Arduino code \_Document

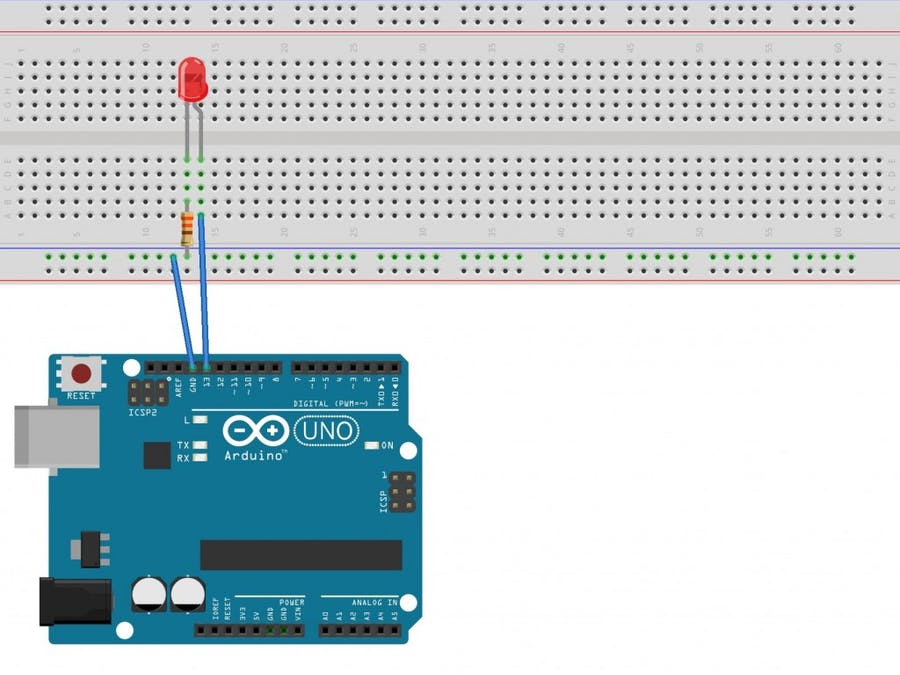
|  |  |  |
| --- | --- | --- |
| **S.no** | **Title** | **Page** |
| 1 | Simple led blink project |  |
| 2 | Multiple led blink project |  |
| 3 | Push button used LED on and off |  |
| 4 | Humidity sensor used value get |  |
| 5 | Water level sensor: |  |
| 6 | Ultra sonic sensor: |  |
| 7 | IR receiver and remote |  |
| 8 | IR receiver and sender |  |
| 9 | MQ2 Sensor |  |
| 10 | PIR Sensor |  |

**CHAPTER 1 : Simple led blink project**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × LED
* 1 × 10k Resistor
* 2 × Jumper

## Procedure



## Code:

**/\***

**\* TamilZorous info tech**

**\*/**

**int led =9;**

**void setup() {**

**pinMode(led, OUTPUT);//pinout**

**}**

**void loop() {**

**digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)**

**delay(1000); // wait for a second**

**digitalWrite(led, LOW); // turn the LED off by making the voltage LOW**

**delay(1000); // wait for a second**

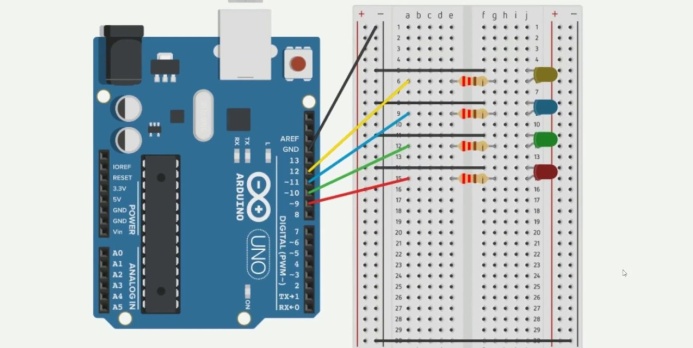
**}**

**CHAPTER 2 : Multiple led blink project:**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 4 × LED
* 4 × 10k Resistor
* 8 × Jumper

