# Top 15 Sensor Types Being Used in IoT

Industries and organizations have been using various kind of sensors for a long time but the invent of The internet of things has taken role of sensors and evolutions of sensors to a completely different level.

IoT platforms function and deliver valorous kind of intelligence and data using a variety of sensors. They serve to collect data, pushing it and sharing it with a whole network of connected devices. All this collected data makes it possible for devices to autonomously function, and the whole ecosystem is becoming

“smarter” every day.

By combining a set of sensors and a communication network, devices share information with one another and are improving their effectiveness and functionality.

Take Tesla vehicles as an example. All of the sensors on a car record their perception of the surroundings, uploading the information into a massive database. The data is then processed and all the important new pieces of information are sent to all other vehicles. This is an ongoing process, through which a whole fleet of Tesla vehicles is becoming smarter every day.

Let’s take a look at some of the key sensors, extensively being used in the IoT world.

### **Temperature sensors**

These sensors have been deployed for a long time in variety of devices. However, with emergence of IoT, they have found more room to be present in an even greater number of devices. Only a couple of years ago, their uses mostly included A/C control, refrigerators and similar devices used for environment control.

However, with the advent of the IoT world, they have found their role in manufacturing processes, agriculture and health industry. In the manufacturing process, many machines require specific environment temperature, as well as device temperature. With this kind of measurement, the manufacturing process can always remain optimal.

On the other hand, in agriculture, the temperature of soil is crucial for crop growth. This helps with the production of plants, maximizing the output.

In the health industry, many experiments, and drug storage temperature conditions have to be respected. With these sensors, maintenance is made much easier, making following the protocols a much simpler endeavor.

### **Proximity sensor**

The car industry is rapidly developing, working on making it possible for vehicles to drive you and not the other way around. This is made possible by an abundance of sensors, some of which we are going to talk about in the passages below. One of the sensors that are quite useful include proximity sensors.

In combination with all the other sensors on the vehicle, it ensures that the vehicle remains away from any obstacles from all sides of the vehicle.

However, it is important to note that this is only one of its applications. Proximity sensors are largely used in the retail industry, as they can detect motion and the correlation between the customer and product they might be interested in. A user is immediately notified about discounts and special offers of nearby products.

They are also used for parking availability in big places such as malls, stadiums or airports.

### **Pressure sensor**

There are plenty of devices that rely on liquid or other forms of pressure. Also, whole systems involve pressure for proper function. These sensors make it possible to create IoT systems that monitor systems and devices that are pressure propelled. Once a drop in pressure occurs, the device notifies the system administrator about any problems that should be fixed.

Deployment of these sensors is very useful in manufacturing, but also in maintenance of whole water systems and heating systems, as it is easy to detect any fluctuation or drops in pressure.

### **Water quality sensor**

Water is practically used everywhere. These sensors play an important role as they monitor the quality of water for different purposes. They are used in a variety of industries.

### **Chemical sensor**

Chemical sensors are applied in a number of different industries. Their goal is to indicate changes in liquid or air chemical changes. They play an important role in bigger cities, where it is necessary to track changes and protect the population.

### **Gas sensor**

Gas sensors are similar to the chemical ones, but are specifically used to monitor changes of the air quality and detect the presence of various gases. Like chemical sensors, they are used in numerous industries such as manufacturing, agriculture and health.

### **Smoke sensor**

Smoke sensors have been in use for a long period of time. However, with the development of IoT, they are now even more effective, as they are plugged into a system that immediately notifies the user about any problem that occurs in different industries.

Beside the accommodation industry, these sensors are used in manufacturing, where there is a high risk of fires. This serves to protect people working in dangerous environments, as the whole system is much more effective in comparison to the older ones.

### **IR sensors**

These sensors have entered the health industry gradually. They are now used in a variety of IoT projects, as they make monitoring of blood flow and blood pressure simple. They are even used in a wide array of regular smart devices such as smartwatches and smartphones.

However, their usage does not end there, they are also a great tool for ensuring high level security in your home. Also, their application includes environment checks, as they can detect a variety of chemicals and heat leaks. They are going to play an important role in the smart home industry, as they have a wide-range of applications.

### **Level sensors**

Like IR sensors, level sensors are present in wide array of industries. They are primarily known for measuring fuel levels, but they are also used in businesses that work with liquid materials. For example, the recycling industry, as well as the juice and alcohol industry rely on these sensors to measure the amount of liquid assets in their possession.

This helps better streamline their businesses, as sensors collect all the important data at all times. With the use of these sensors, any product manager can precisely see how much liquid is ready to be distributed and whether the manufacturing should be stepped up.

### **Image sensors**

An average consumer would think that this is a regular camera, but even though this is not far from the truth, image sensors are connected with a wide range of different devices, making their functionality much better.

One of the best known uses includes the car industry, in which imagery plays a very important role. With these sensors, the system can recognize signs, obstacles and many other things that a driver would generally notice on the road. They play a very important role in IoT industry, as they directly affect the progress of driverless cars.

They are also implemented in improved security systems, where images help capture details about the perpetrator.

In the retail industry, these sensors serve to collect data about customers, helping businesses get a better insight into who is actually visiting their store, race, gender, age are only some of useful parameters that retail owners get by using these IoT sensors.

### **Motion detection sensors**

Motion detection plays an important role in the security industry. Businesses utilize these sensors in areas where no movement should be detected at all times, and it is easy to notice anybody’s presence with thee sensors installed.

On the other hand, these sensors can also decipher different types of movements, making them useful in some industries where a customer can communicate with the system by waving a hand or by performing a similar action. For example, someone can wave to a sensor in the retail store to request assistance with making the right purchase decision.

