Intelligent Assisted Living Systems with Multicore Microcontrollers

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree

of
BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

Submitted by:
Bittu Kumar,
130001008,
Computer Science and Engineering,
Indian Institute of Technology Indore

Supervised by:
Dr. Gourinath Banda,
Assistant Professor,
Computer Science and Engineering,
Indian Institute of Technology Indore

INDIAN INSTITUTE OF TECHNOLOGY INDORE
November 2016

CANDIDATE'S DECLARATION

I hereby declare that the project entitled **"Intelligent Assisted Living Systems with Multicore Microcontrollers"** submitted in partial fulfillment for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** is an authentic work.

The project was supervised by **Dr. Gourinath Banda, Assistant Professor, Computer Science and Engineering, IIT Indore**.

Further, I declare that I have not submitted this work for the award of any other degree elsewhere.

Bittu Kumar
130001008
Discipline of Computer Science and Engineering
Indian Institute of Technology Indore

CERTIFICATE
by BTP Guide

It is certified that the declaration made by the student is correct to the best of my knowledge and belief.

Dr. Gourinath Banda,
Assistant Professor,
Discipline of Computer Science and Engineering,
IIT Indore

PREFACE

This report on “Intelligent Assisted Living Systems with Multicore Microcontrollers” is prepared under the supervision of Dr Gourinath Banda, Assistant Professor, Computer Science and Engineering, IIT Indore.

I have put my best efforts to explain the proposed device. Implementation and testing of the device is also discussed.

Through this report, i have tried to discuss some of the common life challenges faced by aged people and how to solve them using Multicore Microcontrollers. To solve those problems, i have designed a device, which could be easily used by these people in daily life to tackle these problems whose design and implementation is discussed in detail in the report. r

ACKNOWLEDGEMENTS

I would like to express my gratitude towards Dr Gourinath Banda for constantly supporting and motivating me throughout the project and for always giving me new ideas and guiding me in the right direction.

I would also like to thank my family, friends and colleagues for always being there with me and for being a constant source of motivation.

Bittu Kumar
B.Tech IV Year
Discipline of Computer Science and Engineering
IIT Indore

ABSTRACT

Assisted Living is one of the most emerging fields of the technology sector today. It has the potential to change and enhance the everyday lives of people. People are continuously getting more conscious about their health and trying to make their life easier using technology. Which at this time has not been explored much. With intelligent design and right amount of exposure there is going to be a vast market for such devices.

Many organisations are working on developing devices of Assisted Living. We will be discussing about one such device which will be used to tackle some daily life problems of aged people. We will be using multicore microprocessors, sensors and actuators with some other electrical component in the development of this device.

Our device would have the ability to detect some of the health problems and respond accordingly (either alert some preassigned contacts, call ambulance or do some predefined tasks). The design, implementation and testing of the device is discussed in the report with it's future scope.

Contents

List of Figures

Chapter 1

Introduction to Assisted Living

1.1 Buzzword: Assisted Living

An **Assisted Living Residence** or **Assisted Living Facility** (ALF) is a housing facility for people with disabilities or for adults who cannot or chose not to live independently. The term is popular in the United-States but is similar to a retirement home in the sense that facilities provide a group living environment and typically cater to an elderly population.

In the modern world, the technology has empowered us to live more easily, overcoming any difficulty we face. The basic idea behind technology was to ease our life and further improve the quality of living. Assisted living is kind of one focussed step into the same area. It is the step where we try to counter the daily life problems of special needy person, group of persons, gender or community, or just the improvement over the quality of life of general people.

For example we can think of paralysed people who can't move around so a wheelchair is designed for them in which they just need to see on one of the four images shown on the screen and the chair would move by itself into the assigned direction, more cases are like when it comes to gender specific, like in case of mistreatment with females in public places, there are various devices developed which automatically alert the assigned contacts in these situations, sending their location and relevant data to the same, thus enabling females to be more independent and secure. In cases where we improve the quality of our general lifestyle there are a lots of examples like the most common wrist bands nowadays which measure our heartrate and blood presssure and various vital signs in our body which enables us to keep monitoring our health status.

1.2 Motivation to Assisted Living

Assisted Living helps us to remain carefree for our loved ones and gives assurance for their better life. It helps the person to feel more independent and confident about his life e.g. An aged person who faints sometime can be carefree of the fact that if there is always someone there to take care of him or not because he knows that certain device would alert the caretaker at such incidents.

Since Assisted living residences are build for specific group of people, so it would definitely be cheaper than taking all that care personally. With the passing of time people are getting more conscious about their health and they want to carefully plan their future years, both of which brings a big market for Assisted Living devices. Assisted living is also effectively cheaper than most other healthcare services like hospitals, regular clinical checkups, which gives it a lot of audience and again the market.

1.3 Some common Assisted Living equipments

Some common Assisted Living equipments can be seen as hearing aid machines, wheelchair designed for special needs, various bands designed to monitor vitals of the person, special breathing machines, GPS machines(for alzheimer patients) etc.

These devices could also be connected to a central server through mobile phones or directly where data from these are stored for future reference.

FIGURE 1.1: Assisted Living equipments

1.4 Some common challenges faced by old people

1. **Old people faint and fall on the ground:** Due to various diseases and degrading health in old age, people fall on the ground unexpectedly and there are cases whether no one notices it untill it is late or damage has been done.

2. **Old people with Asthma condition encounters situations where they do not have medicine (gaseous spray) with them in time of need:** With the passing of time the memory of old people degrades along with their physical capability, so there are many instances where they forget to refill their asthma medicine which they need after continuous intervals, and by the time they realize it they have not much physical capability to move quickly to refill, and this results them to fall on ground and getting unconscious from short of breath.

3. **Old people usually have problems related to heart, one of these is heart attack:** Since old people have many ailments related to heart and heart attack is one prominent one, also heart is also affected by various other ailments in body, so monitoring heart can help in detecting these problems too.

Chapter 2

Role of Technology

Many daily life problems could be easily solved using our advanced technology. We have got sensors and processors to detect various activities and actuators to respond.

2.1 Technological solution approach to the problems

These problems can be solved using three steps first:

1. **Observing the user:** In this step we measure any measurable quantity about the user or any relevant activity which would be useful in detecting our required activity. These requirement is fulfilled by various sensors

2. **Processing the raw data:** In this step we process the raw data using various logics derived by either experimental methods, hit and trial method or just scientific fact, and then we bring out the relevant result from the raw data. And we finally either get a relevant data or a result that the type of incident we are looking is found or not.

3. **Response to the activity:** In this step we respond, once we have been confirmed that the activity we have been monitoring for is found. There are many ways we can respond to this activity, we can either send this data to other device or just store the result or make some moves by our own like lowering the temperature in AC or starting a warning noise, or any other device, or it could be any custom made device which will perform a definite task when signaled to.

The earlier mentioned problems could be easily solved by monitoring the movement and the heart rate about the user. The data about the movement of the user can be easily obtained by using sensors like accelerometer and the heart rate data can be obtained by using sensors like piezo sensors. These data can be processed at a central processing device for detection of these incidents. Once detected the device could be set to take appropriate action like alerting some other person, calling ambulance, or take some action like injecting the user with the appropriate medicine.

2.2 New Technology

- **Economical Sensors:** For the solution of these daily life problems there are lot of sensors which could be employed depending on the requirement like temperature sensors, proximity sensors, piezo sensor, speed sensor etc. The best thing about these sensors is that there is a tradeoff between accuracy and the cost of the sensor and for most of the daily life solution (of problem faced), we don't need much accuracy so effectively they are cheap components which could be easily used for development of these devices.

- **Advanced Actuators:** Once the problem is detected, then we need to respond appropriately. This could be done in various ways depending on the condition for example we can alert the user about the same using some warning sound made by the speaker placed on the device or it might inject some medicine into the body of the user, or there might be case where it just sends some signal to other device to take some action for itself. The choice and design of actuator will highly depend on the user and the work it is being employed for.

