] Lord SR, Sherrington C, Menz HB. Falls in older people: risk factors and strategies for prevention. Cambridge University Press; 2001.

[50] Tinetti ME, Williams CS. Falls, injuries due to falls, and the risk of admission to a nursing home. N Engl J Med 337(18):1279–84, 1997.

[51] Kannus P, Sievanen H, Palvanen M, Jarvinen T, Parkkari J. Prevention of falls and consequent injuries in elderly people.Vol. 366(9500):1885–93, Lancet 2005.

[52] Vital Statistics Fourth Quarter and Yearly Summary, 2005. Central Statistics Office: Ireland; www.cso.ie, 2005.

[53] Cotter PE, Timmons S, O'Connor M, Twomey C, O'Mahony D. The financial implications of falls in older people for an acute hospital.175(2):vol. No.11–3, Ir JMed Sci 2006;.

[54] M.J. Mathie, A.C. Coster, N.H. Lovell, and B.G. Celler, "Accelerometry: providing an integrated, practical method for long-term, ambulatory monitoring of human movement," Physiol. Meas.;25(2):R1-20, 2004.

[55] G. Williams, K. Doughty, K. Cameron, and D.A. Bradley, "smart fall and activity monitor for telecare applications," Proc. 20th Annual Int.Conf. of the IEEE Engineering in Medicine and Biology Society, 1998.

[56] K. Doughty, R. Lewis, and A. McIntosh, ”The design of a practical and reliable fall detector for community and institutional telecare,” J.Telemed. Telecare, 6 S150-4, 2000.

[57] M. Mathie, J. Basilakis, and B.G. Celler, ”A system for monitoring posture and physical activity using accelerometers,” 23rd Annual Int. Conf. IEEE Engineering in Medicine and Biology Society, 2001.

[58] http://www.analog.com/en/sensors/inertial-sensors/adxl213/products/product.html

[59] http://www.tekscan.com/pdfs/FlexiforceUserManual.pdf

[60] http://www.tekscan.com/pdfs/Brochure-Flexiforce.pdf

[61] Silicon-Labs. “C8051F020 Datasheet pdf - Mixed-Signal 64KB ISP FLASH MCU.”*DatasheetCatalog.org*.http://www.datasheetcatalog.org/datasheets2/43/4308973_1.pdf

[62] http://pdf1.alldatasheet.com/datasheet-pdf/view/84570/ETC/C8051F020.html

[63] Gupta, Sen Gourab, and Tin Chew Moi. *Embedded Programming with Field-Programmable Mixed-Signal μControllers.* 2005.

[64] Digi. “XBee® & XBee-PRO® 802.15.4 OEM RF Modules - Digi International.” http://ftp1.digi.com/support/documentation/90000982_B.pdf (accessed June 6, 2010).

[65] http://ftp1.digi.com/support/documentation/90001003_A.pdf

[66] Finger Pulse Oximeter, www.favoriteplus.com

[67] Doughty, K., Lewis, R., McIntosh, A. The Design of a Practical and Reliable Fall Detector for Community and Institutional Telecare. In *Journal of Telemedicine and Telecare,* 2000

