HUMAN FALL DETCTION AND GPS TRACKER

GROUP NO: D12

Swadesh Choudhary (05D07018) <swadesh@iitb.ac.in>

Ruchit Mehta (05D07017) <ruchit_mehta@iitb.ac.in>

Rahul Dash (05D07014) <rahul_dash@iitb.ac.in>

Supervisors

Abhay Karandikar

Dipankar

D.K.Sharma

Girish Kumar

V.K. Tandon

Abstract

The report discusses the concept, design and implementation of a prototype that does the following:

a) It detects the fall of a person to which the device is attached and sends a SMS to a pre-decided mobile phone, alerting about the fall
b) It detects if a person to whom the device is attached has moved outside a pre-decided polygon-specified by its coordinates-and sends a SMS to a pre-decided mobile phone, alerting about the occurrence
c) On enquiry from a mobile phone in the form of a SMS, the device sends back a SMS to that mobile phone giving the coordinates of the device and thus, of the person wearing it
The prototype that has been built successfully carries out the three actions mentioned above as described in the rest of the report. To increase the efficiency, some future improvements have been suggested.

1. **Introduction**

The main motivation behind the project was to address three critical issues regarding health and safety equipments.

(a) Fall Detection
The first motivation was to develop an instrument or a device that can detect if a person has a fall and then send an SMS to relatives. For e.g. a person may be

walking down slippery stairs in the monsoon, and have a fall. The device would have to detect this and be able to send out a signal for help. The considerations were the different scenarios where this could be used, the design and form factors and the occurrence of false alarms. And accelerometer has been used for this purpose, which has been calibrated in a way that would make it possible to monitor a person's movement under normal conditions, recognize if the conditions of motion have changed, determine if the change is dangerous(e.g. a fall) and send a warning in the form of a SMS accordingly. Presently, we measure the accelerometer o/p and look out for anomalies as a representation for a fall.

(b) Geo-fencing
The second motivation was to have a wearable device that would send an alarm if the person wearing it goes out of a certain pre-defined boundary. This device is aimed especially and little children and mentally challenged people, who need constant attention as they might venture into dangerous areas. The user can give the coordinates defining a polygon which is the "safe area" for the child. If the child moves out of this area, a SMS is automatically sent to the guardian. An efficient algorithm has been written for this functionality that takes care of polygons ("safe-areas") of various shapes etc, as is the case in the real world.

(c) Location On-Demand
This functionality was the first one to be implemented. It basically enables a person to send a SMS to the device asking for the location device (and thus the person wearing the device) A SMS containing the coordinates of the location(generated by a GPS module within the device) is automatically generated and sent by the device and is sent to the other phone in question(via Bluetooth)

2. Report Split

The following pages will give a detailed description of the entire device, with each function treated separately. The 'Location On-Demand' has already been covered in the final report of the previous EDL course. Hence the topics that will be covered are:

a) The various modules used in the device, namely the GPS module, the Bluetooth module and the Accelerometer along with their advantages
b) A description of the "Fall Detection" function along with the algorithm
c) A description of the Geo-fencing function along with the algorithm
d) Battery Management
e) Demonstrations, plots and other results
f) More efficient algorithms for future implementation

