**PSUEDOCODE**

#include <AltSoftSerial.h>

#include <TinyGPS++.h>

#include <Wire.h>

#include <SoftwareSerial.h>

const String EMERGENCY\_PHONE = "+919110243131";

#define rxPin 2

#define txPin 3

SoftwareSerial sim800(rxPin, txPin);

AltSoftSerial neogps;

TinyGPSPlus gps;

#define xPin A1

#define yPin A2

#define zPin A3

int xaxis = 0, yaxis = 0, zaxis = 0;

int vibration = 2;

int devibrate = 75;

int sensitivity = 20;

boolean impact\_detected = false;

unsigned long impact\_time;

unsigned long alert\_delay = 30000;

String latitude, longitude;

// Function prototypes

void makeCall();

void sendAlert();

void sendSms(String text);

void impact();

void setup() {

Serial.begin(9600);

sim800.begin(9600);

neogps.begin(9600);

// Initialize GSM module

sim800.println("AT");

delay(1000);

sim800.println("ATE1"); // Enable echo

delay(1000);

sim800.println("AT+CPIN?");

delay(1000);

sim800.println("AT+CMGF=1"); // Set SMS mode to text

delay(1000);

sim800.println("AT+CNMI=1,1,0,0,0");

delay(1000);

// Initialize accelerometer

xaxis = analogRead(xPin);

yaxis = analogRead(yPin);

zaxis = analogRead(zPin);

}

void loop() {

if (micros() - impact\_time > 1999) impact();

if (impact\_detected) {

impact\_detected = false;

Serial.println("Impact detected!!");

Serial.print("Magnitude:");

Serial.println(sqrt(sq(xaxis) + sq(yaxis) + sq(zaxis)));

getGps();

makeCall();

delay(1000);

sendAlert();

}

while (sim800.available()) {

parseData(sim800.readString());

}

while (Serial.available()) {

sim800.println(Serial.readString());

}

}

void impact() {

impact\_time = micros();

int oldx = xaxis;

int oldy = yaxis;

int oldz = zaxis;

xaxis = analogRead(xPin);

yaxis = analogRead(yPin);

zaxis = analogRead(zPin);

vibration--;

Serial.print("Vibration = ");

Serial.println(vibration);

if (vibration < 0) vibration = 0;

if (vibration > 0) return;

int deltx = xaxis - oldx;

int delty = yaxis - oldy;

int deltz = zaxis - oldz;

int magnitude = sqrt(sq(deltx) + sq(delty) + sq(deltz));

if (magnitude >= sensitivity) {

impact\_detected = true;

vibration = devibrate;

} else {

magnitude = 0;

}

}

void parseData(String buff) {

Serial.println(buff);

unsigned int len, index;

index = buff.indexOf("\r");

buff.remove(0, index + 2);

buff.trim();

if (buff != "OK") {

index = buff.indexOf(":");

String cmd = buff.substring(0, index);

cmd.trim();

buff.remove(0, index + 2);

if (cmd == "+CMTI") {

index = buff.indexOf(",");

String temp = buff.substring(index + 1, buff.length());

temp = "AT+CMGR=" + temp + "\r";

sim800.println(temp);

} else if (cmd == "+CMGR") {

if (buff.indexOf(EMERGENCY\_PHONE) > 1) {

buff.toLowerCase();

if (buff.indexOf("get gps") > 1) {

getGps();

String sms\_data = "Accident Alert!!\r";

sms\_data += "http://maps.google.com/maps?q=loc:";

sms\_data += latitude + "," + longitude;

sendSms(sms\_data);

}

}

}

}

}

void getGps() {

// Can take up to 60 seconds

boolean newData = false;

for (unsigned long start = millis(); millis() - start < 2000;) {

while (neogps.available()) {

if (gps.encode(neogps.read())) {

newData = true;

break;

}

}

}

if (newData) {

latitude = String(gps.location.lat(), 6);

longitude = String(gps.location.lng(), 6);

newData = false;

} else {

Serial.println("GPS data is not available");

latitude = "";

longitude = "";

}

Serial.print("Latitude= ");

Serial.println(latitude);

Serial.print("Longitude= ");

Serial.println(longitude);

}

void sendAlert() {

String sms\_data = "Accident Alert!!\r";

sms\_data += "";

sms\_data += latitude + "," + longitude;

sendSms(sms\_data);

}

void makeCall() {

Serial.println("calling....");

Serial.println("Sending SMS....");

sim800.println("ATD" + EMERGENCY\_PHONE + ";");

delay(20000); // 20 sec delay

sim800.println("ATH");

delay(1000); // 1 sec delay

}

void sendSms(String text) {

sim800.println("AT+CMGF=1");

delay(1000);

sim800.print("AT+CMGS=\"" + EMERGENCY\_PHONE + "\"\r");

delay(1000);

sim800.print(text);

delay(100);

sim800.write(0x1A);

delay(1000);

Serial.println("Waiting for SMS response...");

while (sim800.available()) {

Serial.write(sim800.read());

}

Serial.println("SMS Sent Successfully.");

}