Even though their primary use is correlated with the security industry, as the technology advances, the number of possible applications of these sensors is only going to grow.

### **Accelerometer sensors**

These sensors are now present in millions of devices, such a smartphones. Their uses involve detection of vibrations, tilting and acceleration in general. This is great for monitoring your driving fleet, or using a smart pedometer. In some instances, it is used as a form of anti-theft protection, as the sensor can send an alert through the system if an object that should remain stationary is moved.

### **Gyroscope sensors**

These sensors are always combined with accelerometers. The use of these two sensors simply provides more feedback to the system. WIth gyroscopic sensors installed, many devices can help athletes improve the efficiency of their movements, as they gain access to the athletes movement during sports activities.

This is only one example of its application, however, as the role of this sensor is to detect rotation or twist, its application is crucial for the automation of some manufacturing processes.

### **Humidity sensors**

These sensors usually follow the use of temperature sensors, as many manufacturing processes require perfect working conditions. Through measuring humidity, you can ensure that the whole process runs smoothly, and when there is any sudden change, action can be taken immediately, as sensors detect the change almost instantaneously.

### **Optical sensors**

Optical sensors are loved by IoT experts, as they are practical for measuring different things simultaneously. The technology behind this sensor allows it to monitor electromagnetic energy, which includes, electricity, light and so on.

Due to this fact, these sensors have found use in healthcare, environment monitoring, energy, aerospace and many more industries. With their presence oil companies, pharmaceutical companies and mining companies are in a much better position to track environmental changes while keeping their employees safe.

It is clear that IoT has become incredibly popular, and current trends show that it is the future. It simply helps with automation of various processes, making these systems quite useful for both regular consumers and businesses.

# Switch ON/OFF LED Using ARDUINO UNO and C#

Form Applications

#### Introduction

This practical is for beginners who are interested in the Internet of Things. This practical demonstrates how to create two buttons for switching the LED light ON and OFF, with simple Arduino and C# code.

#### Requirements

1. Arduino Uno R3 + Arduino data cable I already have this broad
2. USB 2.0 Male to B Male cable
3. Arduino Uno R3 Software I have also downloaded
4. One LED light “5MM”
5. 300 Ohm resister
6. Jumper wire only two (Male to Male)
7. Bread broad
8. Visual Studio Any versions

#### Follow my steps

Arduino is available in several versions online but you only have to buy Arduino Uno R3.

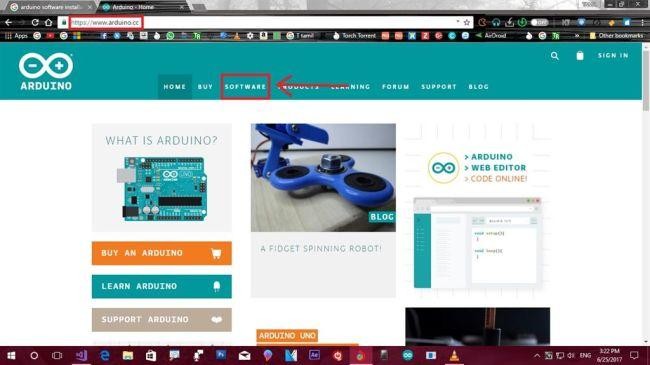
#### Step 1

First connect Arduino from your computer if the flash light is ON it's successfully connected

#### Step 2

Go to any browser type this link *https://*[*www.arduino.cc/*](http://www.arduino.cc/)

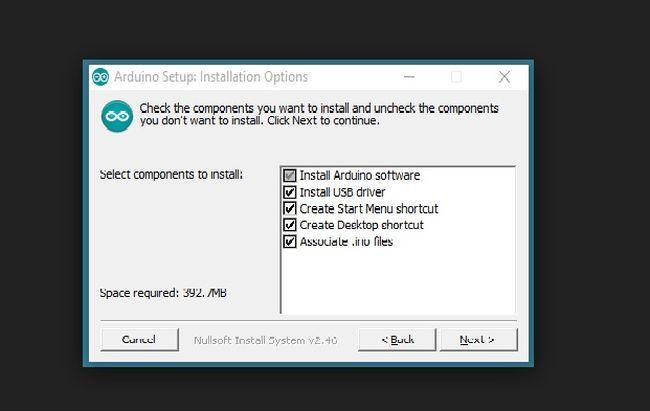
Scroll down on windows installer click to download 96MB Software

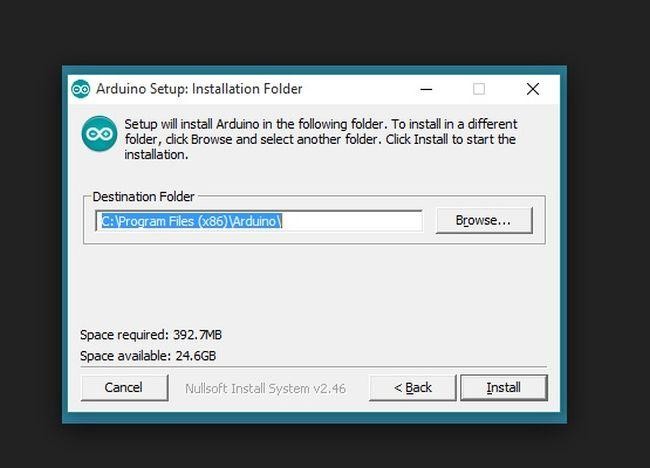
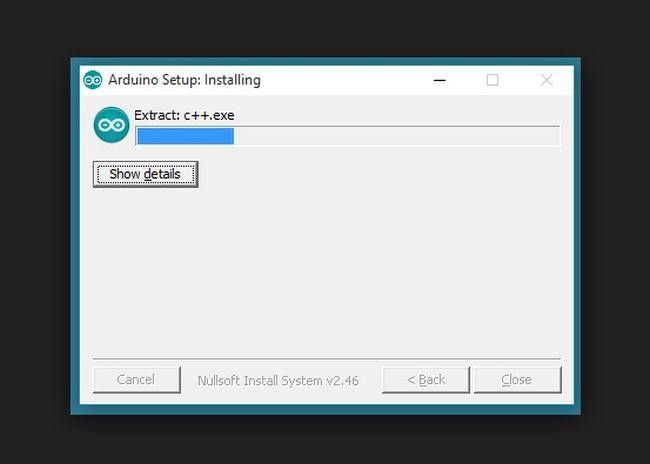