- **Multicore Microcontroller:** Sensors will obtain the data for us, and the actuators will respond for us but who will decide when to take action, that's where multicore microcontroller comes into play. It processes the data by the sensors and then signals the actuators to take appropriate action. The multicore design of this microcontroller enables it to process data from various sensors at the same time which makes it more accurate, also the response time increases effectively, because it will have an independent processesor handling the responses.

Chapter 3

Problems meets Technology

We will discuss how the problems we described in section:1.4 could be tackled using technology and some good combination of sensors and actuators.

3.1 Solution approach to "Old people faint and fall on the ground"

Whenever a person will faint and fall on the ground, we can detect it with monitoring the movement of the person. We will look for incidents where cases close to freefall are encountered, because falling unconsciously is very similar to freefall of the body. Now there are many factors which will affect our result:

- **Position of the device on the body:** For monitoring movement of the person the position of the device is very important. The right placement of device will save us from many unwanted data. Like since we want to detect falling of the person placing it on wrist does not makes sense which will give us too much unuseful data, whereas placing it near belly or chest will save us from this.
- **Future data:** If after detecting free fall we detect normal movement then previous case was surely not a "Faint and Fall" case. So we need to monitor the patient for a while more after detecting free fall before declaring it a "Faint and Fall" case.

  Or we can use certain other vitals of body to confirm the incident and reduce the delay in warning. Heart rate or breathing rate or pulse rate could be used for this purpose, if the person has fainted then fallen then his pulse rate will start going down because he is going unconscious.

3.2 Solution approach to "Alerting in the case of heart attack (both warning the user before and the time of incident)"

Cases like heart attack can be easily detected by monitoring heart rate of the patient. Any unusual activity or behaviour in heart rate can be a potential threat. These cases could again be confirmed by looking at the "movement" activity of the user in past. Like there can be cases where user just had a workout so it's obvious that his heart rate will be above normal for some certain interval. If unusual activity in heart rate is detected then it would be saved in machine.

Now at some particular time of the day it will check how many such activity has been encountered throughout the day and from how many days. It could easily deduce from that if the user is having some heart related problem or not, if it detects then it will alert the user about the same.

3.3 Solution approach to "Old people with Asthma condition with no medicine in emergency"

If we have the information that the user is suffering from Asthma. And there may be cases that the person could not take the medicine in time due to various reasons one of which is he forgot to refill his medicine and now he is having difficulty in breathing and falls on the ground and goes unconscious.

This case is similar to out first problem discussed **"Old people faint and fall on the ground"**, infact it will be more easily detectable because heart rate will act as a more strong indicator here. Here what we can improve is that we can incorporate a certain bursting mechanism which will burst a packet of medicine (so that patient can breathe in the medicine) in these cases which will give a window to the patient to refill his medicine.

3.4 Device requirement

So after all this discussion, the final device which will fulfill our needs described in above proposed solution approaches will need to have to fulfill following criteria:

1. Can monitor the Movement of the patient.
2. Can monitor the HeartRate of the patient.
3. Should have the capability to process data from these both components at the same place.
4. Should have a mechanism to alert the user.

Chapter 4

Implementation

We will discuss about the architecture of the device which will fulfill our requirements for earlier mentioned specifications in section: 3.4 .

4.1 Objectives

1. **Monitor the Movement of the patient:**
   - **Abstract:** Device should be able to monitor the motion of the patient so that whenever a free fall happens we can detect it.
   - **Device Idea:** This requirement could be fulfilled by using Accelerometer. Accelerometer detects acceleration, both static (gravity) and dynamic (produced while doing various activities). So accelerometer can easily detect a freefall if it reads 0 acceleration, but in our more real world scenario the reading would be close to 0.
2. **Monitor the Heart Rate of the patient:**
   - **Abstract:** Device should be able to monitor the heart rate of the patient so that whenever an unusual change in heart rate happens we can detect it.
   - **Device Idea:** This requirement could be fulfilled by using PiezoSensor. PiezoSensor detects vibrations. Since heart also produces vibrations in the body which is proportional to heart rate, we can detect it using piezosensor. PiezoSensor would be place in contact with the body of the patient so that it mostly senses the body vibrations and escape most error signal.
3. **Capability to process data from these both components:**
   - **Abstract:** Device should be able to process the data received from both sensor at the same place because these datas are interrelated when concluding various results like if we detect falling down then we need to check heart rate to see if he is unconscious and when we get high heart rate then we need to check if this is due to some activity the user is doing or it is heart problem using movement data.
   - **Device Idea:** This requirement could be fulfilled by using XMOS, a multicore microcontroller. This microcontroller has 7 cores which can run simultaneously, which could be used in parallel to input data from these sensors, and processing the data simultaneously.
4. **Mechanism to alert the user:**
   - **Abstract:** Once we detect our wanted situation (faint and fall, heart attack etc), we need to respond. We will respond in two steps. First we will alert the user, if he responds means nothing is wrong means wrong detection, but if he doesn't responds then it means that it is indeed our wanted situation and we will alert either the ambulance or some assigned contacts.

- **Device Idea:** For the first requirement of warning the user and a way for user to respond we will use a buzzer and a button. We will start the buzzer as a signal to warn and if the user press the button then he is conscious and nothing is wrong so we will switch the buzzer off but if he doesn't respond under certain interval then we will alert the ambulance or certain preassigned contacts, which could be done by connecting our microcontroller to smartphone and sending data to smartphone to alert someone.

