Design and Development of a Wearable Health Monitoring System for Elderly People

Department of Electrical and Electronic Engineering

A Thesis
Submitted to the Department of Electrical and Electronic Engineering
of
BRAC UNIVERSITY
by

Shakkharatul Haque Mehad	**| 12321044**
Showmik Islam	**| 13221024**
Mashfiqur Rahman	**| 12121032**
Naziur Rahman Khan	**| 12121024**

Supervised by

Dr. Md. Khalilur Rhaman

Associate Professor

Department of Computer Science and Engineering

BRAC University, Dhaka

Inspiring Excellence

Declaration

We do here by declare that the thesis titled 'Design and Development of a Wearable Health Monitoring System for Elderly People' is submitted to the Department of Electrical and Electronic Engineering of BRAC University in partial fulfillment of the Bachelor of Science in Electrical and Electronics Engineering. This is our original work and was not submitted elsewhere for the award of any other degree or any other publication.

Date: 12.12.2016

Signature of Supervisor

____

Dr. Md. Khalilur Rhaman

Associate Professor

Department of Computer Science and Engineering

BRAC University, Dhaka

Signature of Authors

____

Shakkharatul Haque Mehad | 12321044

____

Showmik Islam | 13221024

____

Mashfiqur Rahman | 12121032

____

Naziur Rahman Khan | 12121024

Acknowledgement

We would like to take this opportunity to express our gratitude to Dr. Md. Khalilur Rhaman for his guidance, and instruction that he has given us from time to time throughout our thesis project 'Design and Development of a Wearable Health Monitoring System for Elderly People' under his supervision for the partial completion of our undergraduate program. The skills that we were able to learn from him during our thesis work have benefited us immensely and will continue to do so throughout our future endeavors.

We also thank all our peers and every other individual involved with us for being supportive.

Abstract

Bangladesh is presently experiencing a demographic move and the extent of the populace of the age 60 years and more is drastically increasing. The elderly population of Bangladesh is currently 7.7 percent of the whole population. However, the family size in Bangladesh is decreasing, therefore less people are available at home to take care of their elderly. The number of old homes in the country is in increase. Old home often lacks constant monitoring of the elderlies. Senior citizen who are at risk of cardiovascular diseases and health conditions need to be in constant monitoring. Hence we develop a wearable system for the elderlies which will evaluate their health condition by constantly monitoring heart rate, body temperature and movement activity using triaxial acceleration, and will send these data to a web application. By the help of this system family members can constantly monitor their elderly's health at home. Caregivers at old home can also monitor the conditions of all the elderlies present at old home in real time. The system can also respond to medical situation of the elderly for heart attack or stroke at the moment of occurrence by quickly notifying the caregivers and doctors through a web application.

Table of Contents

List of Abbreviations

HRM Heart Rate Monitor

HR Heart Rate

Bpm Beat per Minute

MCU Microcontroller

MAXHR maximum heart rate

SMS Short Message Service

ADC Analog to Digital Converter

LED Light Emitting Diode

IR Infra-Red

PCB Printed Circuit Board

List of Figures

Chapter - 1
Introduction

1.1 Overview

Throughout our culture and time, people of Bangladesh have always valued family and tradition. In the past it was very common for an entire generation of the family to live together under one roof. This includes aunts, uncles, cousins and most importantly grandparents. Grandparents play a vital role in the family and in the lives of grandchildren. However, the family size in Bangladesh is decreasing. According to ArcGIS, the average number of member per family has decreased to 4.5[9]. Less families live together under one roof resulting there will be less number of people at house to take care of the elderly at home.

Currently the population of Bangladesh over the age of 60 is 7.7 percent of the whole population, according to a long questionnaire survey, census 2011[10]. Because of poverty, physical hard working and, inability and illness, people in this country becomes older before the age of 60[1]. With the decrease in member per family, constant monitoring of elderly is affected. For not being able to give time and proper care, some family will opt for old homes and nursing facilities. However, these places are quite depressing because of the lack of independence. [2]

Introduction to health monitoring system is nothing new. Many researchers have been made in the past. In Takaoka City, Toyama Pref. Japan, researches have been made on health monitoring systems by making experimental rooms where data acquisition system was used. [4] The system consisted of monitoring devices and computer terminals for collecting data.
According report published by Counterpoint Technology Market Research, in Bangladesh, the number of smartphone users in 2015 was 8.2 million [5]. Hence to develop a health monitoring device with an app will accessible and affordable by many in the country.

Elderly with chronic diseases need constant monitoring of their vital signs. For noninvasive diagnosis, research has been done using ECG signal at medical organizations. [6] However these places are already equipped with monitoring devices unlike homes or nursing facilities.

There are many commercial installations for lifestyle monitoring, however the evidence base is relatively weak. [7] Most researches dealt with the technological development and needs to deal with the practical development and user's comfort ability.

The development of such device is motivated because of the increased health care cost and the large rise in aging population [3]. Although other countries are pursuing in this cause, such health monitoring system is not implemented in Bangladesh yet. Implementation of such system will be beneficial towards healthcare industry and the respected elderly of the country.

1.2 Objective

Our objective is to device a wearable system for the elderlies to be worn constantly which will feed three different data: body temperature, heart rate and detect fall; and will send these data to a web server which will translate the information on a web application and feed it to the care givers and guardians to monitor their close ones remotely.

The system will also introduce notification system for emergency medical conditions which will alert the responders immediately in the time of medical emergency.

1.3 Motivation

The aim of this research is to explore recent developments in the field of wearable sensors and systems that are relevant to the field of rehabilitation and medication. The growing body of work focused on the application of wearable technology to monitor older adults and subjects with chronic conditions in the home and community settings justifies the emphasis of this research on summarizing clinical applications of wearable technology currently undergoing assessment of describing the development of wearable sensors and systems.

1.4 Literature Review

1.4.1 Heart Rate

The organ responsible for pumping blood throughout human body is the heart, located in the middle of the thorax, slightly offset to the left and surrounded by the lungs. The human heart is composed of four chambers which are two atriums and two ventricles. The right atrium receives blood returning to the heart from the whole body. That blood passes through the right ventricle and is pumped to the lungs where it is oxygenated and goes back to the heart through the left atrium, and then the blood passes through the left ventricle and is pumped again to be distributed to the entire body through the arteries [15].

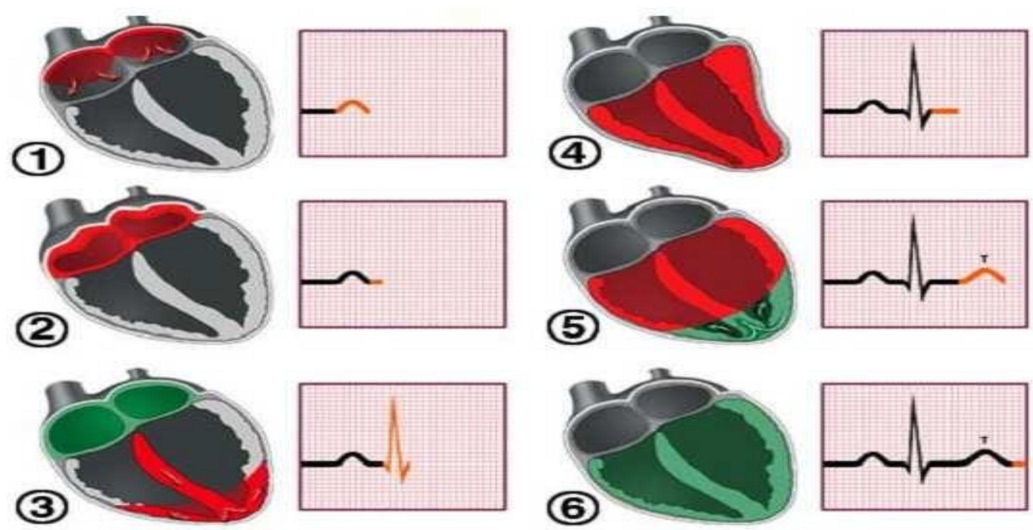

Figure 1.1 heart behavior and part of the generated signal [15]

1.4.1.1 Visual Representation of Electrocardiogram (ECG) signal

An electrocardiogram (ECG), also called an EKG, is a graphic tracing of the voltage generated by the cardiac or heart muscle during a heartbeat. It provides very accurate evaluation of the performance of the heart [14]. The heart generates an electrochemical impulse that spreads out in the heart in such a fashion as to cause the cells to contract and relax in a timely order and, thus, give the heart a pumping characteristic. An actual voltage potential of approximately 1 mV develops between various body points [14].

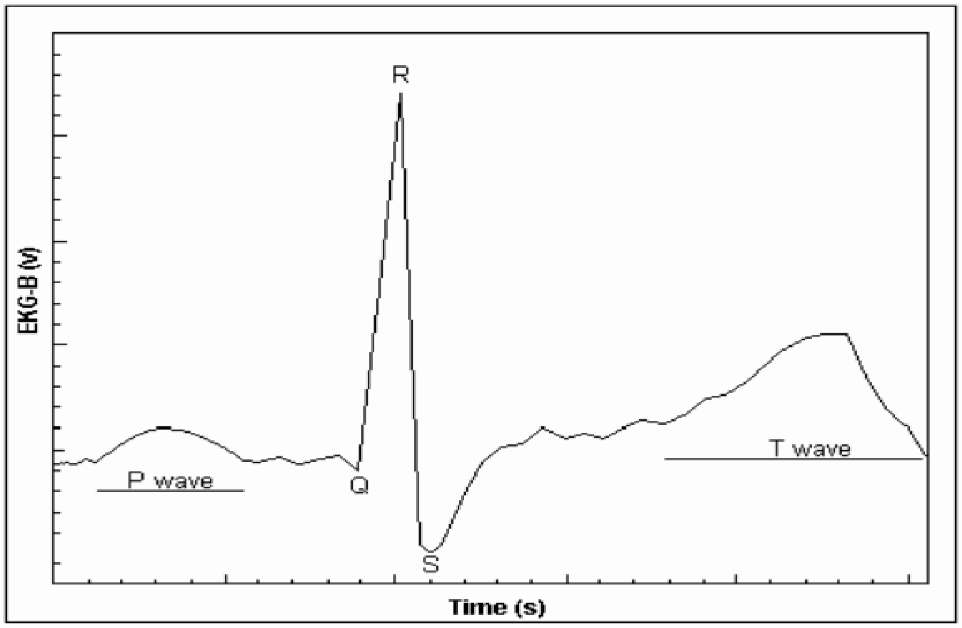

Figure 1.2 The ECG signal waveform

Heart rate is measured in beats per minute (bpm). In measuring heart rate, there are various ways to measure such as by using pulse oximeter, heart rate monitor, an electrocardiograph, and ECG strap. The beats per minute is differ for many people which depending on the ages, body physical condition and environmental factor. Center in the brain is controlled the rate of heart beat [15]. According to information received from muscles and sensors located, this center speeds up or slows down the heart.