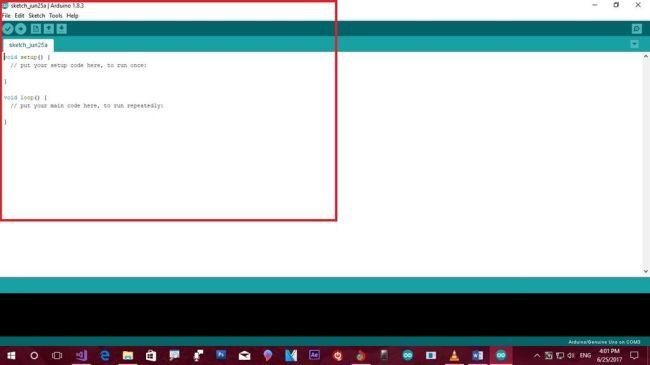
#### Step 3

Download the software



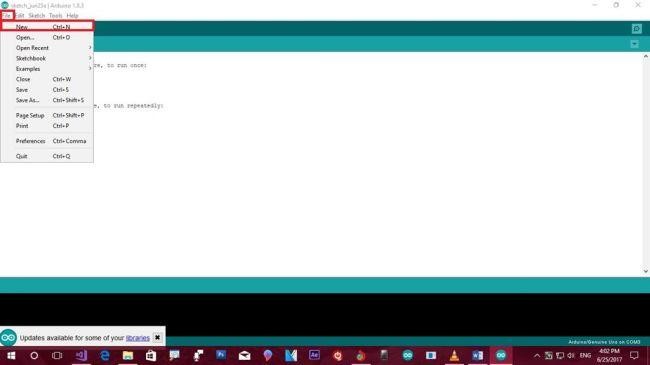


#### Step 4

Then after installation completes this page appears 8

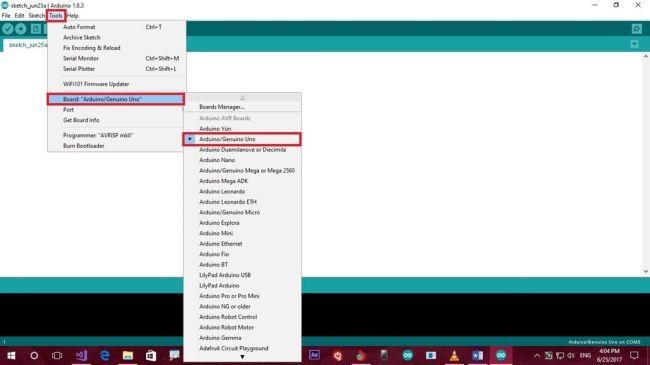
#### Step 5

Then select FILL NEW . This will open a new page



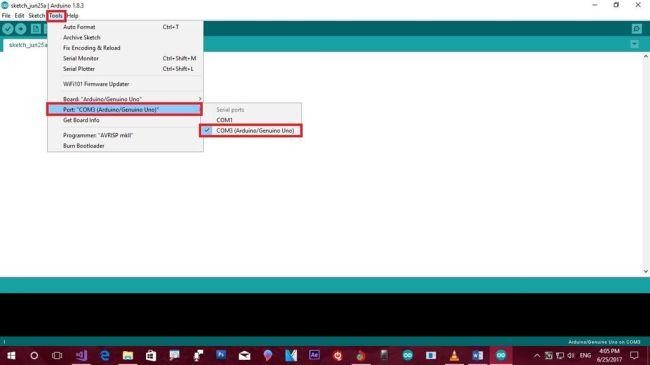
#### Step 6

Then Click Tool box Selected TOOL BROAD ARDUINO/GENUINO UNO



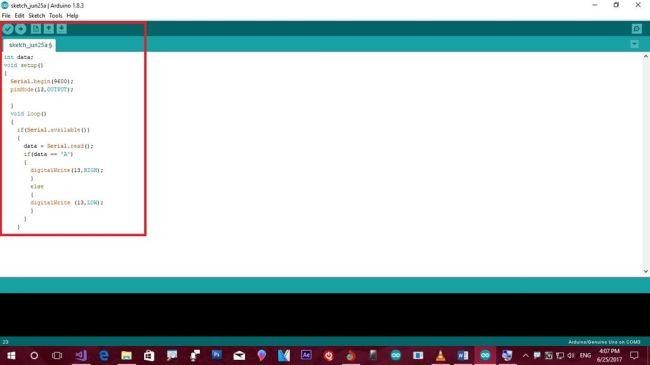
#### Step 7

Then select Tool Box TOOL PORT COM3 ARDUINO UNO



#### Step 8

Given source code type our Arduino Software



#### source code

1. int data;
2. void setup() {
3. Serial.begin(9600);
4. pinMode(13, OUTPUT); 5. }
5. void loop() {
6. if (Serial.available()) {
7. data = Serial.read();
8. if (data == 'A') {
9. digitalWrite(13, HIGH);
10. } else {
11. digitalWrite(13, LOW);

