UKQA2242/UKQF2442 2122/1 ROBOTICS & COMMUNITY

BUKU MODUL PEMBEJARAN TINKERCAD

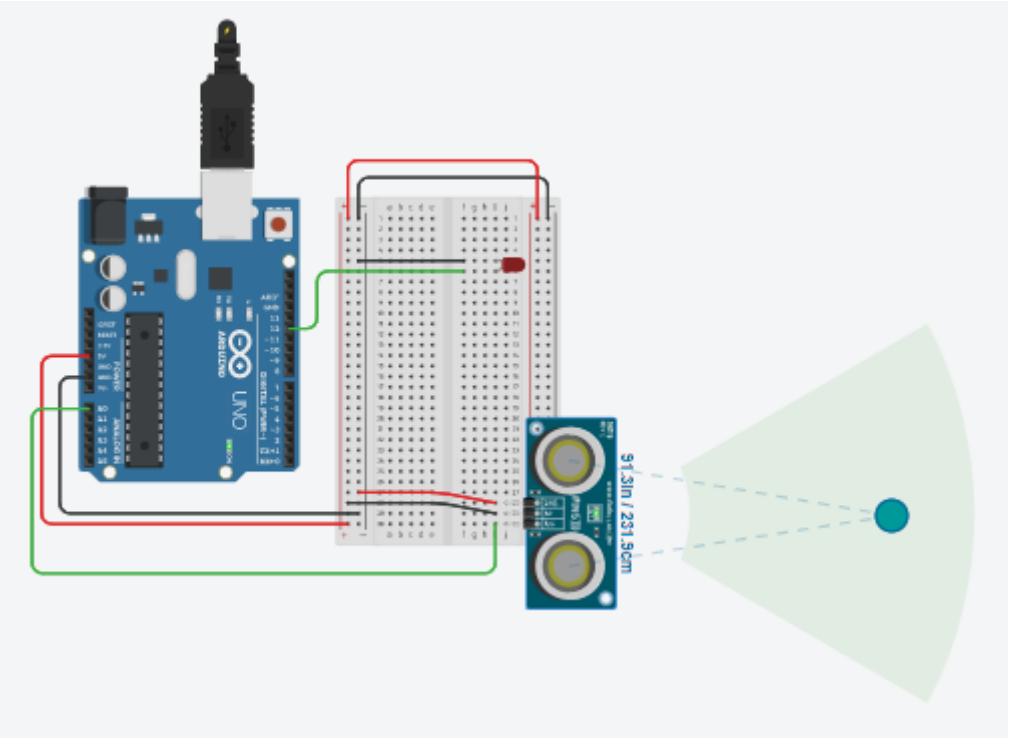
BESERTA CONTOH PROJEK ROBOTIK

UTM UNIVERSITI TEKNOLOGI MALAYSIA
Institute for Life-Ready Graduate (UTM iLeaGue)
Office of Deputy Vice Chancellor
(Academic & International)

Title 1 : Ultrasonic Sensor trigger LED

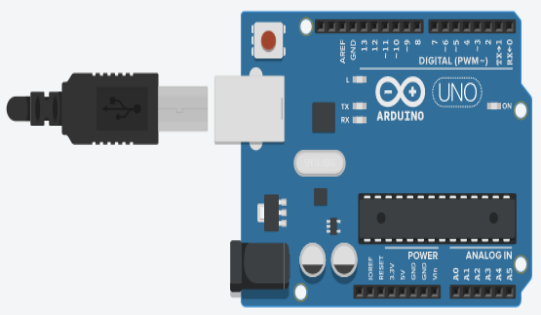
Introduction

In this project we are going to create Ultrasonic Sensor trigger LED. Ultrasonic Sensor in Arduino where we will turn on a led with the variation of distance and print the distance from an object to the serial monitor.

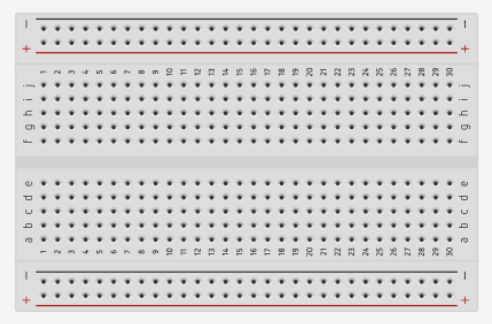


Hardware Requirement :

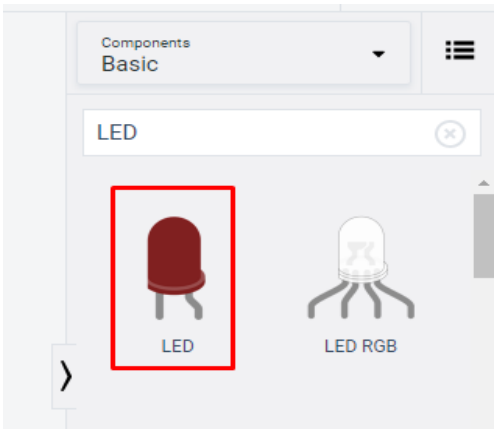
Components: The Parts which we need are -



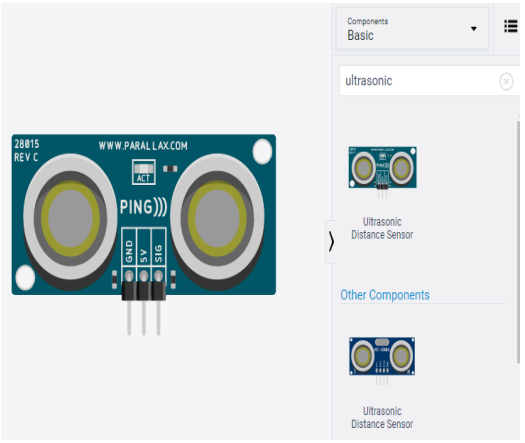
Arduino UNO R3



Breadboard



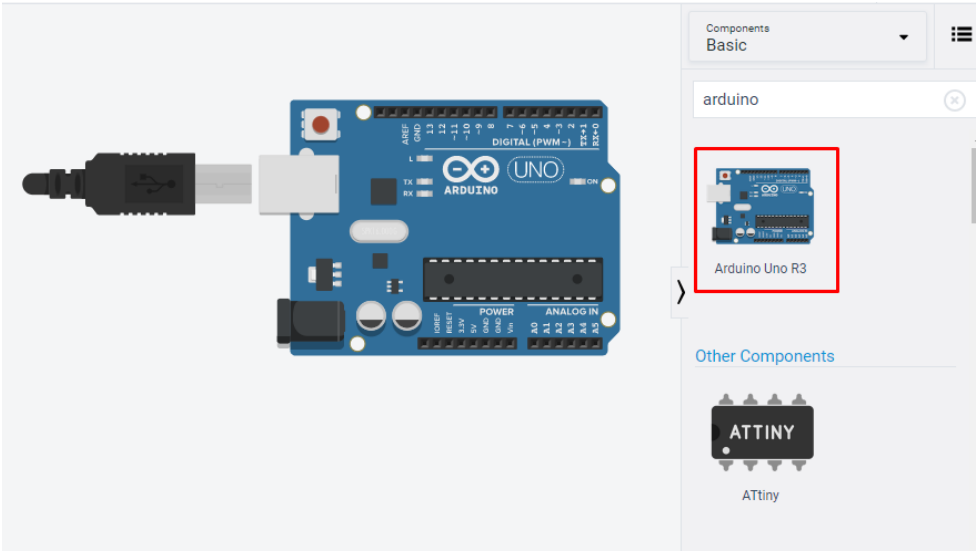
LED



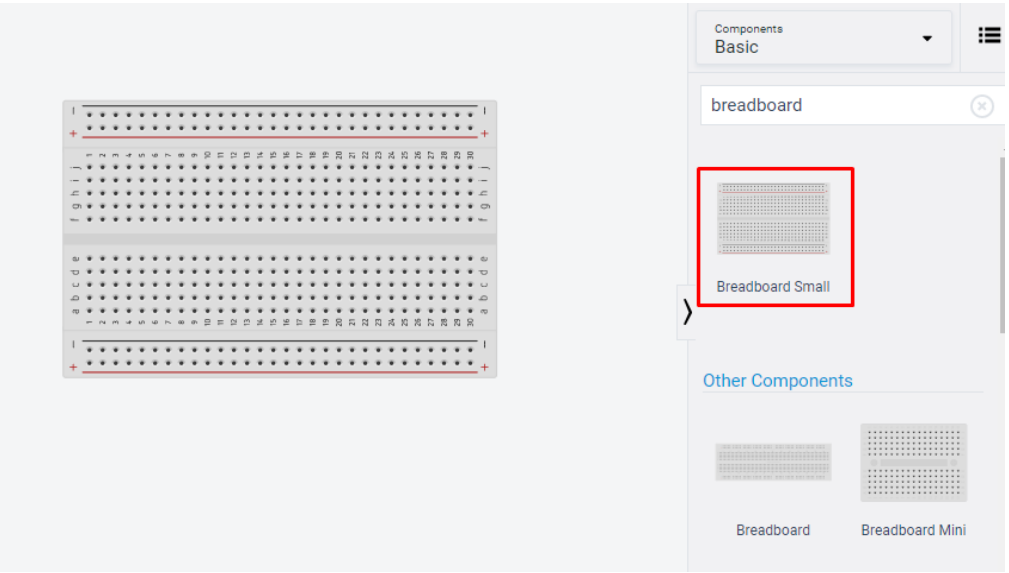
Ultrasonic Distance Sensor

Steps:

Step 1: Get Arduino Uno R3 and Breadboard



Arduino Uno R3

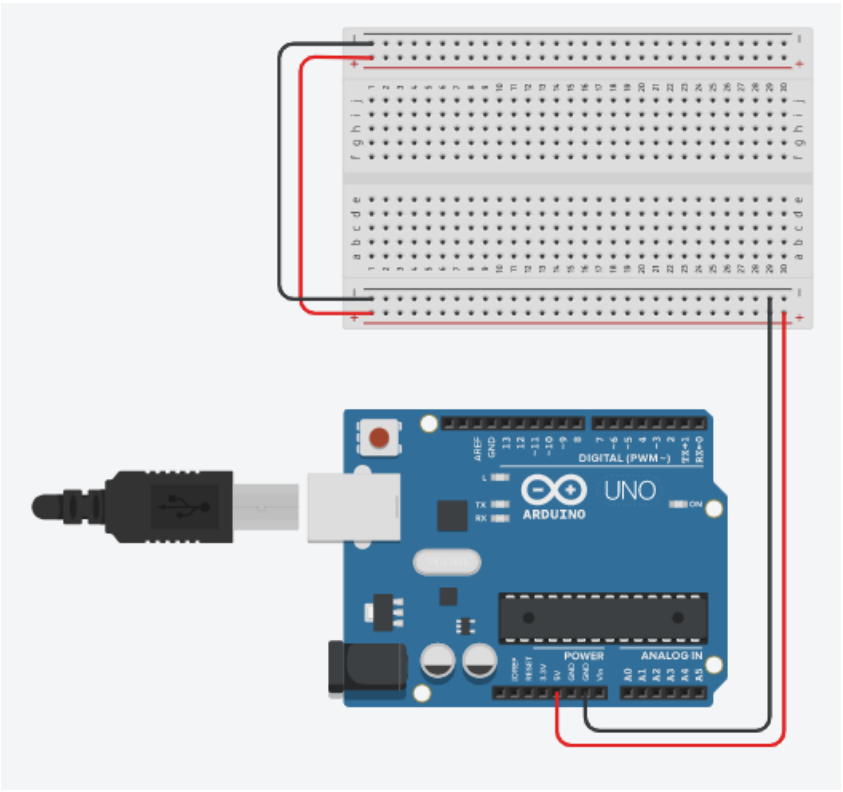


Breadboard

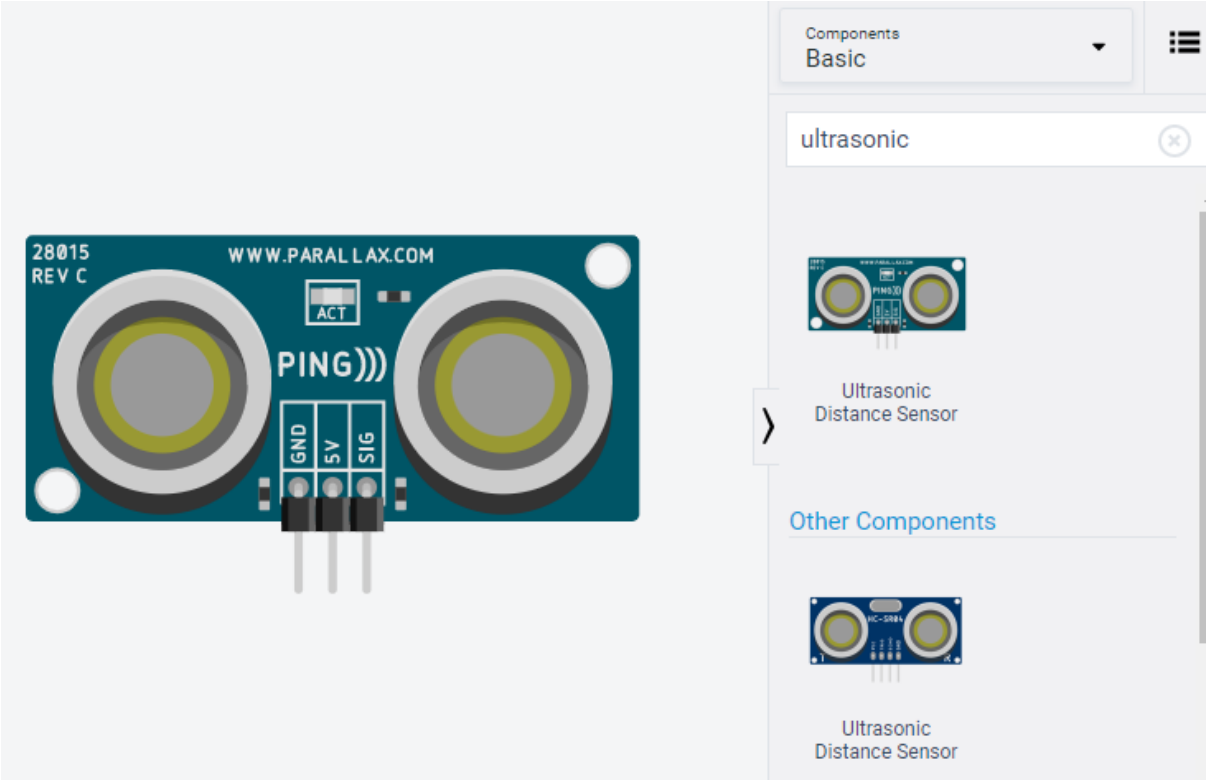
1. From the component menus, select the breadboard and Arduino Uno R3.
2. Arrange them on the work surface.

Step 2:

1. Connect a ground wire from the GND pin of Arduino to the breadboard.
2. Connect 5V wire from the 5V pin of Arduino to the breadboard.