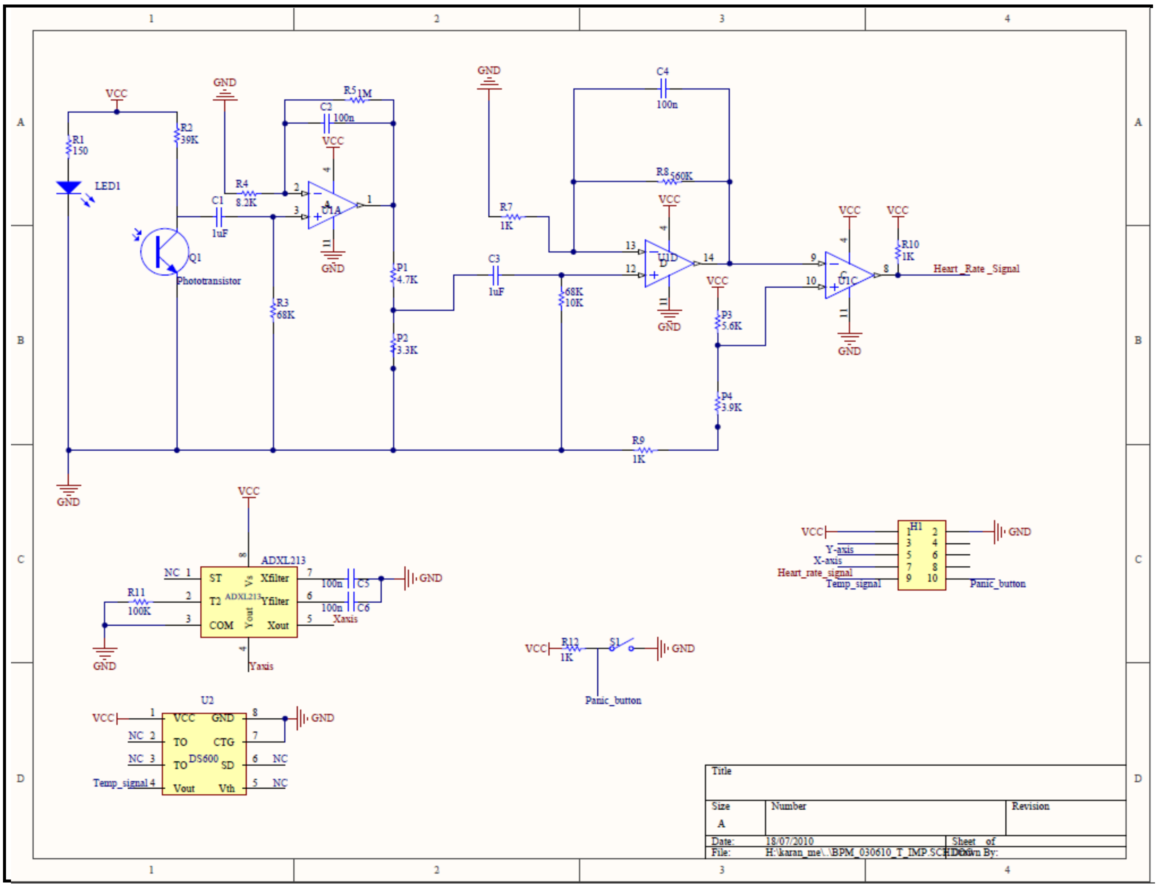
[68] Jay Chen, Karric Kwong, Dennis Chang, Jerry Luk, Ruzena Bajcsy: Wearable Sensors for Reliable Fall Detection. Department of Electrical Engineering and Computer Sciences, University of California, Berkeley Berkeley, CA 94720. Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai, China, September 1-4, 2005

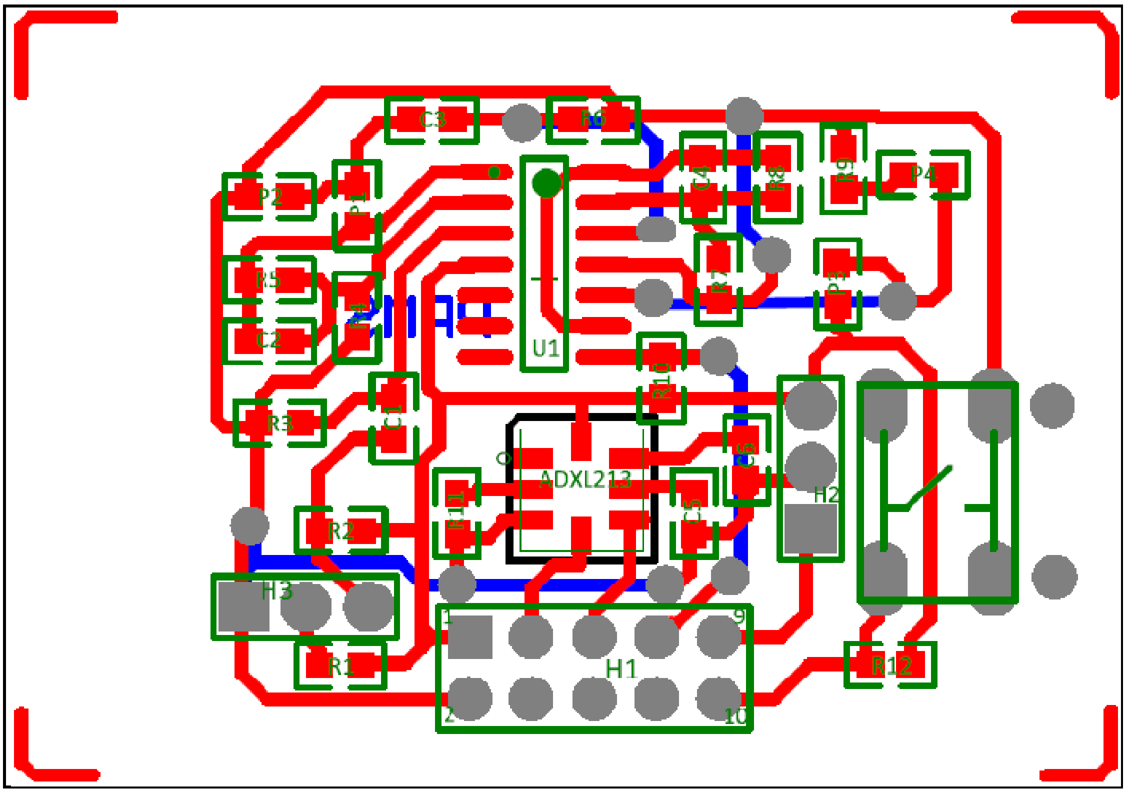
[69] Fall Detection Sensor System, Swiss Centre for Electronics and Microtechnology (CSEM) www.csem.ch