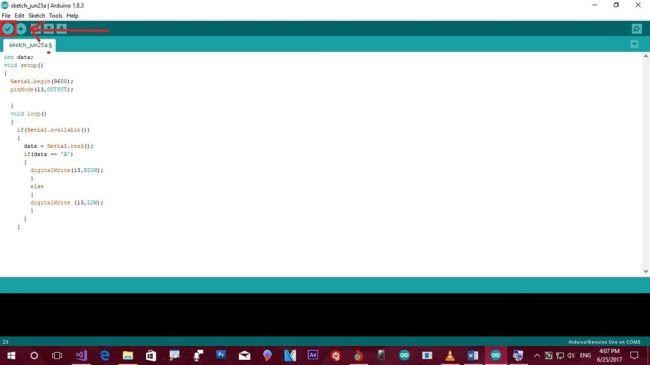
13. }

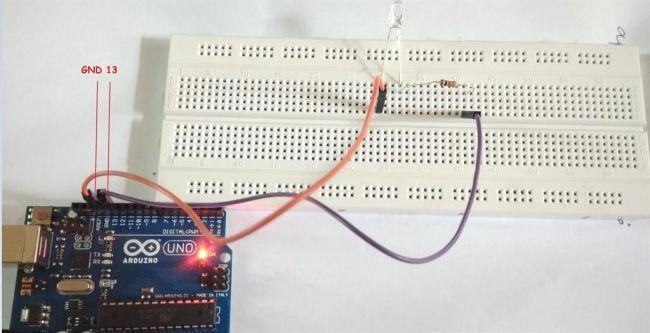
14. }

15. }

#### Step 9

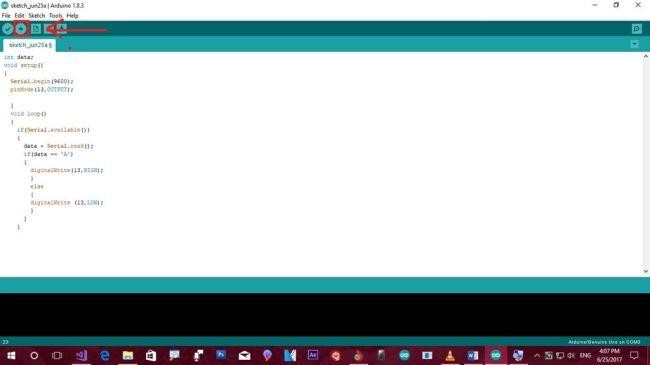
Compiler process is successful





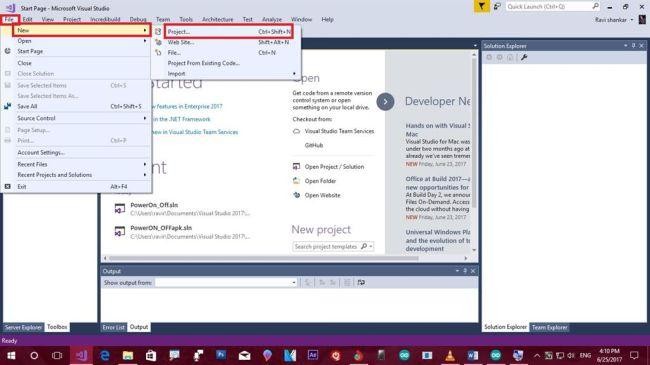
#### Step 10

Then select the upload a program option



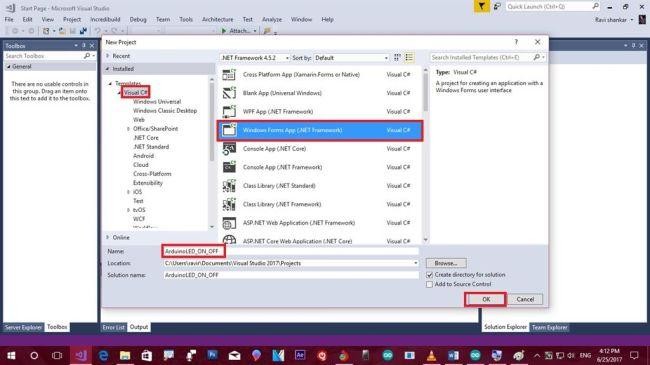
#### Step 11

I also have Visual Studio 2017.Then select FILE NEW PROJECT



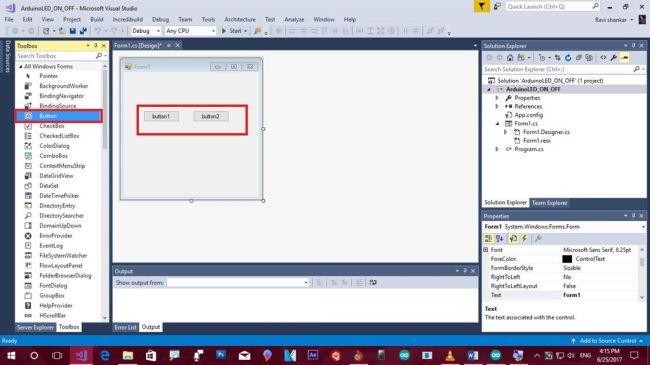
#### Step 12

Select Windows form applications Change file name as per your wish



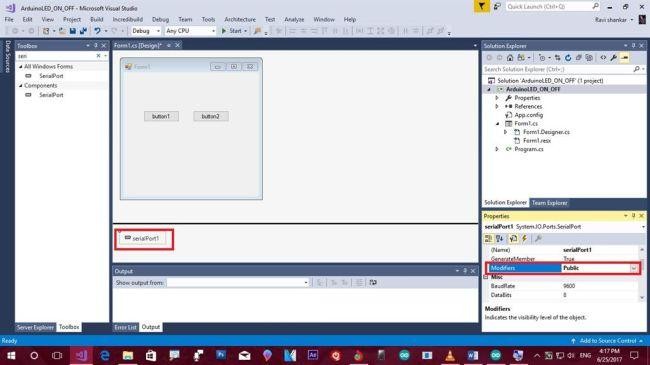
#### Step 13

Then left corner appears on tool box select search button drag and drop two buttons and search Serial Port drag and drop from form applications