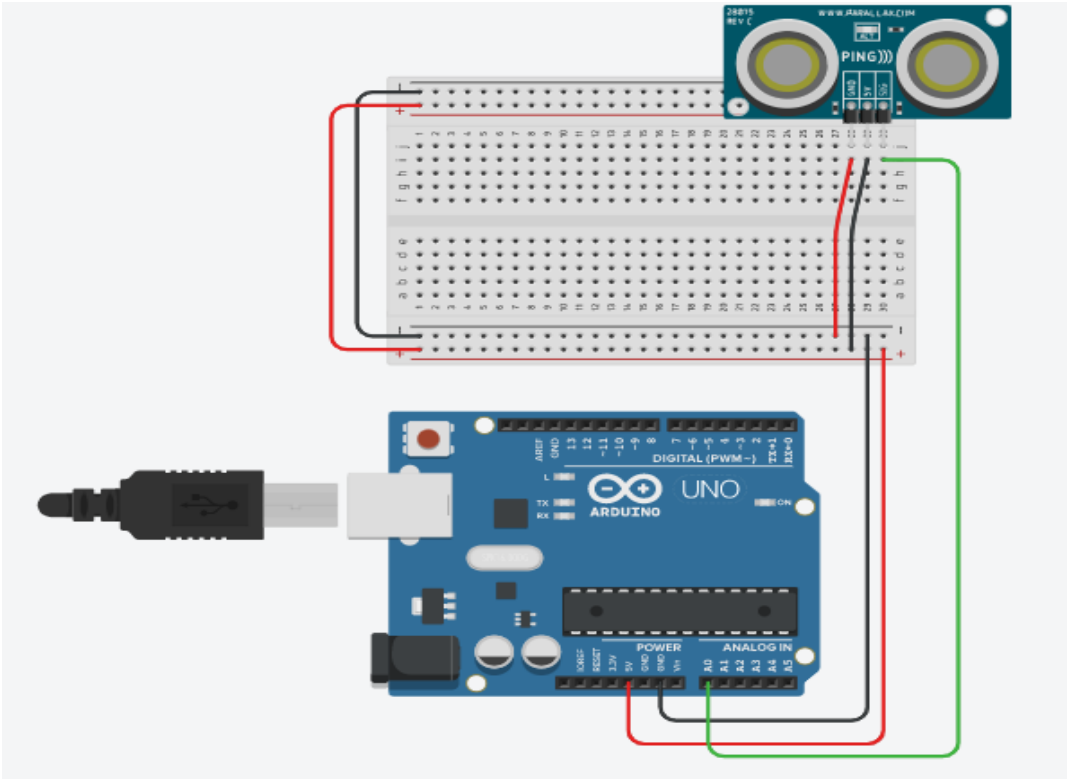


Step 3: Search Ultrasonic Distance Sensor



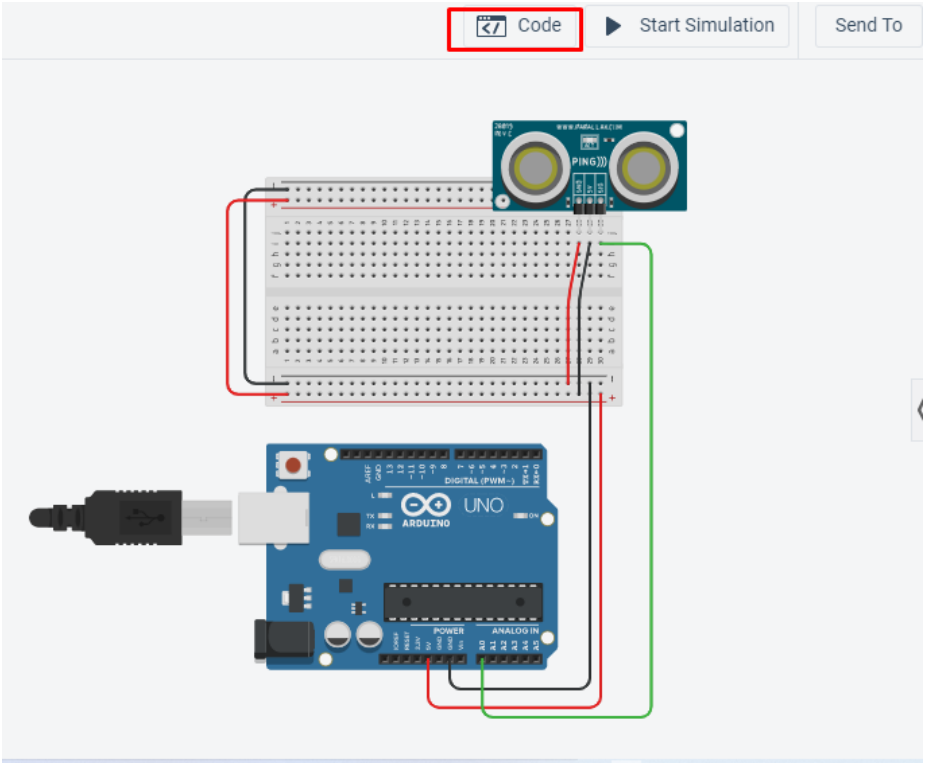
Step 4:

1. 5V and GRD wire connected with Ultrasonic Distance Sensor and add another wire with A0 of Arduino to the Ultrasonic Distance Sensor.

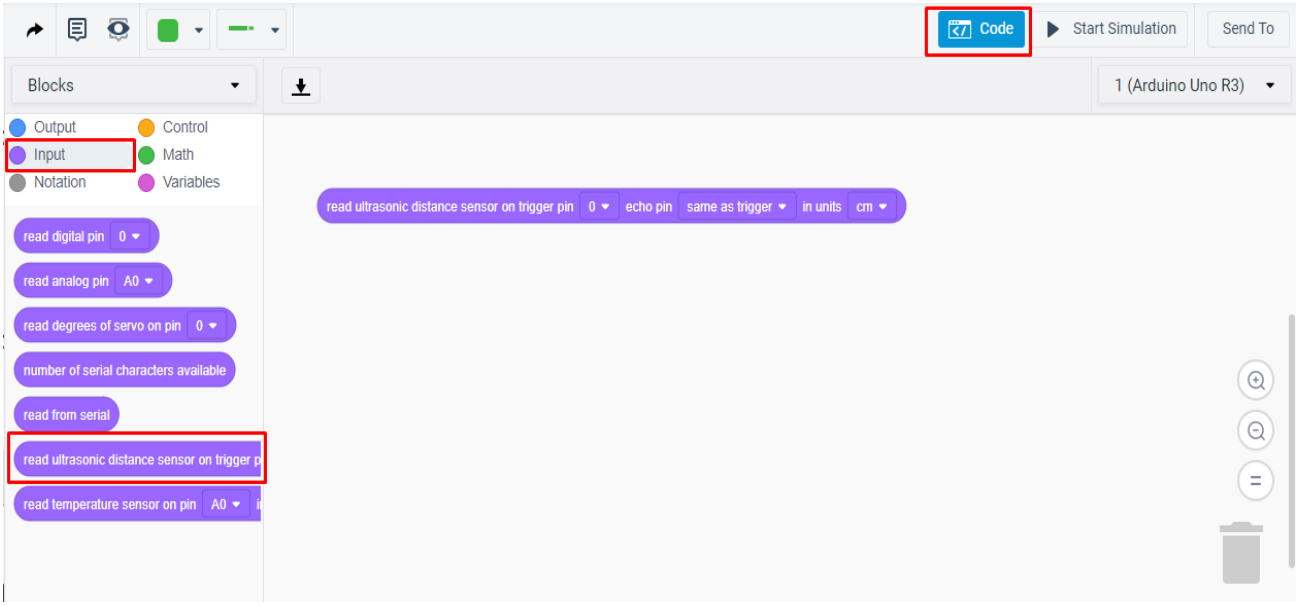


Step 5:

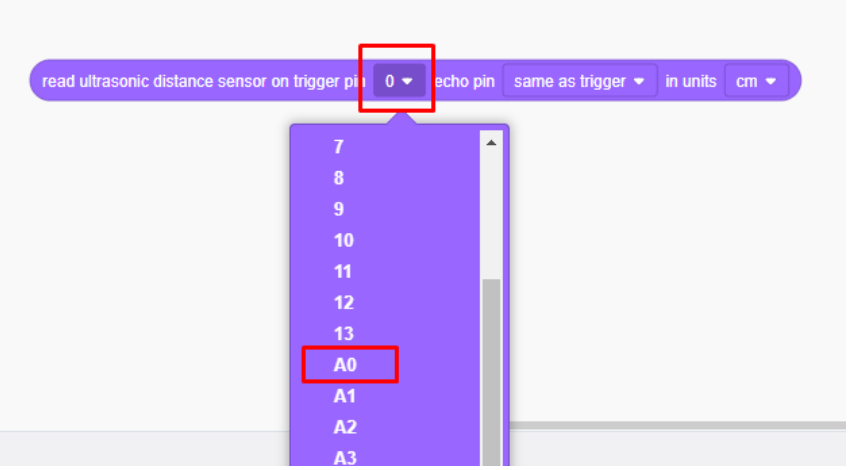
1. After added wire between Ultrasonic Distance Sensor and Arduino Uno R3 and Breadboard then click to the “code” option.



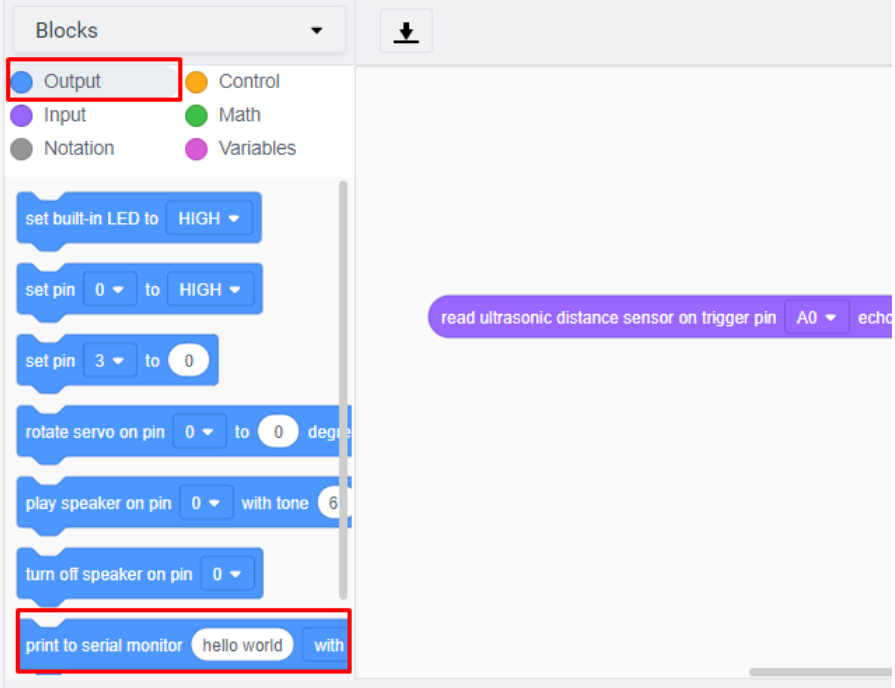
2. Then click on input and select “road Ultrasonic Distance Sensor trigger pin” section.



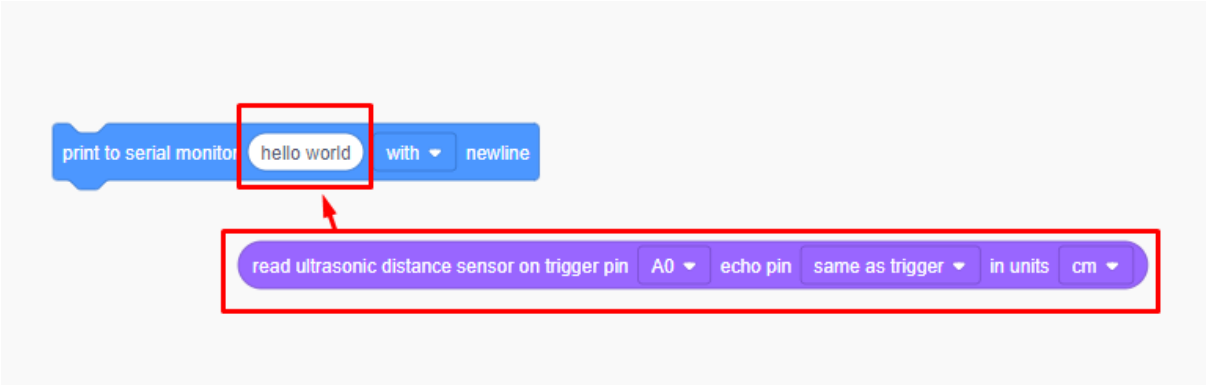
3. Set triggar pin “A0”



4. Going to the output option then click " print to the serial monitor"

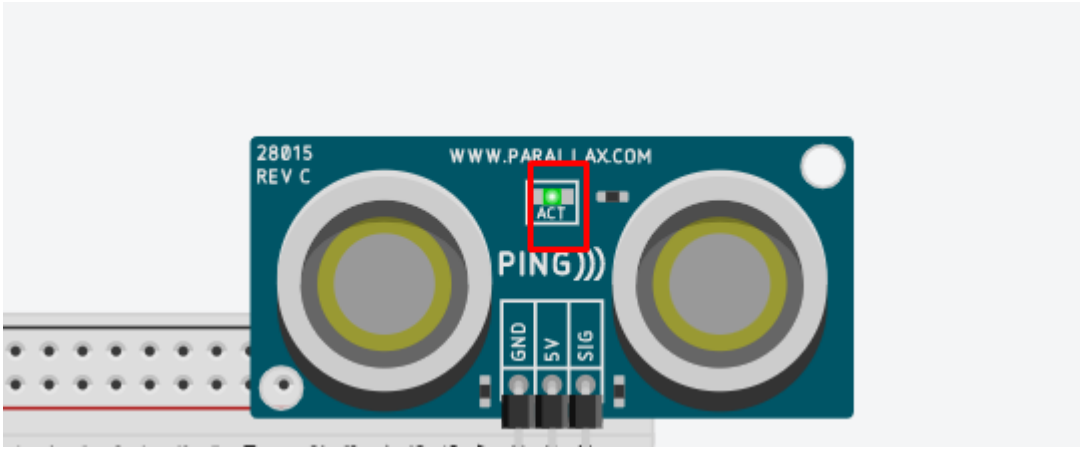


5. Replace input into output

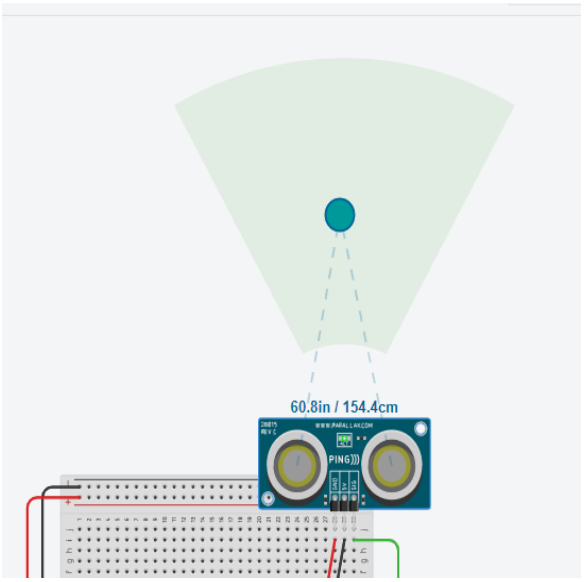


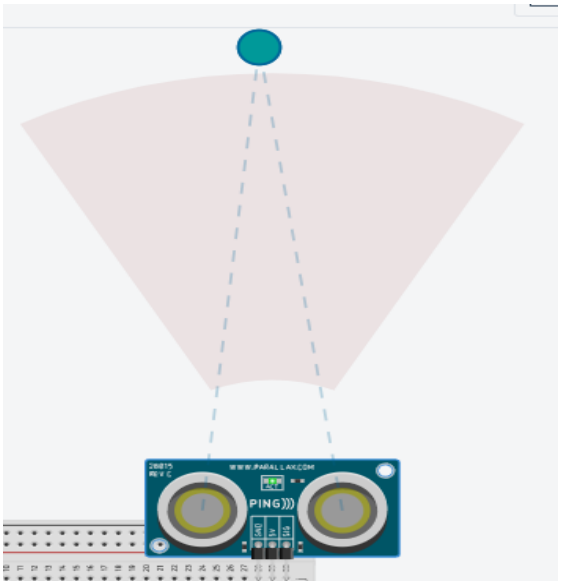
6. Can see the code upto added with sensor, arduino and breadboard connection. If we simulate then we will see

7. Then back to the main board where we create circuit. After click “simulation” section it show green light.



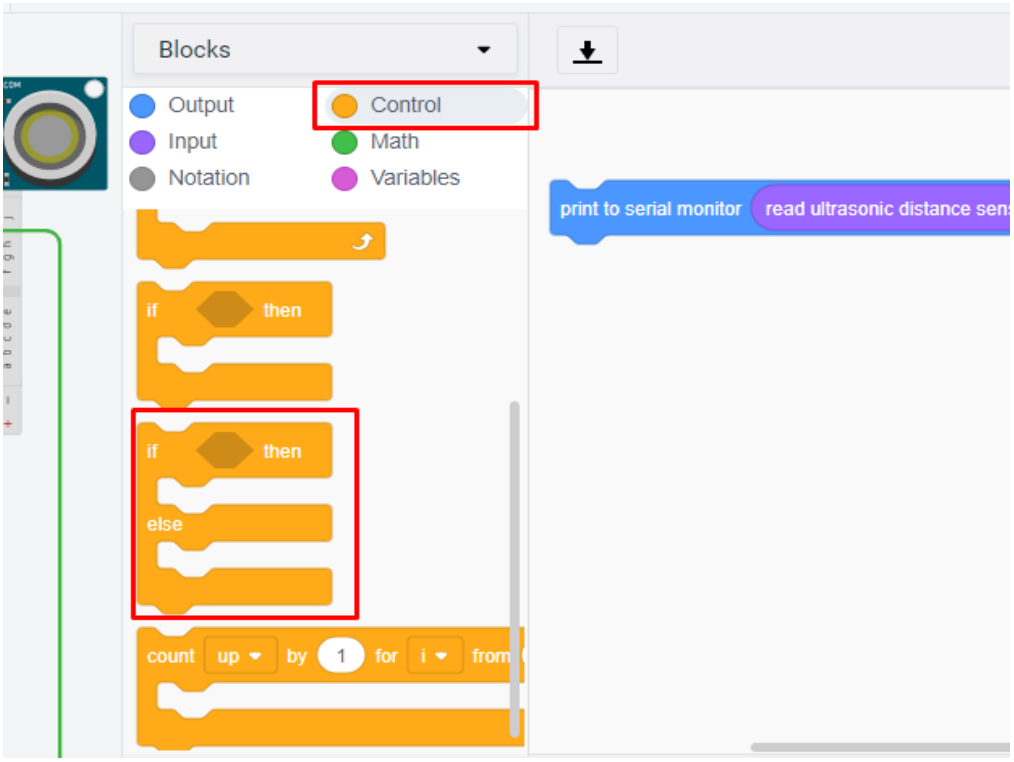
8. After that, it is running and give the value then click on the ultrasonic sensor and drag that thing adrond to adjust and can move the ball. When you get too close but this is really just like ultrasonic sensor. We can use it in the classroom where it has a kind of range.



