

```
/* THE CIRCUIT
```

```
  UltrasoundSensor  Arduino  LED
  VCC                +5V/+3.3V  -
  Gnd                Gnd        -
  Trig               9          -
  Echo               4          -
  -                 11         Long lead of Led1
  -                 12         Long lead of Led2
  -                 13         Long lead of Led3
  -                 Gnd        Smaller lead of all LEDs
```

```
*/
```

```
//u may need a breadboard for the above circuit
```

```
//u may choose to connect a resistor
```

```
//across all the LEDs (i have not chosen to do so)
```

```
int Trig= 9;//could connect to any- i have chosen 9
```

```
//connected to trigger pin on UltraSound sensor
```

```
int Led1=11;
```

```
int Led2= 12;
```

```
int Led3= 13;
```

```
/*Led Connected to these pin -
```

```
LED3 connected to 13 - to get blink on startup
```

```
other leds are connected randomly -
```

```
u may have to change the numbers
```

```
based on the pins u wish to use
```

```
*/
```

```
int Echo= 4;//PulseIn pin
unsigned long echo = 4;
unsigned long final = 4;
//declaring terms on pin 4 - used for pulseIn

void setup() {
Serial.begin(9600);/*
this is required if u want data to be
read onto yourr serial Monitor
- more on that, later in the code
*/
pinMode(Led1, OUTPUT);
pinMode(Led2, OUTPUT);
pinMode(Led3, OUTPUT);
pinMode(Trig, OUTPUT);
//defining OUTPUTs on Pins (that are output)
// - all pins are by default, INPUT
}

unsigned long Value() /*defining 'Value'
- note this is not part of void setup()
*/
{
digitalWrite(Trig, LOW); // Send low signal
delayMicroseconds(1); // Wait for 1 microseconds
digitalWrite(Trig, HIGH); // Send high signal
delayMicroseconds(10); /* delay 10 microseconds -
```

to trigger UltraSound Sensor

*/

digitalWrite(Trig, LOW);

//Switch to low again - to complete triggering pulse

echo = pulseIn(Echo, HIGH,8824);

//using timeout of 8824 or 150cm approx

final = (echo*0.017); //dist in cm

return final;

/* delay(500) [or more/less]

could be introduced for slowing down inputs -

may be required for more

complex systems (so as to not

overwhelm the arduino with data)

*/

}

void loop() {

int x;

x = Value();/*defining 'x' variable

in loop() segment.

Defining x as 'Value ()' function defined above

*/

Serial.println(x);

/*Value() commands are run and values

are printed onto the

serial monitor (ctrl+shift+M)

this can be omitted -

i have put this in for checking

reliability of raw Ultrasound values,

and change delay and

other code segments accordingly

*/

if(x<=20 && x>0)

{

digitalWrite(Led1, HIGH);

/*

close proximity alert - led can be replaced by buzzer

(note relay or arduino shield may be required ,

if buzzer is demanding)

*/

digitalWrite(Led3, LOW);

digitalWrite(Led2, LOW);

delay(100);

/*since proximity is close,

new data input should be fast

(closer the proximity -

greater should be the update rate

of the feedback loop)

```
*/
}
if(x<=75 && x>20)
{
digitalWrite(Led1, LOW);
digitalWrite(Led2, HIGH);//medium proximity LED
digitalWrite(Led3, LOW);
delay(200);
/*since proximity is 'medium'
data input is slower
*/

}
if(x>75)
{
digitalWrite(Led1, LOW);
digitalWrite(Led2, LOW);
digitalWrite(Led3, HIGH);//distant obstacle alert
delay(300);
/*data update is slower for distant objects.
Fast update is not required - puts undue stress on system
*/
}

else
{ /*else is used and not another if function
```

to cover timeout '0' values - else function
defines "out of range" and "error" codes

```
*/
```

```
digitalWrite(Led1,LOW);
```

```
digitalWrite(Led2, LOW);
```

```
digitalWrite(Led3, LOW);
```

```
delayMicroseconds(100);
```

```
/*
```

delay is random very small number

- if error there is an error in reading values

faster data input provides more reliability,

and less 'incorrect'/'undesirable' outputs

```
*/
```

```
}
```

```
}
```

```
/* closing brace brackets from void loop()
```

thereby completing all syntax requirements

code is ready to upload

```
*/
```

//pls refer to video - to see observations after carrying out the project.