4.2 Components Used

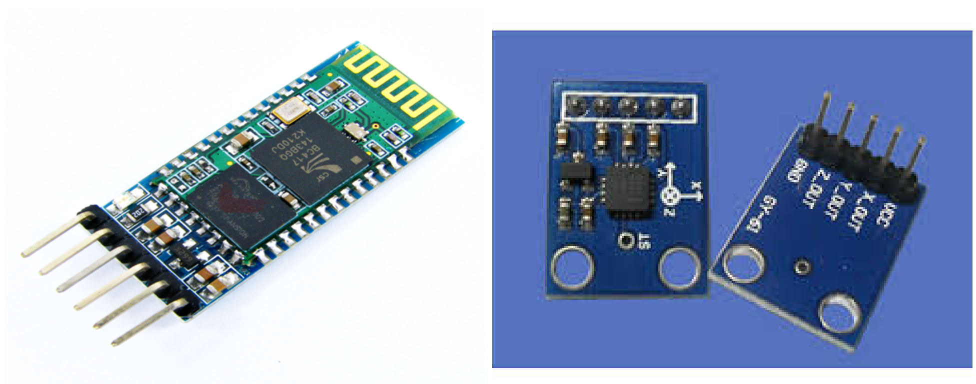

FIGURE 4.1: Blutetooth Module and Accelerometer

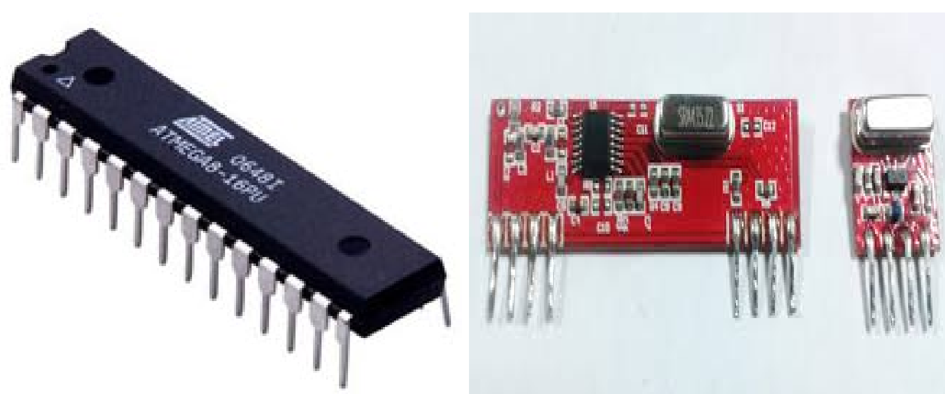

FIGURE 4.2: Atmega8 (Microcontroller) and RF: Receiver & Transmitter

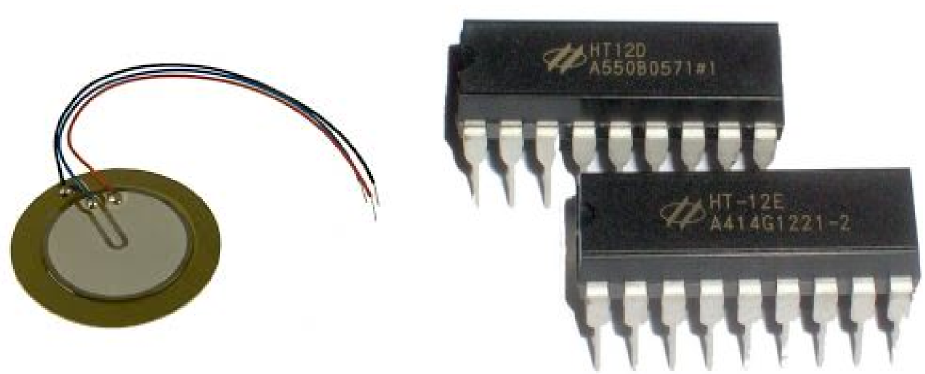

FIGURE 4.3: PiezoSensor and HT12D & HT12E (Encoder & Decoder)

- **LEDs**
- **Resistors**
- **Voltage Regulator (9V to 5V)**

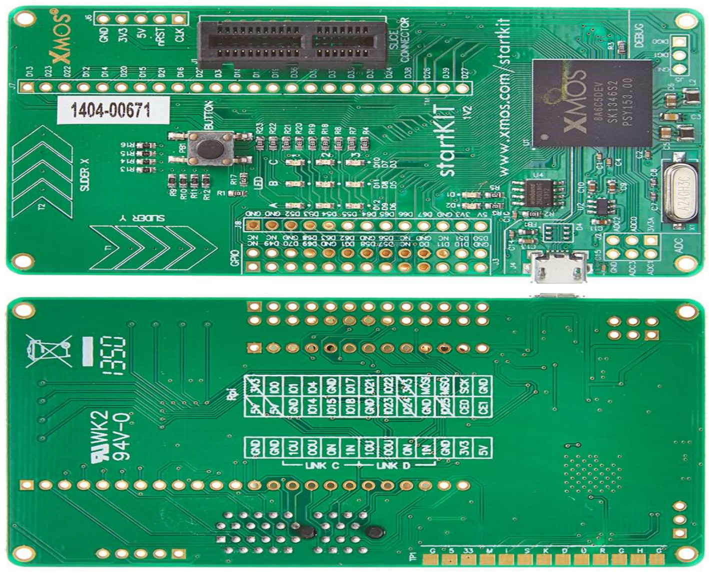

FIGURE 4.4: XMOS (Multicore Microcontroller)

- **Buzzer & Button**
- **9V Battery**

4.3 System Architecture

Our device will have an XMOS which will be continously fed data from the Accelerometer and piezosensor. If this detects any unusual activity then it will turn on a buzzer, if the user doesn't respond within certain interval of time, then it will alert the concerned contacts, or if the user responds within that certain interval interval of time then it will run as normal.

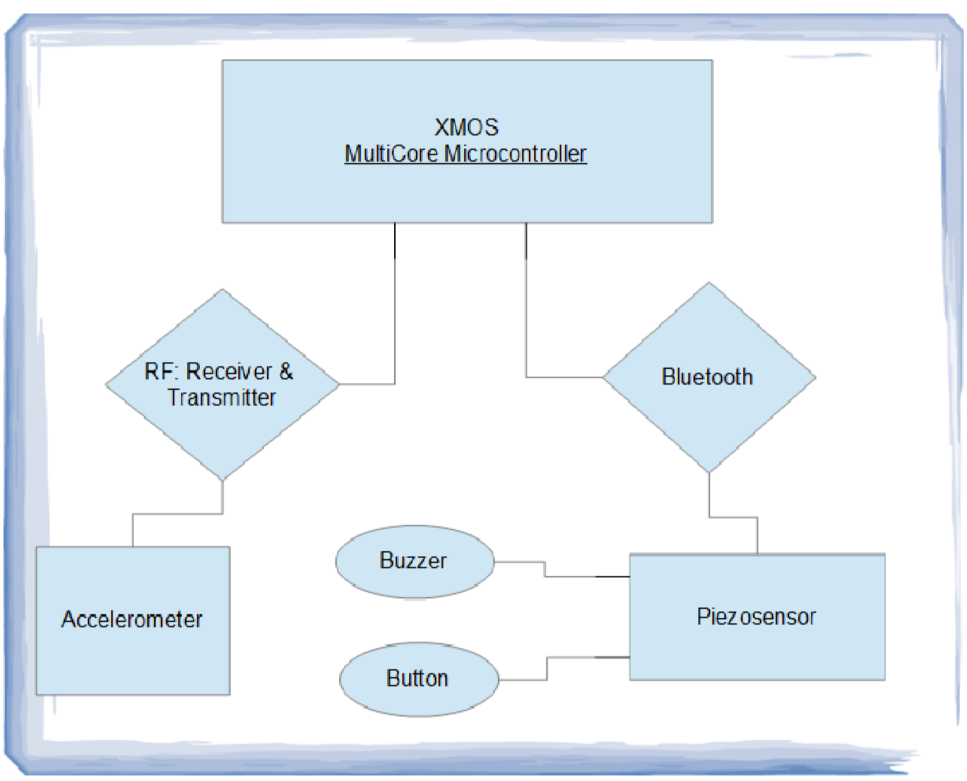

FIGURE 4.5: Abstract Device Design

- **Accelerometer component:** This component will have an accelerometer. It's data would be processed by Atmega8 and converted into 4 feature bits and these bits will be fed to the XMOS for futher processing.

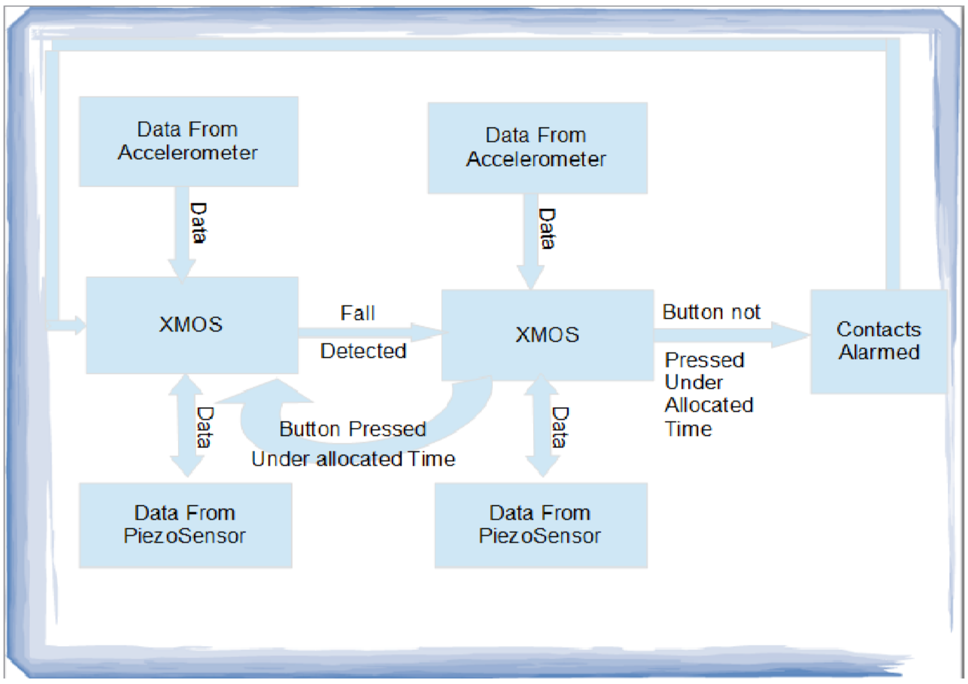

FIGURE 4.6: Mechanism Design of the Device (1)