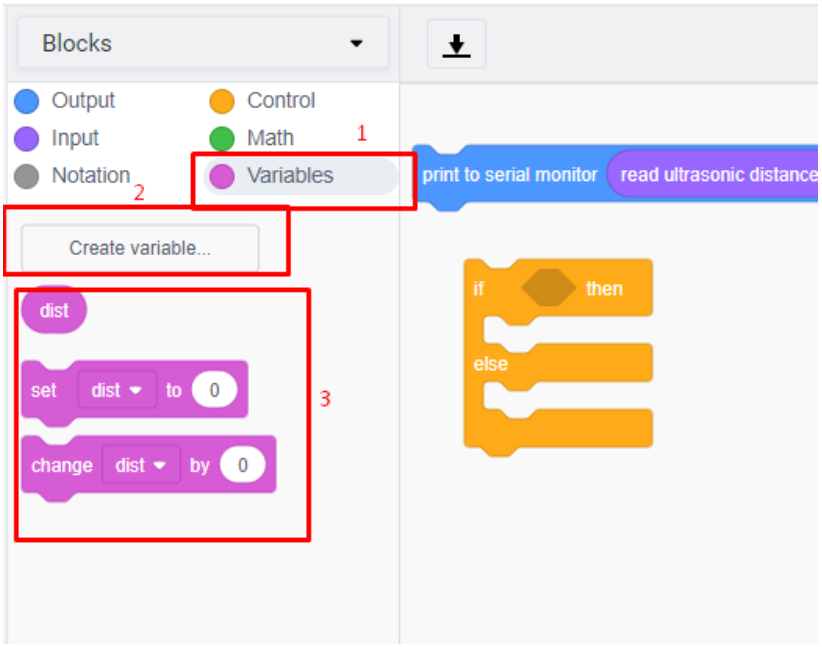


Step 6:

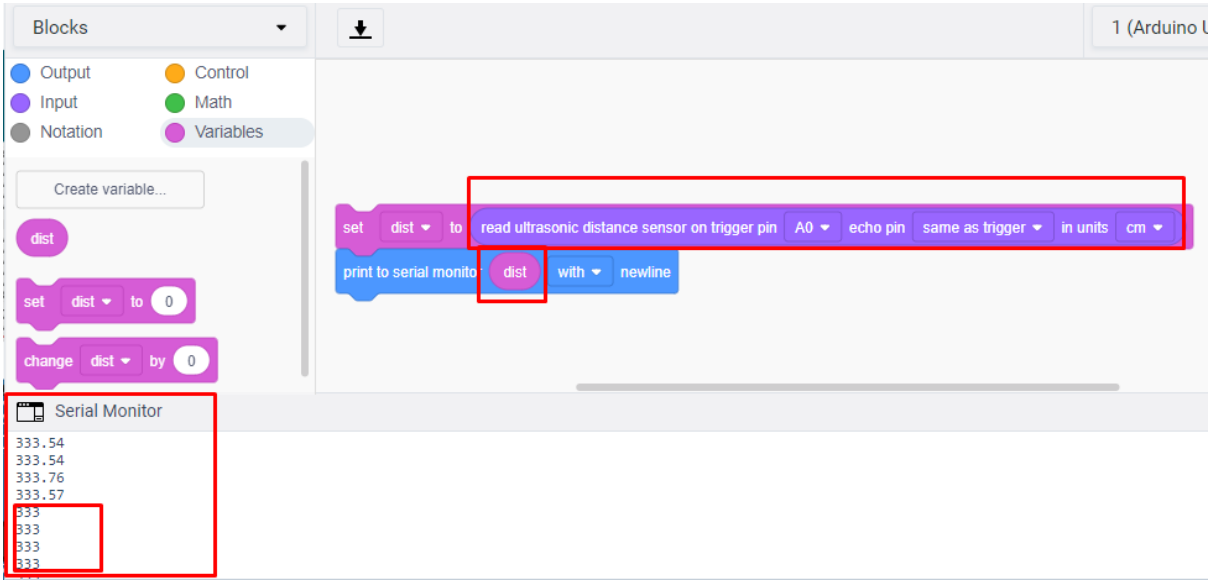
1. If it's less than a certain distance turn it off besides turn it on if it's close so that we have lots of condition statements.



2. Then create variables "dist" means distance.



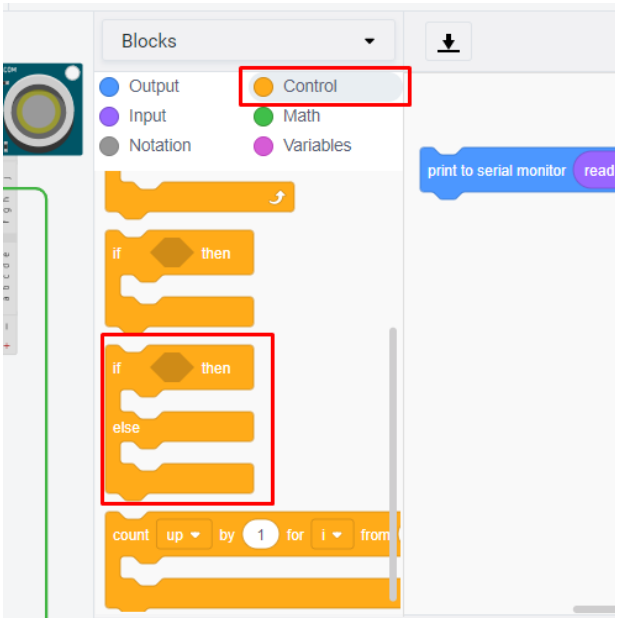
3. After that add some variables into code. If we focus on serial monitor section then we can see when we simulate it it some decimal numbers because all the variables appears inside of here going to then be integers , if set float then will appear floating type numbers.



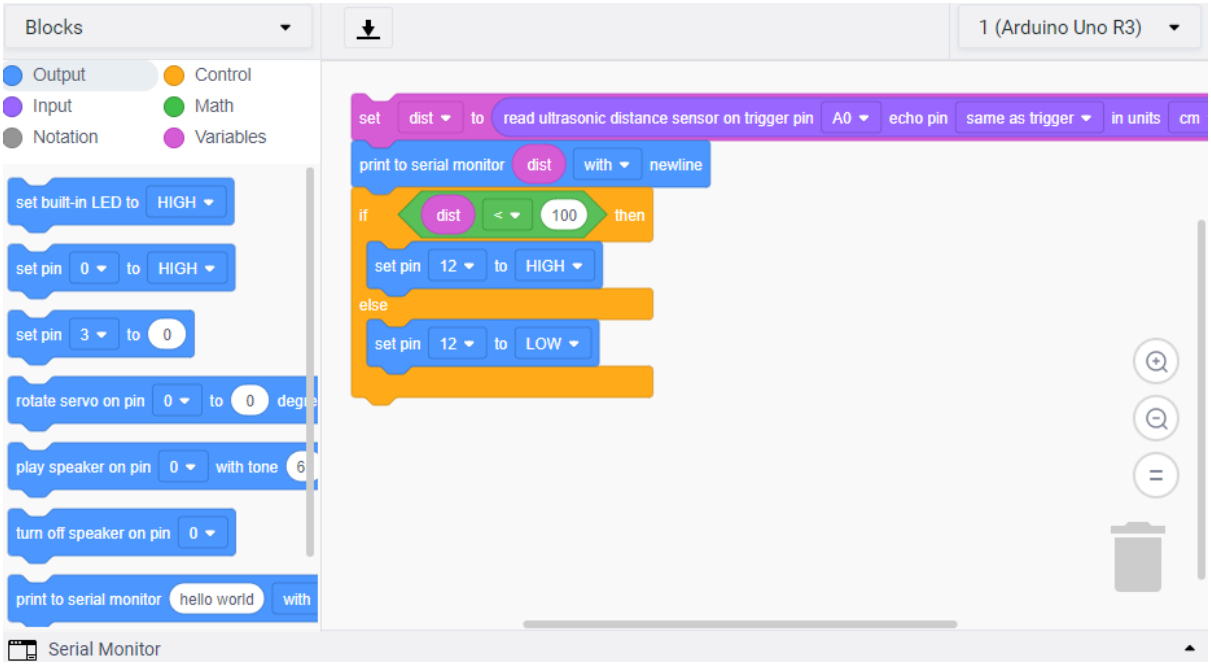
```
  // Reads the echo pin, and returns the sound wave
  return pulseIn(echoPin, HIGH);
}

void setup()
{
  Serial.begin(9600);

}

void loop()
{
  dist = 0.01723 * readUltrasonicDistance(A0, A0);
  Serial.println(dist);
  delay(10); // Delay a little bit to improve simula
}
```

Step 7:

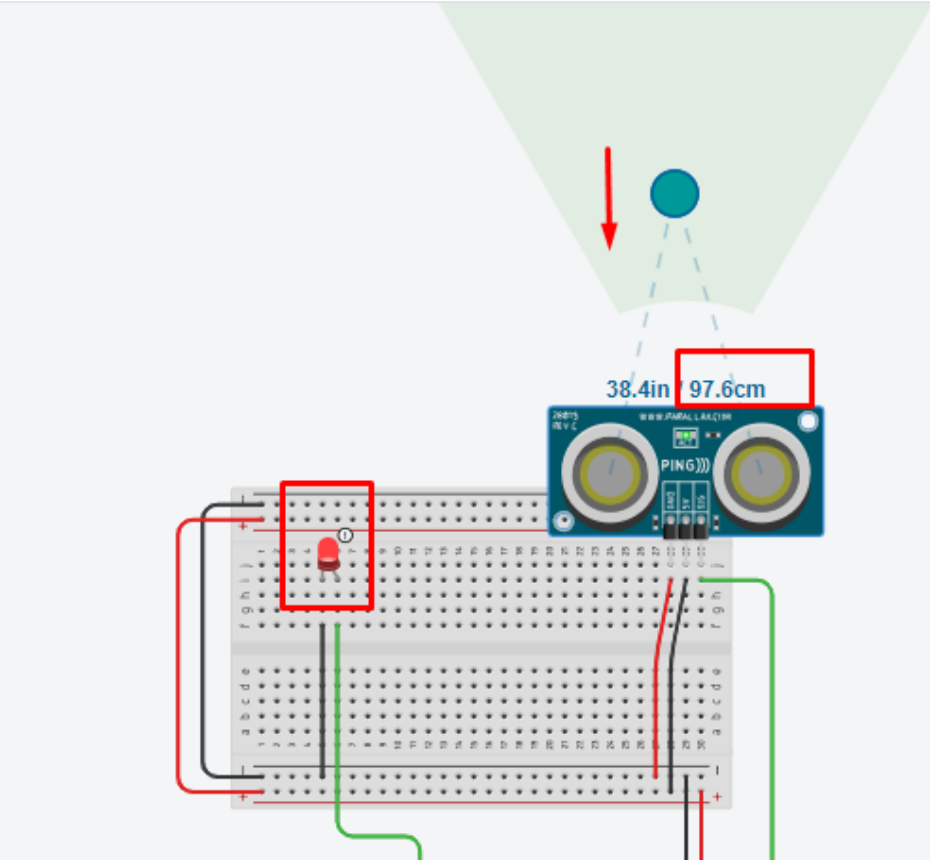
1. If distance of centimeter is fraction then we can put if…else…then conditional statements. If it's less than a certain distance turn it off besides turn it on if it's close so that we have lots of condition statements.



2. Into condition pin do pin 12 set that pin 12 too high that should turn an LED on and just be great and then if want to turn it off I turn it low also choose pin 12. And all just add together. All makes perfect scense.



3. Now put LED. And start simulation then we can see if the ball range under 100cm then LED light on.



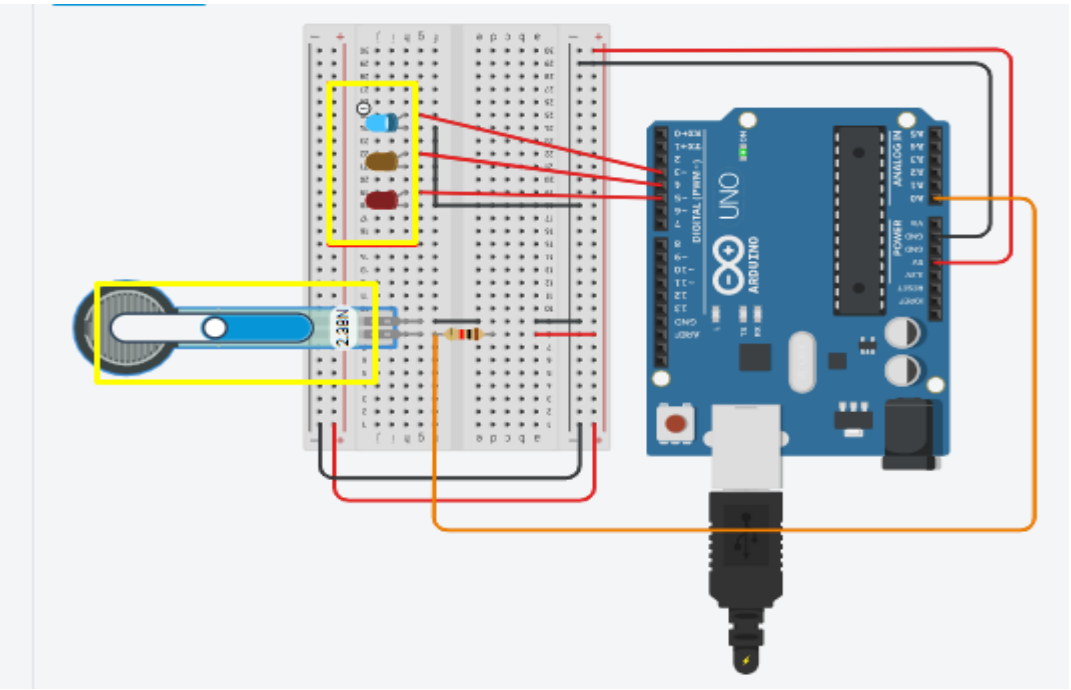
Result: It is really could very easy and all the lights kind of show your distance, could have like a distance meter just lights so if you are far away it would be all light up. And then as you got closer or be down to like one or two lights when you get really close and that'd be really cool kind of like you're little distance detector for it.

Title 2: Force Sensor trigger LEDs

Introduction

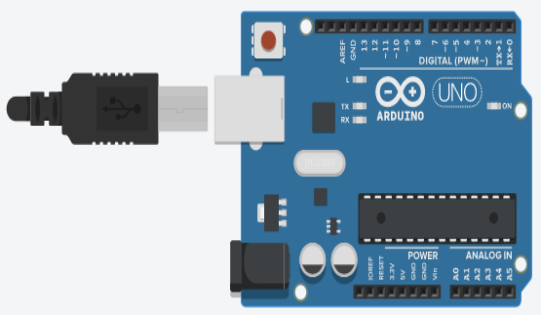
The Force sensor, also known as a force sensing resistor (FSR), will be connected to the Arduino Uno in this project. We've also used Arduino to connect three separate LEDs.

Force Sensor are extremely durable pressure sensors that are employed in a variety of applications. Electronic drums, cellphones, portable gaming systems, and a variety of other portable electronics contain them. These pressure sensors are simple to use and work well.