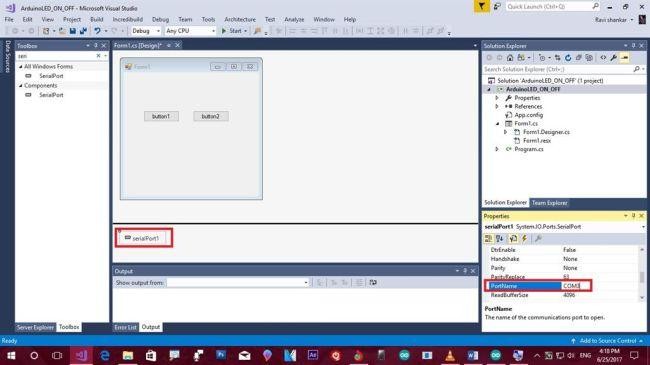


#### Step 14

Then select serial port1 properties; modifier will change for public view

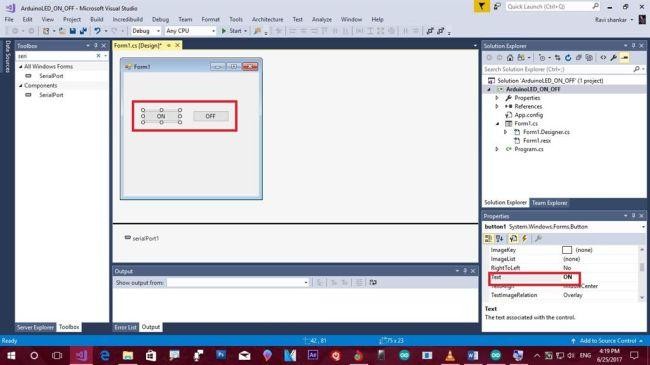


Then scroll down to port name it will change to COM3



#### Step 15

Button1 click to change from properties will change button name for ON Button2 click to change from properties will change button name for OFF

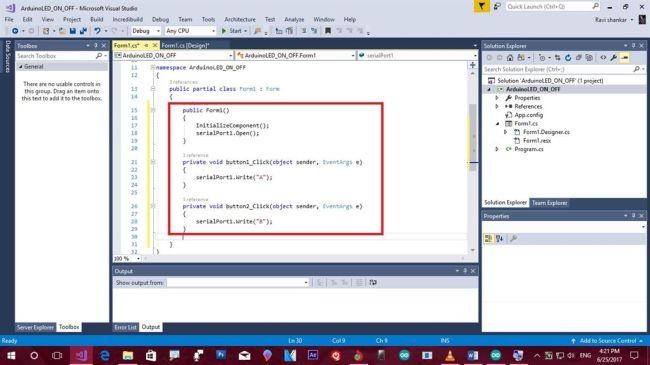


#### Step 16

ON Double click to going to page from Form1.cs file

Again OFF Double click to going to page from Form1.cs file

Then replace this C# source code

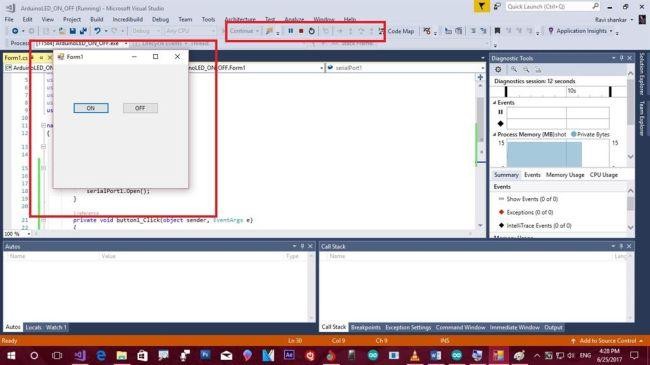


#### Source code

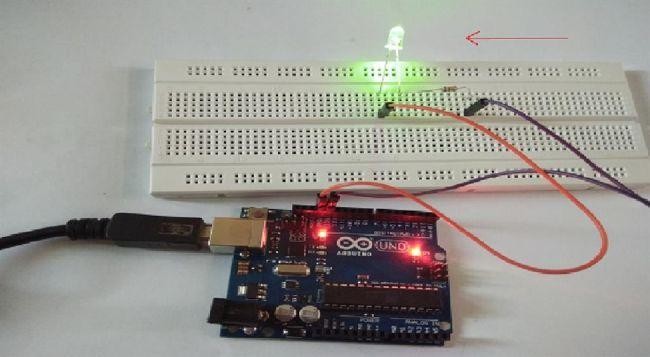
1. public Form1() {
2. InitializeComponent();
3. serialPort1.Open(); 4. }
4. private void button1\_Click(object sender, EventArgs e) {
5. serialPort1.Write("A"); 7. }
6. private void button2\_Click(object sender, EventArgs e) {
7. serialPort1.Write("B");
8. } // This is just a sample script. Paste your real code (javascript or HTML) here.
9. if ('this\_is' == /an\_example/) {
10. of\_beautifier();
11. } else {
12. var a = b ? (c % d) : e[f];

15. }

Then click Start button or Ctrl+F5 and our program will debug



Click LED ON and OFF; the process is working.