1.4.1.2 Measuring the Heart Rate

By detecting the R peaks (shown in figure 1.2) and measuring their frequency, the heart rate can be calculated and then displayed. A person's heart rate before, during and after exercise is the main indicator of their fitness. Measuring this manually requires a person to stop the activity they are

doing in order to count the number of heart beats over a period of time. Measuring the heart rate using an electrical circuit can be done much quicker and more accurately.
Heart rate measurement is one of the very important parameters of the human cardiovascular system. The heart rate of a healthy adult at rest is around 72 beats per minute (bpm). Athletes normally have lower heart rates than less active people. Babies have a much higher heart rate at around 120 bpm, while older children have heart rates at around 90 bpm. The heart rate rises gradually during exercises and returns slowly to the rest value after exercise. The rate when the pulse returns to normal is an indication of the fitness of the person. Lower than normal heart rates are usually an indication of a condition known as bradycardia, while higher than normal heart rates are known as tachycardia [16].

Endure athletes often have very low resting heart rates. Heart rate can be measured by measuring one's pulse. Pulse measurement can be achieved by using specialized medical devices, or by merely pressing one's fingers against an artery (typically on the wrist or the neck). It is generally accepted that listening to heartbeats using a stethoscope, a process known as auscultation, is a more accurate method to measure the heart rate. There are many other methods to measure heart rates like Phonocardiogram1 (PCG), ECG, blood pressure wave form and pulse meters but these methods are clinical and expensive [16].

1.4.2 Maximum Heart Rate

The maximum Heart Rate (Max HR) is the fastest of heart can beat for one minute. A generalized rule anchors Max HR using a mathematical formula. Inside each zone, there are different exercise changes which occur as the result of spending training time in the zone. Heart zones is expressed as a percentage of the maximum heart rate, reflect exercise intensity and the result benefit.

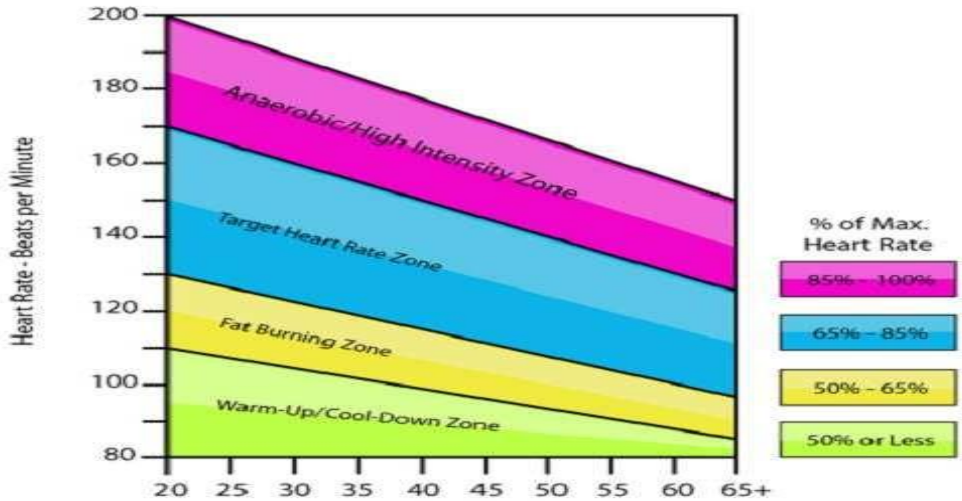

Figure 1.3 Exercise target zone chart

1.4.3 Fingertip sensor

Use of light to measure heart rate is a field of study where abundant research has been done in the past few decades. Fingertip sensor relies on measurement of a physiological signal called Photoplethysmogrpahy (PPG) [17], which is an optical measurement of the change in blood volume in the arteries. Fingertip sensor acquires PPG signals by irradiating wavelength of light through the tissue, and compares the light absorption characteristics of blood under these wavelengths.

1.4.3.1 Transmittance vs. Reflectance

Fingertip sensor has traditionally been done in two methods: transmittance and reflectance of light. In transmittance fingertip sensor, light is shone through the tissue using an LED and is detected on the other end using a photodetector. In contrast, reflectance fingertip sensor uses a photodetector on the same side as the LED to detect the light reflected by the tissue (Figure 1.4).

2 Interbeat interval is a scientific term used in reference to the time interval between individual beats of the mammalian heart. Interbeat interval is abbreviated "IBI". It is also sometimes referred

to as "beat-to-beat" interval. IBI is generally measured in units of milliseconds. Individual human heart IBI values can vary from as short as 5 milliseconds to as long as 70 milliseconds.

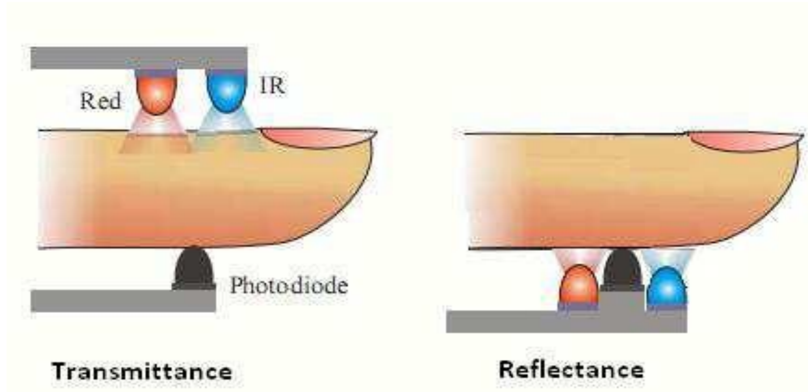

Figure 1.4 Transmittance and Reflectance configurations of transducer [16]

The opposite affect is on the reflected light. This can be intuitively justified, as the more blood there is in the tissue, the more the light passing through the tissue gets blocked. Since this improves the amount of light reflecting back, the signal observed in the reflectance configuration increases. Similarly, as the light gets blocked, not enough light reaches the photodetector in the transmittance configuration, and therefore a decline in the signal is observed [18].

In terms of the application, the transmittance configuration is more suited to the areas of the body that lend themselves better to light transmittance through them, e.g. fingers or ear lobes. However, transmittance configuration cannot be used in other areas of the body as the transmittance of light is significantly less when there are obstacles such as bone or muscle in the way, besides the fact that the path of light is much longer than in thin areas such as the ear lobes. In such scenarios, reflectance configuration is more useful, provided that vasculature is available close to the surface of the skin, e.g. forehead, wrist or forearm.

Reflectance configuration is not limited to areas where the transmittance configuration cannot be used. It can be employed to measure PPG signal from the ear lobes or the fingers just as the transmittance configuration. However, due to their thin cross-sectional area, fingers and ear lobes transmit much of the light shone through them, resulting in lower signal intensity in the reflectance configuration [18].

1.4.4 Embedded Systems

The basic idea of an embedded system is a simple one. If we take any engineering product that needs control, and if a microcontroller is incorporated within that product to undertake the control, then we have an embedded system. An embedded system can be defined as: A system whose principal function is not computational, but which is controlled by a computer embedded within it. Embedded processors can be broken into two broad categories: ordinary microprocessors and microcontrollers.

1.4.5 Microcontrollers

A microcontroller [19] (also microcontroller unit, MCU) is a small computer on a single integrated circuit; its function is determined by a program loaded in it. Like all computers microcontrollers are equipped with a central processing unit or CPU, a memory system, an input/output system, a clock or timing system, and a bus system to interconnect constituent systems. The bus system consists of an address bus, a data bus, and a control bus. In Figure (2.1) we have provided the block diagram of a generic microcontroller. We would like to emphasize that all systems shown in the diagram are contained within the confines of a single integrated circuit package.

Chapter - 2
System Overview

On Spring of 2016, we started planning to do our project on healthcare systems. At first we researched on medical healthcare devices that were connected to the internet which allowed to visualize patient data on an internet host. This included internet connected services like weight machine, blood pressure measurement, geo-location tracking, movement activity, heart rate detection. Our research also scoped on the latest trends of messenger bots, which worked as a conversational application giving users a more human feeling. From here we narrowed our scope to vital recognition to monitor patient's heart rate, temperature and fall detection.

Our very first step to work on the project was to work our way with microcontrollers. We chose the Arduino family, a versatile open-source electronic prototyping platform that offered wide ranges of variation models. Our prime objective was to keep it small and light. The two options we found was to work with Arduino pro mini and the Arduino nano. Both of them were almost identical in features however pro mini was more compact because it came solderless in packaging, however, because pro mini lacked the USB serial interface we chose to work with Arduino nano version.

To collect readings from the patient we experimented with three different types of sensors. First, we worked with MLX90614 Infrared Temperature sensor to collect body temperature data. Second, we worked with SEN-11574 Pulse Sensor to measure heart rate data. And lastly, we worked with MPU6050 motion processing module to detect sudden fall by the patient.