3. Description of modules used

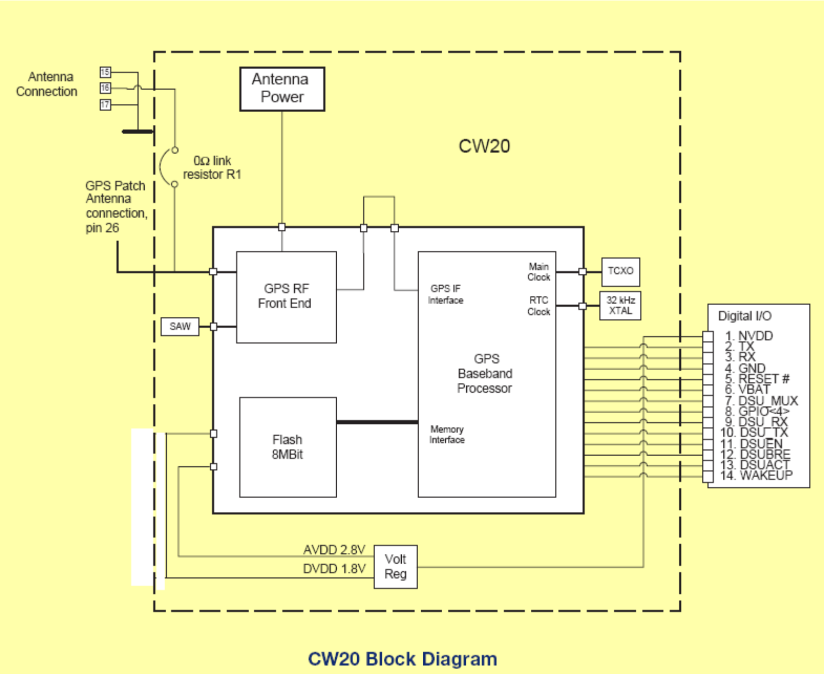
(a) Global Positioning System unit

The Global positioning system is used worldwide to obtain accurate positional information. Tracking with the help of commercially available GPS modules is omnipresent. To include the benefits of tracking into the system and the location of the person, GPS module is used. The device would contain a GPS unit to obtain the coordinates of the user in real time. During a fall the most recent location of the person wearing the device would sent along with the fall-alert. This would help in critical emergency situations.

The module used is the CW20 GPS receiver from NAVSYNC. The benefits and functions offered are:-

1. Supports either active or passive antenna
2. Low power consumption: 19 mA avg fully active
3. Static accuracy – 1.2m to 3m
4. Compact design: 21 x 16.4 x 2.4 mm
5. NMEA format output and UART compatible.
6. Enabling tracking down to -152 dBm,

The GPS module supports a large number of commands for message sequence control, power control and reset. These have been used (as explained in program flow) while keeping the low power and robust nature of tracking into consideration.



Block Diagram of the GPS Receiver module

For the prototype testing we have used an active antenna – Taoglas Titan AA150
It has good tracking range, even down to -162dBm. So tracking was possible inside the antenna lab.

(b) Bluetooth Module

Bluetooth Initialization involves the setup of a communication channel between the Bluetooth module and the cell phone coupled to the device. Steps include:

i) Inquiry: An inquiry is sent out for the devices in range and a check is performed for the Bluetooth device address of the cell phone.
ii) Connect Service Discovery Access Protocol (SDAP): This is a command which is used to create a temporary connection to the mobile phone and determine which RF comm. Channel to use for communication.
iii) Browse SDAP and Disconnect SDAP: Once the com port has been found via the Browse SDAP command, we disconnect the SDAP connection.
iv) Create Serial Port Profile: A permanent link is established to the cell phone.
v) Transparent Mode: Once the confirm for the link established has been received, we set the Bluetooth module in transparent mode so that anything sent to it via the MSP is not interpreted as a command but forwarded to the mobile phone. Now the device is ready to send commands to the GSM module of the mobile phone.
vi) A series of AT commands are sent to the phone to get it ready for the SMS send – receive functions i.e. where the received messages will be stored and read from

(c) ADC

ADC Initialization: Once the Bluetooth connection is successfully established (we used various status flags to indicate the right responses from the module and phone), we proceed by initializing and setting the parameters of the ADC to sample and convert the voltage levels of the accelerometer. We are using three channels of the ADC (port pins 6.3, 6.5 & 6.7 on the MSP). Rather than continuously sampling the channels and interrupting the program flow, we use trigger the ADC at regular intervals and at each trigger we take in 30 samples from the ADC. During ADC Initialization, we assume that the user will be wearing the device in the standard position at his waist, and we do 5 rounds of calibration to store this orientation of the accelerometer. The algorithm of detecting a Fall is explained later.

