

## GPS detection (latitude & longitude):-

```
#include<SoftwareSerial.h>
```

```
SoftwareSerial Serial1(2,3); //make RX arduino line is pin 2, make TX arduino line is pin 3.
```

```
SoftwareSerial gps(10,11);
```

```
#include<LiquidCrystal.h>
```

```
LiquidCrystal lcd(4,5,6,7,8,9);
```

```
#define x A1
```

```
#define y A2
```

```
#define z A3
```

```
int xsample=0;
```

```
int ysample=0;
```

```
int zsample=0;
```

```
#define samples 10
```

```
#define minVal -50
```

```
#define MaxVal 50
```

```
int i=0,k=0;
```

```
int gps_status=0;
```

```
float latitude=0;
```

```
float logitude=0;
```

```
String Speed="";
```

```
String gpsString="";
```

```
char *test="$GPRMC";
```

```
void initModule(String cmd, char *res, int t)
```

```
{
```

```
while(1)
{
    Serial.println(cmd);
    Serial1.println(cmd);
    delay(100);
    while(Serial1.available()>0)
    {
        if(Serial1.find(res))
        {
            Serial.println(res);
            delay(t);
            return;
        }

        else
        {
            Serial.println("Error");
        }
    }
    delay(t);
}
```

```
void setup()
{
    Serial1.begin(9600);
    Serial.begin(9600);
    lcd.begin(16,2);
    lcd.print("Accident Alert ");
    lcd.setCursor(0,1);
    lcd.print("  System  ");
}
```

```
delay(2000);  
  
lcd.clear();  
  
lcd.print("Initializing");  
  
lcd.setCursor(0,1);  
  
lcd.print("Please Wait...");  
  
delay(1000);
```

```
  
Serial.println("Initializing....");  
  
initModule("AT","OK",1000);  
  
initModule("ATE1","OK",1000);  
  
initModule("AT+CPIN?","READY",1000);  
  
initModule("AT+CMGF=1","OK",1000);  
  
initModule("AT+CNMI=2,2,0,0,0","OK",1000);  
  
Serial.println("Initialized Successfully");  
  
lcd.clear();  
  
lcd.print("Initialized");  
  
lcd.setCursor(0,1);  
  
lcd.print("Successfully");  
  
delay(2000);  
  
lcd.clear();  
  
lcd.print("Callibrating ");  
  
lcd.setCursor(0,1);  
  
lcd.print("Acceleromiter");  
  
for(int i=0;i<samples;i++)  
{  
  
    xsample+=analogRead(x);  
  
    ysample+=analogRead(y);  
  
    zsample+=analogRead(z);  
  
}  
  
  
xsample/=samples;
```

```
ysample/=samples;
```

```
zsample/=samples;
```

```
Serial.println(xsample);
```

```
Serial.println(ysample);
```

```
Serial.println(zsample);
```

```
delay(1000);
```

```
Lcd.clear();
```

```
Lcd.print("Waiting For GPS");
```

```
Lcd.setCursor(0,1);
```

```
Lcd.print("  Signal  ");
```

```
delay(2000);
```

```
gps.begin(9600);
```

```
get_gps();
```

```
show_coordinate();
```

```
delay(2000);
```

```
Lcd.clear();
```

```
Lcd.print("GPS is Ready");
```

```
delay(1000);
```

```
Lcd.clear();
```

```
Lcd.print("System Ready");
```

```
Serial.println("System Ready..");
```

```
}
```

```
void loop()
```

```
{
```

```
    int value1=analogRead(x);
```

```
    int value2=analogRead(y);
```

```
    int value3=analogRead(z);
```

```
int xValue=xsample-value1;

int yValue=ysample-value2;

int zValue=zsampl-value3;
```

```
Serial.print("x=");

Serial.println(xValue);

Serial.print("y=");

Serial.println(yValue);

Serial.print("z=");

Serial.println(zValue);
```

```
if(xValue < minVal || xValue > MaxVal || yValue < minVal || yValue > MaxVal || zValue < minVal
|| zValue > MaxVal)
```

```
{

    get_gps();

    show_coordinate();

    lcd.clear();

    lcd.print("Sending SMS ");

    Serial.println("Sending SMS");

    Send();

    Serial.println("SMS Sent");

    delay(2000);

    lcd.clear();

    lcd.print("System Ready");

}

}
```

```
void gpsEvent()

{

    gpsString="";

    while(1)
```

```

{
while (gps.available(>0)      //Serial incoming data from GPS
{
char inChar = (char)gps.read();
gpsString+= inChar;          //store incoming data from GPS to temporary string str[]
i++;
// Serial.print(inChar);
if (i < 7)
{
if(gpsString[i-1] != test[i-1])    //check for right string
{
i=0;
gpsString="";
}
}
if(inChar=='\r')
{
if(i>60)
{
gps_status=1;
break;
}
else
{
i=0;
}
}
}
if(gps_status)
break;
}