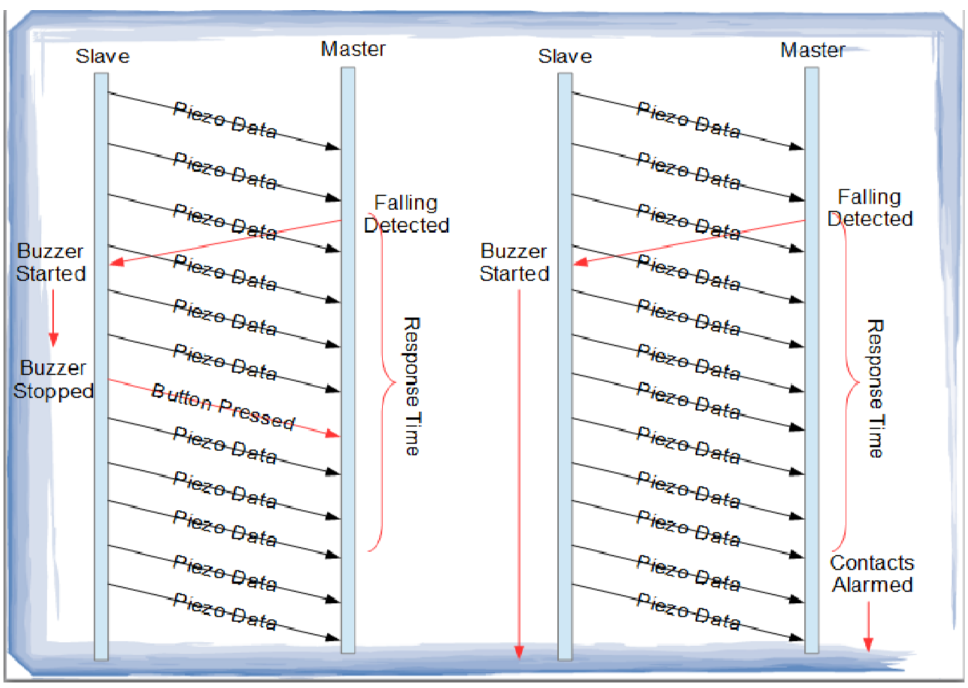

FIGURE 4.7: Mechanism Design of the Device (2)

- **PiezoSensor component:** This component will have an piezoSensor. It's data would be processed by Atmega8 and converted into 8bit wide heartrate data which will be fed further to the XMOS for further processing.
- **Actuator:** This component will have a buzzer and a button, and these will be integrated with the piezoSensor component.
- **XMOS:** This will act as a central hub for processing data and responding when certain events are detected. Data will be continuously fed from the sensor components to this.
- **Master-Slave design:** The multicore microcontroller will take processed data from both sensor components for final decision making and record keeping. Since data has already been processed at the sensor side itself so it will make it more power efficient since now it (the sensor components) would send less data, and also dependency on the type of sensor reduces by a great margin and sensors could be used as a replacable component. Like in our case we will be sending 4 bit wide data of feature extracted from accelerometer instead of the raw data itself and in the second component we will send the

pulse rate (8 bit wide data) of the user instead of the raw piezo data to the XMOS. Both the component would be sending data by wireless methods with baud rate of 9600.

4.4 System Programming

The Atmega8 is programmed in arduino uno environment. It is burnt with Atmega8 bootloader and then the code using USBasp programmer. While burning the code the following board specification was used:

- atmega8.name=ATmega8 (8MHz internal OSC)
- atmega8.upload.protocol=stk500
- atmega8.upload.maximum_size=7680
- atmega8.upload.speed=9600
- atmega8.bootloader.low_fuses=0xe4
- atmega8.bootloader.high_fuses=0xc2
- atmega8.bootloader.path=atmega8
- atmega8.bootloader.file=ATmegaBOOT.hex
- atmega8.bootloader.unlock_bits=0x3F
- atmega8.bootloader.lock_bits=0x0F
- atmega8.build.mcu=atmega8
- atmega8.build.f_cpu=8000000L
- atmega8.build.core=arduino
- atmega8.build.variant=standard

The XMOS will be progrmmed in C along with some of it's own defined libraries for handling threads or it's multicore features.

Chapter 5

Accelerometer Module

This component will obtain data from Accelerometer and then process this data into four bits, where every one bit will represent the status of a feature and then this data will be finally fed to the XMOS.

We will have two parts of this module:

- **Transmitter Side:** It will have sensor along with atmega8 and encoder and rf transmiter. It will basically process the accelerometer data and send it through rf transmiter.
- **Receiver Side:** It will have decoder and rf receiver. It will basically receive data transmitted and will make it available for XMOS.

5.1 Features of Accelerometer

- We will be using ADXL335.
- Detects Acceleration and outputs it as an analog data.
- The ouput is along three axes (X, Y & Z) perpendicular to each other. Which when combined represents the final acceleration vector.
- It detects Dynamic as well as Static acceleration such as gravity.
- It detects -3g to 3g, where 'g' represents acceleration due to gravity.
- The output is mapped to a number in range 0-1023. It means 0 acceleration lies near 612.

5.2 RF: Receiver & Transmitter

We will be using **RF: Receiver & Transmitter** for transfering data between the circuit at sensor side to the circuit at Central Processor side. RFs can be used to transfer data in only one direction which makes it fit for our purpose where we need to send data in only direction i.e. from sensor to the central processor.

We will be using it along side with HT12D & HT12E (4bit encoder & decoder), which will enable us to send 4 bit wide data across.

5.3 Position of the component

The device will be placed near the chest of the user. Since placing it on wrist or any place on arm will produce many unwanted data whenever the user is doing any domestic chores. Again placing it near belly will produce unwanted data whenever the user sits or leans, because the body bends at that point which will change the orientation of the accelerometer which will result in change in output by accelerometer because acceleration is a vector quantity.

So we will place it near the chest of the user which will get rid of all these unwanted abberations in the data for us. We will place it in such a way such that X-axis of accelerometer points directly downwards with Z-axis pointing in the front of the user. In this way while standing gravity will only affect x-axis output data of accelerometer.

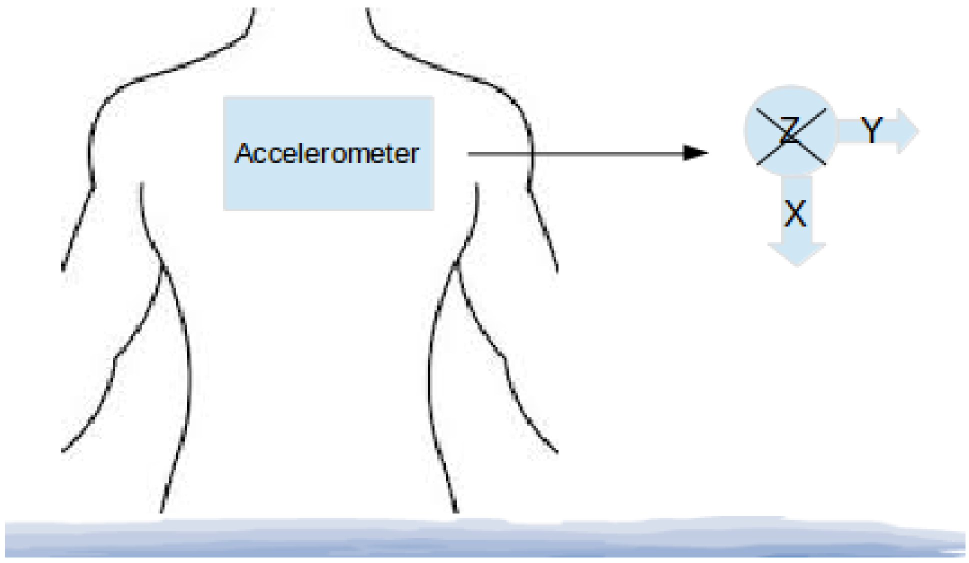

FIGURE 5.1: Position of Accelerometer

5.4 Component Architecture

There will be two circuit for this component Data Transmitter circuit and Data Receiver circuit.