As this project is part of 'Internet of Things' activity we need to connect it to the internet. The available GSM and internet shields available in the market was much bulkier than our project could handle, hence we needed a new improvisation. We introduced ESP8266 version 1 to our project. It is a wifi-module of self-contained system-on-a-chip with integrated TCP/IP protocol stack that can give any microcontroller access to a WiFi network. We then combined the ESP8266 with Arduino nano to send to a website successfully.

By combining the sensors to the Arduino and the Arduino to ESP8266, wifi module we were able to send the data to a local website implemented with XAMP. The data were send to a php script with GET method which were then translated onto the webpage as values, graphs and status conditions.

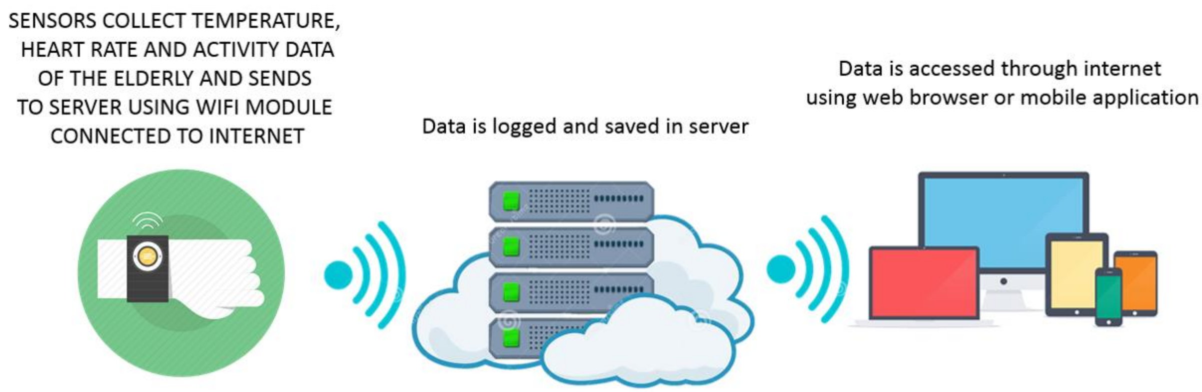

Figure 2.1 Mechanism of the overall System

After every one of these experiments and research we made about the entire framework, we divided our goal into distinctive parts and started working on it. The general outline and procedure of the total system is clarified in the next sections.

2.1 GENERAL OVERVIEW OF THE SYSTEM

The entire framework has the wearable device consisting of the three sensors and connectivity module to connect the system to the internet. The temperature information, heart rate condition data and fall detection alarm data is send to the web database by the wearable system. The transmission to the internet is one way. The device only records the data and sends them to the internet.

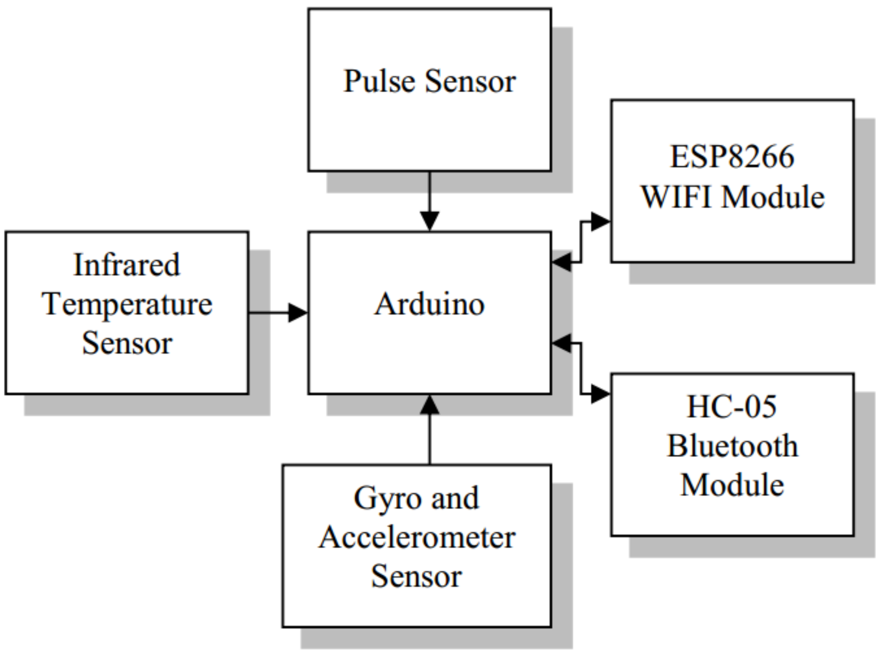

Figure 2.2 Block Diagram of the Wearable Hardware Device

The overall System can be divided into three parts: Sensor Module, Communication Module, Web Application.

The sensor module is a combination of three sensors: SEN-11574 Pulse Sensor, MLX90614 Infrared Temperature Sensor, InvenSense MPU-6050 sensor.

The SEN-11574 pulse sensor is connected to the elderly's thumb from where it records the pulse by using infrared technology. The pulse sensor sends the value as an analog value to Arduino where it determines the pulse rate, BPM, and IBI. [11]

The MLX90614 infrared temperature sensor is designed for non-contact temperature sensing. It detects the temperature of the body with precision up to 0.5c on human wrist.

InvenSense MPU-6050 is a gyro and accelerometer sensor. The MPU-60X0 is an integrated 6-axis Motion Tracking device that combines a 3-axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor™ [13]. This sensor any kinds of movement from the point of view of wrist. So when the user walks and moves their hand, it generates signals, which are then recorded and passed to Arduino. This module is connected to Arduino in i2c communication protocol.

The communication module includes the ESP8266 WIFI module and the HC-05 Bluetooth module. These are used to send the recorded data to the server and connect to the app to visualize data on mobile

The sensor module and the communication module are connected to the microcontroller module. In this project, we are using Arduino nano as the microcontroller. Arduino dictates all the logical operation to monitor data and transfer it to the server.

2.3 Web Application

The purpose of the web application is to visualize all the data of the elderly in graphical and statistical manner. From the application, the elderly, care giver, family members will be able to check the elderly's current personal health condition. The app will display the heart rate, movement activity, IBI, BPM and temperature of the elderly. If the pulse rate and temperature is in the proper range of the elderly's age the app will status the condition as good and healthy. However, if the conditions of the elderly reaches danger level it will status the caregivers and relatives as critical. This will bring the elderly to proper medical attention.

For the web based management, an HTTP Apache server with PHP script intercepts all HTTP request through a defined address. The Arduino sends all the data through a POST request which saves the data according to the device id to the MySQL server. A frontend website is developed using jQuery and PHP that provides a clean Graphical User Interface to the end user. The website consists of a graph box on the left that shows live feed of the heart rate. On the right is shows the BPM and IBI. We can find the value of the temperature down below. The data is fetched from the MySQL server to display on the website. The web app requires password log in. Interface of the system.

CHAPTER - 3
System Implementation

We have configured our proposed health monitoring system as a wrist band, which is to be worn in hand. The band will consist a combination of sensors, arduino, communication modules and battery. The device will send the data to its server. Users can view the demographics through a web based app on their mobile phone. Each device will be coded with a device ID. The server will store data in order by device ID. Users have sign in with the device ID to see their demographics.

3.1 Hardware

3.1.1 Sensor Module:

As mentioned earlier, we have worked with three sensors to collect data form the patient which we shall discuss here briefly.

3.1.1.1 Temperature Measurement

To measure the temperature of the patient we compared two different temperature sensors for the most accurate reading. First we used LM35 temperature sensor. The LM35 series are precision integrated-circuit temperature sensors that requires for the medium to be in contact with the package of the sensor to accurately measure the temperature of the medium. The time required by the sensor to provide stable and correct output varies with medium. One has to wait for more than a minute to have a stable output from LM35 for the human body temp measurement. However, there are drawbacks on using analog sensors for remote sensing for that application, unless cabling is properly shielded and twisted. The LM35 provides 10mV/C degrees, and considering that the resolution for human temperature is normally assessed in the range of 0.1C degrees, this means that the noise expected should not exceed 1mV.

Figure 2.3 MLX90614 Infrared Temperature Sensor

This is why we then opted for MLX90614 Infrared Temperature Sensor which is used for non-contact temperature measurements. The digital output can be configured to be pulse width modulation (PWM). As a standard, the 10-bit PWM is configured to continuously transmit the measured temperature in range of -20 to 120°C, with an output resolution of 0.14°C.

3.1.1.2 Heart Rate Measurement

To measure the heart rate, we worked our way with SEN-11574 Pulse Sensor. It is a plug and play sensor for the Arduino series that can be used to collect heart rate data. It essentially combines a simple optical heart rate sensor with amplification and noise cancellation circuitry making it fast and easy to get reliable pulse readings. It sips power with just 4mA current draw at 5V hence it is perfect for this project.

Figure 3.2 SEN-11574 Pulse Sensor

3.1.1.3 Fall Detection Alarm

For the detection of motion and free falling we experimented with InvenSense MPU-6050 sensor. The InvenSense MPU-6050 sensor contains a MEMS accelerometer and a MEMS gyro in a single chip. It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefor it captures the x, y, and z channel at the same time. The sensor uses the I2C-bus to interface with the Arduino.

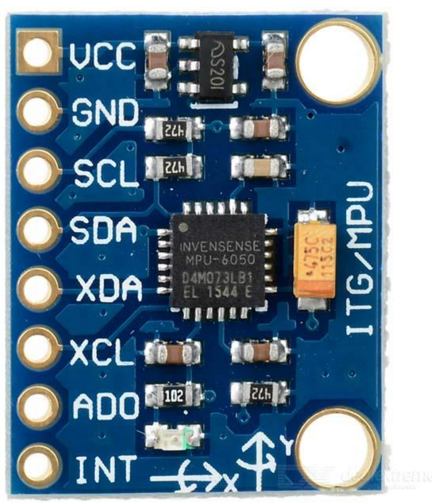

Figure 3.3 InvenSense MPU-6050 sensor

3.2. Communication Module

We have implemented the communication of the wearable device to the cloud server through ESP8266 WIFI module. ESP8266 is a self-contained SOC with integrated TCP/IP protocol which gives it the power to embed within other systems using Wifi capabilities as well as can function standalone application [8].