#### Summary

LED light switch ON to LED is ON then switch LED LED is OFF. Simple project will be created and deployed.

# Turn ON OFF MULTIPLE LEDs using Arduino and

visual studio

## Introduction:

This Practicle will help us to understand to turn ON/OFF LEDs using ARDUINO based on the selection using C# Programming Language (Windows App). Before starting this, reader should have basic understanding about serial interface and its communication between computer and peripheral devices.

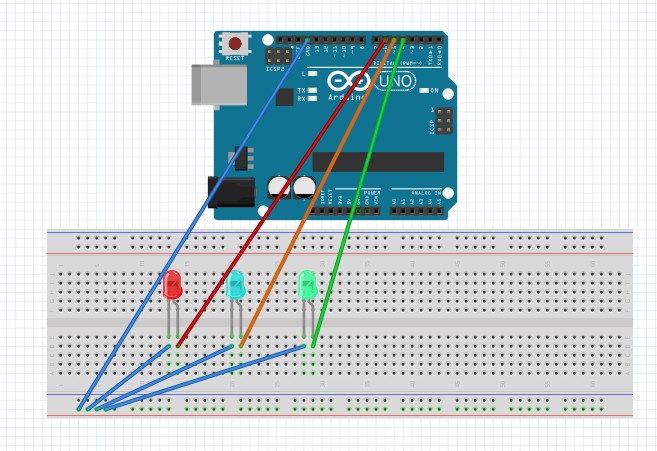
**Serial Interface**: (This project uses USB interface)

Serial Interface is a communication interface between two digital systems that transmits data as a series (bit by bit) voltage pulses. Usually information will be sent Bit by bit over a cable. Some of Modern Serial interfaces are USB, fireware and Ethernet.

## Required Components:

1. ARDUINO
2. 7 to 8 Jumper Male Pins
3. Bread Board
4. USB Cable – Used to connect ARDUINO with Computer to send commands.

## Circuit Design:



**ARDUINO Source Code**:

Given below the code to **Turn ON/OFF LEDs**. Copy and Paste it in ARDUINO Code explorer and burn the code in ARDUINO by selecting required COM Port. From Windows App, send commands from windows application.

#define BaudRate 9600

char inchar; //Will hold the incoming character from the Serial Port.

// Start: I am going to control the below LEDs

int isFanOn = 0; int isLightOn = 0; int isBoilerOn = 0;

// End

// Start: Below LED Planes int led1 = 4;

int led2 = 5; int led3 = 6;

// End

void setup()

{

// serial communication Serial.begin(BaudRate);

// prepare the digital output pins pinMode(led1, OUTPUT); pinMode(led2, OUTPUT); pinMode(led3, OUTPUT);

// initially all are off digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

}

void loop()

{

inchar= Serial.read();

// Serial.println(inchar);

if(inchar==’O’)

{

isFanOn = 0;

isLightOn = 0;

isBoilerOn = 0;

digitalWrite(led1, LOW); digitalWrite(led2, LOW); digitalWrite(led3, LOW);

Serial.println(“O”);

}

if(inchar==’F’ && isFanOn==0)

{

isFanOn = 1; digitalWrite(led1, HIGH); Serial.println(“F1”);

}

else if (inchar==’F’ && isFanOn==1)

{

isFanOn = 0; digitalWrite(led1, LOW); Serial.println(“F0”);

}

if(inchar==’L’ && isLightOn==0)

{

isLightOn = 1; digitalWrite(led2, HIGH); Serial.println(“L1”);

}

else if (inchar==’L’ && isLightOn==1)

{

isLightOn = 0; digitalWrite(led2, LOW); Serial.println(“L0”);

}

if(inchar==’B’ && isBoilerOn==0)

{

isBoilerOn = 1; digitalWrite(led3, HIGH);

Serial.println(“B1”);

}

else if (inchar==’B’ && isBoilerOn==1)

{

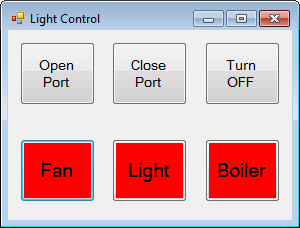
isBoilerOn = 0; digitalWrite(led3, LOW); Serial.println(“B0”);

}

}

**Windows Application Interface**: (Interface to communicate with Arduino)

1. Open Port – It will be used to open the COM Port
2. Close Port – It will be used to close the COM Port
3. Fan, light and Boiler Buttons – It will send Volt to LEDs. These buttons will act as ON/OFF switch.
4. Turn OFF – It will make all the LEDs OFF.



We need to drag and drop the “**Serial Port**” control into the form and set the BaudRate to 9600 and Port Name to COM3 (it is based on the port which you have selected for ARDUINO). Given below the Windows Application (C#) code.

public partial class Form1 : Form

{

int isFanOn = 0; int isLightOn = 0; int isBoilerOn = 0;

public Form1()

{

InitializeComponent();

SetDefaultValues();

}

private void SetDefaultValues()

{

isFanOn = 0;

isLightOn = 0;

isBoilerOn = 0;

btnFan.BackColor = Color.Red; btnLight.BackColor = Color.Red; btnBoiler.BackColor = Color.Red;

}

private void btnOpenPort\_Click(object sender, EventArgs e)

{

try

{

}

serialPort1.Open(); //Open serialPort MessageBox.Show("Serial Port Opened!");

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void btnClosePort\_Click(object sender, EventArgs e)