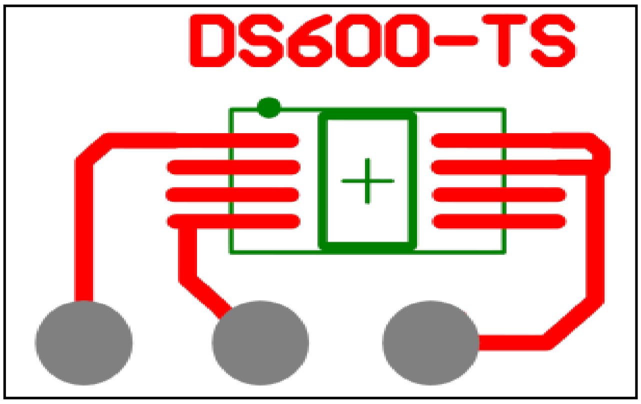
Chapter 10: Appendix

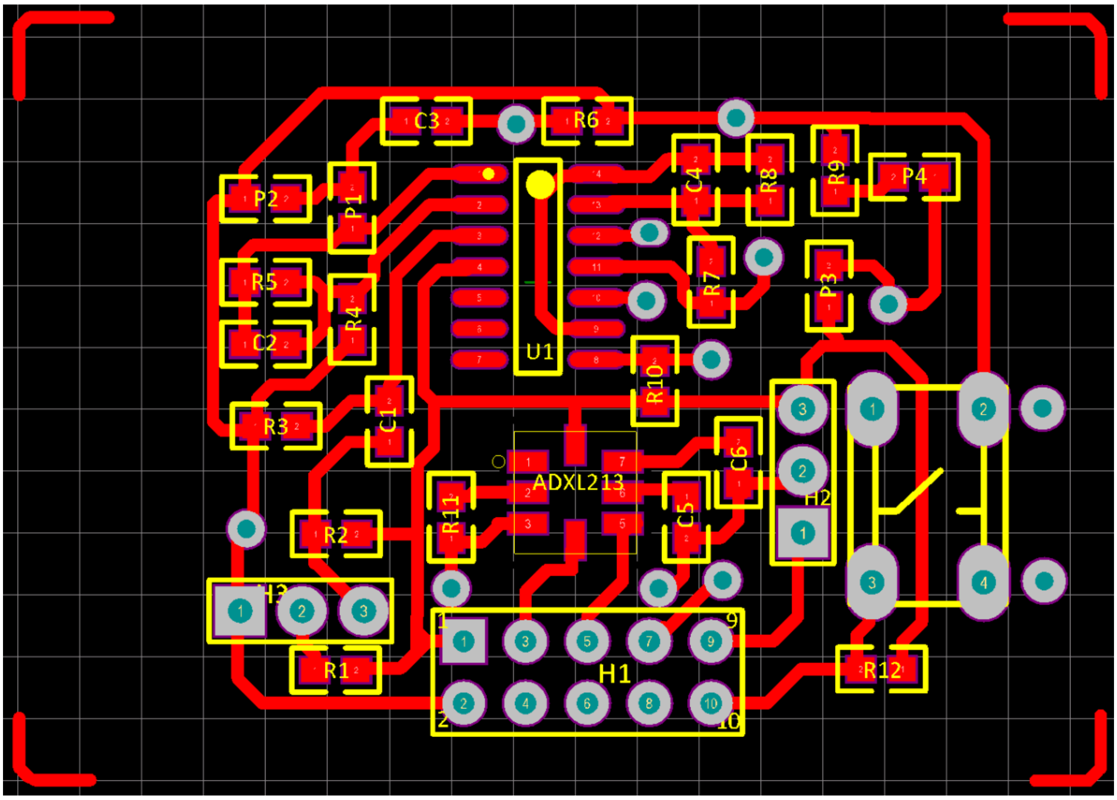
10.1 Schematics and PCB Layouts

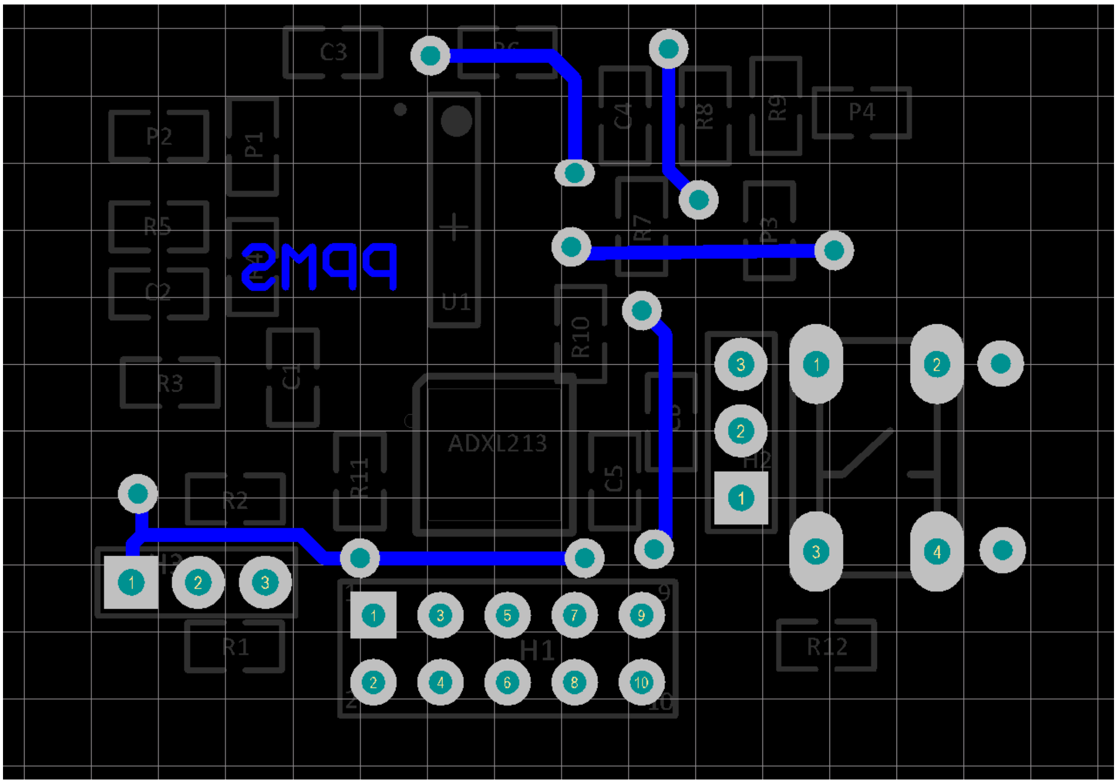
10.1.1 Sensor Unit




C3
C6
P2
P1
R5
R4
C2
C1
R3
R2
H3
R1
U1
ADXL213
R11
R10
C5
C4
R8
R9
P4
R7
P3
H2
H1
1
2
9
10
R12


DS600-TS


C3
R6
P2
P1
R5
C2
R4
R3
C1
U1
C4
R8
R9
P4
R7
P3
R10
H2
C6
ADXL213
R11
C5
R2
H3
R1
H1
R12


C3
P2
P1
R5
C2
R3
C1
U1
C4
R8
R9
P4
R7
P3
R10
H2
ADXL213
R11
C5
R2
H3
R1
H1
R12

10.2 Parts List

Comment	Description	Designator	LibRef	Quantity	Value
LM324		U1	LM324	1	
Cap	Capacitor	C1, C2, C3, C4, C5, C6	Cap	6	10n, 100n, 1u
Photo Sen	Phototransistor	Q1	Photo Sen	1	
DS600	Temperature IC		DS600	1	
Header 10	Header, 10-Pin	H1	Header 10	1	
LED1	Typical RED GaAs LED	LED1	LED1	1	
Header 3	Header, 3-Pin	Q1	Header, 3	2	
Res1	Resistor	R1, R2, R3, R4, R5, R6, R7, R8, R9,R10, R11, R12, P1,P2, P3, P4	Res1	16	1k, 3.3K 4.7K, 5.6K, 8.2K, 10K, 39K,100K, 560K, 320
SW_PB		sw1	SW_PB	1	
ADXL312		ADXL312	Sheet1	1	