After we connect the proposed system to the internet, it will send the data of the reading taken from the elderlies to its personal server at regular intervals. The device shall store and update the data in cloud at regular intervals from which the app will access the data from. In case of emergency medical condition, the device will send the data to its server instantaneously to alarm the elderly's caregiver and doctor to take immediate medical action.

The ESP8266 has an operating voltage of 3.3v, however the Arduino has an operating voltage of 5v. Hence we use a voltage divider circuit in between Arduino’s TX pin to ESP8266’s RX pin to shift the voltage level from 5v to 3.3v.

The proposed system also includes HC-05 Bluetooth module. HC-05 module is a simple to utilize Bluetooth SPP (Serial Port Protocol) module, intended for straightforward remote serial connection setup. [12]. The Bluetooth module is connected in software serial with Arduino. With this feature, user can connect their smart phone to the wearable system to connect with the app.

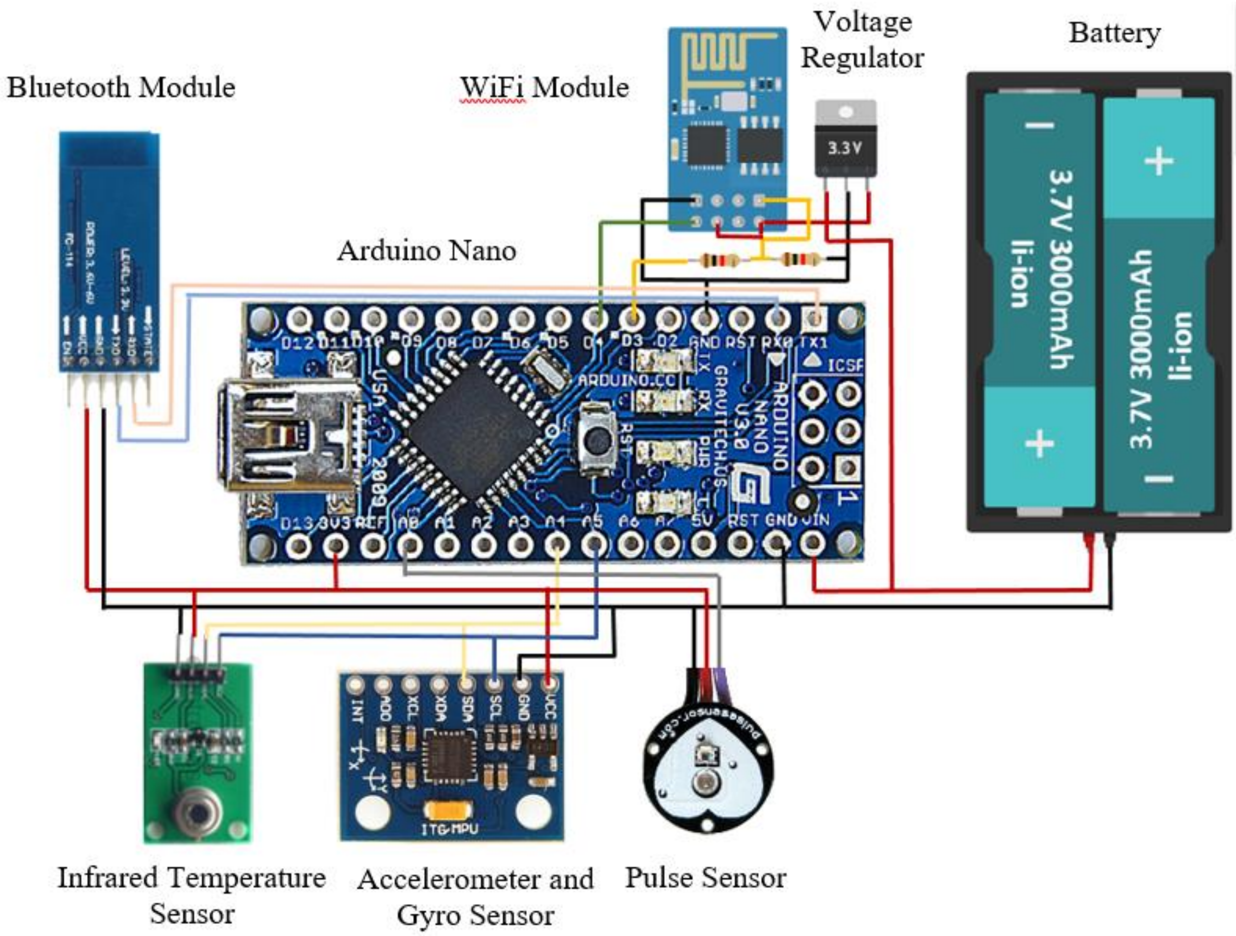

Figure 3.4 Circuit Diagram of the Wearable System

3.2 Control

The control system relies upon the mechanism of the sensors reading the value accurately. One of the main critical reasoning was to determine if the sensor was getting any value. The temperature module accurate the decision of precision as the sensors have the most vital role here and their accuracy determines the proper execution of the system. The control process is shown in fig 3.5

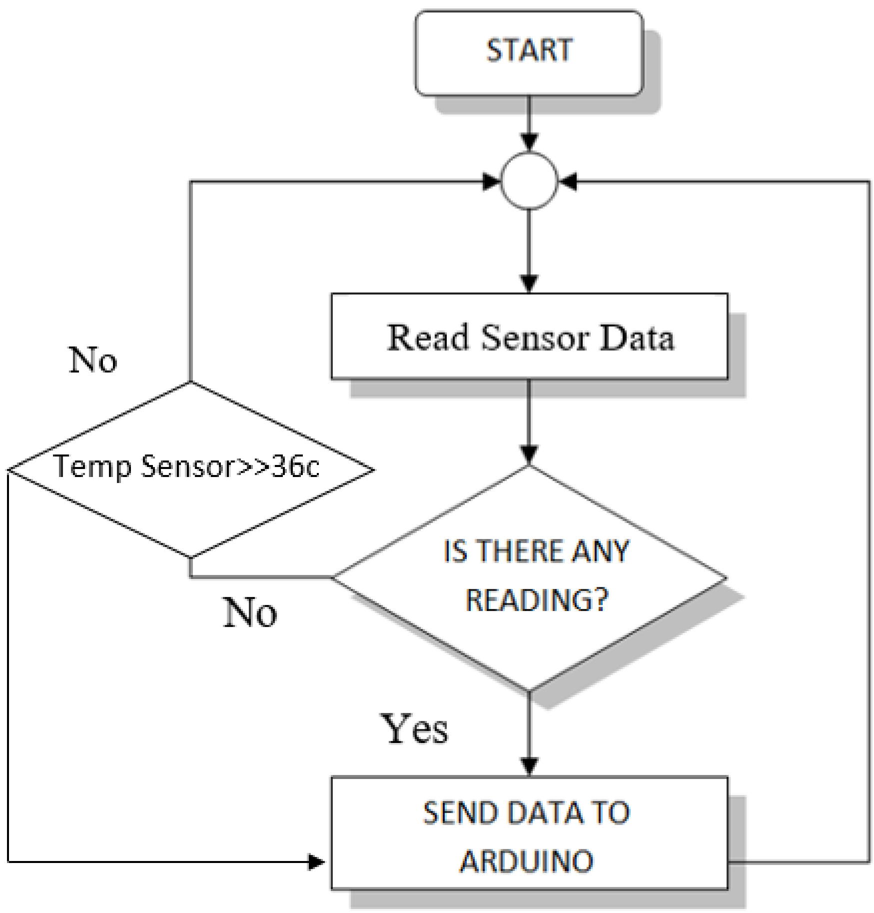

Figure 3.5 Control Logic of the sensors

To provide the status of elderly's health condition, we measure the heart rate, BPM, temperature and IBI and provide the status in web app. We also determine if the elderly has fallen down or not by measuring the vertical acceleration. If it reaches a critical level instead of a steady level the device will detect the elderly has fallen. the control process of the health band is shown in fig 3.6