{

try

{

}

serialPort1.Close(); //Close serialPort MessageBox.Show("Serial Port Closed!");

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

private void btnFan\_Click(object sender, EventArgs e)

{

try

{

}

serialPort1.Write("F");

//serialPort1.Write("0#100");

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnLight\_Click(object sender, EventArgs e)

{

try

{

}

serialPort1.Write("L");

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnBoiler\_Click(object sender, EventArgs e)

{

try

{

}

serialPort1.Write("B");

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

}

private void btnTurnOffAll\_Click(object sender, EventArgs e)

{

serialPort1.Write("O");

}

private void serialPort1\_DataReceived(object sender, SerialDataReceivedEventArgs e)

{

string callBackString = serialPort1.ReadLine(); callBackString = callBackString.Replace("\r", ""); switch (callBackString)

{

case "F0":

isFanOn = 0;

btnFan.BackColor = Color.Red; break;

case "F1":

isFanOn = 1;

btnFan.BackColor = Color.Green; break;

case "L0":

isLightOn = 0; btnLight.BackColor = Color.Red; break;

case "L1":

isLightOn = 1;

btnLight.BackColor = Color.Green; break;

case "B0":

isBoilerOn = 0; btnBoiler.BackColor = Color.Red; break;

case "B1":

isBoilerOn = 1; btnBoiler.BackColor = Color.Green; break;

case "O":

SetDefaultValues(); break;

}

// MessageBox.Show(callBackString);

}

}

That’s all, Follow the steps to Turn ON/OFF LEDs.

1. Make sure you have connected to correct port with ARDUINO.
2. Burn the ARDUINO code.
3. Make sure you have connected the ARDUINO.
4. Run the Application (windows App).
5. Click on buttons as mentioned above.

Once you familiar with the ARDUINO and its working methods, we can use Relays to connect to turn ON/OFF electrical house appliances (Extra care should be taken when dealing with 240V, please don't attempt if you are not confident).

# Home Automation In Windows Forms Using

Visual Studio

#### Requirements

* 1. Visual Studio.
  2. Arduino IDE.
  3. Connecting wires (10 numbers or more).
  4. Breadboard.
  5. DC motors which can run by just 5 volts power supply (2 numbers).
  6. LED light (1 number).

#### Introduction

We expect advancement in our day to day life. That is why I have thought of implementing a voice control technology using Arduino to take control over the electrical appliances at my home. Hence, in this article, I will explain how to create an application which helps you control the devices in your home, with your voice commands. So, to make it simple, I will explain how to connect a door which can be opened and closed and a fan along with a light which can be turned on and off, using the voice commands.

**Note**

**Step 1 Programming the Arduino**

* Open the Arduino IDE if already installed (or download and install it )
* Now, open the IDE and enter the following code in it.
  + 1. char incomingdata;
    2. void setup() {
    3. pinMode(2, OUTPUT);
    4. pinMode(4, OUTPUT);
    5. pinMode(12, OUTPUT);
    6. pinMode(13, OUTPUT);
    7. Serial.begin(9600); 8. }

1. void loop() {
2. incomingdata = Serial.read(); {
3. if (incomingdata == 'a') {
4. digitalWrite(2, HIGH);
5. } else if (incomingdata == 'b') {
6. digitalWrite(2, LOW);
7. } else if (incomingdata == 'c') {
8. digitalWrite(4, HIGH);
9. } else if (incomingdata == 'd') {
10. digitalWrite(4, LOW);
11. } else if (incomingdata == 'e') {
12. digitalWrite(12, HIGH);
13. digitalWrite(13, LOW);

22. delay(3500);

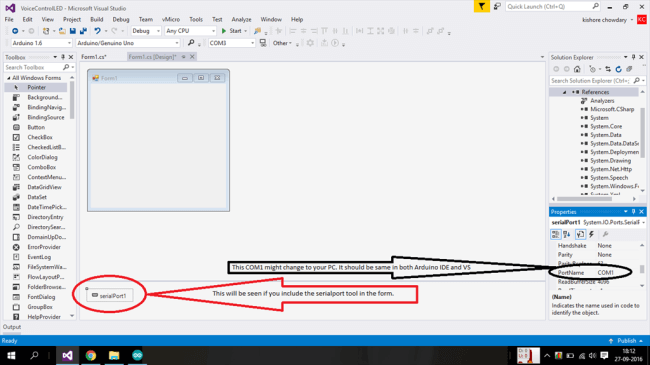
1. digitalWrite(12, LOW);
2. } else if (incomingdata == 'f') {

|  |  |  |  |
| --- | --- | --- | --- |
| 25. |  |  | digitalWrite(12, LOW); |
| 26. |  |  | digitalWrite(13, HIGH); |
| 27. |  |  | delay(3500); |
| 28. |  |  | digitalWrite(13, LOW); |
| 29. |  | } |  |
| 30. | } |  |  |
| 31. } |  |  |  |

* In this program, I have used a delay of 3.5 seconds in the last two conditions. This is because of the length of the door which I have used in my prototype. You can change the delay length according to your prototype.
* Save this sketch at any location and verify the code.
* Now, connect your Arduino board and upload this sketch into the board.