10.3 Microcontroller Code

10.3.1 Wrist Band

```
/*=WristBand Unit
=========================================================
  * Author(s): Karan Malhi

//
//    program for physiological parameter monitoring on a patient
//    contains temperature & BPM measurement, fall detection and a
      panic button
//
//    Tool chain: SDCC
//

//------------------------------------------------------------------
// Includes
//------------------------------------------------------------------
#include <c8051f020.h>                  // SFR declarations
#include <stdio.h>
#include <string.h>

//------------------------------------------------------------------
// Global CONSTANTS
//------------------------------------------------------------------

#define BAUDRATE        9600        // Baud rate of UART in bps
#define SYSCLK          22118400    // approximate SYSCLK  freq. in Hz
#define SAMPLE_DELAY    1000
#define INT_DEC         50           // sample rate (Integrate and
                                     decimate ratio)
#define SAR_CLK         25000       // Desired SAR clock speed
```

```
#define SAMPLE_RATE     50000       // Sample frequency in Hz
#define Input_BPM       P0_2        // Inputport P0.2 for BPM
#define Input_Accel     P0_3        // Inputport P0.3 for Accelerometer
#define     button      P0_4        // Input port for emergency button


//-----------------------------------------------------------------
// Global Variables
//-----------------------------------------------------------------
char unit=99;
char out[30];
char i;
char k=0,v=0;
char adc_ready=0;

unsigned long temperature;
unsigned long BPM_T_count=0,BPM_T_count_value=0;
unsigned long T_count_value_y=0,T_count_y=0;
unsigned int Duty_y;
unsigned int Frequency;
unsigned int Duty_count_y=0,BPM;
unsigned int D_count_value_y=0;

xdata unsigned int temperature_array[INT_DEC] = {0};
                                    //--collect ADC Results in array
                                    //--- filtered Results
idata unsigned int average=0;
idata unsigned long sum=0;          //--average of colleted Results
idata unsigned long value=0;
idata unsigned int Result;          // ADC0 decimated value

//---------------------------------------------------------------
// Function PROTOTYPES
//--------------------------------------------------------------------
---------------
void small_delay (char d);
void large_delay (char d);
void huge_delay (char d);

void Port_Init (void);
void ADC0_Init (void);
void Timer3_Init (unsigned int counts);
void SYSCLK_Init (void);

void Init_UART0(void);                    //-- configure and initialize
the UART0
void OSCILLATOR_Init (void);
void send_serial(void);
void Init_Timer0(void);                   // Initialise the timer 0

//--------------------------------------------------------------------
---------------
// MAIN Routine
//--------------------------------------------------------------------
---------------
void main (void)
{
EA =0;
```

```
                  // disable watchdog timer
WDTCN = 0xde;
WDTCN = 0xad;

OSCILLATOR_Init();
Port_Init();
Init_Timer0();                            // Initialise the timer 0

Timer3_Init (SYSCLK/12/50);         // Init Timer3 to generate
interrupts
Init_UART0();
ADC0_Init();                        // Init ADC

EA =1;

huge_delay(5);    //-- give time to the sensors to settle down

while (1)
{
if (adc_ready)
{
sum =0;

// making a simple average
//-----------------------------------------------------------

for (i=0; i<INT_DEC; i++)
{
sum=sum+temperature_array[i];
}
average=sum/INT_DEC;                //-average of values in array

value=0;k=0;

//---------------------------------------------------------------
// sort out the poor value from the temperature array
//delete the values which are too far away from the average
//---------------------------------------------------------------

for (i=0; i<INT_DEC; i++)
{
if ((temperature_array[i] <= (average+200)) && (temperature_array[i] >=
(average-650)))
{
value+=temperature_array[i];
k++;
}
}
if (k != 0)
Result=value/k;
else
Result = 0;

//------------------------------------------------------------------
// calculate the temperature out of the collected and averaged ADC0
data