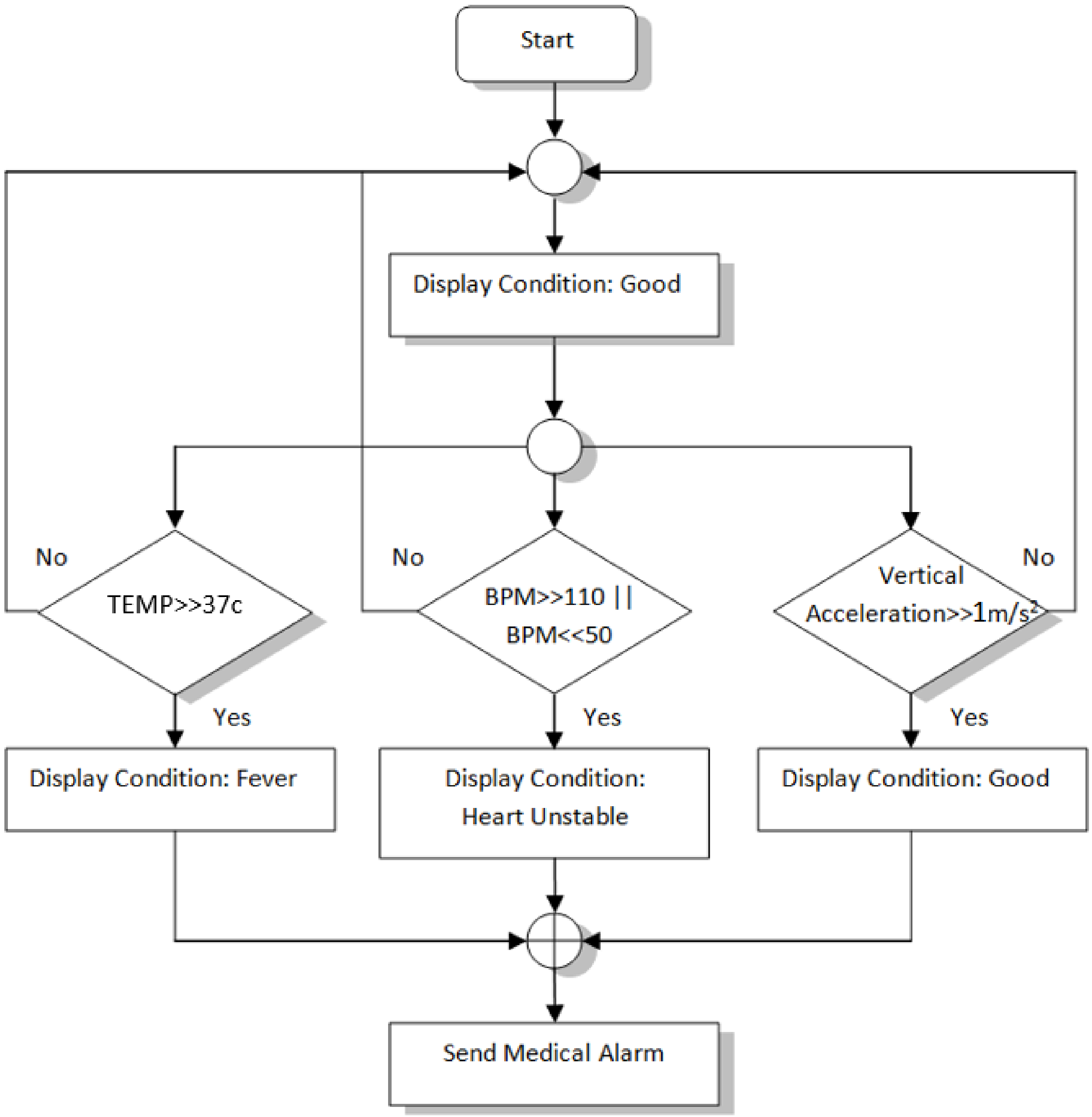

Figure 3.6 Control Logic of the Wearable System

3.3 Communication

For connecting the device to internet we used ESP8266. The ESP8266 is a low-cost Wi-Fi chip with full TCP/IP stack and MCU (Micro Controller Unit) capability produced by Shanghai-based Chinese manufacturer, Espressif Systems.

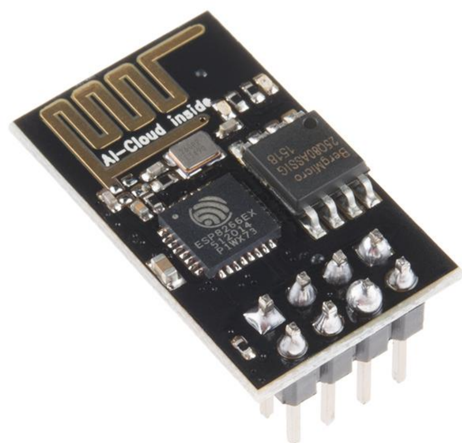

Figure 3.7 ESP8266 Wifi Module

To send data to the internet AT commands are used. First we use the command to connect to the internet network. Then we use the AT commands to send data to the internet webpage.

`AT+CWJAP` - Connect to AP

Variant	Command	Response	Function
Query	AT+CWJAP?	+ CWJAP: `ssid` OK	Prints the SSID of Access Point ESP8266 is connected to.
Execute	AT+CWJAP= `ssid` , `pwd`	OK	Commands ESP8266 to connect a SSID with supplied password.

Parameters:

- `ssid` : String, AP's SSID
- `pwd` : String, not longer than 64 characters

Figure 3.7 ESP8266 AT Commands to Connect to AP

`AT+CIPSEND` - Send data

Variant	Command	Response	Function
Test	AT+CIPSEND=?	OK	
Set	AT+CIPSEND= `length`	SEND OK	Set length of the data that will be sent. For normal send (single connection).
Set	AT+CIPSEND= `id` , `length`	SEND OK	Set length of the data that will be sent. For normal send (multiple connection).
Execute	AT+CIPSEND		Send data. For unvarnished transmission mode.

Normal Mode

Parameters:

- `id` : ID no. of transmit connection
- `length` : data length, MAX 2048 bytes

Figure 3.8 ESP8266 AT Commands to Send Data

3.4 Power

To power the whole circuit, we use two low discharge rate 3000mAh li-ion battery with 3.7V in series. The battery is connected to the Arduino. Arduino has an operating voltage of 5v and has recommended input voltage of 7-12v. The sensors are powered from the arduino's 5v and 3.3v terminal. The pulse sensor, MPU6050, temp sensor and HC-05 is connected to 3.3v. ESP8266 WIFI module also has an input voltage threshold of 3.3v. However, the Arduino cannot supply the total power needed by the module. Hence we use 3.3v regulator circuit connected to the battery to regulate the power and supply the required power to the module.

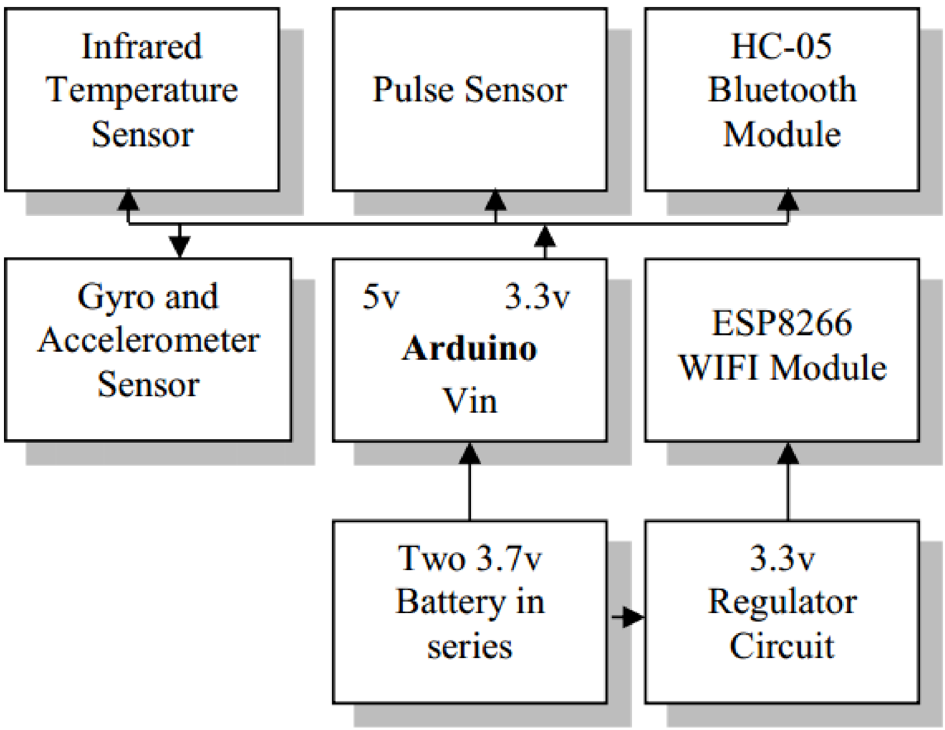

Figure 3.8 Power Supply Diagram of the Wearable System

3.5 Application

The main purpose of the web application is to visualize all the data of the elderly in graphical and statistical manner. From the application, the elderly, care giver, family members will be able to check the elderly's current personal health condition. The app will display the heart rate, movement activity, IBI, BPM and temperature of the elderly. If the pulse rate and temperature is in the proper range of the elderly's age the app will status the condition as good and healthy. However, if the conditions of the elderly reaches danger level it will status the caregivers and relatives as critical. This will bring the elderly to proper medical attention.

For the web based management, an HTTP Apache server with PHP script intercepts all HTTP request through a defined address. The Arduino sends all the data through a POST request which saves the data according to the device id to the MySQL server. A frontend website is developed using jQuery and PHP that provides a clean Graphical User Interface to the end user. The website consists of a graph box on the left that shows live feed fof the heart rate. On the right is shows the BPM and IBI. We can find the value of the temperature down below. The data is fetched from the MySQL server to display on the website. The web app requires password log in.

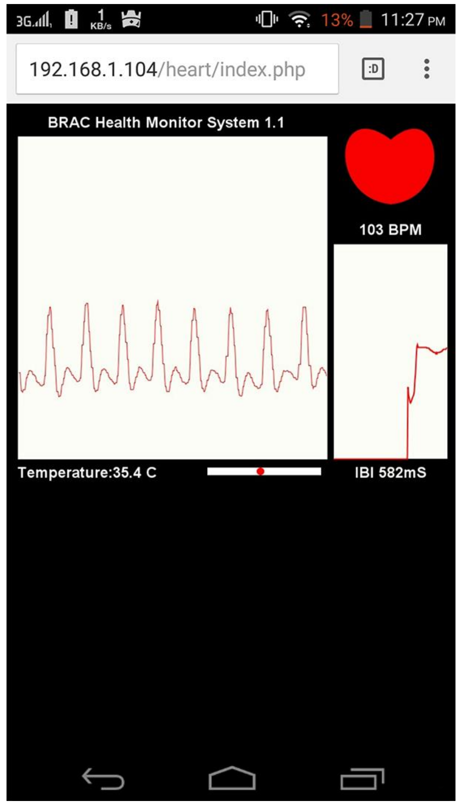

Figure 3.8 Web Application of the Wearable System

3.6 Twilio Messaging API

To send message for emergency notification we need to work with a messaging API. The system detects the emergency and then sends the request to the website called Twilio. Twillio is a cloud communications platform as a service (PaaS) company based in San Francisco, California. Twilio allows the device to programmatically make and receive phone calls and send and receive text messages using its web service APIs. Twilio's services are accessed over HTTP and are billed based on usage. For the purpose of our thesis we used a free account which can send message to only a dedicated number.