(d) Accelerometer

The advantages of using an accelerometer for fall detection are:-

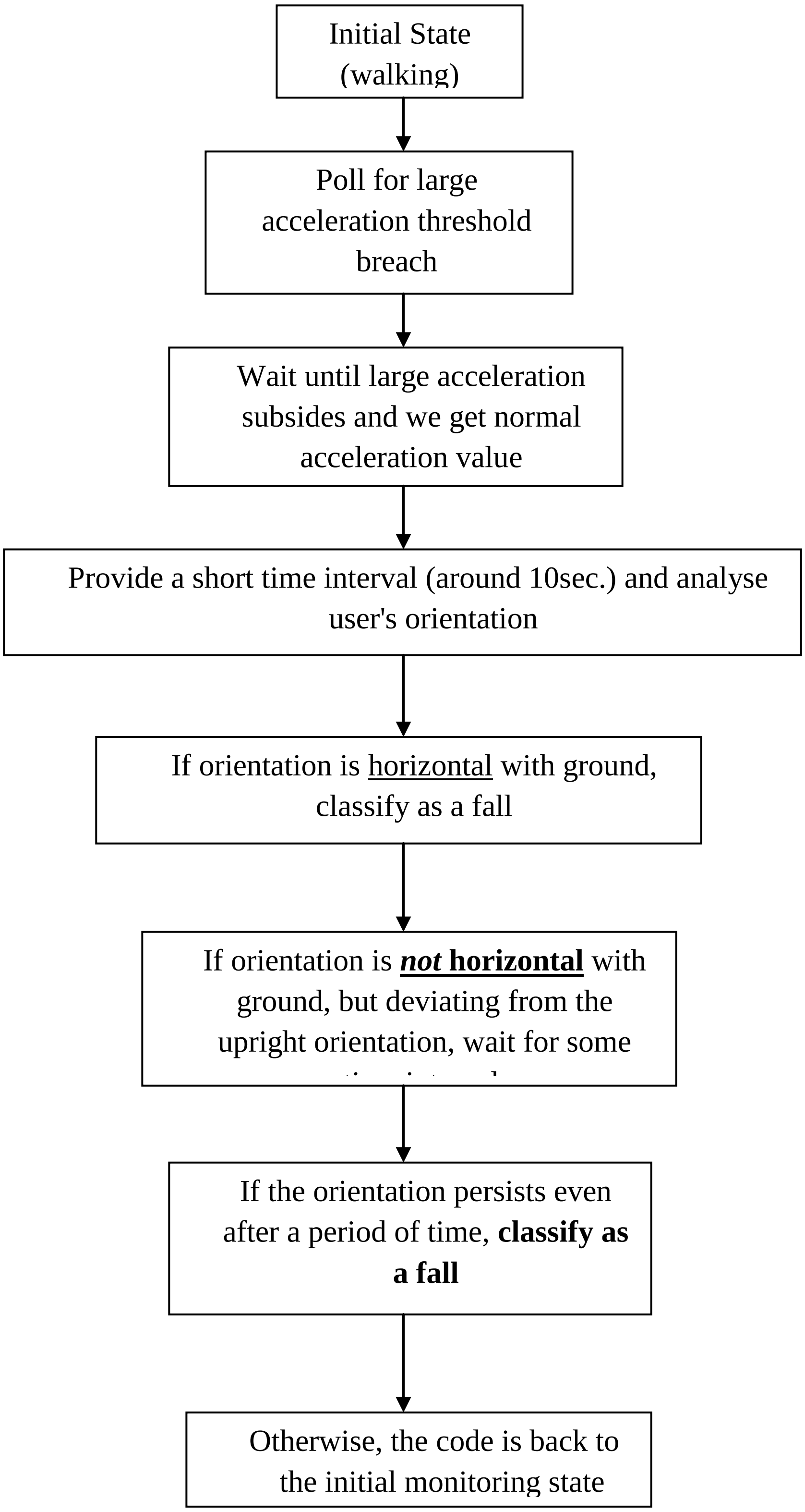
1. They are small and can be mounted easily on the body
2. Can be easily interfaced with a portable processing unit
3. They are low power sensors. Ideal for handheld devices.

Specifications of the 3-axis Accelerometer – Free scale MMA7260QT

- Low Current Consumption: 500 μA
- Sleep Mode current: 3 μA
- Low Voltage Operation: 2.2 V – 3.6 V
- 6mm x 6mm x 1.45mm QFN – small size
- High Sensitivity (800 mV/g @ 1.5g)

4. Fall Detection Algorithm

The idea is to provide information that helps determine if a person has suffered an accident (if the person has fallen) and to provide information related to the fall to determine the magnitude and characteristics of the accident.