//------------------------------------------------------------------
```

```
temperature =  Result;
temperature = (((temperature/16) *2430/4095 )-509);
                  //(temperature/16) *2430/4095 ) 16= gain;
                  //max. signal 2430mV corresponds to 4095 digital
                  steps
temperature = (temperature*10000/645);
                  // temperature calculation according to

                  //manufacturer "^C = (Vout - 509mV)/ (6.45mV/^C)"

adc_ready=0;
}

//------------------------------------------------------------------
//  calculating the BPM & Duty cycle for accelerometer
//------------------------------------------------------------------

if (BPM_T_count_value !=0)
{
Frequency = 10000000/BPM_T_count_value;
BPM = (Frequency*6)/10;
}
else
BPM = 0;

if (T_count_value_y != 0)
Duty_y = (D_count_value_y*100)/T_count_value_y;
else
Duty_y = 0;


if (button==1)
{
button==1;
sprintf(out,"%03d-%05u-%05u-%05u-%05u",unit, (unsigned int)
temperature, BPM, Duty_y,'01110');
send_serial();
button==0;
}

else
{
//-------------------------------------------------------------------
// write the data into the string and send it via serial port to
terminal server
//-------------------------------------------------------------------
sprintf(out,"%03d-%05u-%05u-%05u-%05u",unit, (unsigned int)
temperature, BPM, Duty_y, button);
send_serial();
}
huge_delay(20);
}

}


//------------------------------------------------------------------
// PORT_Init
```

```
//----------------------------------------------------------------
void Port_Init (void)
{
XBR0        = 0x04;         // Enable crossbar
XBR1        = 0x14;         // Enable INT0 & INT1
XBR2     = 0x40;            // Enable crossbar,  weak pull-ups


P0MDOUT     = 0x01;      // enable TX0 as a push-pull output
// P0MDOUT4 = 0;         // set P0.4 as input (panic)
//   P0.4 = 1;          // '' '' '' '' '' '', enable it


IT0=1;                  // INT0 edge triggered
IT1=1;                  // INT1 edge triggered
EX0=1;                  // enabled external interrupt INT0
EX1=1;                  // enabled external interrupt INT1

P0 |=0x08;              // P0.3 as input for Impactsensor
P0 |=0x10;              // P0.4 as an input for panic button
}


//----------------------------------------------------------------
//----------------------------------------------------------------

void OSCILLATOR_Init (void)
{
int i;                           // delay counter

OSCXCN = 0x67;                   // start external oscillator with
                                 // 22.1184MHz crystal
for (i=0; i < 256; i++) ;        // wait for oscillator to start

while (!(OSCXCN & 0x80)) ;       // Wait for crystal osc. to settle

OSCICN = 0x88;                   // select external oscillator as
                                    SYSCLK
                              // source and enable missing clock
                              // detector
}


//----------------------------------------------------------------
// Timer3_Init
//----------------------------------------------------------------
// Configure Timer3 to auto-reload and generate an interrupt at
interval
// specified by <counts> using SYSCLK/12 as its time base.
//----------------------------------------------------------------
void Timer3_Init (unsigned int counts)
{

TMR3CN = 0x00;      //-- Stop Timer3; Clear TF3;
//-- use SYSCLK/12 as timebase
//    TMR3CN |= 0x02;         //-- if you want to use SYSCLK (NOT
SYSCLK/12)
```

```
TMR3RL  = -counts;  //-- Init reload values
TMR3    = 0xffff;   //-- set to reload immediately
EIE2   |= 0x01;     //-- enable Timer3 interrupts
TMR3CN |= 0x04;     //-- start Timer3 by setting TR3 (TMR3CN.2) to 1
}


//-----------------------------------------------------------------
//-----------------------------------------------------------------
void Init_UART0(void)
{
PCON |= 0x80;           //SMOD0=1 UART0 baud rate divided by 2 disabled
TMOD  |= 0x20;          // TMOD: timer 1, mode 2, 8-bit reload
CKCON |= 0x10;          // Timer1 uses SYSCLK as time base, T1M = 1
SCON0   = 0x50;         // SCON0: mode 1, 8-bit UART, enable RX

TH1 = 0x70;             // reload value for baudrate 9600
TR1    = 1;             // start Timer1

IE |= 0x10;
IP |=0x10;             //Enable UART interrupt and set high