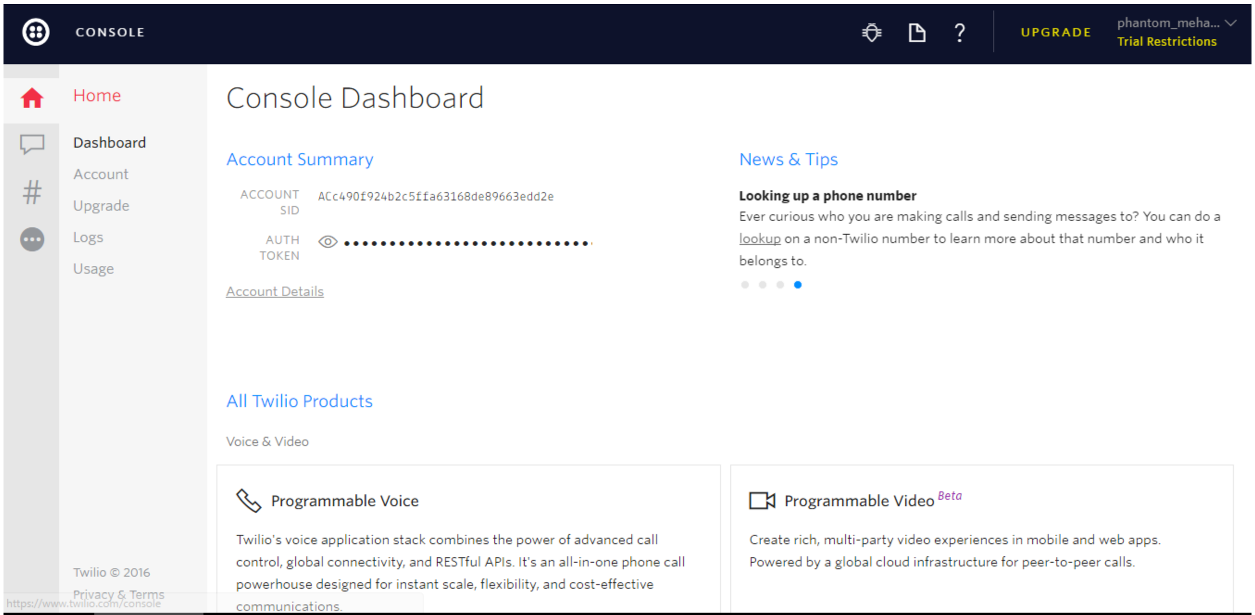

Figure 3.8 Twilio Messaging API Dashboard to send message

Chapter - 4
Data Analysis

4.1 Temperature Detection

To measure the temperature of the patient we used the temperature sensor and recorded the body temperature at different conditions at different temperature. The result is almost as the graph registered below provided by Texas Instruments. [20]

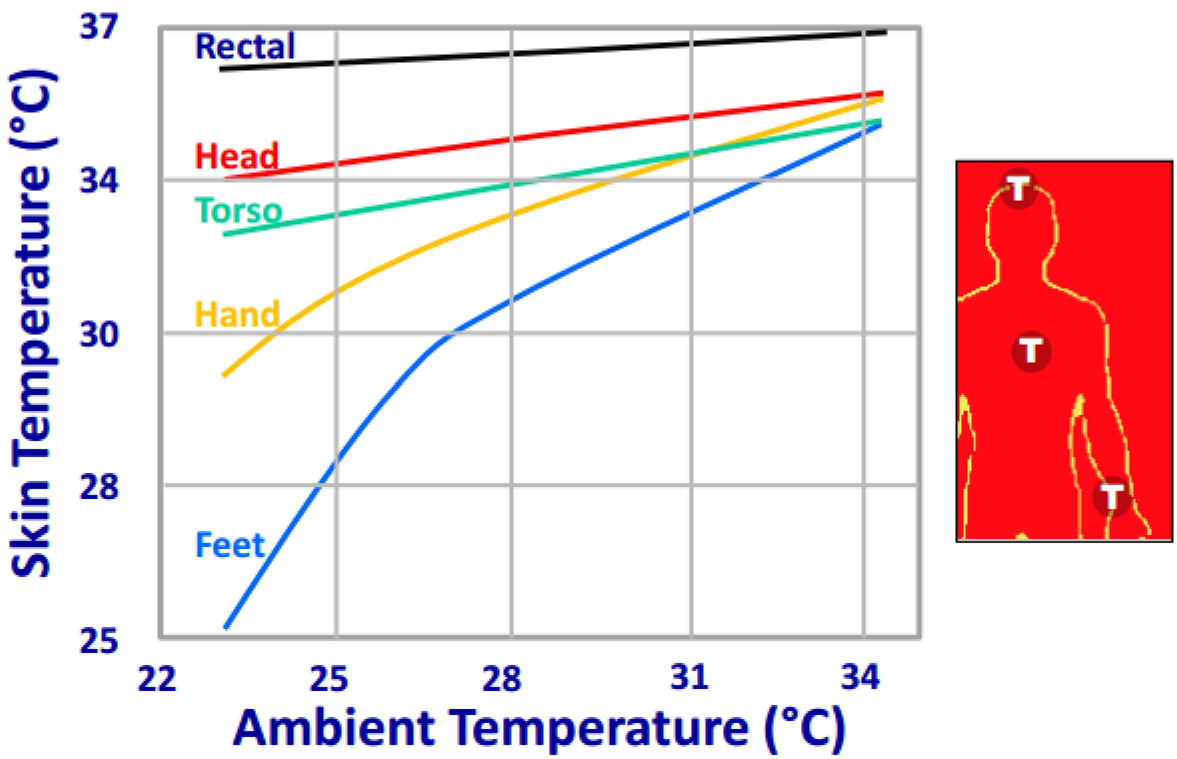

Figure 4.1 Correlation of Body Temperature with ambient Temperature at different Heat Points

4.2 Pulse sensor

The output signal is very important for the testing, the signal is correct, the testing can go on to next step. For this reason, checking the signal before testing is done at the first step. The sensor is supplied by 3.3V power. As considering the stable of infrared sensor, finger needs to place very close to sensor.

4.2.1 User heart beat pulse testing

After the assembly of the system test were done to find the accuracy of the device. For this experiment, ten elderlies were chosen randomly and their heart rate was measured at certain instance. The results are tabulated below.

Gender	HR on display	HR on oscilloscope	Error rate (%)
Male	67	96	1.03
Male	83	81	2.41
Male	78	78	0
Male	80	77	3.33
Male	70	69	1.25
Female	77	77	0
Female	104	103	0.96
Female	75	75	0
Female	69	71	2.81
Female	83	85	2.35

Figure 4.2 Experiment of Pulse Detection

4.3 Fall Detection Analysis

The system depends on the free fall acceleration towards the ground. Since we cannot decide how fast or slow fall takes place, we experimented by dropping the device from different heights to determine its accuracy. The experiment recorded the approximate time to fall at recorded heights. The results of the experiment are tabulated in Fiugre 4.3. An experiment was also conducted to realize the confirmation of fall detection which is tabulated in Figure 4.4.

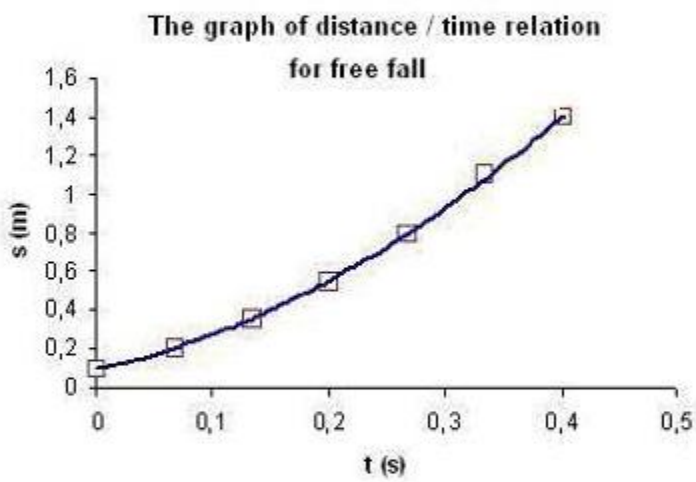

Figure 4.3 Experiment of Accuracy of Fall Detection System

Height(feet)	**Response**
1	Negative
1.5	Negative
2	Negative
2.5	Negative
3	Positive
3.5	Positive
4	Positive
4.5	Positive
5	Positive
5.5	Positive

Figure 4.4 Experiment of Accuracy of Fall Detection System

The result shows the device will register a fall if the device worn is above minimum 3 feet of the ground.

4.4 SMS TEST

Other test performed to ensure that the warning procedures will be applied, when the heartrate exceed the maximum allowable level. This is done by entering incorrect information of lower threshold that anyone can exceed. The testing was successful and the SMS was sent to the phone number which was specified previously. Below is a table of experiment for how long it took the sms to reach. A screenshot of the received message in shown in Figure 4.4.