- **Data Transmitter Circuit:** The data flow of this circuit is described below.
  1. Analog data is produced by accelerometer and this is fed into Atmega8.
  2. Atmega8 processes the data and converts into 4 bits (which represents different features).
  3. Then this 4-bit wide data is fed into encoder which serializes this data and further transmits using rf transmitter.

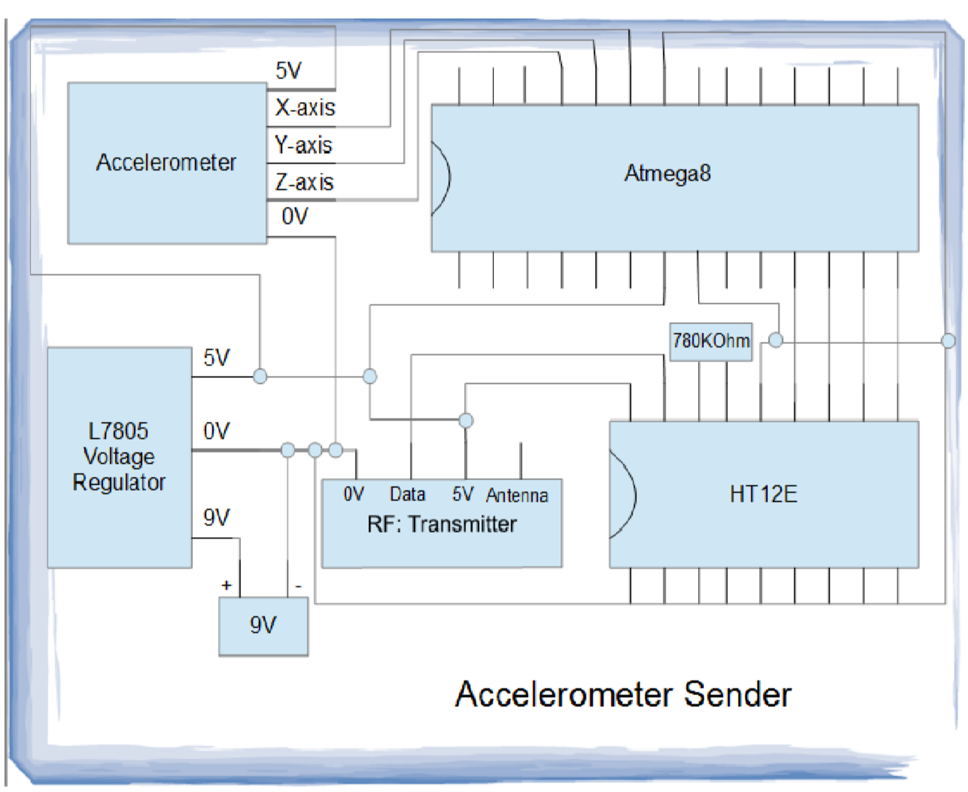

FIGURE 5.2: RF Transmitter circuit

- **Data Receiver Circuit:** The data flow of this circuit is described below.
  1. The Transmitted data is received by rf receiver and is further fed into decoder.
  2. Decoder decodes the data and make it availabe at four ports for futher operation.

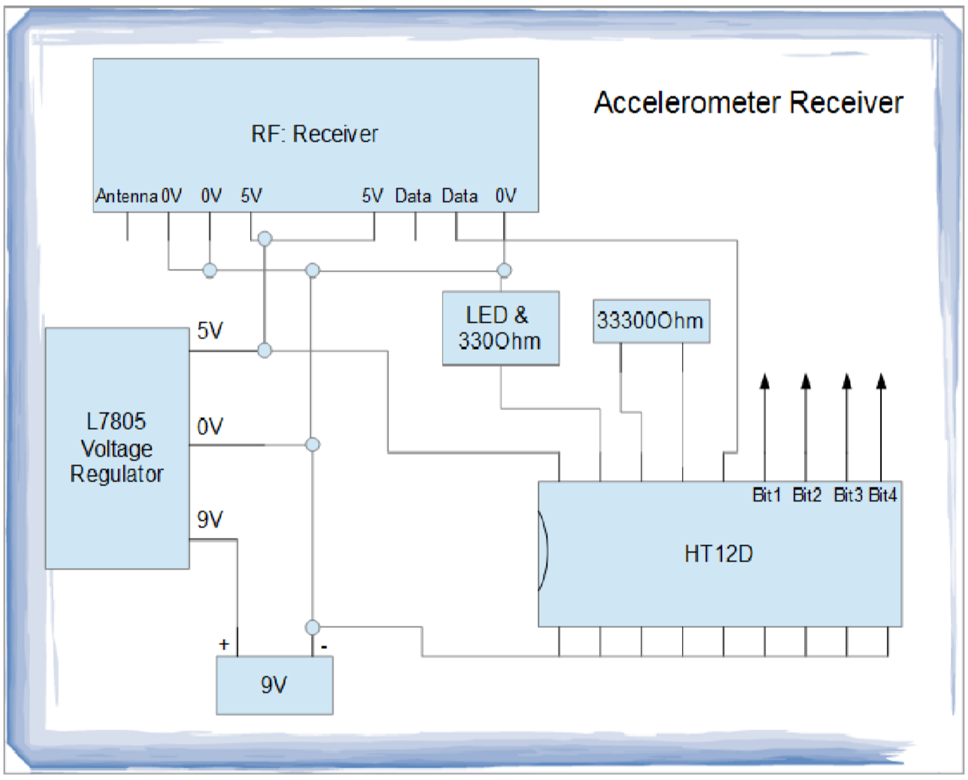

FIGURE 5.3: RF Receiver circuit

5.5 Calibration of the Sensor

The sensors need to be calibrated before use. In calibration we will find the maximum and minimum value of the data given by sensor in our experiments like for our purpose we don't need to cover cases where the magnitude of acceleration is more than 1.3g (g = acceleration due to gravity). After finding these values we map this band of values to 0-255.

Some Constants found Experimentally

- Before Calibration:
  - Minimum value of X-axis data: 239
  - Maximum value of X-axis data: 480
  - Minimum value of Y-axis data: 191
  - Maximum value of Y-axis data: 462
  - Minimum value of Z-axis data: 244
  - Maximum value of Z-axis data: 520
- After Calibration
  - Value of X-axis data with gravity in same direction: 40
  - Value of X-axis data with gravity in opposite direction: 180
  - Value of X-axis data with gravity perpendicular to this axis: 110
  - Value of Y-axis data with gravity in same direction: 70

- Value of Y-axis data with gravity in opposite direction: 200
- Value of Y-axis data with gravity perpendicular to this axis: 130
- Value of Z-axis data with gravity in same direction: 28
- Value of Z-axis data with gravity in opposite direction: 152
- Value of Z-axis data with gravity perpendicular to this axis: 80