## Procedure



**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**int led1 =9;**

**int led2 =10;**

**int led3 =11;**

**int led4 =12;**

**void setup() {**

**Serial.begin(9600);**

**pinMode(led1, OUTPUT);//pinout 9**

**pinMode(led2, OUTPUT);//pinout 10**

**pinMode(led3, OUTPUT);//pinout 11**

**pinMode(led4, OUTPUT);//pinout 12**

**}**

**void loop() {**

**//led1**

**digitalWrite(led1, HIGH); // turn the LED on (HIGH is the voltage level)**

**delay(1000); // wait for a second**

**digitalWrite(led1, LOW); // turn the LED off by making the voltage LOW**

**delay(1000); // wait for a second**

**//led2**

**digitalWrite(led2, HIGH);**

**delay(500);**

**digitalWrite(led2, LOW);**

**delay(500);**

**//led3**

**digitalWrite(led3, HIGH);**

**delay(1000);**

**digitalWrite(led3, LOW);**

**delay(1000);**

**//led4**

**digitalWrite(led4, HIGH);**

**delay(200);**

**digitalWrite(led4, LOW);**

**delay(200);**

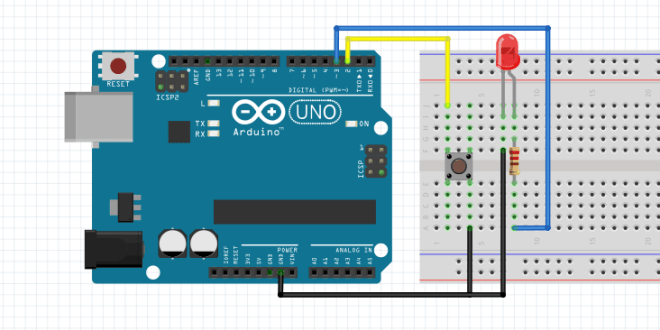
**}**

**CHAPTER 3 : Push button used LED on and off**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × LED
* 1 × 10k Resistor
* 1 × Push button
* 4 × Jumper