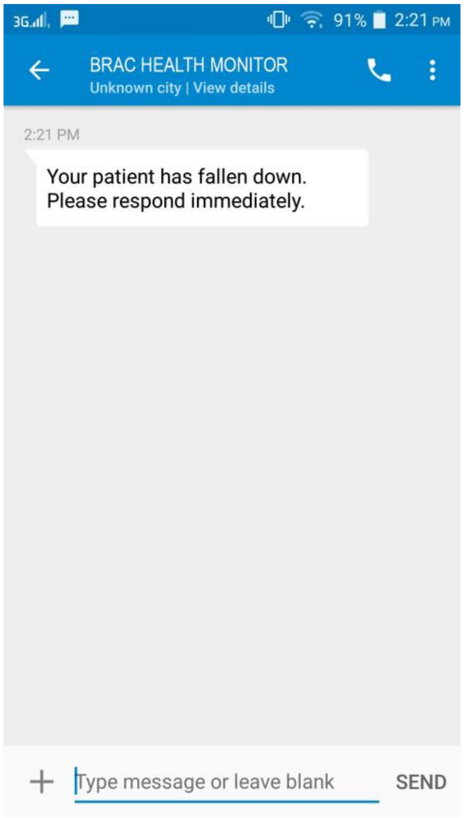

Figure 4.4 SMS to phone by Fall Detection System

SMS No.	Time(seconds)
1	14
2	11
3	12
4	8
5	15
6	11
7	13
8	10
9	12
10	13

Figure 4.5 Experiment of Time for SMS Notification

Hence we can observe it takes roughly 15 seconds for the system to notify though messaging.

Chapter - 5
Discussion

This chapter discusses the suggestion of future work for the project and conclusion will be made according to the project development. This thesis has discussed the development of the wearable health monitor system that collects heart rate, body temperature and fall detection of the patient and remotely sending the data visualize over a web application and emergency notification by text message to phone.

6.1. Discussion

Infrared sensor instead of pressure sensor was used in Heart Rate Monitor, it has higher sensitivity, and the output signal is more stability. The infrared sensor depends on the fact that the speed of blood is proportional to heart rate. That is to say, when people are in high active, the heart rate will change. As the same theory, as previous research, heart rate changes when people are excited. When people took exercise, the heart rate will increase; sometimes it is out of healthy range. When people take a break, the normal heart rate will recover.

I

n totally, heart rate is physiological indicators to show what is going on in body. People need to care about their health condition with the help of Heart Rate Monitor.

6.2. Problems

Throughout the project life cycle several problems and difficulties were encountered from several internal and external causes. The following is brief description of most problems and the ways to overcome them.

1) The fingertip sensor and its accuracy is the big problem since its output signal was weak and boxed in by movement artifacts noise of the ring and the 50 HZ AC mains interference. The solution was filtering block these higher frequency noises present in the signal.

2) The component provided in market were not up to medical standard hence sensor modules would need to be upgraded to make it more reliable.

6.3. Accomplishments

All in all, this project achieved a lot of its goals. The project implemented a low cost, low power heart rate monitoring with fall detection and alarm system using microcontroller technology. Lists of accomplishments include:

- Adequately acquiring biological signal
- Adequately amplifying biological signal
- ADC conversion of analog signal
- Semi functional heart rate meter
- Functional notification and alarm system
- Web application status display
- Emergency text notification

6.4. Future work

To ensure the accuracy of heart rate monitor device, more testing can be performed to larger number of people with different ages and weights.

In terms of making the device more portable, the device would be miniaturized onto a printed circuit board making it light weight and more stable.

6.5 Conclusion

We have designed a wearable simple wrist band device that can measure the vital signs of the 41elderly. In long run a lot of modifications and improvement can be adjusted based on requirement. For people with dementia and Alzheimer disease, global positioning system can be added to track the position of the elderly along with a GSM module to keep communication through mobile network. [7] Furthermore more sensors can be added to include more accuracy. Also portable environmental sensors like PIR (passive infrared sensor) sensors to detect in house movement between rooms can be added to the system to provide further assistance to the elderly and caregivers at home or nursing homes.

Chapter - 6
Reference

[1] K. M. Mustafizur Rahman, B. Noble, and I.N. Sneddon, "Unnayan Onneshan Policy Brief on Present Social Context and Elderly Population in Bangladesh"

[2] Ms. S. Selvakanmani, "Android Based Health Monitoring System for Elderly People," in International Journal of Trend in Research and Development, Volume 2(4), ISSN 2394-9333, July-August 2015.

[3] Center of Assistive Research Technologies (CART), " Wearable/Portable Health Monitoring System," Wright State University

[4] Tamura, T., Togawa, T., Ogawa, M., and Yoda, M., Fully automated health monitoring system in the home. Med. Eng. Phys. 20(8):573–579, 1998.

[5] Ishtiaq Husain, "Smartphone users on the rise," Dhaka Tribune, January 2016

[6] Dong-Her, S., Hsiu-Sen, C., Binshan, L., and Shih-Bin, L., An Embedded Mobile ECG Reasoning System for Elderly Patients. IEEE Trans. Inf. Technol. Biomed. 14(3):854–865, 2010.

[7] Roine, R., Ohinmaa, A., and Hailey, D., Assessing telemedicine: a systematic review of the literature. Can. Med. Assoc. J. 165(6):765–71, 2001.

[8] Espressif Systems IOT Team, "Espressif Systems IOT Team Version 4.3," http://bbs.espressif.com, 2015

[9] ArcGIS. "Bangladesh Average Household Size," http://www.arcgis.com/home/item.html?id=692cee7e5a5e47dd86531ab0c6a00cff

[10] Bangladesh Bureau Of Statistics (BBS), "Elderly Population In Bangladesh: Current Features And Future Perspectives," Novermber 2015

[11] Sparkfun, "Pulse Sensor SEN-11574 ," https://www.sparkfun.com/products/11574

[12] "ITead Studio, "HC-05-Bluetooth to Serial Port Module"

[13] InvenSense Inc., "MPU-6000 and MPU-6050 Product Specification Revision 3.4,"

[14] Ken Li CHONG, David HOLDEN, Tim OLIN , "ANALOGUE ELECTRONICS - Heart Rate Monitor", p2.

[15] Grajales, L. and I. Nicolaescu., Wearable multisensor heart rate monitor, 2006: IEEE Journal, pp. 1-4.

[16] M.M.A.Hashem, Rushdi Shams, Md. Abdul Kader, and Md. Abu Sayed, Design and Development of a Heart Rate Measuring Device using Fingertip, International Conference on Computer and Communication Engineerin,ICCCE2101, Kuala Lumpur, Malaysia ,2010.

[17] G. M. Azmal, A. Al-Jumaily and M. Al-Jaafreh, "Continuous Measurement of Oxygen Saturation Level using Photoplethysmography Signal," biomedical and Pharmaceutical Engineering, 2006. ICBPE 2006.

[18] Yousuf Jawahar, Design of an Infrared based Blood Oxygen Saturation and Heart Rate Monitoring Device, McMaster University Hamilton, Ontario, Canada April 10, 2009.

[19] "ATMEGA32 datasheet", 8-bit AVR Microcontroller, 32KB Flash, 40/44-pin, ATMEL production.

[20] Emmy Denton, Temperature Sensor Applications, Texas Instruments, March 17, 2015

Chapter - 7
Appendix

Code Section:

7.1 Arduino Source Code for Heart Rate Monitor:

```
/*

Signal :  int that holds the analog signal data straight from the sensor. updated every 2mS.

IBI :   int that holds the time interval between beats. 2mS resolution.

BPM :   int that holds the heart rate value, derived every beat, from averaging previous 10 IBI values.

QS :    boolean that is made true whenever Pulse is found and BPM is updated. User must reset.

Pulse :   boolean that is true when a heartbeat is sensed then false in time with pin13 LED going out.


Pin 13 LED will blink with heartbeat.


IBI: Inter-Beat Interval

*/


#include "SoftwareSerial.h"

String ssid ="xxx"; //Wifi Network ID

String password="xxxx"; //Wifi Password


SoftwareSerial esp(6, 7);// RX, TX
```

```
String data;

String server = "www.mehad.cf"; // www.example.com
String url = "/brac/heartdata.php";// path

// VARIABLES

int pulsePin = 0;        // Pulse Sensor purple wire connected to analog pin 0
int blinkPin = 13;       // pin to blink led at each beat
int fadePin = 5;         // pin to do fancy classy fading blink at each beat
int fadeRate = 0;        // used to fade LED on with PWM on fadePin

//temperature
int val;
int tempPin = 1;
float TEM;

// these variables are volatile because they are used during the interrupt service routine!
volatile int BPM;          // used to hold the pulse rate
volatile int Signal;       // holds the incoming raw data
volatile int IBI = 600;    // holds the time between beats, the Inter-Beat Interval
volatile boolean Pulse = false;   // true when pulse wave is high, false when it's low
volatile boolean QS = false;    // becomes true when Arduoino finds a beat.
```

```
void setup(){
 pinMode(blinkPin,OUTPUT);     // pin that will blink to your heartbeat!
 pinMode(fadePin,OUTPUT);     // pin that will fade to your heartbeat!
 Serial.begin(115200);      // we agree to talk fast!
 interruptSetup();        // sets up to read Pulse Sensor signal every 2mS
  // UN-COMMENT THE NEXT LINE IF POWERING The Pulse Sensor AT LOW VOLTAGE,
  // AND APPLY THAT VOLTAGE TO THE A-REF PIN
  //analogReference(EXTERNAL);

 esp.begin(9600);
 Serial.begin(9600);
 reset();
 connectWifi();
}

void reset() {
 esp.println("AT+RST");
 delay(1000);
 if(esp.find("OK") ) Serial.println("Module Reset");
}

void connectWifi() {
 String cmd = "AT+CWJAP=\"" +ssid+"\",\"" + password + "\"";
 esp.println(cmd);
```

```
 delay(4000);
 if(esp.find("OK")) {
 Serial.println("Connected!");
}
else {
connectWifi();
Serial.println("Cannot connect to wifi"); }
}

void sendData() {
data = "SIGNAL=" + Signal + "&BPM=" + BPM + "&IBI=" + IBI + "&TEM" + TEM;// data sent must be under
this form //name1=value1&name2=value2.
httppost();
delay(1000);
 Serial.println("Data sent");
}
else {
sendData();
Serial.println("Cannot send data"); }
}

void ledFadeToBeat(){
  fadeRate -= 15;             // set LED fade value
  fadeRate = constrain(fadeRate,0,255);  // keep LED fade value from going into negative numbers!
```

```
 analogWrite(fadePin,fadeRate);     // fade LED
}

void temp(){
val = analogRead(tempPin);
float mv = ( val/1024.0)*5000;
float cel = mv/10;
float farh = (cel*9)/5 + 32;

Serial.print("TEMPRATURE = ");
Serial.print(cel);
Serial.print("*C");
Serial.println();
delay(1000);

/* uncomment this to get temperature in farenhite
Serial.print("TEMPRATURE = ");
Serial.print(farh);
Serial.print("*F");
Serial.println();
*/
}
```

```
void loop(){
 temp();
 if (QS == true){          // Quantified Self flag is true when arduino finds a heartbeat
    sendData(Signal, BPM, IBI, TEM);   // send Sensor data
    QS = false;          // reset the Quantified Self flag for next time
   }


 ledFadeToBeat();


 delay(50);              // take a break
}
```