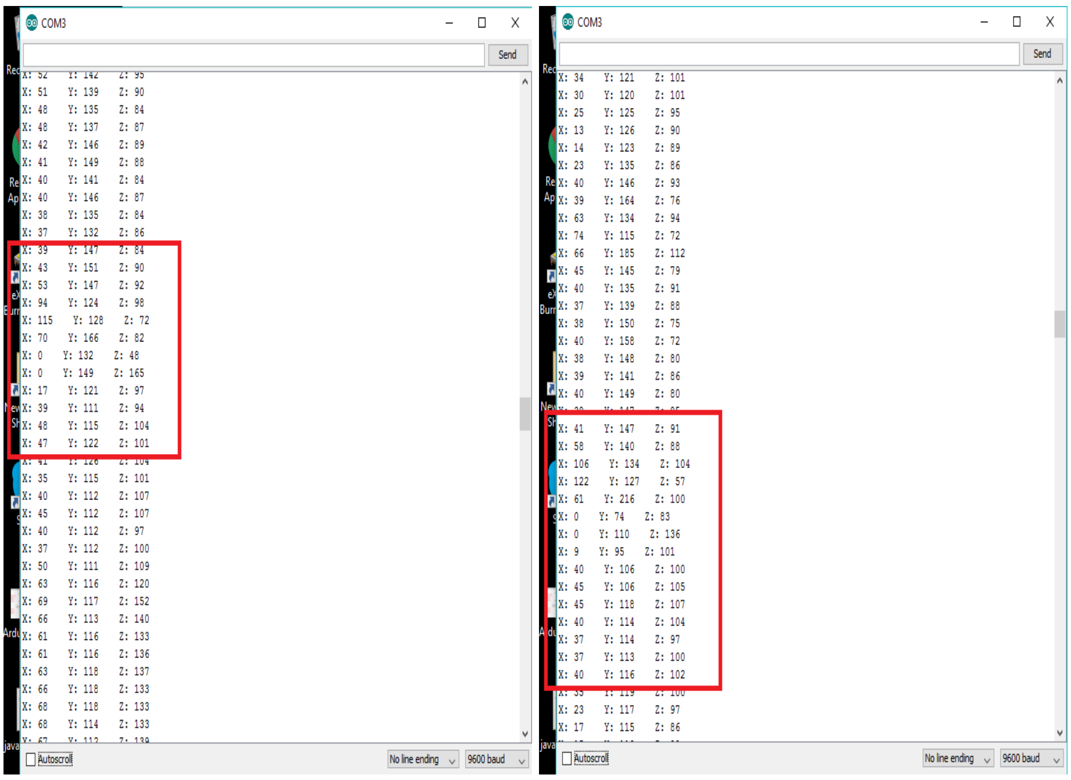

FIGURE 5.4: Accelerometer data while falling down after calibration

5.6 Data Transferred

We can send a raw data of 4 bit (using encoder) to XMOS. But 4 bit data will diminish the resolution of the original analog data very much which cannot be used for intelligent and efficient result extraction further, so instead we will process the data here itself and send data about some features to the central processing unit.

Four Bits:

1. **First bit:** Whether the person is standing or not.

   This could be easily extracted from the accelerometer data, when the person will be standing then gravity will be acting only on the x-axis, and we have this constant with us beforehand, so we just need to check if the data received is approx equal to the recorded (constant) data.

2. **Second bit:** Whether there is free fall or not, by checking total acceleration is less than 0.7g (g = acceleration due to gravity).

   Since we have the values of different axes with full effect of gravity on it in either positive or negative direction, so by taking the magnitude of difference of data to the normal value(value where there is no effect of gravity on that axis), we can find the magnitude of component of gravity acting on it currently, with this data on all three axes we can do vector addition to find the overall acceleration given by the device. Now we will compare it with gravity, and if it's less than 0.7g then we will assume that he is in free fall state.

3. **Third bit:** Whether the total acceleration is approx equal to g.

   Same as before we have the total acceleration in our hand we just need to compare it with the gravity and set the bit accordingly.

4. **Fourth bit:** Whehter there is a peak in X-axis data from accelerometer.

   So this will be calculated by taking the sum of square of difference in consecutive data produced by the x-axis of the accelerometer. This feature is taken by looking at the data produced while falling (experimentally), as while free falling the x-axis will be affected most because gravity is parallel to x-axis at this point and when the free fall ends the gravity will start to act again which leads a sudden transformation in the data of this axis and sum of the square of the difference will effectively capture it.

5.7 Algorithm to Detect Falling

Final algo for detecting falling down case from these feature data array in the XMOS:

- If the first bit changes from 1 to 0 and remains.

  When a person falls then it change it's position from standing to lying down and remains lied down.

  - If the second bit changes to 1 in this small instant of time.
    When a person falls down then for that small instant instant of time the body experiences free fall like situation.
    - If the fourth bit also changes to 1 in this small instant of time.
      If the person starts falling then the data of x-axis of accelerometer will change drastically because it will be affected by changing relative vector of gravity.
      - After this small interval the third bit changes to 1 and remains.
        If the person has fallen down and gone unconscious then after falling on the ground he will lie there without any other external acceleration than gravity.

Chapter 6

PiezoSensor Module

This component will obtain data from PiezoSensor and extract out values proportional to the heart rate of the person represented in the range 0-252, then this data will be finally fed to the XMOS.

We will have two parts of this module:

- **Transmitter Side:** It will have sensor along with atmega8, bluetooth, button and buzzer. It will process the PiezoSensor data and extract out value proportional to heart rate of the user and transmit it using bluetooth. And it will keep listening whether it should start the buzzer. The buzzer will go on whenever bluetooth receives a warning signal and the button will switch off the buzzer and also result in transmitting a signal through that the button has been pressed.

  The baud rate of bluetooth is set to 9600 and it is configure to act as slave and connects automatically to bluetooth on the receiver side.

- **Receiver Side:** It will have bluetooth and atmega8. It will basically receive data transmitted of heart rate and button from transmitter circuit and will make it available for XMOS. And it will keep sending whether the receiver side should start the buzzer or not.

  It is configured to act as Master and connects automatically to our slave bluetooth whenever both are in range. And it's baud rate is also set to 9600.

6.1 Features of PiezoSensor

- PiezoSensor basically detects vibrations.
- It works on the principle of piezoelectric effect.

  If there is a vibration then the crystal (or any material used) also vibrates which results in compression and expansion of the material and this is converted into electric signal.

  The higher the amplitude of vibration, higher the output.
- This sensor is extremely sensitive to heat.
- It should be handeled carefully while soldering wires on this sensor.

6.2 Bluetooth

We will be using bluetooth to transfer data between the transmiter circuit (containing sensor) and the receiver circuit (connected to XMOS). With the help of bluetooth we can do two way communication, which is needed in this case.

Bluetooth can take two roles either "Master" or "Slave". No two slave can talk to each other directly. One slave can connect to only one master. One master can connect to upto 7 slaves.

Bluetooth works in 3 different modes. For changing the attributes of bluetooth we need to get it into command mode. To change it into command mode we need to restart the bluetooth with the button (on the bluetooth) pressed. After we get into command mode we need to use these commands to change settings from the serial monitor (Bluetooth is connected to the arduino). To check if the bluetooth is in command mode or not , use command AT.

- For bluetooth at receiver side:
  1. AT+UART=9600,0,0
     Sets the baud rate to 9600.
  2. AT+CMODE=1
     Now the bluetooth will connect automatically to the binded device if password is same.
  3. AT+BIND=aaa,bbb,ccc,ddd
     It will bind this bluetooth to the device with mac address aaa:bbb:ccc:ddd.
  4. AT+ROLE=1
     It will set the role of this bluetooth as "Master".
- For bluetooth at transmiter side:
  1. AT+UART=9600,0,0
     Sets the baud rate to 9600.
  2. AT+CMODE=1
     Now the bluetooth will connect automatically to the binded device if password is same.
  3. AT+ROLE=0
     It will set the role of this bluetooth as "Slave".
  4. AT+ADDR
     It will print the MAC address of the device which will be used later for binding it with master.

6.3 Position of the component

There are two candidate positions for this component.

1. Placing it near chest:
   - **Pro:** Placing it near the heart increases the magnitude of data produced in case of heartbeat.
   - **Con:** Since it is so close to heart the blood flow through different veins also causes high vibrations, which means lots of noise in the data.
2. Placing it near wrist:
   - **Pro:** Placing it near the wrist get rid of all the continuous flow of noise from our data
   - **Con:** Since it is far from heart the magnitude of heartbeat signal is very low compared to data produced due to other vibrations, like in muscles when doing stuff with hands. And since people do many daily life stuff with hands so again a lot of noise.

6.4 Component Architecture

There will be two circuit for this component Data Transmitter circuit and Data Receiver circuit.