## Procedure

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#define led 3**

**#define pushButton 2**

**void setup() {**

**Serial.begin(9600);**

**pinMode(led, OUTPUT);**

**pinMode(pushButton, INPUT);**

**}**

**void loop() {**

**if (digitalRead(pushButton) == HIGH) {**

**digitalWrite(led, HIGH);**

**}**

**else {**

**digitalWrite(led, LOW);**

**}**

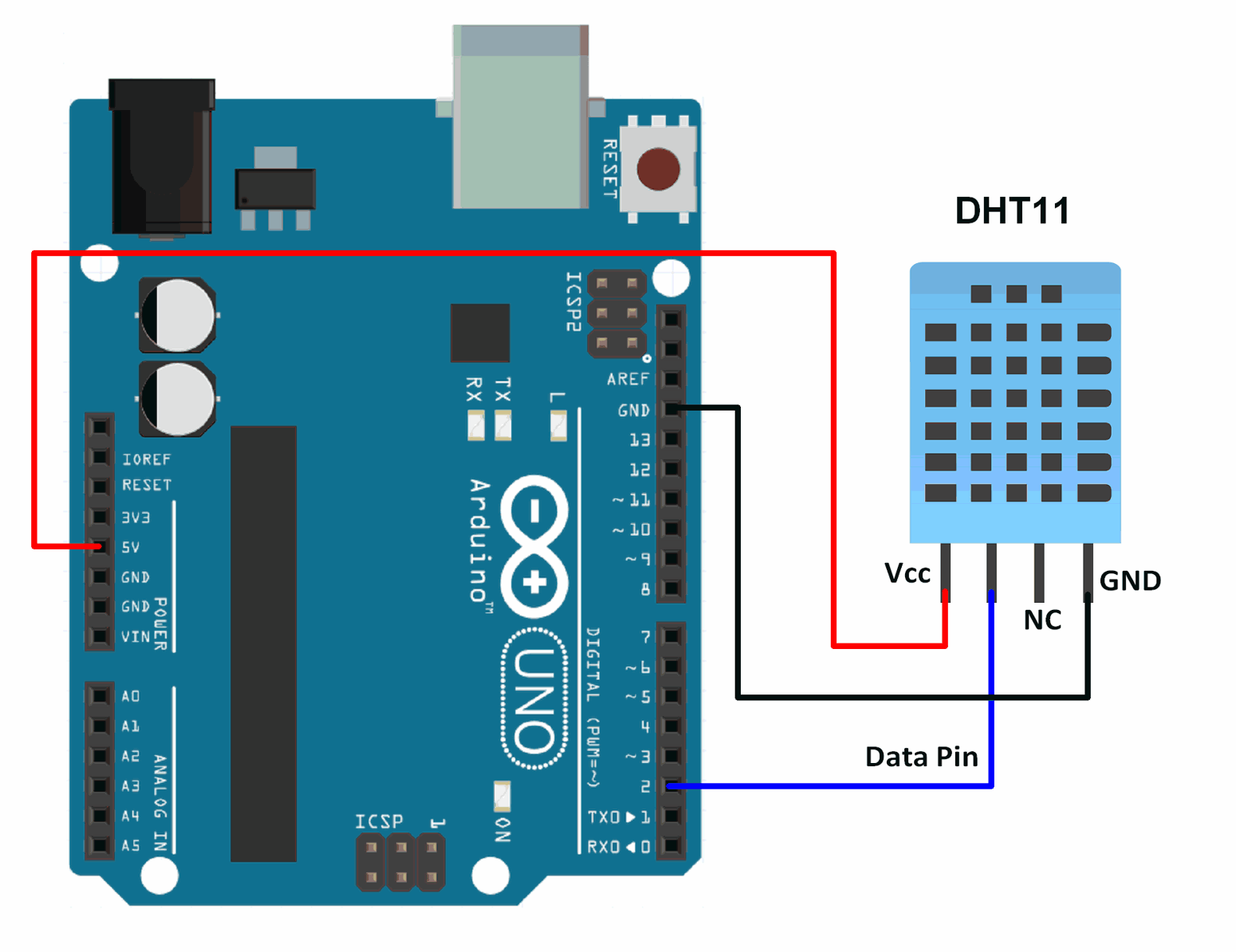
**}**

**CHAPTER 4 : Humidity sensor used value get**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno R3
* 1 × DHT11
* 1 × 10K ohm resistor
* 3 × Jumper

## Procedure:



**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#include "DHT.h"**

**#define DHT11PIN 5 //D1 input**

**#define DHT11TYPE DHT11**

**DHT dht11(DHT11PIN, DHT11TYPE);**

**void setupAP(void);**

**void setup() {**

**Serial.begin(115200); //Serial connection**

**dht11.begin();**

**}**

**void loop() {**

**float h11 = dht11.readHumidity();**

**float t11 = dht11.readTemperature();**

**if (isnan(t11) || isnan(h11)) {**

**Serial.println("Failed to read from DHT #1");**

**} else {**

**Serial.print("Humidity 11: ");**

**Serial.println(h11);**

**//Serial.print(" %\t");**

**Serial.print("Temperature 11: ");**

**Serial.println(t11);**

**Serial.println(" \*C");**

**}**

**delay(1000);**

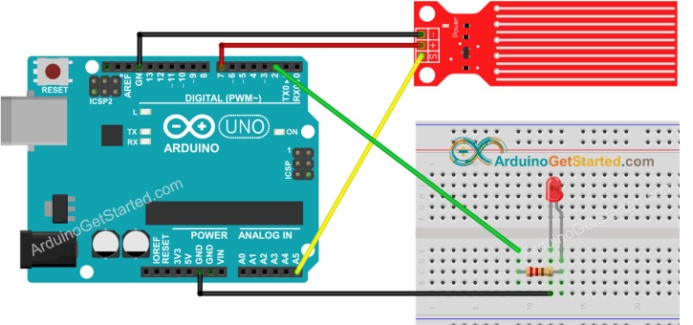
**}**

**CHAPTER 5 : Water level sensor:**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × LED
* 1 × 10k Resistor
* 1 × Water level sensor
* 5 × Jumper