5. **Geo-fencing**

Objective:

A major problem many people face in their day to day lives is that their child or a mentally challenged person in their family may wander off into territories deemed too dangerous for them. For example, crossing the boundary of the house, moving close to the pool etc. It is extremely difficult to keep an eye on them 24 hours a day. The present facility wishes to address this problem. The aim is to provide a built in facility in our device which enables the user to set any number of coordinates to form a well defined polygon of any shape. Whenever the carrier of the device crosses the specified boundary, a SMS is sent to the user.

The Process:

In very basic terms, the algorithm is supposed to take in a user defined polygon and send an SMS to the user whenever the carrier of the device ventures out of the region defined by the polygon. The best way to define a polygon is to take in the coordinates of the vertices of the polygon in form of the latitude, longitude of the points. Thus the problem at hand transforms itself into the more general one of finding whether a given point (the carrier of the device in this case) is inside the polygon or outside it. In case it is outside, a SMS should be sent.

The various algorithms that we thought could work at the start were the following:

1) We know the coordinates of all the vertices of the polygon and the coordinate of the point in question. Thus we can find out the angles subtended by the point at all the sides of the polygon. If the algebraic sum of these angles turns out to be 360 degrees, then the point lies within the polygon. If the sum turns out to be 0, then the point lies outside the polygon. However this algorithm is not efficient for the following reasons:
   a) In order to calculate the angle subtended, we need to calculate a dot product, divide it by a magnitude and involve the inverse cosine function.
   b) Further, in order to calculate the sign of the angle considered, we also need to involve the inverse sine function. This makes the algorithm slow.

2) This algorithm is based on the famous Jordan Curve Theorem, which says that any simple, closed figure in a plane divided the plane into two regions, the inside and the outside. All that we do is this: We let a ray originate from the point in question and intersect the polygon's sides in well defined points. We count the total number of such intersections possible. If the number is odd, then the point lies within the polygon. If the number is even, the point lies outside the polygon. Studies have shown that on an average, this algorithm is about 60 times faster than the preceding one.

   However, that does not imply that this algorithm hasn't got problems. The following problems have to be tackled in case this algorithm is to be implemented:

   a) In case the given ray intersects a given side at a vertex or in case the ray itself lies along a side, we need modifications in the algorithm to address the problem.

b) If we take an arbitrary ray and let it intersect the sides, before the problem of counting the number of intersections, we need to find out whether the ray intersects a particular side at all.

In order to take care of these difficulties and to make the algorithm efficient, we make the following modifications:

a) We let the polygon and the point remain as they are but let the ray be parallel to the y axis.
b) Since the algorithm depends only on the number of intersections that the ray makes with the various sides, it is utmost crucial to decide what we consider as a count. We decide to count an intersection only if the following two conditions are met:

(i) The point of intersection of the ray with the side lies to the right hand side of the side. By right this, we mean that if x1 and x2 are the x coordinates of the two vertices of the side and if x0 is the x coordinate of the point, then x0>x1 and x2>=x0

(ii) The point of intersection should lie above the point in consideration. By this condition we mean that in case the two vertices are (x1,y1) and (x2,y2) and if the point is (x0, y0); then (y0-y1)/(x0-x1) > (y2-y1)/(x2-x1) or, (y0-y1)(x2-x1) > (y2-y1)(x0-x1)

Thus, our algorithm runs as follows:

a) We consider each side of the polygon one at a time. We apply the above conditions. In case they are satisfied, we increase the 'intersection counter' by 1, else we don't

b) After we run out of sides, we check whether the counter is even or odd

c) It is easy to observe that the problem of intersection at vertex is easily taken care of by this intersection. Also, as we are not finding out intersection points of the ray with every side, we do not have to worry about the problem of a large number of sides