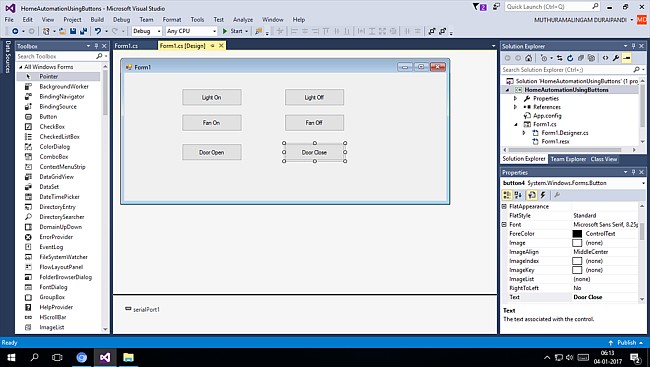
#### Step 2 Programming in Visual Studio

* Open Visual Studio and create a new Windows Form application with any name you want.
* Once you have created a new form application, open the designer window of Form1.
* Now, from the tool box, drag and drop the SerialPort tool into the form1.
* This will help your program communicate with the Arduino board.
  + Now, drag and drop six buttons in the form to make use of them for controlling the lights and fan.



#### Step 3 Creating buttons

* Drag and drop six buttons in the form.
* Name them as LIGHT On, LIGHT Off, FAN On, FAN Off, DOOR Open, and DOOR Close.
  + Once you finish these all, the form looks like below.



#### Step 4 Writing code for controlling and using buttons

* The controls will work as they are if you build your prototype by connecting a DC motor with a small fan wing to work like a fan and another DC motor with a door which can be moved up and down like a shutter door and also an LED which can work like a light.
* Remember again that you are building a prototype for an automated house and not a real house.
* Double click on the "Light On" button. You will get to a coding page. There, you must add a small piece of code for the button click event.
* Add the button click event code for each and every separate button.
* Have a look over the code.

1. using System;
2. using System.Windows.Forms;
3. namespace HomeAutomationUsingButtons {
4. public partial class Form1: Form {
5. public Form1() {
6. InitializeComponent(); 7. }
7. private void button1\_Click(object sender, EventArgs e) {
8. serialPort1.Open();
9. serialPort1.Write("a");
10. serialPort1.Close();

12. }

1. private void button6\_Click(object sender, EventArgs e) {
2. serialPort1.Open();
3. serialPort1.Write("b");
4. serialPort1.Close();

17. }

1. private void button2\_Click(object sender, EventArgs e) {
2. serialPort1.Open();
3. serialPort1.Write("c");
4. serialPort1.Close();

22. }

1. private void button5\_Click(object sender, EventArgs e) {
2. serialPort1.Open();
3. serialPort1.Write("d");
4. serialPort1.Close();

27. }

1. private void button3\_Click(object sender, EventArgs e) {
2. serialPort1.Open();
3. serialPort1.Write("e");
4. serialPort1.Close();

32. }

1. private void button4\_Click(object sender, EventArgs e) {
2. serialPort1.Open();
3. serialPort1.Write("f");
4. serialPort1.Close();

37. }

38. }

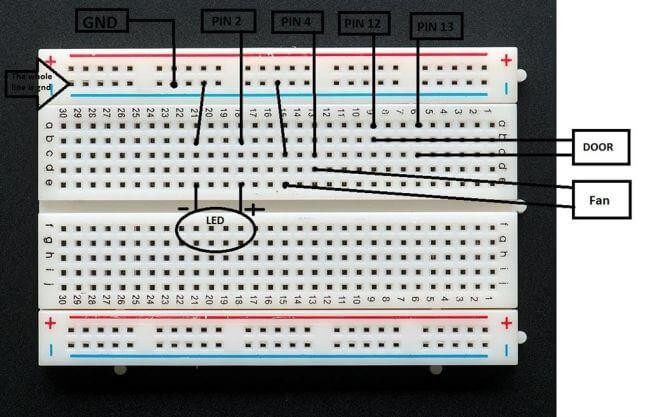
39. }

#### Step 5 Giving the connections in the breadboard

* According to the program, I have used pin 2 of the Arduino for light, pin 4 for the fan, and pin 12, 13 for the door.
* Since the door needs to move in both directions, we need to make the dc motor of the door run in both directions.
* Hence, the door is given with two pin connections.

#### Connecting the setup

1. Take a ground connection from the Arduino and connect it to the breadboard.
2. Take pin 2 and connect it to the LED along with the ground connection.
3. Take pin 4 and connect it to the DC motor along with the ground connection which will act as a fan.
4. Take pins 12, 13 and connect each of them to one end of the motor so that it can run in both directions.
5. Take a look over the image for clarification.



#### Working of the setup

1. When you click on Light On button, lights will get turned on and when you click on Light Off button, lights will get turned off.
2. When you click on Fan On button, fan will turn on and when you click on Fan Off button, fan will turn off.
3. When you click on Door Open button, the motor of the door will rotate in a direction and then turn off after a few seconds.
4. When you click on Door Close button, the motor will rotate in another direction and then turn off after few seconds.

#### Final step

* If you are unable to build a simulated model, you can just give connections for the LEDs and the DC motors and check the output.
* Here, for the door, the motor will just rotate in bi-directions. Hence, you need to create the door such that it can move up and down.
* Once again, check all your code and the connections.
* Now, click on the buttons and check the output.

To measure distance between two objects by ultrasonic sensor with Arduino.

int trigPin = 9;

int echoPin = 10;

int led = 7;

void setup() {

Serial.begin(9600);

pinMode(led, OUTPUT);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

// put your setup code here, to run once:

}

void loop() {

long duration, distance;

digitalWrite(trigPin,HIGH);

delayMicroseconds(1000);

digitalWrite(trigPin, LOW);

duration=pulseIn(echoPin, HIGH);

distance =(duration\*0.034/2);

Serial.print(distance);

Serial.println("CM");

delay(10);

if((distance<=20))

{

digitalWrite(led, HIGH);

Serial.println("Detect Something");

}

else if(distance>10)

Serial.println("Detect Nothing");

{

digitalWrite(led, LOW);

}

}