## Procedure:

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#define led 2**

**#define POWER\_PIN 7**

**#define SIGNAL\_PIN A5**

**#define THRESHOLD 300**

**int value = 0; // variable to store the sensor value**

**void setup() {**

**Serial.begin(9600);**

**pinMode(led, OUTPUT); // configure D2 pin as an OUTPUT**

**pinMode(POWER\_PIN, OUTPUT); // configure D7 pin as an OUTPUT**

**digitalWrite(POWER\_PIN, LOW); // turn the sensor OFF**

**digitalWrite(led, LOW); // turn LED OFF**

**}**

**void loop() {**

**digitalWrite(POWER\_PIN, HIGH); // turn the sensor ON**

**delay(10); // wait 10 milliseconds**

**value = analogRead(SIGNAL\_PIN); // read the analog value from sensor**

**digitalWrite(POWER\_PIN, LOW); // turn the sensor OFF**

**if (value > THRESHOLD) {**

**Serial.print("The water is detected");**

**digitalWrite(led, HIGH); // turn LED ON**

**} else {**

**digitalWrite(led, LOW); // turn LED OFF**

**}**

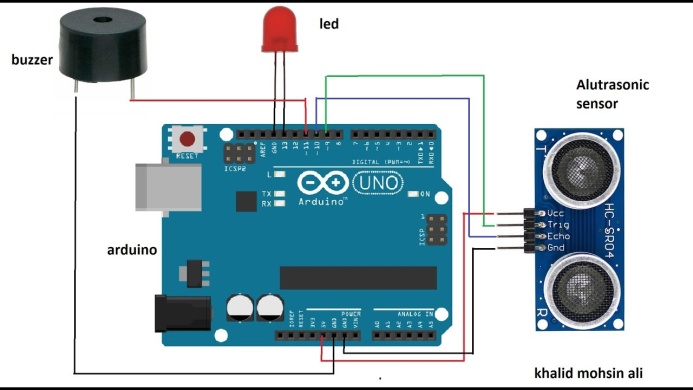
**}**

**CHAPTER 6: Ultra sonic sensor:**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × Buzzer
* 1× led
* 1 × Ultra sonic sensor
* 8 × Jumper

## Procedure:

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#define echoPin 10 // attach pin D3 Arduino to pin Echo of HC-SR04**