```

```
}
```

```
void get_gps()
```

```
{
```

```
  lcd.clear();
```

```
  lcd.print("Getting GPS Data");
```

```
  lcd.setCursor(0,1);
```

```
  lcd.print("Please Wait.....");
```

```
  gps_status=0;
```

```
  int x=0;
```

```
  while(gps_status==0)
```

```
  {
```

```
    gpsEvent();
```

```
    int str_lenth=i;
```

```
    coordinate2dec();
```

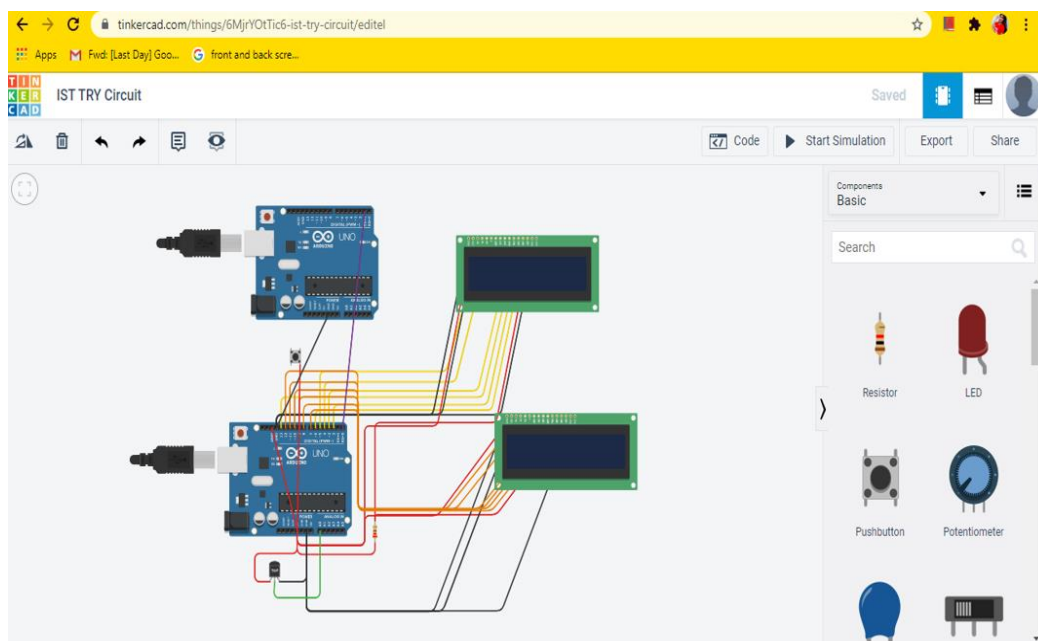
```
    i=0;x=0;
```

```
    str_lenth=0;
```

```
  }
```

```
}
```