- **Data Transmitter Circuit:** The data flow of this circuit is described below.
  1. Analog data is produced by piezoSensor and this is fed into Atmega8.
  2. If the button is pressed , it turns off the buzzer and this data is also fed into Atmega8.
  3. Atmega8 processes the data and extracts data which is proportional to the heartrate of the user.
  4. This heartrate data is fed to bluetooth for transmitting if there is no button press.
  5. If button is pressed then it's signal is transmited instead.
  6. The bluetooth keeps listening for warning signal.
  7. If it receives a warning signal, it turns on the buzzer.

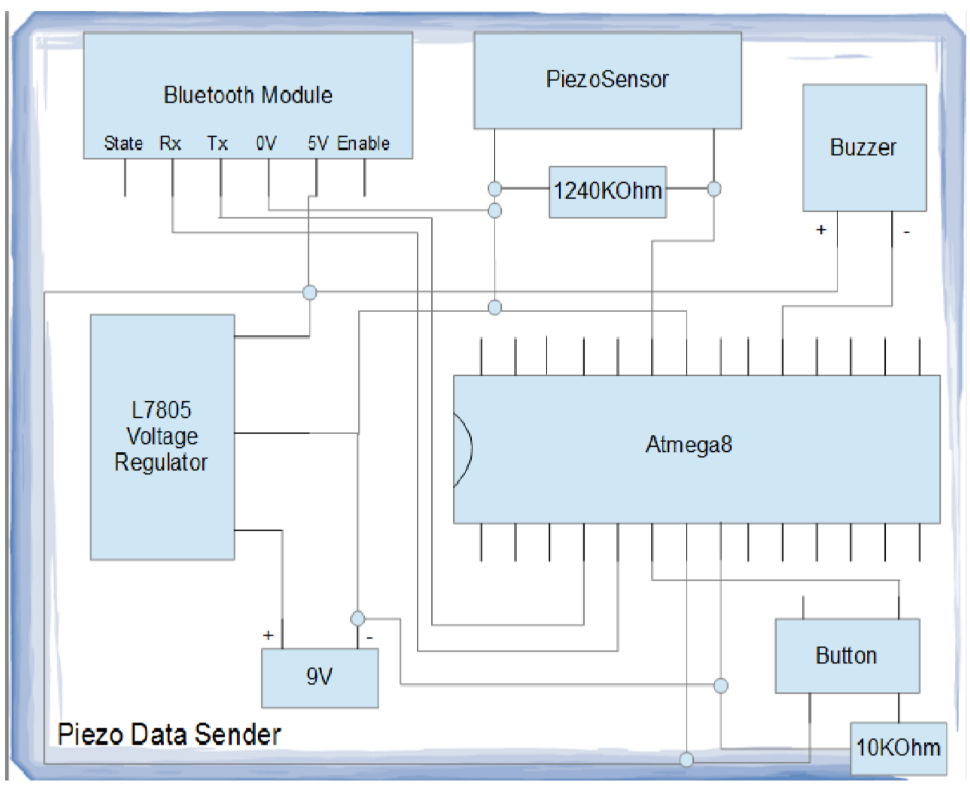

FIGURE 6.1: Bluetooth Transmitter circuit

- **Data Receiver Circuit:** The data flow of this circuit is described below.
  1. The Transmitted data is received by bluetooth and fed into Atmega8.
  2. This data of heartrate or button pressed is directly passed to XMOS
  3. The Atmega8 takes input from XMOS about warning signal and transmits it as it is.

6.5 Calibration of the Sensor

The sensors need to be calibrated before use. In calibration we will find the maximum and minimum value of the data given by sensor in our experiments like for our purpose we don't need to cover cases where the amplitude of vibration is too high (relative to the amplitude of data produced by heart beat). After finding these values we map this band of values to 0-252.

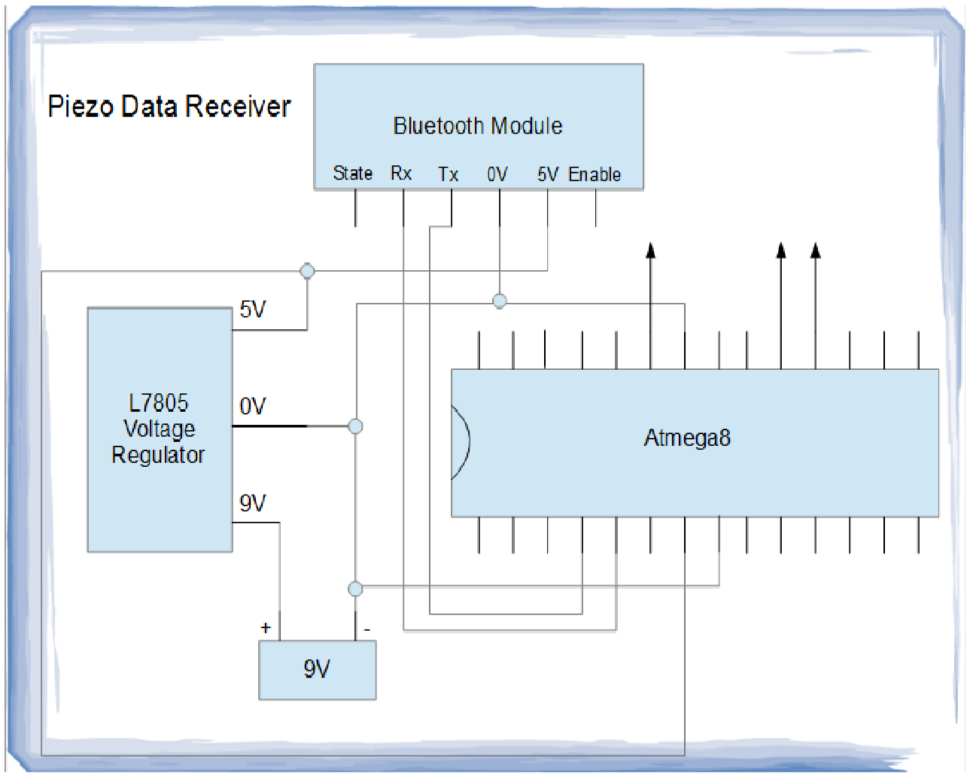

FIGURE 6.2: Bluetooth Receiver circuit

6.6 Data Transferred

Since the data through the piezoSensor will come at very high rate and we will need this all data to extract heartrate out of it, so it will be a bad idea to just send raw data (because it will consume too much power). Instead we will extract heartrate out of this data and send this heartrate instead.

Also we will use value 254 as a marker to tell that the button has been pressed.