**#define trigPin 9 //attach pin D10 Arduino to pin Trig of HC-SR04**

**int buzzer =11;//attach relays in to D5 of Arduino(DC Pump is connected to Arduino)**

**int redLED =13;**

**int duration, distance;**

**int i=0;**

**int maxnum=10;**

**int count=0;**

**void setup() {**

**Serial.begin(9600);**

**pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT**

**pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT**

**pinMode(buzzer, OUTPUT);// sets the relaypin as OUTPUT**

**pinMode(redLED,OUTPUT);**

**}**

**void loop() {**

**// Clears the trigPin condition**

**digitalWrite(trigPin, LOW);**

**delayMicroseconds(2);**

**// Sets the trigPin HIGH (ACTIVE) for 10 microseconds**

**digitalWrite(trigPin, HIGH);**

**delayMicroseconds(10);**

**digitalWrite(trigPin, LOW);**

**// Reads the echoPin, returns the sound wave travel time in microseconds**

**duration = pulseIn(echoPin, HIGH);**

**// Calculating the distance**

**distance = duration \* 0.034 / 2; // Speed of sound wave divided by 2 (go and back)**

**if((distance<10)&&(count<maxnum))**

**{**

**digitalWrite(buzzer, HIGH);//LOW for the relay to be on**

**delay(1000 );//For how many milliseconds your Pump works for pumping the liquid**

**digitalWrite(buzzer, LOW);//HIGH for the relay to be off**

**delay(2500);//For how many milliseconds your Pump NOT works for pumping the liquid**

**count++;**

**Serial.print("distance :");**

**Serial.println( distance);**

**Serial.print("count :");**

**Serial.println( count);**

**}**

**else if(count == 10){**

**digitalWrite(redLED, HIGH);//LOW for the relay to be on**

**delay(1000 );//redLED blink and alart**

**Serial.println( "Alert");**

**digitalWrite(redLED, LOW);//HIGH for the relay to be off**

**delay(500);**

**count=0;**

**//exit(0);**

**}**

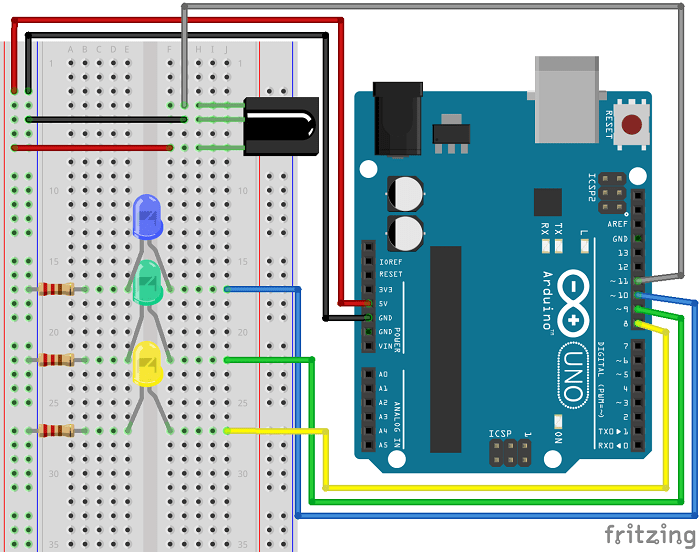
**}**

**CHAPTER 7 :IR receiver and remote :**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × Remote
* 1 × IR receiver
* 3 × 10 k resistor
* 3× led
* 11 × Jumper

## Procedure:

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#include <IRremote.h>**

**int RECV\_PIN = 11;**

**IRrecv irrecv(RECV\_PIN);**

**decode\_results results;**

**void setup()**

**{**

**Serial.begin(9600);**

**irrecv.enableIRIn(); // Start the receiver**

**}**

**void loop() {**

**if (irrecv.decode(&results)) {**

**Serial.println(results.value, HEX);**

**irrecv.resume(); // Receive the next value**

**}**

**delay(100);**

**}**

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**#include <IRremote.h>**

**int IR\_Recv = 11; //IR Receiver Pin 3**

**int bluePin = 10;**

**int greenPin = 9;**

**int yellowPin = 8;**

**IRrecv irrecv(IR\_Recv);**

**decode\_results results;**

**void setup(){**

**Serial.begin(9600); //starts serial communication**

**irrecv.enableIRIn(); // Starts the receiver**

**pinMode(bluePin, OUTPUT); // sets the digital pin as output**

**pinMode(greenPin, OUTPUT); // sets the digital pin as output**

**pinMode(yellowPin, OUTPUT); // sets the digital pin as output**

**}**

**void loop(){**

**//decodes the infrared input**

**if (irrecv.decode(&results)){**

**long int decCode = results.value;**

**Serial.println(results.value);**

**//switch case to use the selected remote control button**

**switch (results.value){**

**case 551520375: //when you press the 1 button**

**digitalWrite(bluePin, HIGH);**

**break;**

**case 551495895: //when you press the 4 button**

**digitalWrite(bluePin, LOW);**

**break;**

**case 551504055: //when you press the 2 button**

**digitalWrite(greenPin, HIGH);**

**break;**

**case 551528535: //when you press the 5 button**

**digitalWrite(greenPin, LOW);**

**break;**

**case 551536695: //when you press the 3 button**

**digitalWrite(yellowPin, HIGH);**

**break;**

**case 551512215: //when you press the 6 button**

**digitalWrite(yellowPin, LOW);**

**break;**

**}**

**irrecv.resume(); // Receives the next value from the button you press**

**}**

**delay(10);**

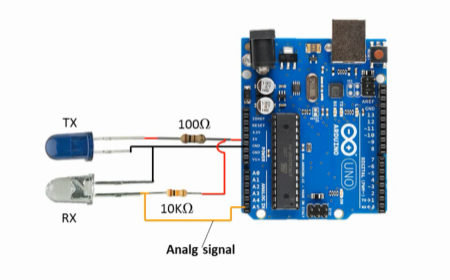
**}**

**CHAPTER 8 :IR receiver and sender :**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × Sender
* 1 × IR receiver
* 1 × 10 k resistor
* 1× led
* 8 × Jumper