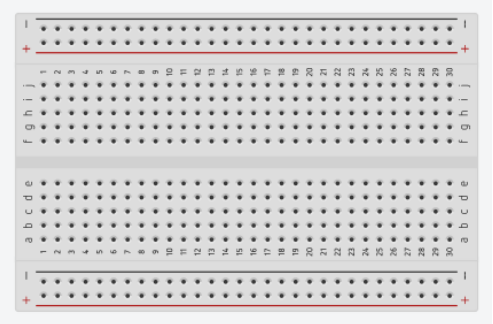


Hardware Requirement :

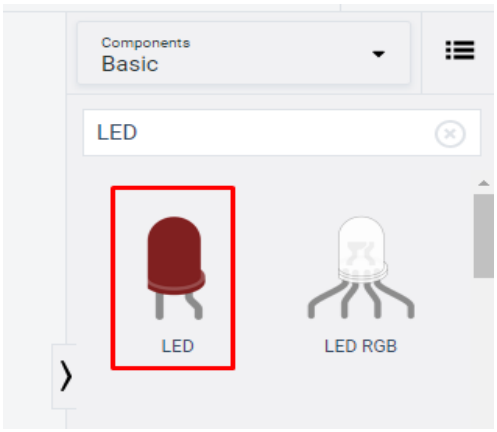
Components: The Parts which we need are -



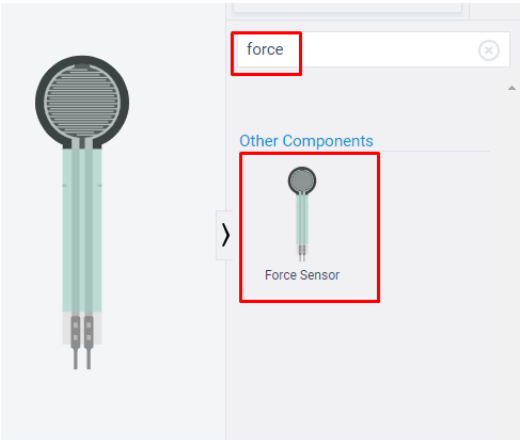
Arduino UNO R3



Breadboard

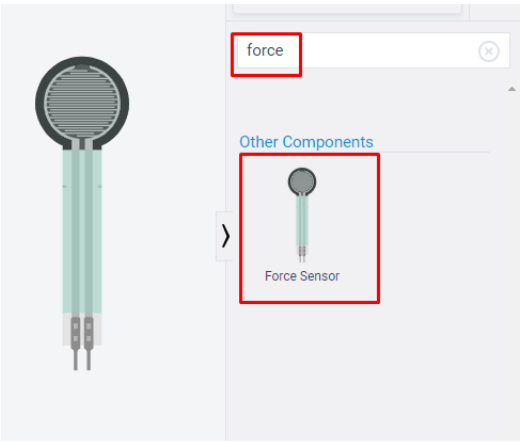


LED



Ultrasonic Distance Sensor

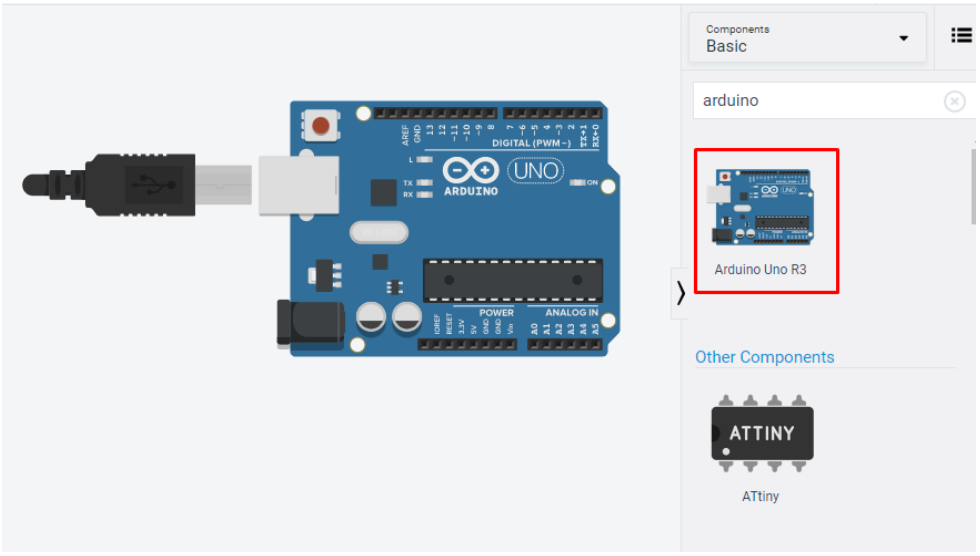
Resistor



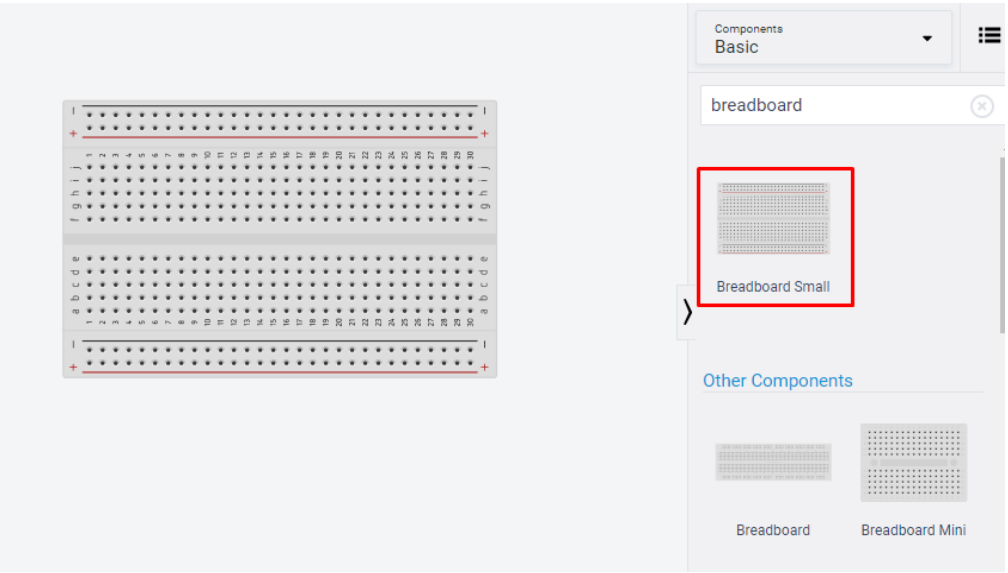
Ultrasonic Distance Sensor

Step:

Step 1: Get Arduino Uno R3 and Breadboard



Arduino Uno R3

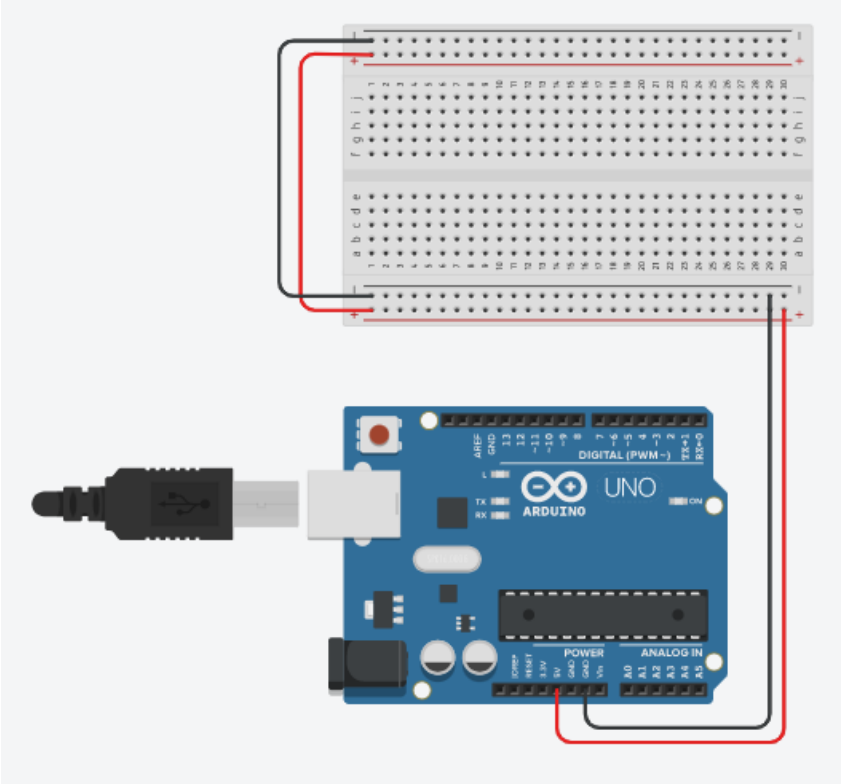


Breadboard

1. From the component menus, select the breadboard and Arduino Uno R3.
2. Arrange them on the work surface.

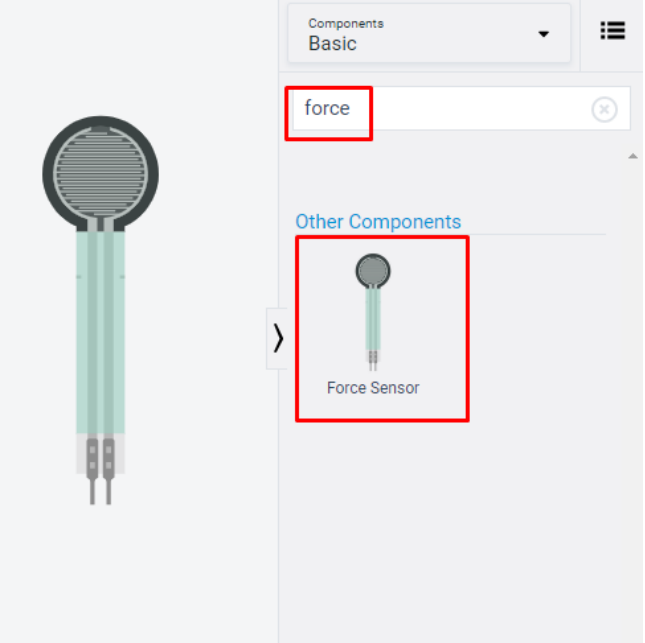
Step 2:

1. Connect a ground wire from the GND pin of Arduino to the breadboard.
2. Connect 5V wire from the 5V pin of Arduino to the breadboard.



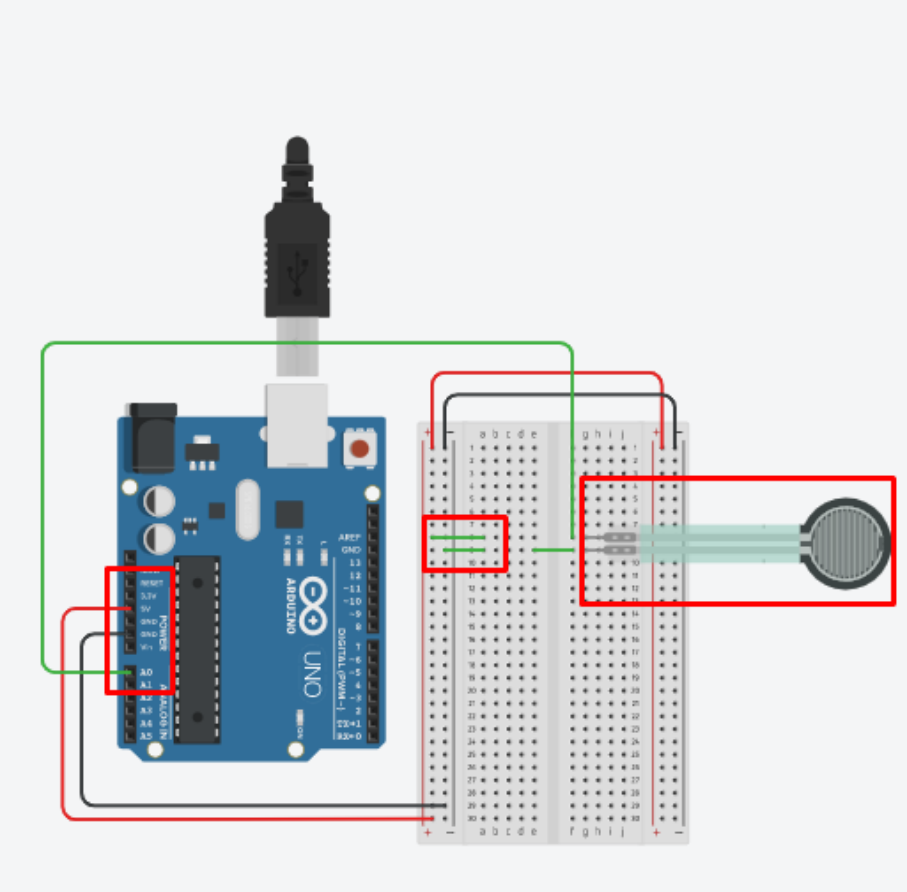
Step 3:

1. Search Force Sensor

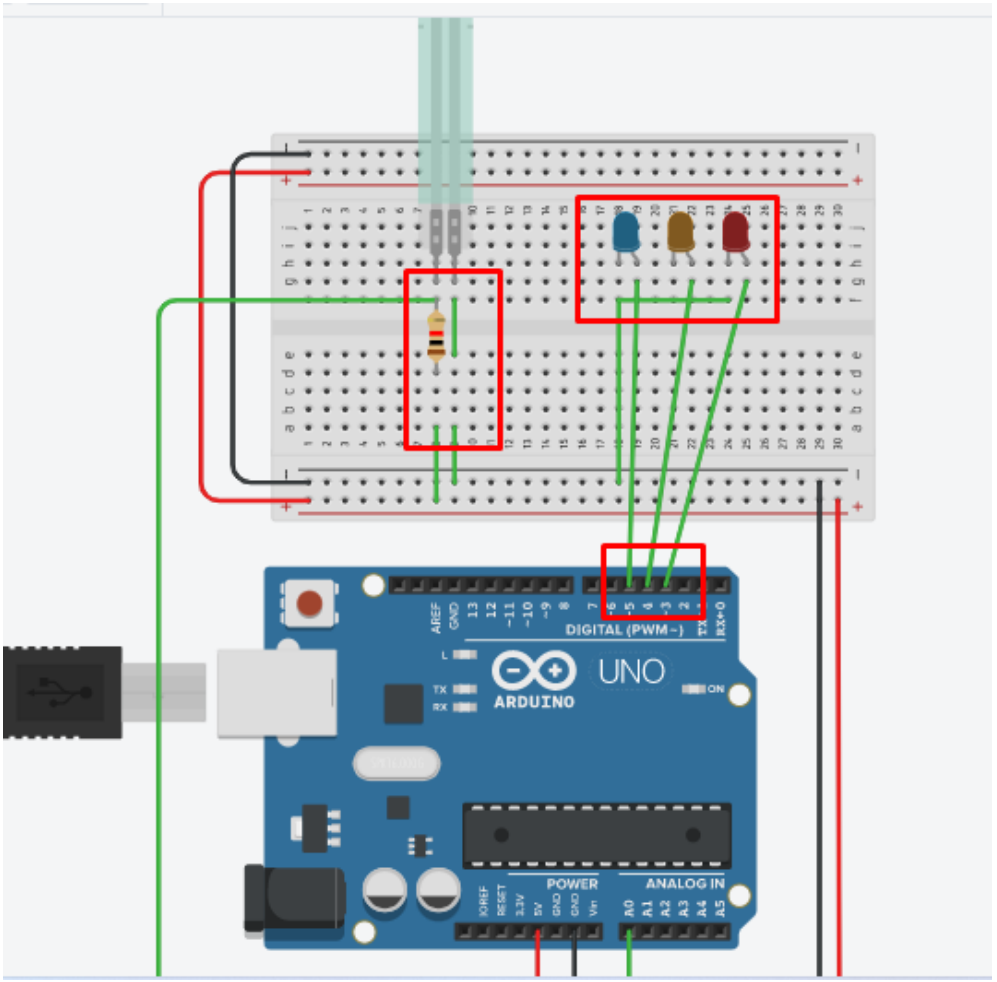


Step 4:

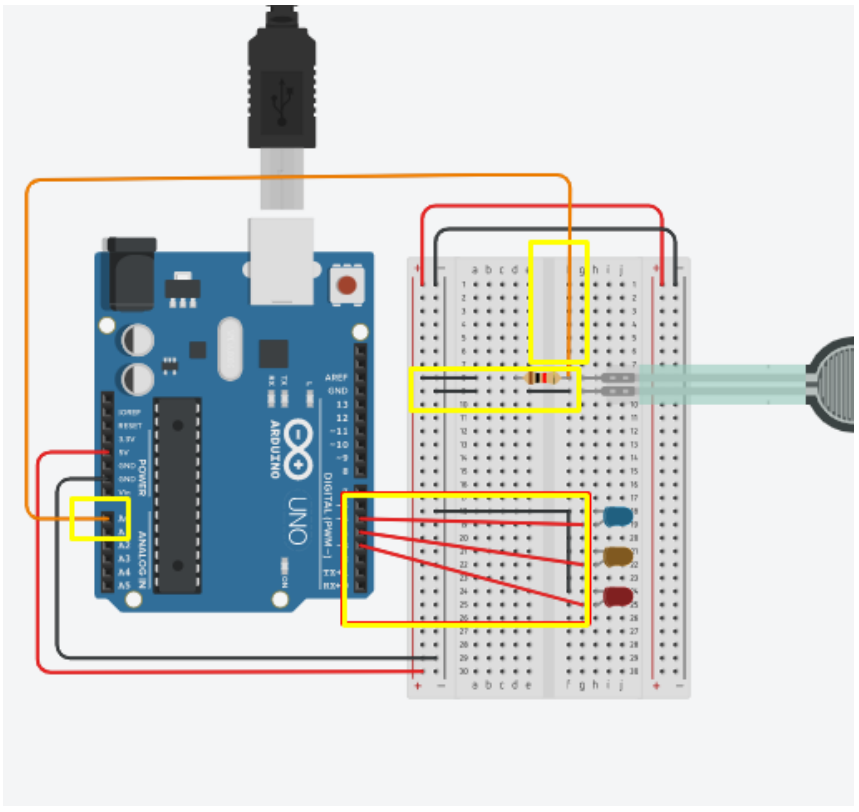
1. 5V and GRD wire connected with Force Sensor and add another wire with A0 pin of Arduino to the Force Sensor.