6.7 Algorithm to extract HeartRate

FIGURE 6.3: Human Pulse

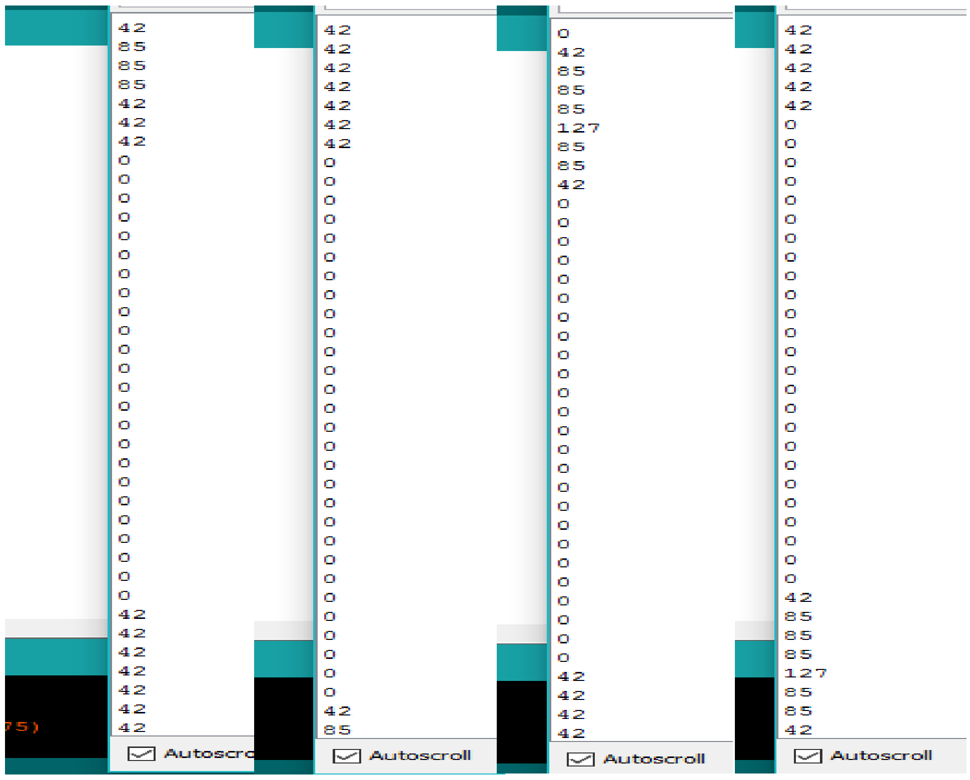

FIGURE 6.4: Ideal Pulse Data without noise

The piezo produces an array of positive numbers when detecting a beat, we will call this array as band. we will calculate the number of bands in the raw data received and divide it with a constant. This constant is equal to 4-6, which would be set experimentally.

Since there is a lot of noise in this data. Some ideas to remove these noise:

- We can employ the fact that disturbance due to heartbeat would not be longer than a certain small interval of time which is proportional to the length of band. And if a band is found with length greater than that (certain experimentally found constant) then it is most probably the data due to noise.
- Also for elimination criteria we will use the fact that heartbeat won't give a value greater than a certain limit (the amplitude of vibration due to pulses would be low, this constant would be found experimentally) so any data containig value greater than that threshold is error prone ,so we will ignore that data too.
- Instead taking the total number of bands in 1 minute for calculating heartrate we will check in windows of small gap (ignoring windows containing noise data) and then average it out for the entire period.

6.8 Algorithm to Detect Abnormal HeartRate

We will simply store the data extracted when piezo is placed on a normal person and will measure the heartrate of the person manually. We will take this data for three different heart rate and use this three point data to approx a two degree function (relation between our extracted data and actual heartbeat.

Now since we know the maximum and minimum heartbeat of a person. We will put these values in the function and get the minimum and maximum (in normal situation) data of a normal person.

Any data out of this band will be considered abnormal heart rate.

Chapter 7

Work Status

7.1 Work already done:

The accelerometer is working fine (checked) , the RF Transmitter & Receiver is also working good (checked) , features are designed and extracted from this data to detect falling, Code has been burnt into the programmer, Bluetooth module is working fine (checked), it's design and circuit is finalized and assembled, PiezoSensor is checked for it's input along with some other components like button and buzzers and also the transmitter and receiver circuit of bluetooth has been finalized and assembled.

7.2 Future Work:

Piezo data need to be filtered to give more accurate pulse data, then all components need to be integrated with the XMOS, with programming XMOS with all the logics to detect required events.

Chapter 8

Conclusion and Future Scope

The device we discussed seems to be very promising at solving the problems we took in hand. We took some common life problems of aged people and tried to tackle it with the use of technology and logics. Sensors and multicore microcontrollers played an important role in this endeavour of ours, with help from actuators.

We used accelerometer to detect movements in the body of the user and to detect the falling of the user, and it was later confirmed with the use of heartrate data we got from the piezosensor and once again it was confirmed by the user with the help of button and buzzer. Similarly in the case of where asthma patients were there we just changed the actuators specially designed for the patients which bursts the packet of medicine when "faint and fall" case is detected. We tackled the problem of abnormal heartrate in the situation of heart attack and various other heart related problems using data from the piezosensors (by experimentally finding the minimum and maximum normal data to be expected out of the device and later checking all the future data that whether it lies inside this range or not).

We extracted some features from the accelerometer data, these features were selected depending on the cases we will be encountering and some experimental features like one feature that there will be a bump in the data of the x-axis while falling was solely taken by looking at the data obtained when a person falls. Remaining all features were designed logically and later on it was checked for agreement. We obtained heartrate from piezosensors simply by counting the number of bands ofpositive numbers in the data array we get (and a few processing after that), although we filtered this data of noises present due to various factors, like vibrations due to outside objects, vibrations due to muscles in body.

Since our device is very modular in design so we can use it with many different component designed to give abstract data to the central processor, where only simple logic will be used to identify the device and their corresponding features and logics would be used to derive results from that data. Like in our case we have developed components using piezoSensors and other with accelerometer we can similarly use other sensors to detect various physical quantities and similarly do a low level processing at the sensor side itself, and then send the processed data to the XMOS. It will enable user to efficiently use the power of the device, since he can remove the unwanted sensors when not needed and add it with the one needed instead.

Bibliography

[1] *Bluetooth Basics [Online]. Available: https://learn.sparkfun.com/tutorials/bluetooth-basics/how-bluetooth-works*.

Appendices

Appendix A

Calibration of Sensors

```
while (startTime + GAP < currentTime())
{
        data = getData();

        if (data < minData)
              minData = data;

        if (data > maxData)
              maxData = data;
}

newData = getData();
mappedData = map (newData, minData, maxData, 0, 255);
mappedData = constrain (mappedData, 0, 255);
print(mappedData);
```

To calibrate the sensors simple logic is used, we first observe the data for some time interval and get the minimum and maximum value of the data observed . Then we use map() function to map the data to our defined range then we constrain the data using constrain() function to our defined range. Now we can use the data.

Appendix B

Calculate Total Acceleration

For any axis "axisX" we have:

- XU : value when the gravity is in opposite direction to this axis.
- XD : value when the gravity is in same direction to this axis.
- XN : value when the gravity is normal to the direction of this axis.

```
dataA = getDataA();

// The magnitude of acceleration.
if (dataA > AN)
        diffA = dataA − AN;
else
        diffA = AN − dataA;

// The amount of acceleration relative to gravity.
diffA = 2 * diffA  / (AD − AN);

// Now similarly we will have data for all three axes
// let's say other two are B and C.
// Now we will do vector sum to find
//ObservedAcceleration/AccelerationDueToGravity.

accRelativeToGravity = sqrt (diffA * diffA + diffB * diffB
                              + diffC * diffC);
```

Since we want to know the magnitude of the acceleration observed we first take the difference from XN which tells us a number proportional to the amount of acceleration. Now AD-AN represents 2g (g = acceleration due to gravity). So using this value we will calculate the relative acceleration in this axis.

In last we just did vector addition. If this value is 1 then the magnitude of acceleration is same as g, or let's say 'x' is the final result then magnitude of acceleration is x*g.

Appendix C

Calculating Sudden Peak in Data array

```
// data[10] : Stores the previous 10 data.
for (i = 0; i < 9; i++)
        diff = (data[i] - data[i+1]) \^ 2;

// Threshold set experimentally by looking at data.
if (diff > Threshold)
        print("Peak found");
else
        print("No Peak detected");
```

For detecting sudden changes in signal or say to detect peak in data array, we are using the fact that there will be large changes in adjacent data values. So we will sum the square of the differences. Squaring it will magnify our result. If this final value is greater than certain preset Threshold then we have found our peak.

Appendix D

Detecting Bands

```
// data[10] : contains previous 10 data.

count = 0;
for (i = 0; i < 10; i++)
        if (data[i])
                count++;

if (disable == 0)
        if (count > 8)
        {
                print("Band Found.");
                skip = 10;
        }

if (disable == 1)
        {
                skip--;
                if (!skip)
                        disable = 0;
        }
```

To detect bands we will simply check an array of data if there are more than 8 positive number in an array of 10 data values then we will consider it as band and we will disable this check for sometime because we will have this same partial data array for next sometime. Skipping some data means next time when we will check we will have fresh new data array.