                        priority level

TI0 = 1;               // Indicate TX0 ready
RI0 = 0;               //clear received interrupt flag
}

//-----------------------------------------------------------------
//-----------------------------------------------------------------
void Init_Timer0 (void)
{
CKCON |= 0x08;    // Timer 0 uses system clock T0M = 1
TMOD |= 0x02;     // Timer 0 in Mode 2 auto reload
TL0 = 0xFF;
TH0 = 0x22;
TR0 = 1;               // Start Timer 0
ET0=1;                 // Enable interrupt request for timer 0

}


//------------------------------------------------------------------
//------------------------------------------------------------------
void ADC0_Init (void)              // 12bit converter
{
ADC0CN = 0x05;               // ADC0 disabled; normal tracking
            // mode; ADC0 conversions are initiated
            // on overflow of Timer3; ADC0 data is
            // left-justified

REF0CN = 0x03;              // Enable temp sensor, on-chip VREF,
            // and VREF output buffer

AMX0CF = 0x00;              // AIN inputs are single-ended (default)

AMX0SL = 0x01;              // Select AIN0.1 pin as ADC mux input

ADC0CF = 0x40; // SAR frequency = 2.5 MHz approximately,
```

```
               //gain = 1 (speed of conversion)...
               //time of how fast conversion takes place
               //ADC0CF = 0x80;
               // SAR frequency = 941 KHz approximately, gain = 1

EIE2 |= 0x02;                // enable ADC interrupts
AD0EN = 1;                  // Enable ADC

}



//-------------------------------------------------------------------
// Interrupt service routines
//-------------------------------------------------------------------
//-------------------------------------------------------------------
// Timer3_ISR
//-------------------------------------------------------------------
// This routine starts ADC conversion whenever Timer3 overflows.
//-------------------------------------------------------------------
void Timer3_ISR (void) interrupt 14
{
TMR3CN &= ~(0x80);           // clear TF3
}

//------------------------------------------------------------------
//------------------------------------------------------------------
void ADC0_ISR (void) interrupt 15
{
AD0INT = 0;             // Clear ADC conversion complete
                        // indicator

temperature_array[v]=ADC0;      // collecting data from ADC0

if(temperature_array[v]<=22000) // discard readings which are too high
{
v++;
if(v>(INT_DEC-1))
{
v=0;
adc_ready=1;
}
}
}


//-------------------------------------------------------------------
//-------------------------------------------------------------------
void Timer0_ISR (void) interrupt 1
{
TF0 = 0;                            // reset timer overflow flag
BPM_T_count++;                      // time count for BPM
T_count_y++;                        // time count for impactsensor

if (Input_Accel == 1)         // if the accelerometer signal is high
increment Duty_count_y
{
Duty_count_y++;
```

```
}
}

//-------------------------------------------------------------------
//-------------------------------------------------------------------
void INT0_ISR(void)  interrupt 0    //-- for BPM
{
BPM_T_count_value = BPM_T_count;
BPM_T_count = 0;
}


//-------------------------------------------------------------------
//-------------------------------------------------------------------
void INT1_ISR (void) interrupt 2    //-- for accelerometer
{

D_count_value_y = Duty_count_y;
Duty_count_y = 0;

//-- for the total period of the signal from accelerometer

T_count_value_y = T_count_y;
T_count_y = 0;
}


//------------------------------------------------------------------
//  serial port
//------------------------------------------------------------------
//

void send_serial(void)                   // sending serial string
{
char j=0;
while (out[j]!='\0')                // write string into SBUF
{
SBUF=out[j];
j++;
while(!TI);
TI=0;
}
}

//------------------------------------------------------------------
// delay functions
void small_delay (char d)
{
while (d--);
}

void large_delay (char d)
{
while (d--)
small_delay (255);
}

void huge_delay (char d)
```

```
{
while (d--)
large_delay (255);
}
```