2. Into breadboard set 3 LEDs and Resistors and properly wiring.

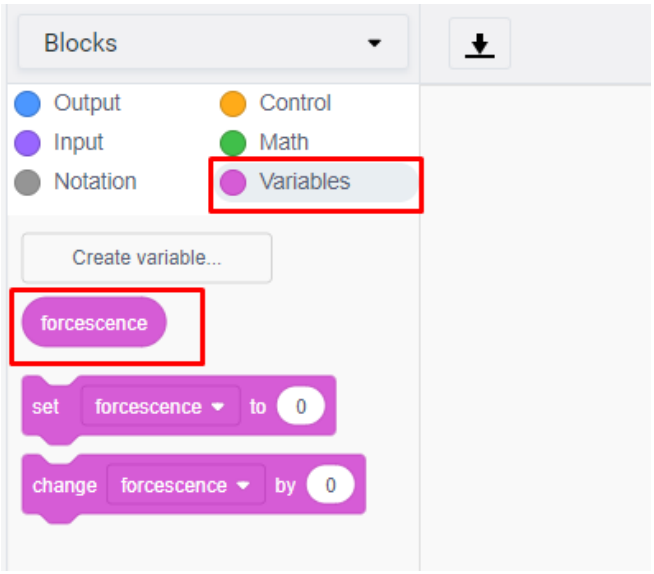


3. Then change the wire color. According to 5v, GRD and AO point.

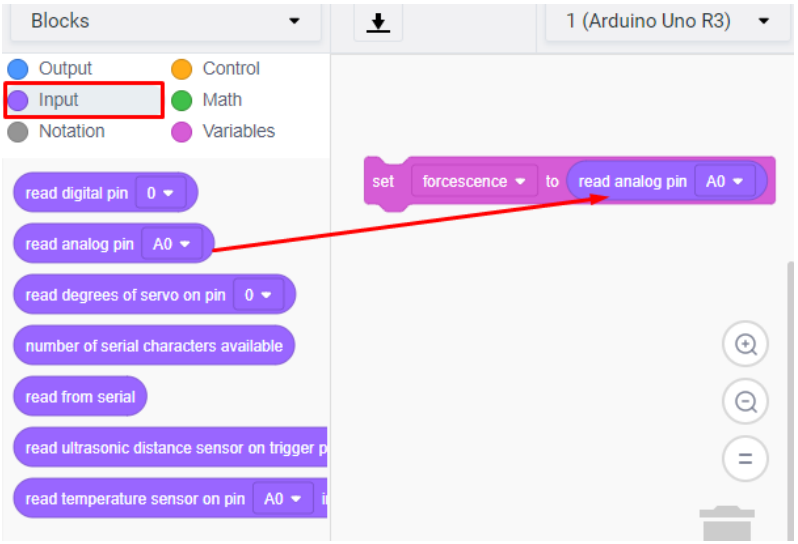


Step 5: Code Explanation

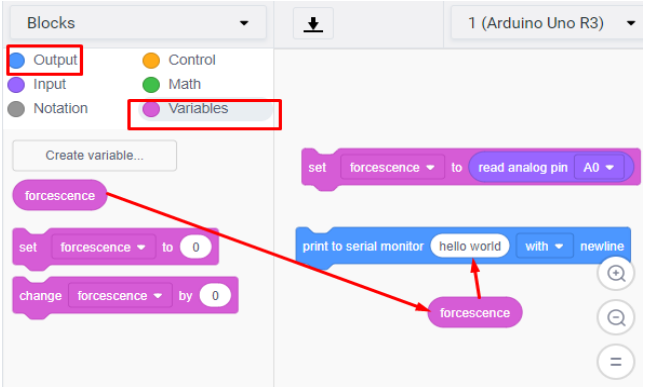
1. To store the sensor information, we create a variable called forcesense. Click the variable block and create a variable to declare the variable.

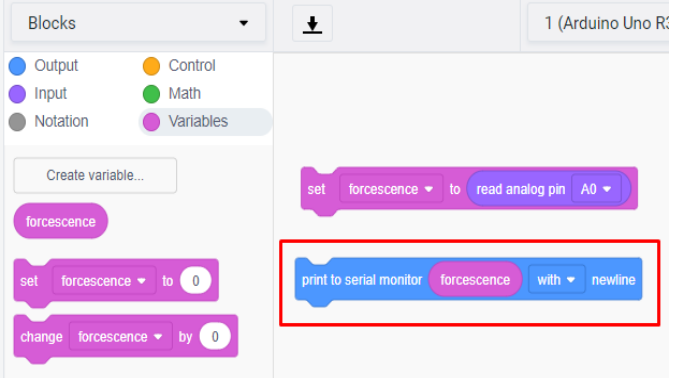


2. Drag out the set block and set it to the read analog pin A0 after defining the variable.

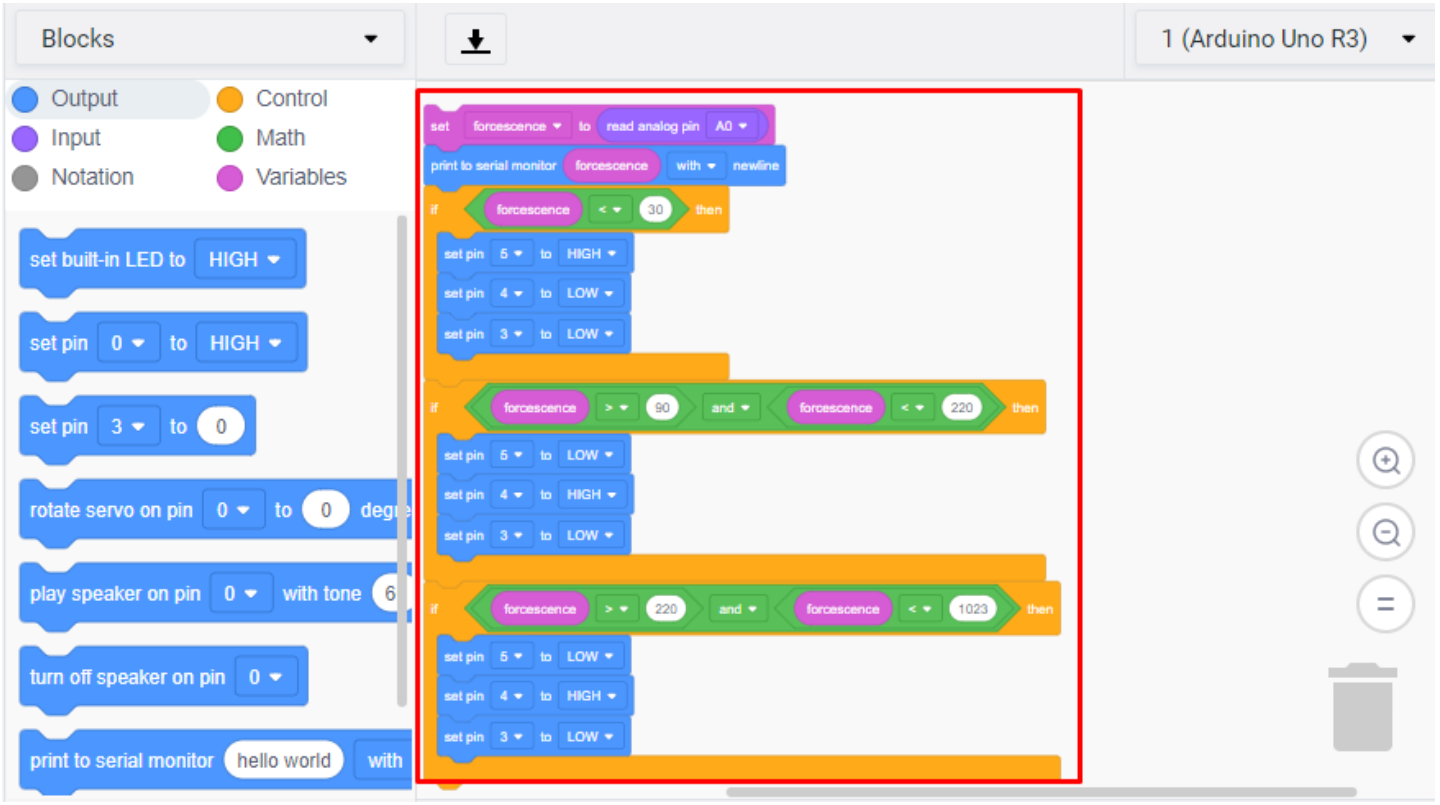


3. To print the sensor value on the screen, remove the serial monitor block.



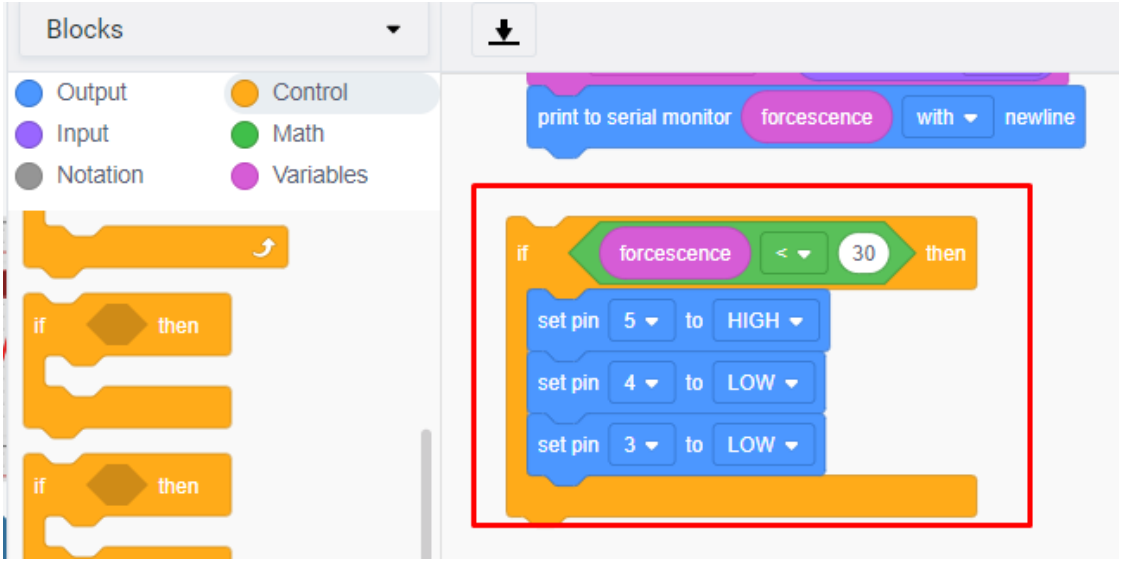


4. Let's implement the program's basic conditional statement, the 'if' block.



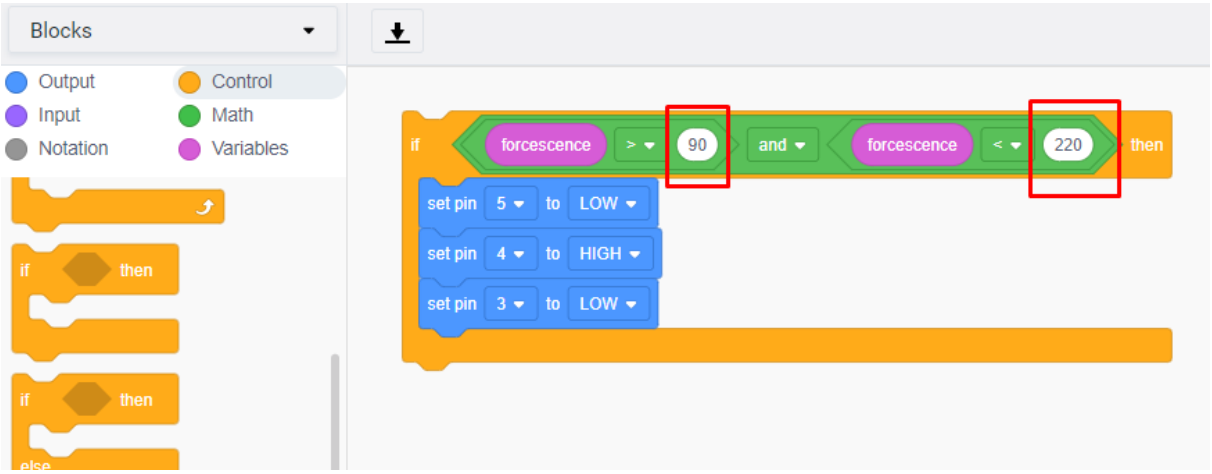
4.1 The first if statement block is as follows:

Pin 5 will be Low if the FSR value is less than 30, and pins 4 and 3 will be Low if the FSR value is less than 30. This will turn on the blue LED and turn off the other two LEDs, indicating that there is minimal pressure applied.



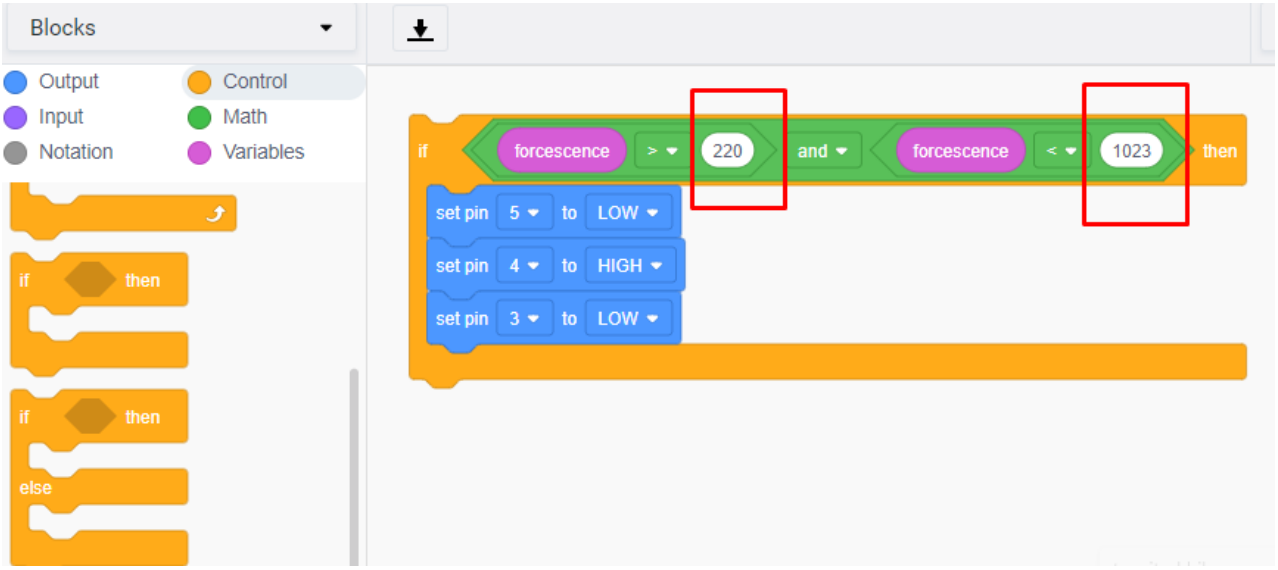
4.2 The second if statement block is as follows:

Pin 4 will be high and pins 5 and 3 will be low if the applied force value is less than 220 and also if the applied force value is more than 90. This will turn on the yellow LED and turn off the other two LEDs, indicating that the medium range force has been applied to the FSR.



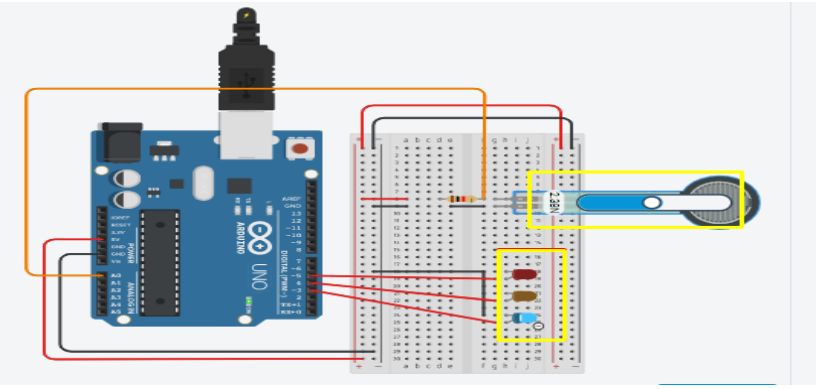
4.3 The third if statement block is as follows:

Pin 3 will be high and pins 5 and 4 will be low if the applied force value is less than 1023 and also if the applied force is more than 220. This will turn on the red LED and turn off the other two LEDs, indicating that the FSR has been subjected to a significant amount of force.



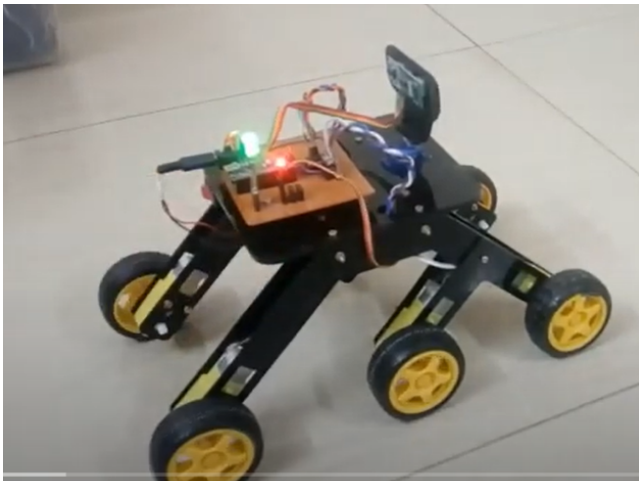
Result:

Simulation time we can see the result. When FSR value is less than 30 then we see turn on the blue LED. Again if the value is more than 90 then turn on the yellow LED and finally when force is more than 220 then we see the turn on the red LED.