6) **Battery Management**

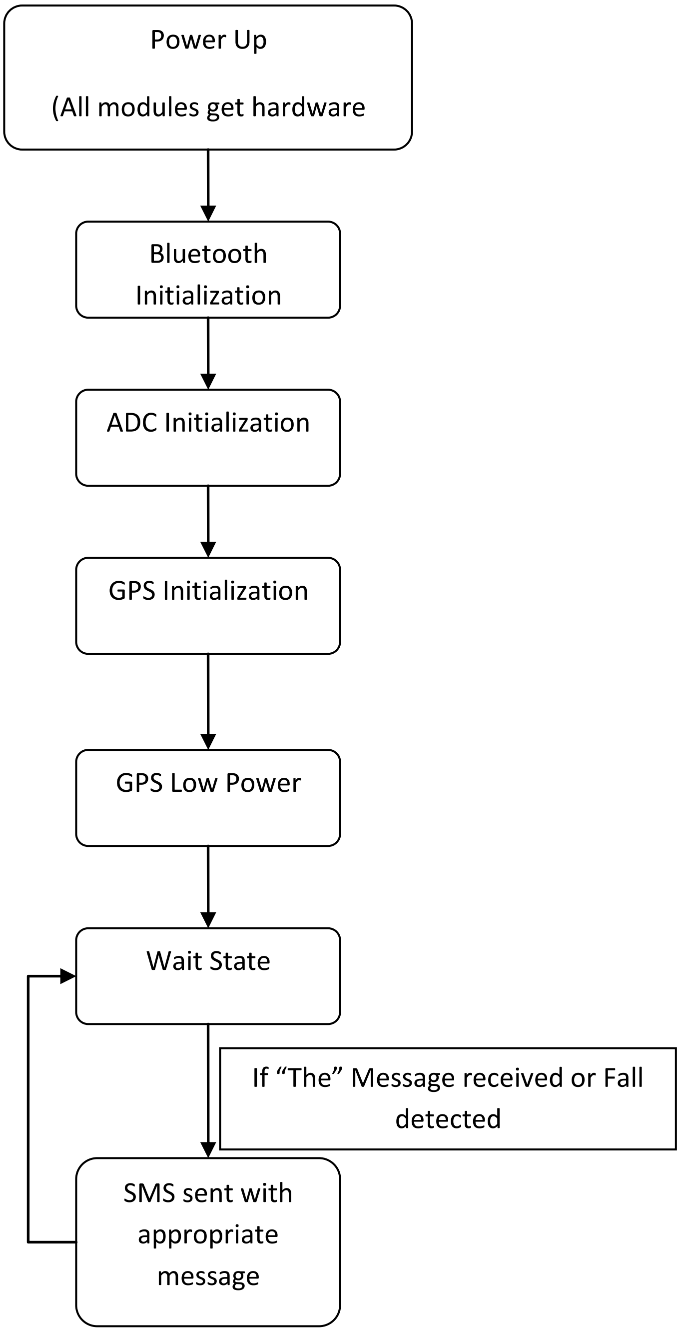
We have included the battery management system which, for Lithium ion polymer batteries, gives an indication when the battery goes below a certain critical value measured against the reference voltage given as the third terminal on the battery pack.

BQ2023 is the IC used.

7. Demonstrations, Plots and Results

(a) Program Flow Chart

Program Flow Chart: -



(b) Demonstration

We have made a prototype of the device. Hence there are some constraints during the demonstration. The final product would have the GPS and Bluetooth modules integrated on the same PCB as the msp and the accelerometer. For the prototype, we had to make separate PCBs for testing and hence while integrating we have attached different modules on the same PCB. This makes the PCB fragile and hence for fall detection testing we take the accelerometer module out from the final PCB with wires. There are four indicator LEDs on the PCB. One for Bluetooth detection(blue), one for the GPS detecting sufficient number of satellites(yellow) , one for the GPS getting the valid data(green) and one for the battery indicator(red).

The end of the initialization phase is indicated by the green and yellow LEDs glowing. We have also interfaced the module with the serial port of the computer so that we can keep track of the GPS data and SMS status. If the accelerometer is dropped from a good height(about a feet), and its orientation has been changed for more than 10 seconds, then an SMS is sent to the target phone stating that the patient has fallen.