## Procedure:



**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**int analogPin = A5;**

**int out = 13;**

**int sensorValue = 0;**

**void setup() {**

**Serial.begin(9600);**

**pinMode(out, OUTPUT);**

**pinMode(analogPin, INPUT);**

**}**

**void loop() {**

**sensorValue = analogRead(analogPin);**

**Serial.print("sensor =");**

**Serial.println(sensorValue);**

**delay(200);**

**if( sensorValue>700)**

**{**

**digitalWrite(out,1);**

**}**

**else**

**{**

**digitalWrite(out,0);**

**}**

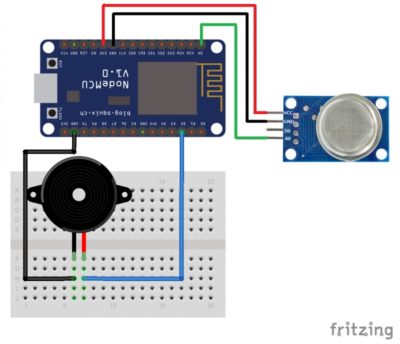
**}**

**CHAPTER 9 :MQ2 Sensor :**

## Components Required

* 1 × Breadboard
* 1 × NodeMCU
* 1 × MQ2
* 1 × Buzzer
* 5 × Jumper

## Procedure:

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**int buzzer = D2;**

**int smokeA0 = A0;**

**// Your threshold value. You might need to change it.**

**int sensorThres = 600;**

**void setup() {**

**pinMode(buzzer, OUTPUT);**

**pinMode(smokeA0, INPUT);**

**Serial.begin(9600);**

**}**

**void loop() {**

**int analogSensor = analogRead(smokeA0);**

**Serial.print("Pin A0: ");**

**Serial.println(analogSensor);**

**// Checks if it has reached the threshold value**

**if (analogSensor > sensorThres)**

**{**

**tone(buzzer, 1000, 200);**

**}**

**else**

**{**

**noTone(buzzer);**

**}**

**delay(100);**

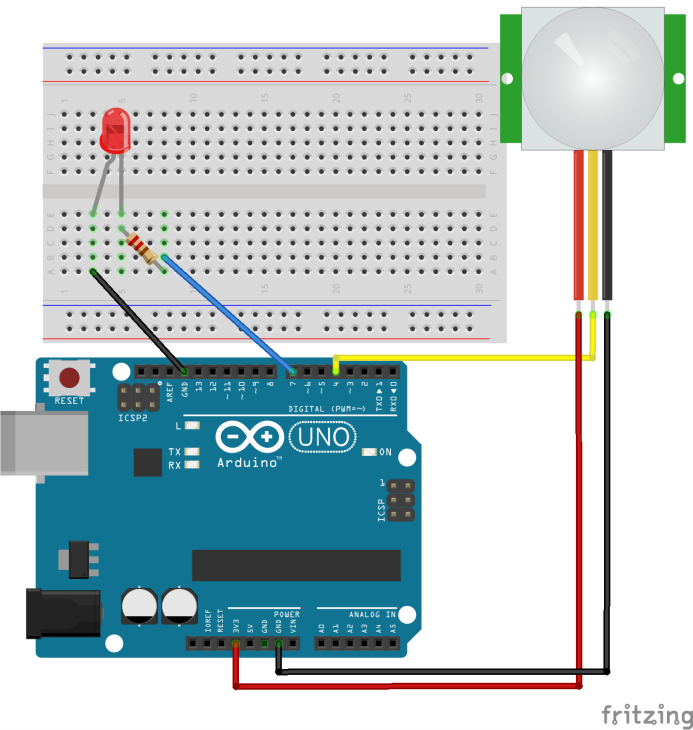
**}**

**CHAPTER 10 : PIR Sensor**

## Components Required

* 1 × Breadboard
* 1 × Arduino Uno
* 1 × PIR Sensor
* 1 × LED
* 1 × 10k Resistor
* 5 × Jumper

## Procedure

****

**Code:**

**/\***

**\* TamilZorous info tech**

**\*/**

**int led = 13; // the pin that the LED is atteched to**

**int sensor = 2; // the pin that the sensor is atteched to**

**int state = LOW; // by default, no motion detected**

**int val = 0; // variable to store the sensor status (value)**

**void setup() {**

**pinMode(led, OUTPUT); // initalize LED as an output**

**pinMode(sensor, INPUT); // initialize sensor as an input**

**Serial.begin(9600); // initialize serial**

**}**

**void loop(){**

**val = digitalRead(sensor); // read sensor value**

**if (val == HIGH) { // check if the sensor is HIGH**

**digitalWrite(led, HIGH); // turn LED ON**

**delay(500);**

**if (state == LOW) {**

**Serial.println("Motion detected!");**

**state = HIGH; // update variable state to HIGH**

**}**

**}**

**else {**

**digitalWrite(led, LOW); // turn LED OFF**

**delay(500); // delay 200 milliseconds**

**if (state == HIGH){**

**Serial.println("Motion stopped!");**

**state = LOW; // update variable state to LOW**

**}**

**}**

**}**