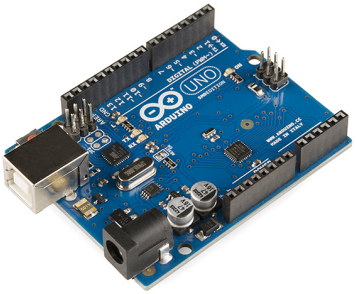
Title 3 : Ultrasonic sensor to avoid obstacles

Introduction



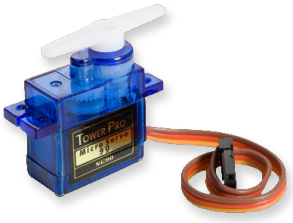
The ultrasonic distance sensor is always used in the robot to avoid obstacles. Ultrasonic distance sensor is a component that can detect the distance of obstacles with ultrasonic beams. The principle of the sensor works is to note down the time taken needed to transmit the ultrasonic beam and receive the ultrasonic beam after heating the surface of obstacles. In this topic, the hardware requirements and steps will be explained briefly.

Hardware Requirement :



Arduino Uno R3

HC-SR04 Ultrasonic Distance Sensor



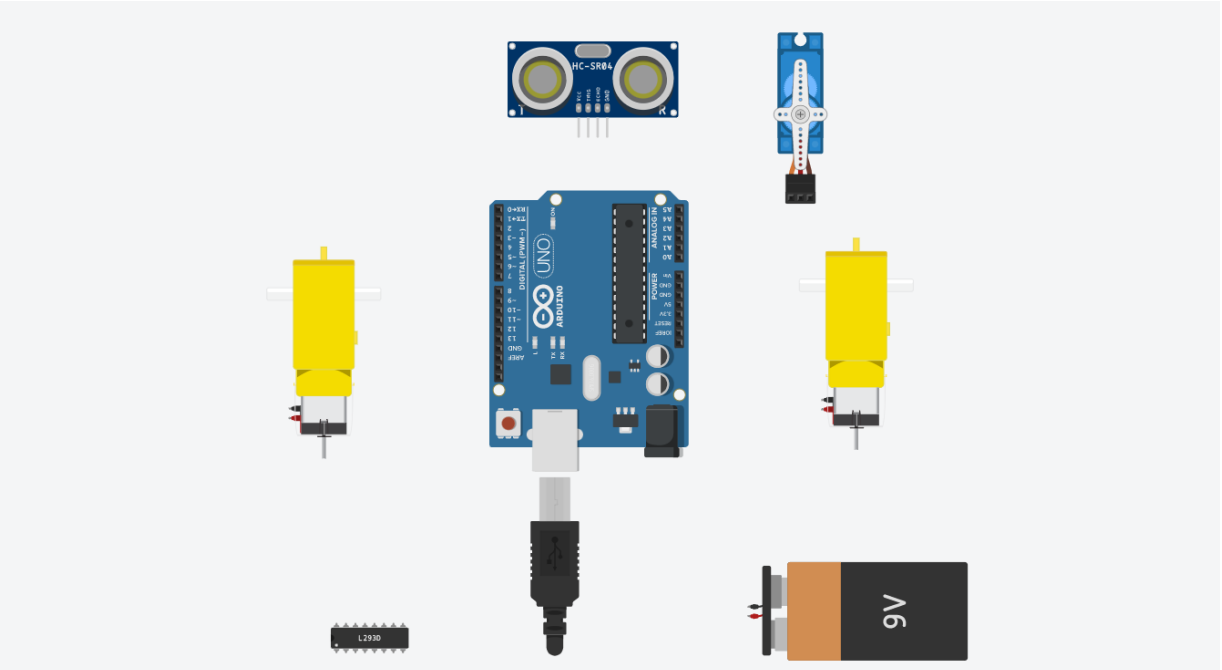
Micro Servo

Hobby Gearmotor

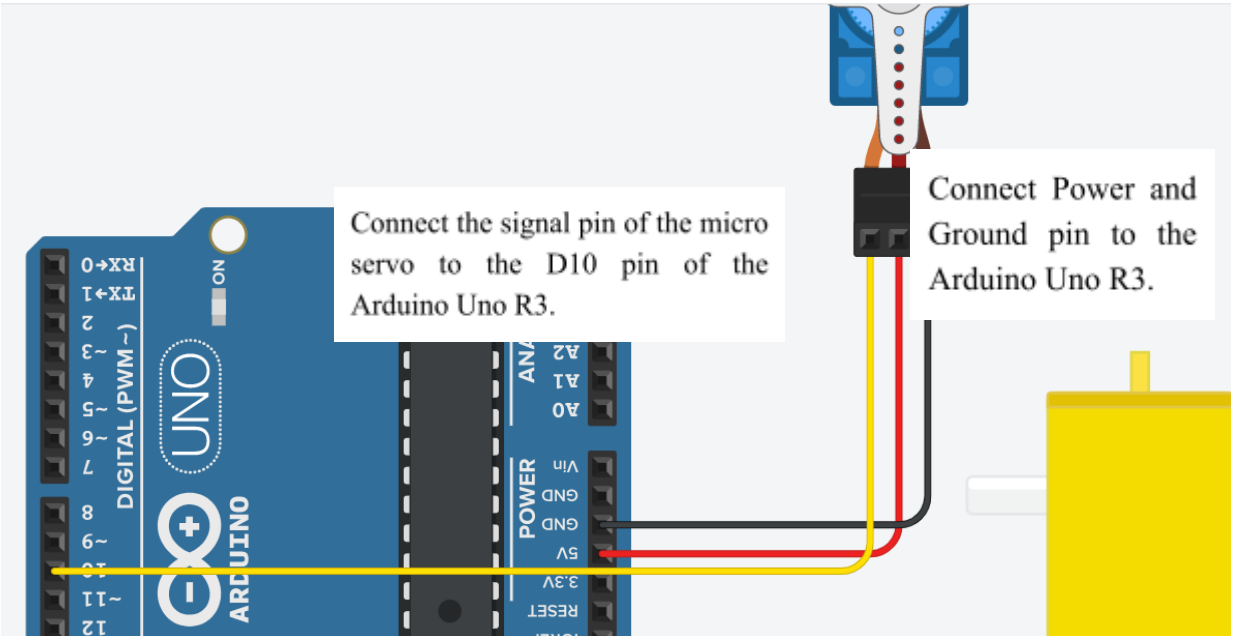
9V Battery

L293D H-bridge Motor Driver

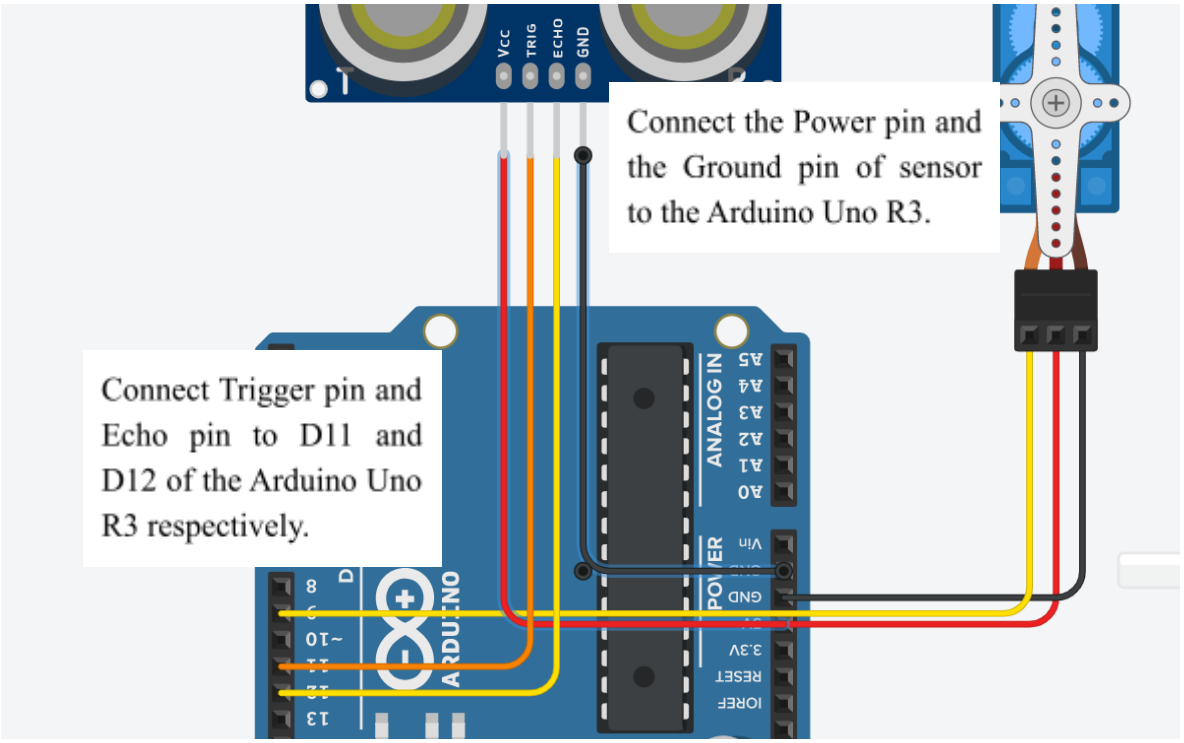
Step 1:
Add all required components into the TINKERCAD space.



Step 2:
Connect all the components with Arduino Uno R3. First of all, we connect the micro servo.

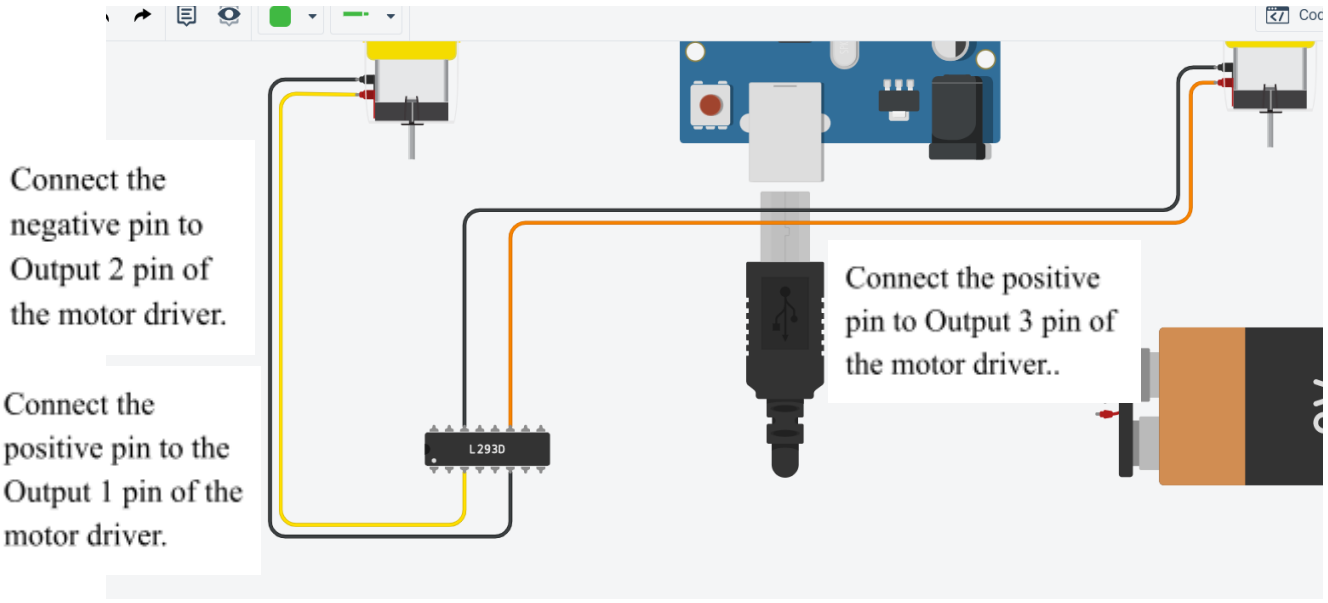


Step 3:
Next, connect the ultrasonic distance sensor with the Arduino Uno R3.

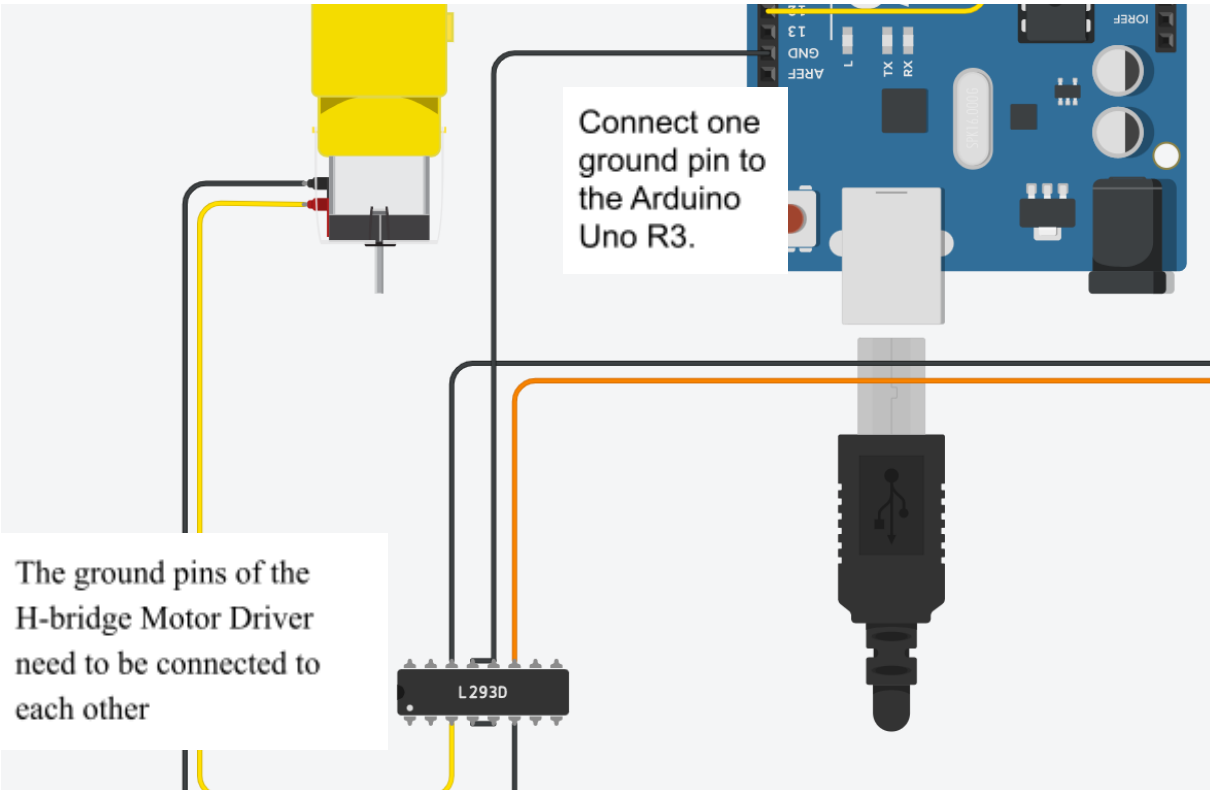


Step 4:
Then, connect the Hobby Gearmotor with the H-bridge Motor Driver.

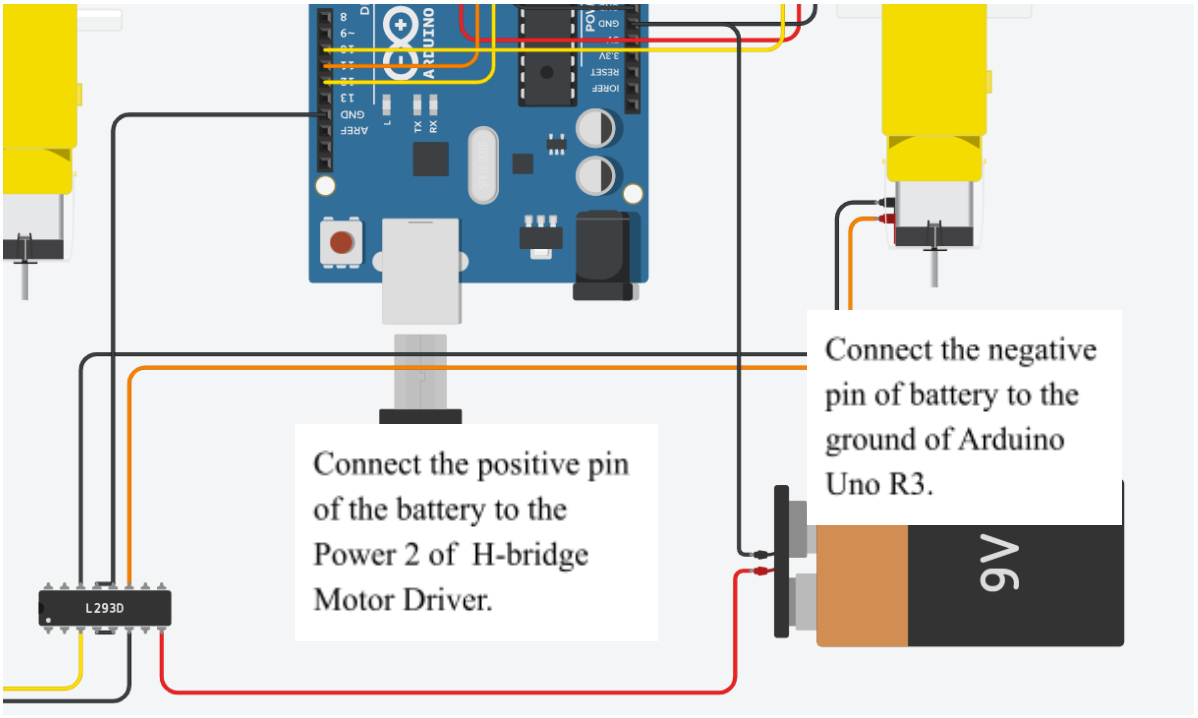
Connect the negative pin to Output 4 pin of the motor driver.