Also, we can send a message with the matter “Test.” To the coupled phone and expect a message in return with the coordinates of the device.

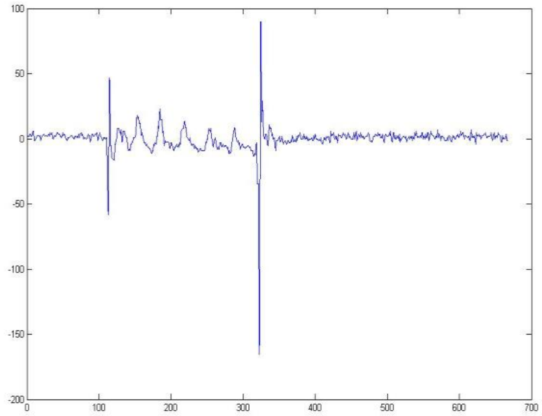
We have also implemented Geofencing as a separate code. Being restricted in mobility, for testing purposes, we define a boundary and if the coordinate is within the boundary, a flag is cleared. Alternately, we modified the boundary such that our location in the lab lies outside it, and the flag is set in this case. The boundary is specified by giving 4 points (each having a latitude and longitude).

Current Usage by the module: - We made a measurement of the current used by the module at various stages of operation. When the initialization begins, the first step is the Bluetooth connection which takes an average of 62 mA for a time of 20 seconds.

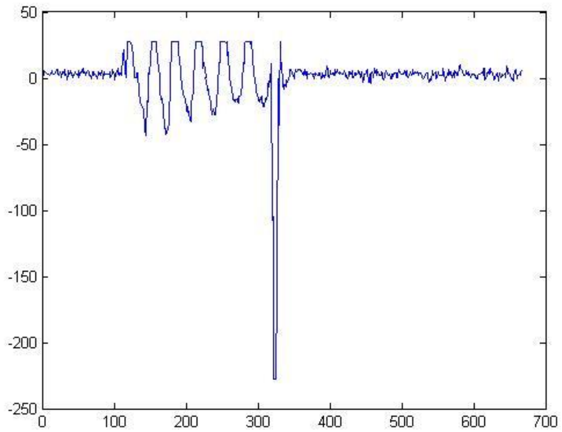
The next step is the ADC initialization and GPS initialization which consumes 50mA for around a minute (depending on how quickly the data is received). After that, if the GPS is in low power mode 30mA is consumed.

(c) **Accelerometer Calibration**

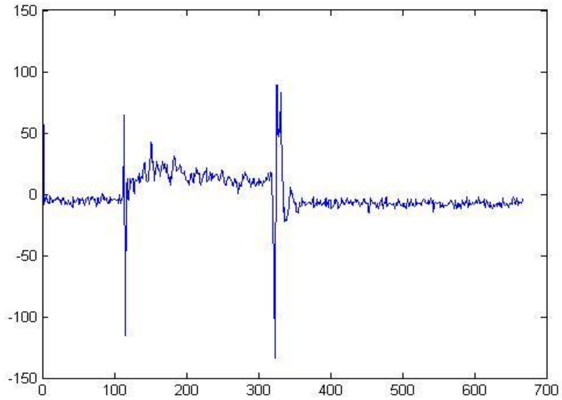
The accelerometer board has been setup and calibration started. The following graphs were obtained. The peaks show the high acceleration encountered during a fall. Threshold is nominally set to +_150 units in the msp code.