10.4 Receiver Unit (ZigBee GUI)

C Sharp code

```
#region comPort_DataReceived
/// <summary>
/// method that will be called when theres data waiting in the buffer
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
void comPort_DataReceived(object sender, SerialDataReceivedEventArgs e)
{
//char buffer[10];
//determine the mode the user selected (binary/string)
switch (CurrentTransmissionType)
{

//user chose string
case TransmissionType.Text:
//read data waiting in the buffer
string msg = comPort.ReadExisting();
string s_buffer = String.Empty;     // name
string msg_buffer = String.Empty;   // Temperature
string msg_buffer2 = String.Empty;  // BPM
string msg_buffer3 = String.Empty;  // Impact
string msg_buffer4 = String.Empty;  // Button
int old = 0;


buffer = buffer + msg;
if (msg.Length >= 27 || buffer.Length >= 27)
{
if (msg.Length > 27)
buffer = String.Copy(msg);
//buffer = buffer.Substring(0,buffer.Length - 1);
OleDbConnection con = new
OleDbConnection(@"Provider=Microsoft.Jet.OLEDB.4.0;Data
Source="+VarClass.path);

try
{
con.Open();


string[] words = buffer.Split('-');
int i = 0,k=0;
foreach (string word in words)
{
```

```
// name
if (i == 0)
{
s_buffer = word.TrimStart('0'); ;
if (word.Length <= 0 || word.Length > 4)
k = 0;
else if (word.Length >= 1)
{
int quantity;
if (int.TryParse(s_buffer, out quantity) == false)
    s_buffer = "0";
k = int.Parse(s_buffer);
}
}
// Temperature Sensor
//(Remove all zeros to the left, and expected to be 4 or 5 char left)
else if (i == 1)
{

msg_buffer = word.TrimStart('0');
if (word.Length < 4 || word.Length>5)
k = 0;
if (word.Length == 4)
msg_buffer=msg_buffer+"0";      // Adding a zero to the right
int quantity;
if (int.TryParse(msg_buffer, out quantity) == false)
msg_buffer = "0";
int one = int.Parse(msg_buffer);

}
// BPM Sensor
// (Remove all zeros to the left, and expected to be 5 char left)
if (i == 2)
{

msg_buffer2 = word.TrimStart('0');

if (word.Length < 5 || word.Length > 5)
k = 0;                  // i.e. There is an error in the name

int quantity;
if (int.TryParse(msg_buffer2, out quantity) == false)
msg_buffer2 = "0";
}
// Impact Sensor
//(Remove all zeros to the left, and expected to be 5 char left)
if (i == 3)
{

msg_buffer3 = word.TrimStart('0');
if (word.Length < 5 || word.Length > 5)
k = 0;
int quantity;
if (int.TryParse(msg_buffer3, out quantity) == false)
msg_buffer3 = "0";

}
// Button
```

```
// (Remove all zeros to the left, and expected to be 5 char left)
if (i == 4)
{

msg_buffer4 = word.TrimStart('0');
if (word.Length < 5 || word.Length > 5)
k = 0;
int quantity;

if (int.TryParse(msg_buffer4, out quantity) == false)
msg_buffer4 = "0";
if (quantity < old - 100)
{
msg_buffer4 = msg_buffer4 + "0"; // Adding a zero to the right

}
old = quantity;
}
DisplayData(MessageType.Incoming, word);
i++;
}


// if it passes (name, denoted by the int k, cannot be 0 or any 3-digit
int)
if (k != 0 && k<=999)
{
string strSQL = "INSERT INTO t_save
(sensor,times,data1,data2,data3,data4) VALUES (" + k + "," +
conv_Date2Timestam() + "," + msg_buffer + "," + msg_buffer2 + "," +
msg_buffer3 + "," + msg_buffer4 + ")";
OleDbCommand cmd = new OleDbCommand(strSQL, con);

cmd.ExecuteNonQuery();
con.Close();



_displayS1.Invoke(new EventHandler(delegate
{
    _displayS1.Text = msg_buffer;
}));
_displayS2.Invoke(new EventHandler(delegate
{
    _displayS2.Text = msg_buffer2;
}));
_displayS3.Invoke(new EventHandler(delegate
{
    _displayS3.Text = msg_buffer3;
}));
_displayS4.Invoke(new EventHandler(delegate
{
    _displayS4.Text = msg_buffer4;
}));
switch (k)
{
case 1:
_displayS1.Invoke(new EventHandler(delegate
```

```
{
    _displayS1.Text = msg_buffer;
}));

break;
case 2:
_displayS2.Invoke(new EventHandler(delegate
{
   _displayS2.Text = msg_buffer3;
}));
break;
case 3:
_displayS3.Invoke(new EventHandler(delegate
{
   _displayS3.Text = msg_buffer;
}));
break;
case 4:
_displayS4.Invoke(new EventHandler(delegate
{
   _displayS4.Text = msg_buffer;
}));
break;
default:
break;

}
}
buffer = string.Empty;

}

catch (OleDbException exep)
{
}}
//display the data to the user
//DisplayData(MessageType.Incoming, msg + "\n");


//int n = atoi(buffer);


break;
//user chose binary
case TransmissionType.Hex:
//retrieve number of bytes in the buffer
int bytes = comPort.BytesToRead;
//create a byte array to hold the awaiting data
byte[] comBuffer = new byte[bytes];
//read the data and store it
comPort.Read(comBuffer, 0, bytes);
//display the data to the user
DisplayData(MessageType.Incoming, ByteToHex(comBuffer) + "\n");
break;
default:
//read data waiting in the buffer
string str = comPort.ReadExisting();
//display the data to the user
```

```
DisplayData(MessageType.Incoming, str + "\n");
break;
}
}
#endregion
}
}
```