7.2 Arduino source code For Fall Detection:

```
#include <Adafruit_ESP8266.h>

#include <SoftwareSerial.h>

#include <Wire.h>

#include <MPU6050.h>


MPU6050 mpu;


boolean ledState = false;

boolean freefallDetected = false;

int freefallBlinkCount = 0;


// Connect ESP TX pin to Arduino pin 3

#define ESP_RX  3

// Connect ESP RX pin to Arduino pin 4

#define ESP_TX  4

#define ESP_RST 8

SoftwareSerial softser(ESP_RX, ESP_TX);

// Must declare output stream before Adafruit_ESP8266 constructor; can be

// a SoftwareSerial stream, or Serial/Serial1/etc. for UART.

Adafruit_ESP8266 wifi(&softser, &Serial, ESP_RST);

// Must call begin() on the stream(s) before using Adafruit_ESP8266 object.


#define ESP_SSID "xxx" // Wifi Network name here
```

```
#define ESP_PASS "xxxxxxx" // Wifi Network password here

\

const char PHONE_FROM[] = "xxxxxxxxxxx"; // Twilio's phone number, including country code

const char PHONE_TO[] = "xxxxxxxxxxx"; // Any phone number including country code,

const char TWILIO_ACCOUNT_SID[] = "xxxxxxxxxxxxxxx"; // Twilio's ACCOUNT SID

const char TWILIO_TOKEN[] = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxx"; // Twilios TOKEN

const char TEXT_MESSAGE_BODY[] =

"Your+patient+has+fallen+down.+Please+respond+for+medical+emergency"; // URL encoded text

message.

char data[200]; // HTTP POST data string. Increase size as required.


#define HOST "iot-https-relay.appspot.com"

#define PORT 80


void setup() {

 Serial.begin(57600);

 Serial.println("Initialize MPU6050");

 while(!mpu.begin(MPU6050_SCALE_2000DPS, MPU6050_RANGE_16G))

 {

  Serial.println("Could not find a valid MPU6050 sensor, check wiring!");

  delay(500);

 }

 mpu.setAccelPowerOnDelay(MPU6050_DELAY_3MS);

 mpu.setIntFreeFallEnabled(true);
```

```
 mpu.setIntZeroMotionEnabled(false);
 mpu.setIntMotionEnabled(false);
 mpu.setDHPFMode(MPU6050_DHPF_5HZ);
 mpu.setFreeFallDetectionThreshold(17);
 mpu.setFreeFallDetectionDuration(2);
checkSettings();

pinMode(4, OUTPUT);
digitalWrite(4, LOW);

 attachInterrupt(0, doInt, RISING);
 char buffer[50];
 // by providing a string to ID the tail end of the boot message:
 wifi.setBootMarker(F("Version:0.9.2.4]\r\n\r\nready"));

    strcat(data,"From=");
    strcat(data,PHONE_FROM);
    strcat(data,"&");

    strcat(data,"To=");
    strcat(data,PHONE_TO);
    strcat(data,"&");

    strcat(data,"sid=");
```

```
  strcat(data,TWILIO_ACCOUNT_SID);
  strcat(data,"&");

  strcat(data,"token=");
  strcat(data,TWILIO_TOKEN);
  strcat(data,"&");

  strcat(data,"Body=");
  strcat(data,TEXT_MESSAGE_BODY);

softser.begin(9600); // Soft serial connection to ESP8266
while(!Serial); // UART serial debug

Serial.println(F("Adafruit ESP8266 Phone Call"));

// Test if module is ready
Serial.print(F("Hard reset..."));
if(!wifi.hardReset()) {
 Serial.println(F("no response from module."));
 for(;;);
}
Serial.println(F("OK."));

Serial.print(F("Soft reset..."));
```

```
if(!wifi.softReset()) {
 Serial.println(F("no response from module."));
 for(;;);
}
Serial.println(F("OK."));

Serial.print(F("Checking firmware version..."));
wifi.println(F("AT+GMR"));
if(wifi.readLine(buffer, sizeof(buffer))) {
 Serial.println(buffer);
 wifi.find(); // Discard the 'OK' that follows
} else {
 Serial.println(F("error"));
}

Serial.print(F("Connecting to WiFi..."));
if(wifi.connectToAP(F(ESP_SSID), F(ESP_PASS))) {

 // IP addr check isn't part of library yet, but
 // we can manually request and place in a string.
 Serial.print(F("OK\nChecking IP addr..."));
 wifi.println(F("AT+CIFSR"));
 if(wifi.readLine(buffer, sizeof(buffer))) {
   Serial.println(buffer);
```

```
  wifi.find(); // Discard the 'OK' that follows

  Serial.print(F("Connecting to host..."));

  Serial.print("Connected..");
  wifi.println("AT+CIPMUX=0"); // configure for single connection,
              //we should only be connected to one SMTP server
  wifi.find();
  wifi.closeTCP(); // close any open TCP connections
  wifi.find();

  if(wifi.connectTCP(F(HOST), PORT)) {
  Serial.print(F("OK..."));

  while(1);
  } else { // TCP connect failed
  Serial.println(F("Fail"));
  }

} else { // IP address check failed
 Serial.println(F("error"));
}
```

```
 } else { // WiFi connection failed
  Serial.println(F("FAIL"));
 }
}



void doInt()
{
 freefallBlinkCount = 0;
 freefallDetected = true;
}

void checkSettings()
{
 Serial.println();

 Serial.print(" * Sleep Mode:        ");
 Serial.println(mpu.getSleepEnabled() ? "Enabled" : "Disabled");

 Serial.print(" * Motion Interrupt:    ");
 Serial.println(mpu.getIntMotionEnabled() ? "Enabled" : "Disabled");

 Serial.print(" * Zero Motion Interrupt:    ");
 Serial.println(mpu.getIntZeroMotionEnabled() ? "Enabled" : "Disabled");
```

```
Serial.print(" * Free Fall Interrupt:    ");
Serial.println(mpu.getIntFreeFallEnabled() ? "Enabled" : "Disabled");

Serial.print(" * Free Fal Threshold:     ");
Serial.println(mpu.getFreeFallDetectionThreshold());

Serial.print(" * Free FallDuration:      ");
Serial.println(mpu.getFreeFallDetectionDuration());

Serial.print(" * Clock Source:      ");
switch(mpu.getClockSource())
{
  case MPU6050_CLOCK_KEEP_RESET:   Serial.println("Stops the clock and keeps the timing generator in
reset"); break;
  case MPU6050_CLOCK_EXTERNAL_19MHZ: Serial.println("PLL with external 19.2MHz reference");
break;
  case MPU6050_CLOCK_EXTERNAL_32KHZ: Serial.println("PLL with external 32.768kHz reference");
break;
  case MPU6050_CLOCK_PLL_ZGYRO:   Serial.println("PLL with Z axis gyroscope reference"); break;
  case MPU6050_CLOCK_PLL_YGYRO:   Serial.println("PLL with Y axis gyroscope reference"); break;
  case MPU6050_CLOCK_PLL_XGYRO:   Serial.println("PLL with X axis gyroscope reference"); break;
  case MPU6050_CLOCK_INTERNAL_8MHZ: Serial.println("Internal 8MHz oscillator"); break;
}
```

```
Serial.print(" * Accelerometer:       ");
switch(mpu.getRange())
{
 case MPU6050_RANGE_16G:     Serial.println("+/- 16 g"); break;
 case MPU6050_RANGE_8G:      Serial.println("+/- 8 g"); break;
 case MPU6050_RANGE_4G:      Serial.println("+/- 4 g"); break;
 case MPU6050_RANGE_2G:      Serial.println("+/- 2 g"); break;
}

Serial.print(" * Accelerometer offsets:   ");
Serial.print(mpu.getAccelOffsetX());
Serial.print(" / ");
Serial.print(mpu.getAccelOffsetY());
Serial.print(" / ");
Serial.println(mpu.getAccelOffsetZ());

Serial.print(" * Accelerometer power delay: ");
switch(mpu.getAccelPowerOnDelay())
{
 case MPU6050_DELAY_3MS:     Serial.println("3ms"); break;
 case MPU6050_DELAY_2MS:     Serial.println("2ms"); break;
 case MPU6050_DELAY_1MS:     Serial.println("1ms"); break;
 case MPU6050_NO_DELAY:      Serial.println("0ms"); break;
```

```
 }

 Serial.println();
}

void(* resetFunc) (void) = 0; //declare reset function @ address 0

void loop() {

Vector rawAccel = mpu.readRawAccel();
 Activites act = mpu.readActivites();

 Serial.print(act.isFreeFall);
 Serial.print("\n");

 if (freefallDetected)
 {
  ledState = !ledState;

  digitalWrite(4, ledState);

  freefallBlinkCount++;

  if (freefallBlinkCount == 20)
```

```
 {
  freefallDetected = false;
  ledState = false;
  digitalWrite(4, ledState);
  wifi.httpPost("iot-https-relay.appspot.com","/twilio/Messages.json",data);
  wifi.closeTCP(); // close any open TCP connections
  delay(100);
  resetFunc(); //call reset
 }
}

delay(100);
}
```