## OUTPUT:-



## Code for Accelerometer:

```
void setup()
{
  Serial.begin(9600);
  pinMode(12,OUTPUT);
  pinMode(12,INPUT);
  pinMode(2,INPUT);
  pinMode(6,OUTPUT);
  attachInterrupt(0,function,RISING);
  sei();
}

float atotal;

void loop()
{
  int x = analogRead(A0);
  int y = analogRead(A1);
  int z = analogRead(A2);
  float vx = (5.0*x)/1023.0;
  float vy = (5.0*y)/1023.0;
  float vz = (5.0*z)/1023.0;
  float gx = (vx-1.65)/0.206;
  float gy = (vy-1.65)/0.206;
  float gz = (vz-1.65)/0.206;
  float atotal = sqrt(gx*gx + gy*gy + gz*gz);

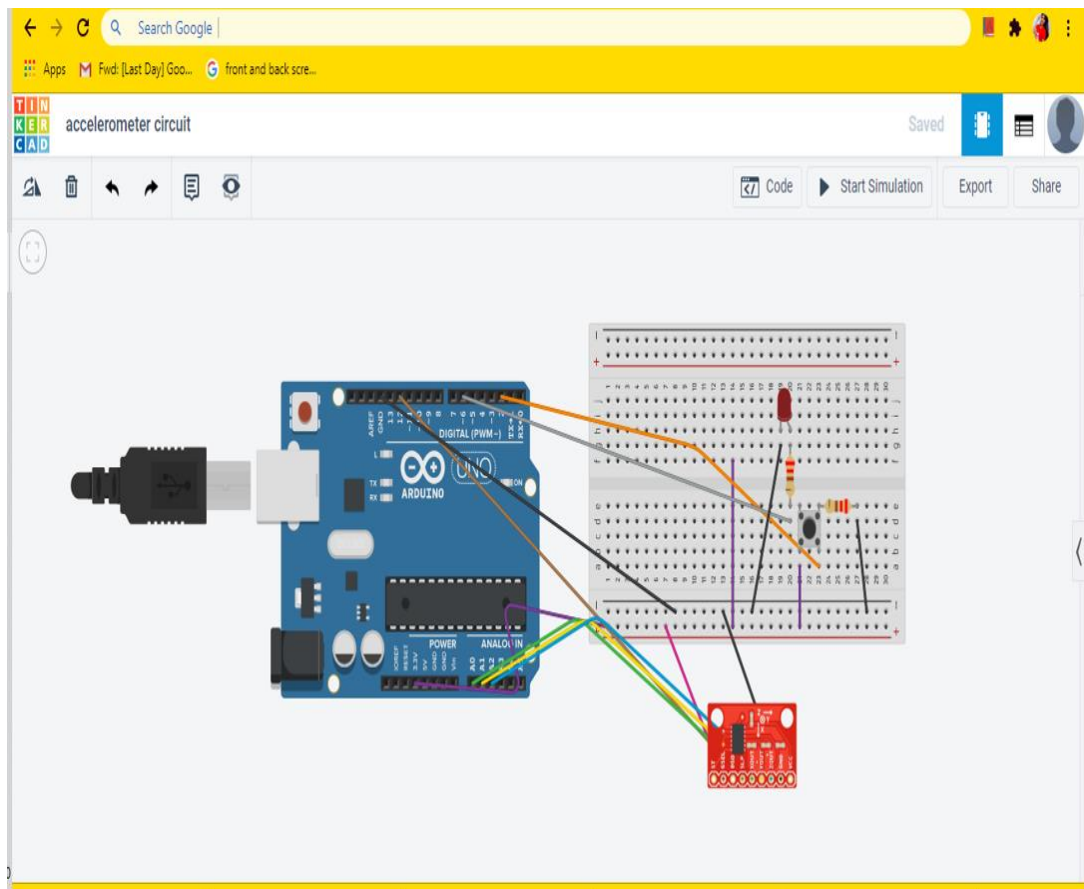
  Serial.print("x: ");
  Serial.print(x);
  Serial.print(", voltx: ");
```



```
Serial.print(vx);  
Serial.print(" , g pres x: ");  
Serial.print(gx);  
Serial.print(" ||y: ");  
Serial.print(y);  
Serial.print(" , volty: ");  
Serial.print(vy);  
Serial.print(" , g pres y: ");  
Serial.print(gy);  
Serial.print(" ||z: ");  
Serial.print(z);  
Serial.print(" , voltz: ");  
Serial.print(vz);  
Serial.print(" , g pres z: ");  
Serial.println(gx);
```

```
if(atotal > 1.4){  
    digitalWrite(6,HIGH);  
}  
if(digitalRead(12)==HIGH){  
    Serial.print("Aceleracion Total: ");  
    Serial.println(atotal);  
}  
delay(100);  
}  
void funcion(){  
    if(digitalRead(12)== HIGH){  
        digitalWrite(12,LOW);  
    }else{  
        digitalWrite(12,HIGH);  
    } }  
}
```

## OUTPUT:-



## Code for Buzzer :

```
void setup() {  
  pinMode(7, OUTPUT);  
}
```

```
void loop(){  
  tone(7, 220, 100);  
  delay(200);  
}
```

## **OUTPUT:-**