X



Y



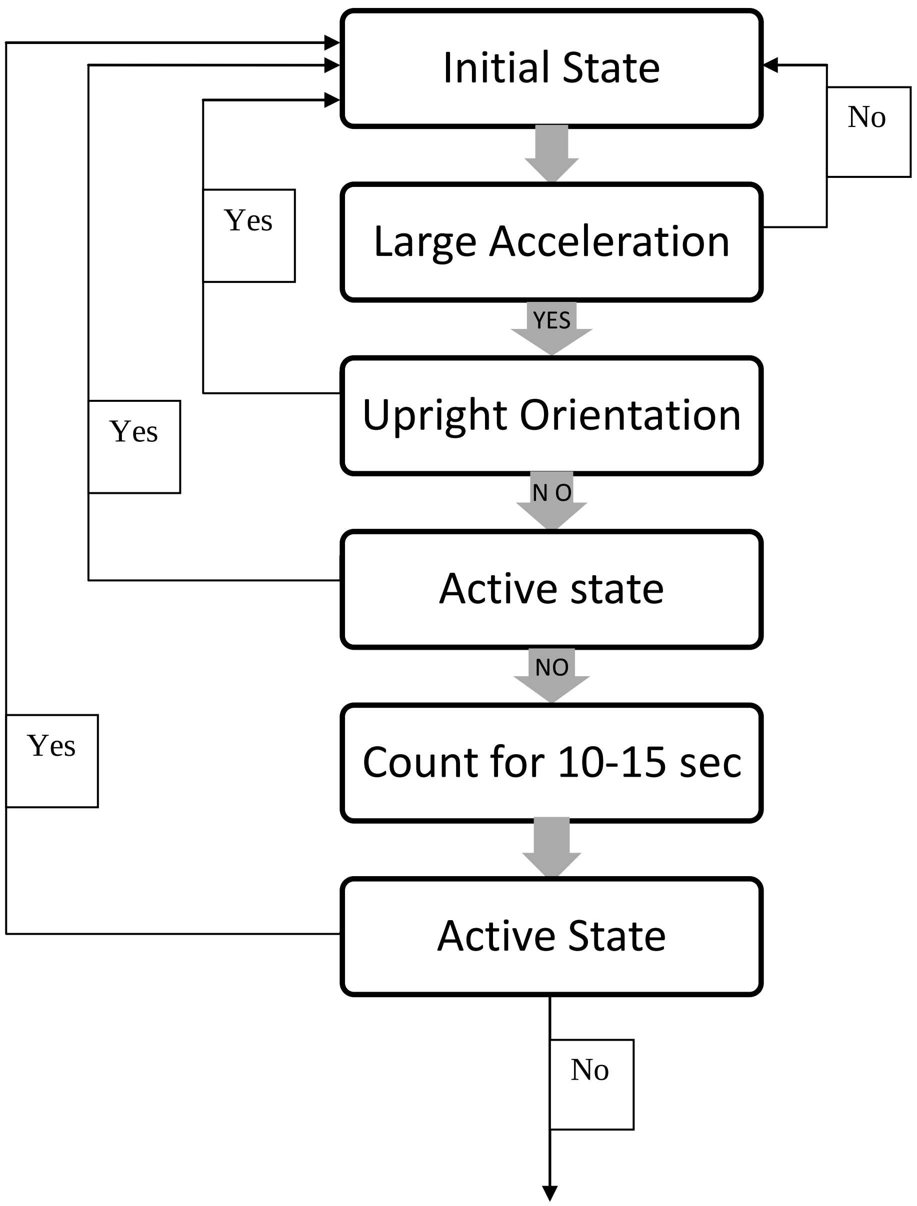
Z

The different plots show the acceleration values converted through the ADC of the MSP430 and displayed on MATLAB through the serial port. Observe the Y axis plot to see the pattern

similar to normal walking and then a sudden jerk or fall. After this fall the person may get up or stay in the horizontal/oblique position. This can be sensed through the following algorithm and processed on the MCU. An alert is sent on the completion of the algorithm depending upon the state of the person.
A median filter has been applied to smooth the data. A high pass filter with very small cut-off frequency can be used to remove noise floor from the output of the accelerometer.

Calibration in a real life scenario would be needed to ascertain the fall from false alarms in a more robust manner. A fall at different places would create different acceleration graphs. So various thresholds and time intervals have to be set to distinguish a normal chore from a fall.

8. Future Implementation Idea



9. Conclusions

Thus, our prototype is able to do most of the functions we intended to have in our project. The next stage would include implementation of the new state diagram as shown above as also the full integration of the geo-fencing algorithm with the broader code implementing the entire device.

10. References

Texas Instruments Data-sheets:

(A) MSP 430F149

(B) TPS 79133

(C) TPS7815

(D) BQ2023

National Semiconductors Datasheets

LMX9838

Free-scale Semiconductor Datasheets

MMQTA7260