Step 5:
Connect the H-bridge Motor Driver to the ground of Arduino Uno R3.

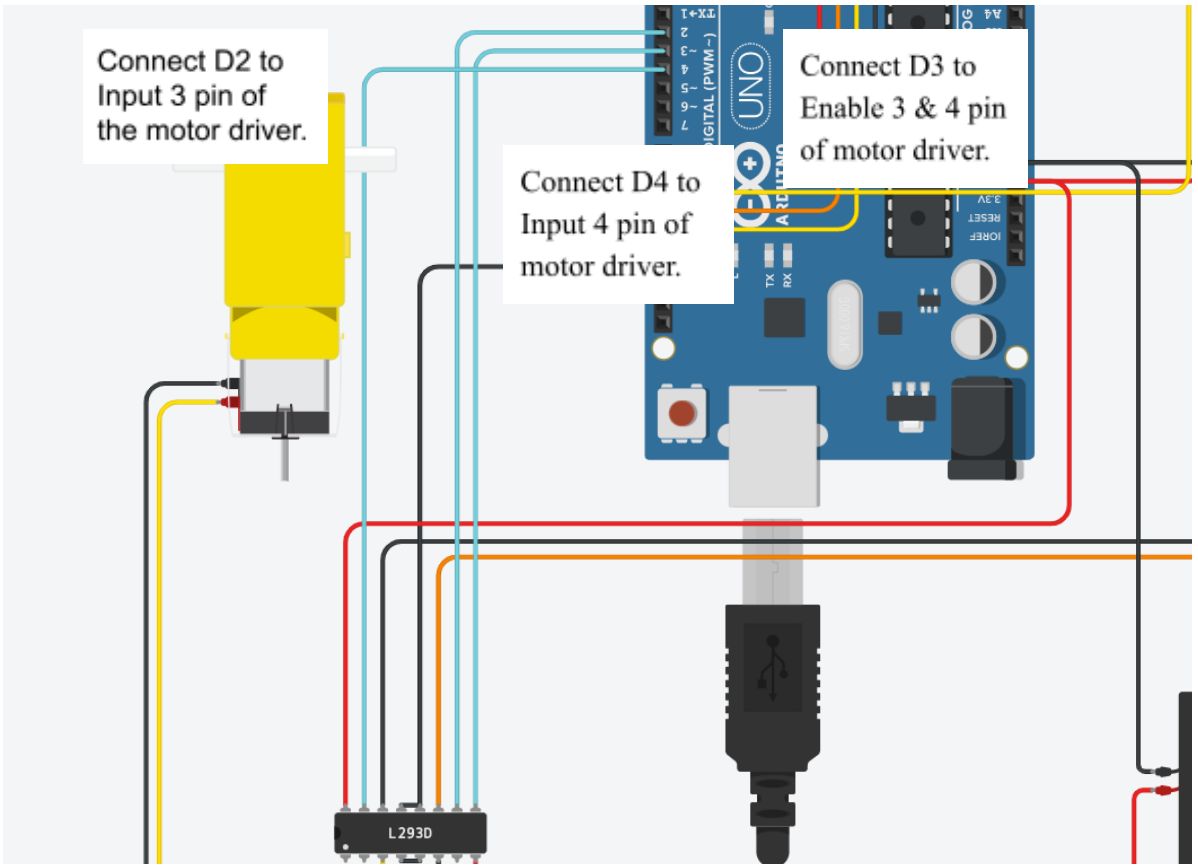


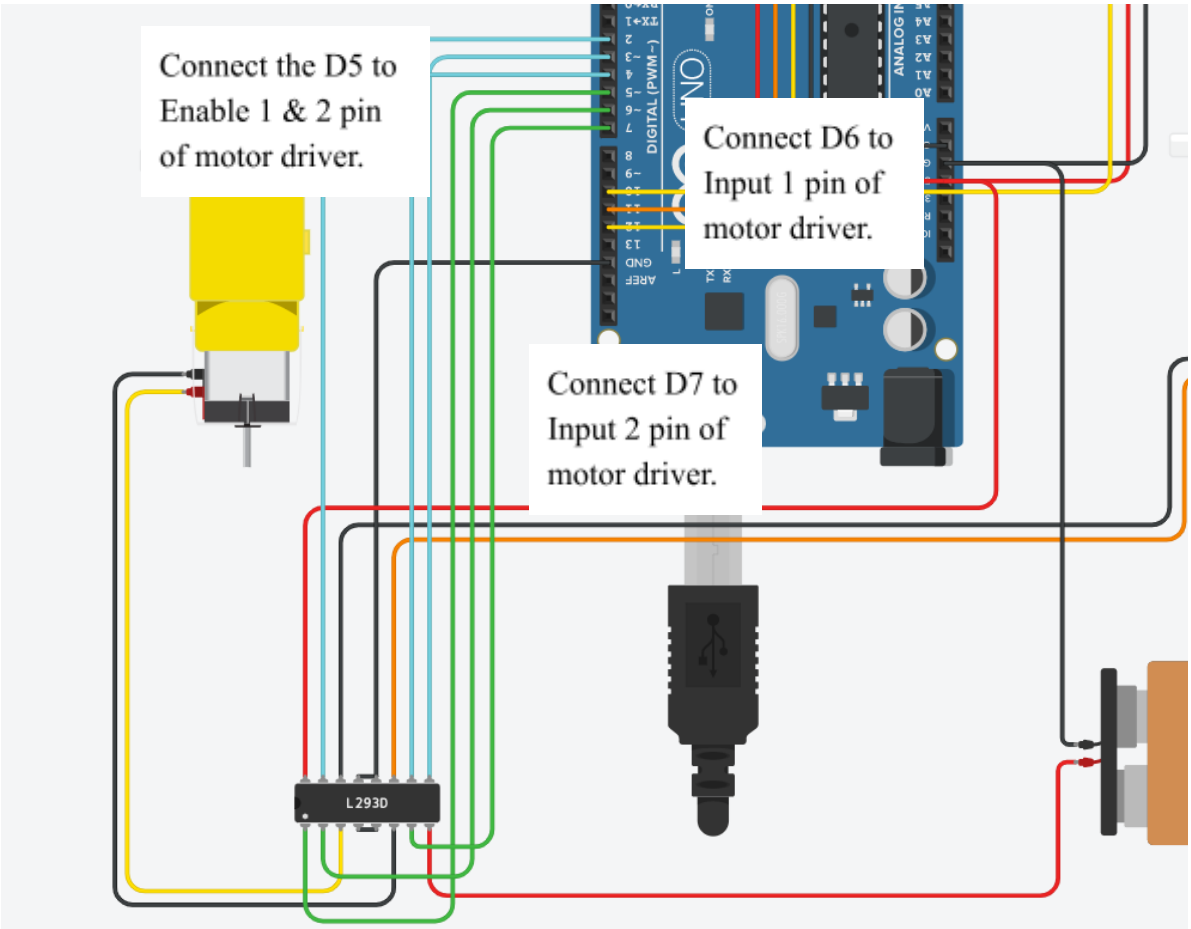
Step 6:
Connect battery to the Arduino Uno R3 and the H-bridge Motor Driver.



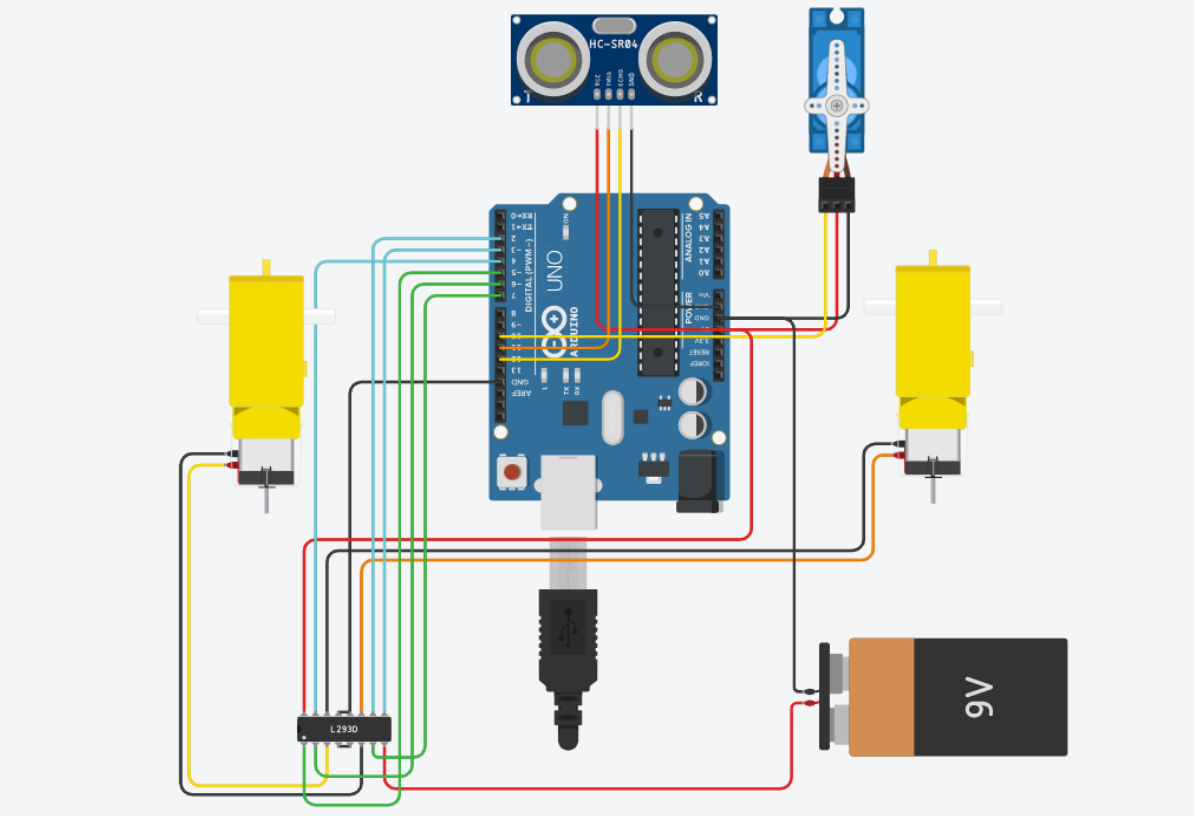
Step 7:

Connect the Power 1 pin of Motor Driver to the power of Arduino Uno R3.





The complete circuit diagram is as below.



Step 8:

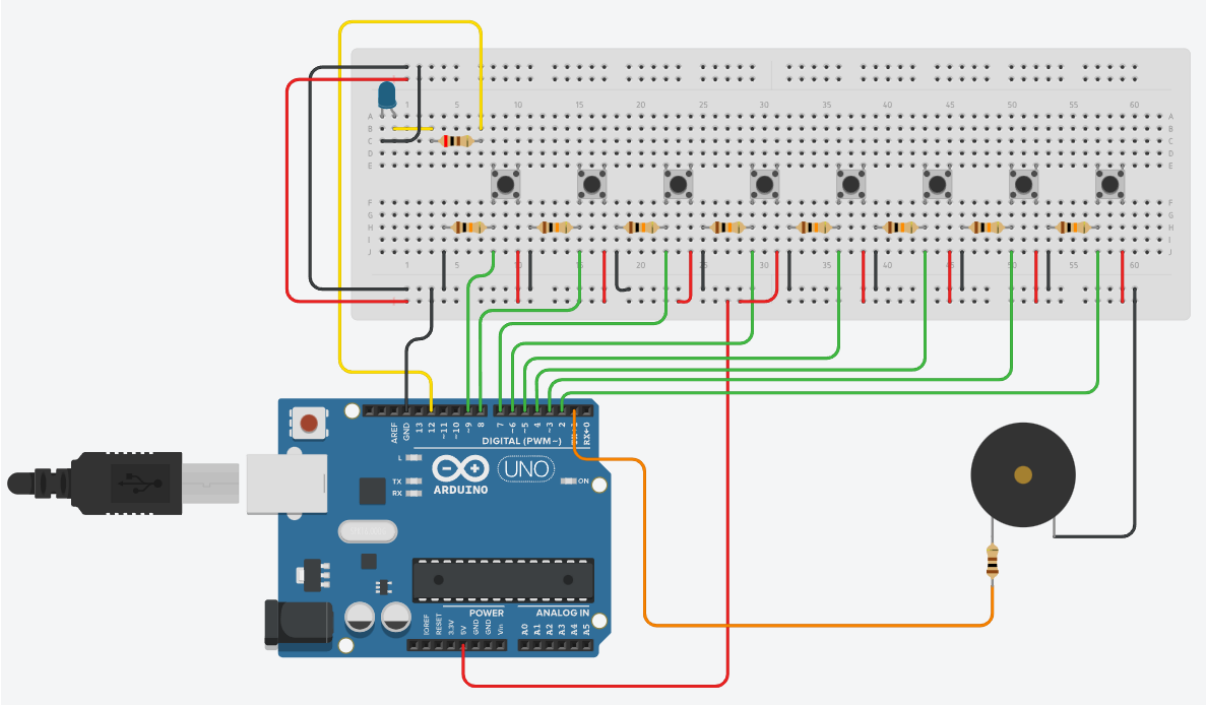
The coding to stimulate the circuit is as below.

Result:

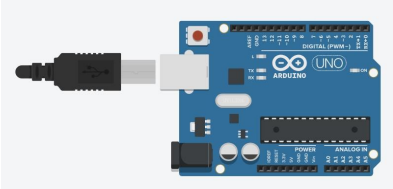
Title 4 : Piano 8 keys

Introduction

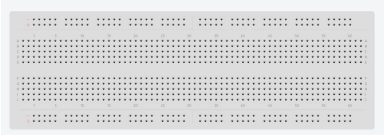
Piano is one of the oldest and most expensive musical instruments. It was invented by Bartolomeo Cristofori in 1700. Nowadays, it is commonly used by the musicians to write and create the music as well as perform the concert. To create a piano, it is particularly difficult. Despite this, the concept of the piano implementation is easy to understand. Hence, this topic will teach you step by step to create the 8 keys piano by using TinkerCad. Diagram below is the sample for Piano 8 keys:



Hardware Requirement



Arduino Uno R3



Breadboard

Led

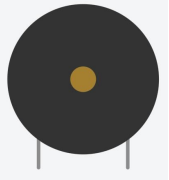
Push button x8

Resistor

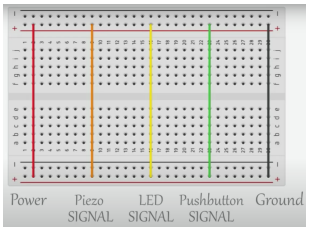
100Ω x1

200Ω x1

10kΩ x8



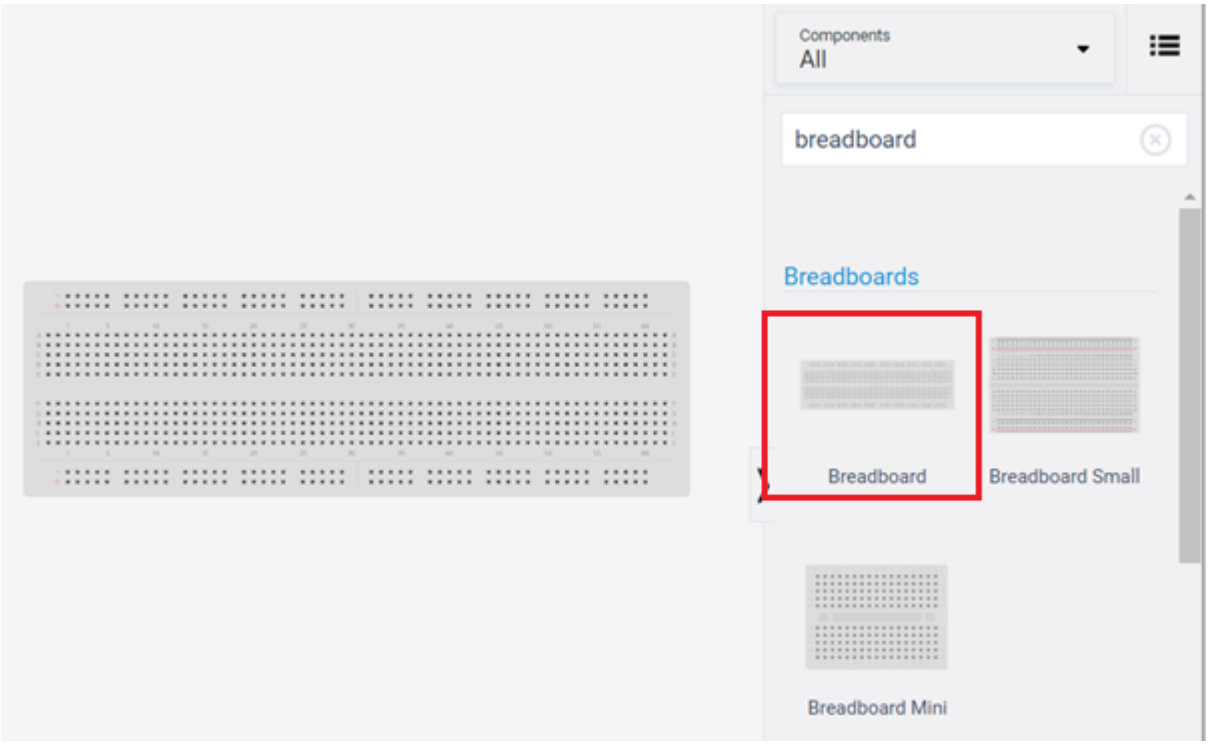
Piezo

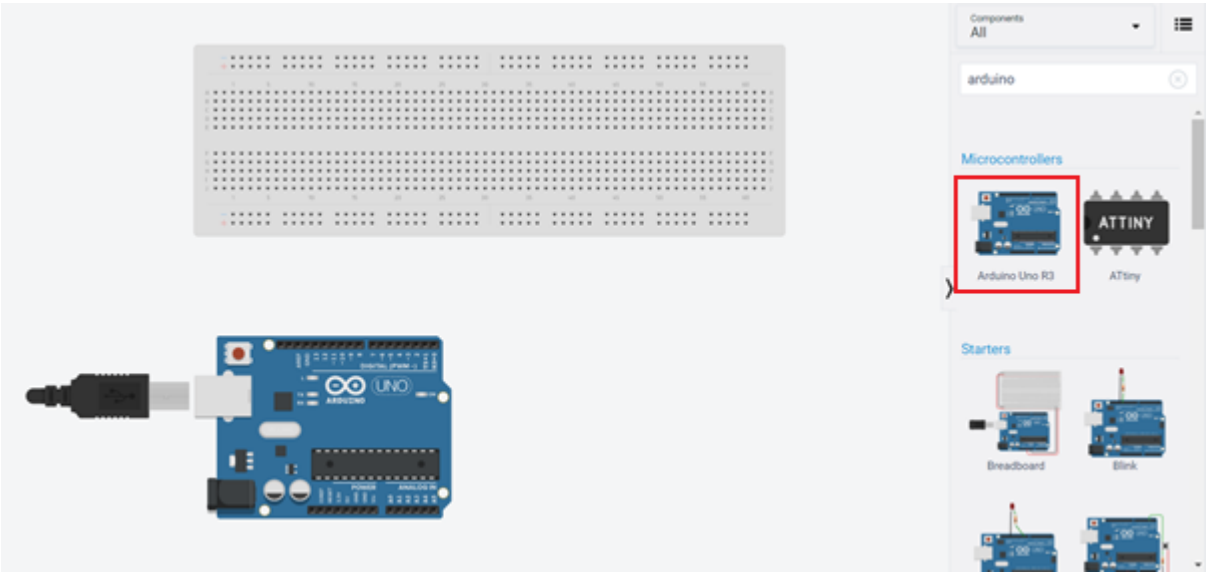


Wire

Step

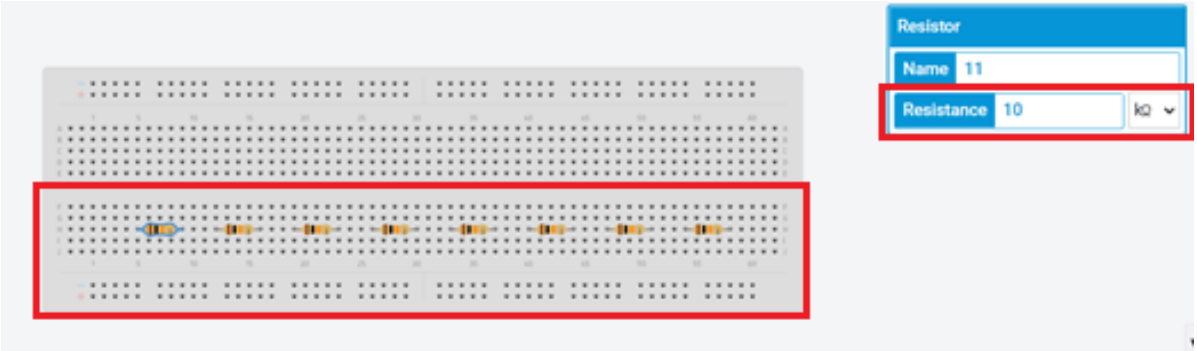
Step 1:Get Arduino Uno R3 and Breadboard





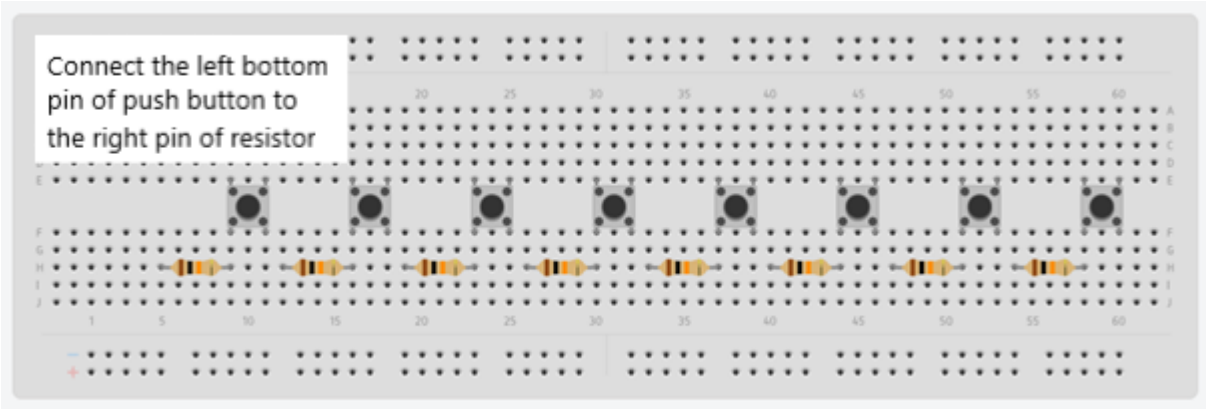
1. Find the breadboard and Arduino Uno R3 from the component panels.
2. Place them on the working board.

Step 2: Place all the 10kΩ Resistors on the Breadboard



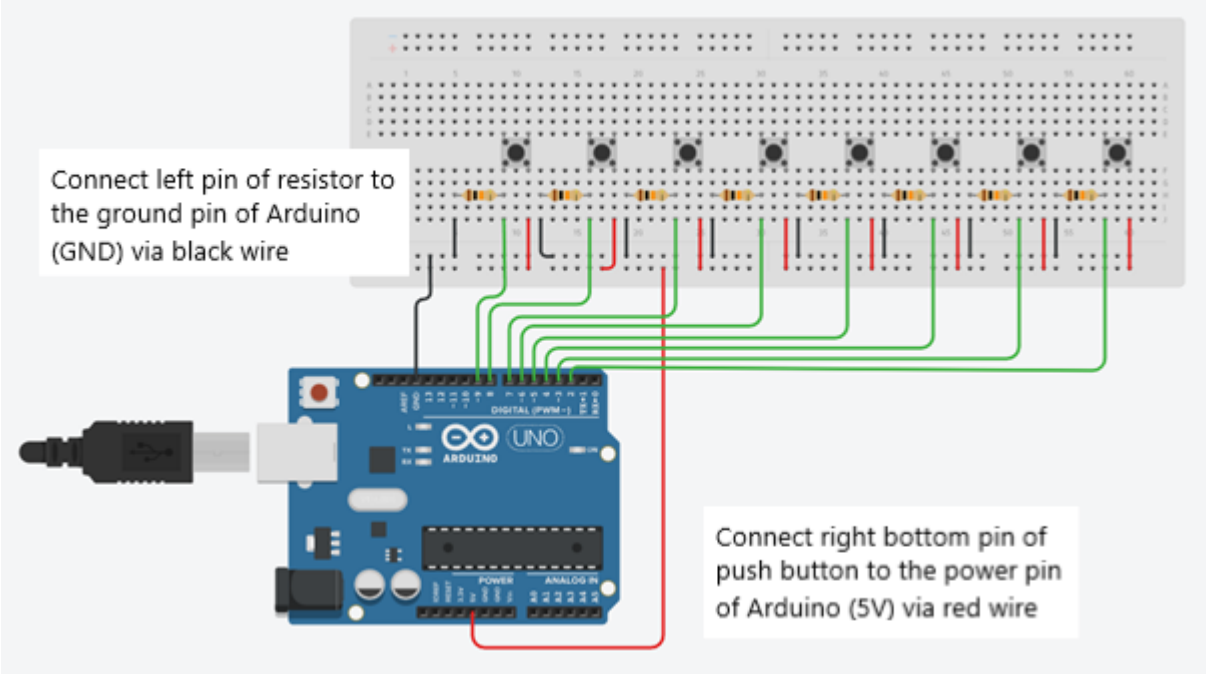
1. Select a resistor and place it on the breadboard.
2. Set resistor to 10kΩ.
3. Repeat the process for 7 times by copy 10kΩ resistor and paste it to the desired position on the breadboard.

Step 3: Place all the Push Button on the Breadboard

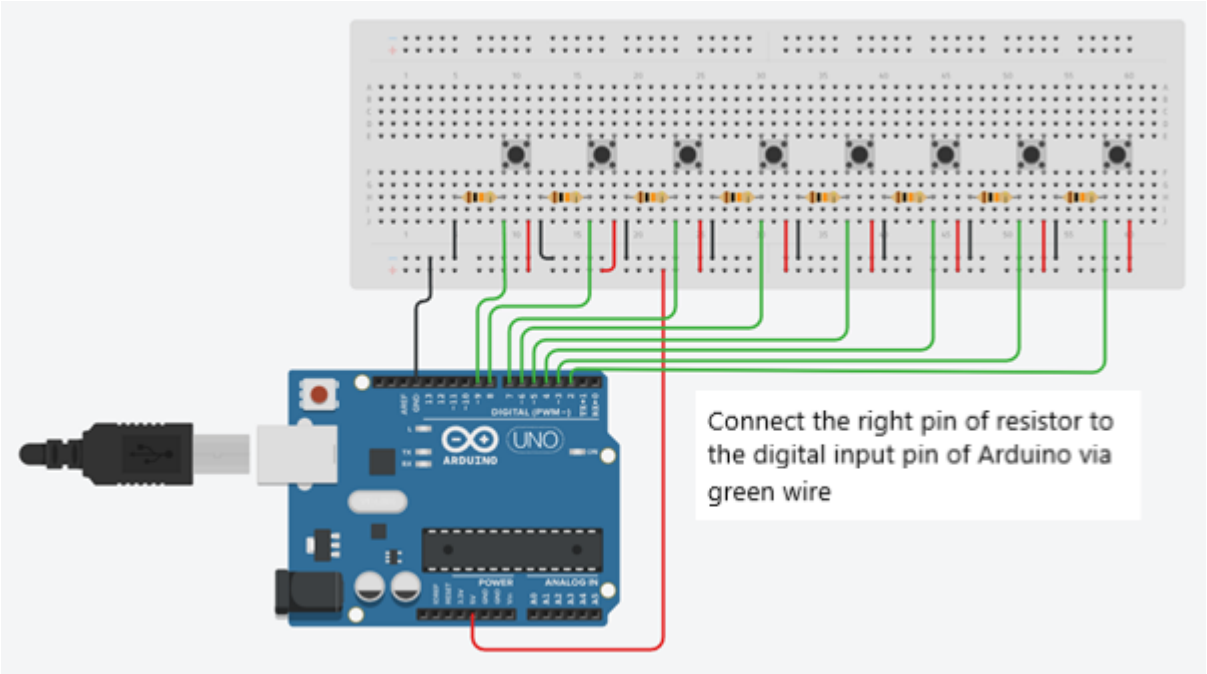


1. Select a push button.
2. Connect push button with the resistor in the breadboard
3. Repeat the process 2 for 7 times by copy and paste the push button on the breadboard.

Step 4: Connect Resistors and Push Button to the Arduino

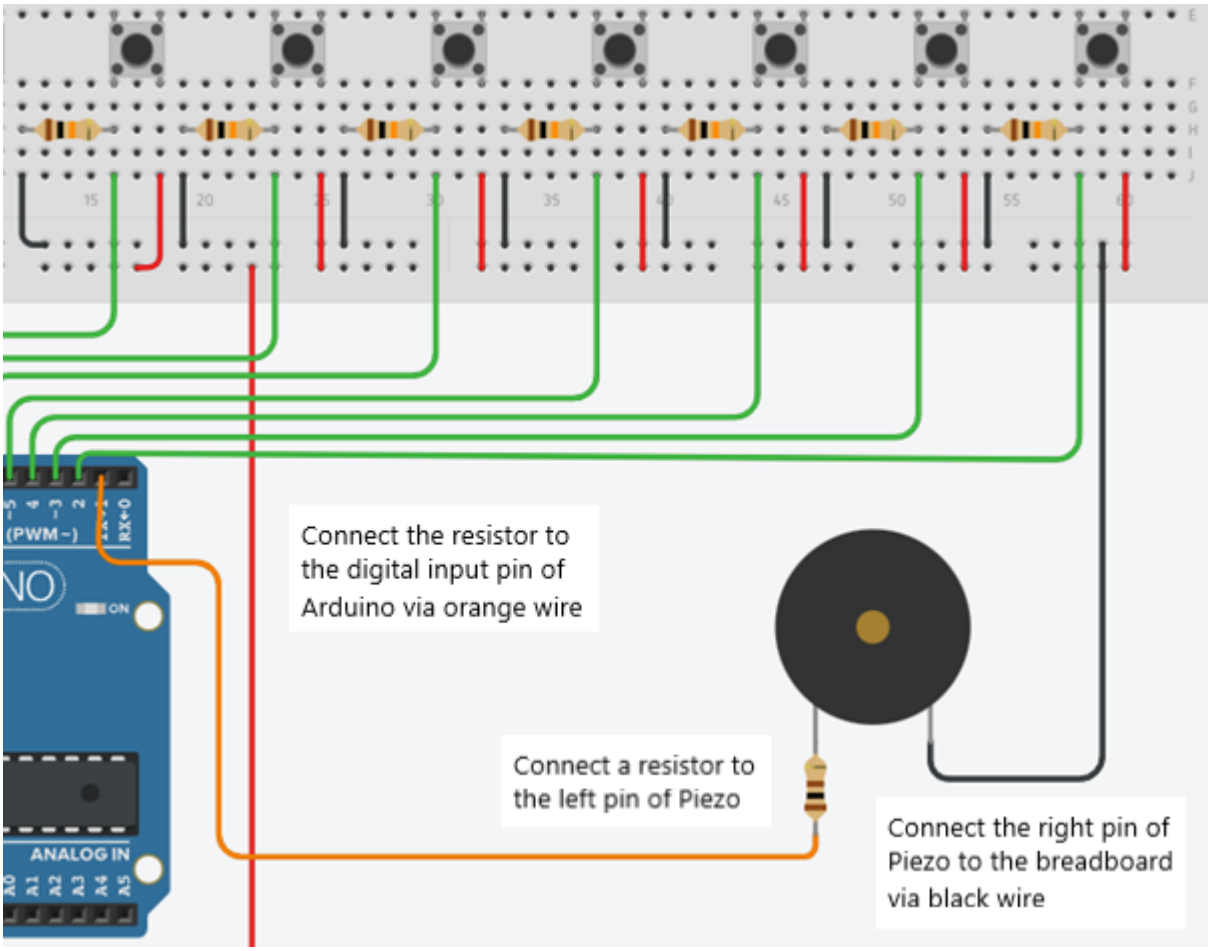


1. Connect a ground wire from the GND pin of Arduino to the breadboard.
2. Extend the ground wire from the breadboard to the first resistor.
3. Repeat the process 2 for 7 times according to the position of the following resistors.
4. Connect a power wire from the 5V pin of Arduino to the breadboard.
5. Extend the power wire from the breadboard to the first push button.
6. Repeat the process 5 for 7 times according to the position of the following push buttons.



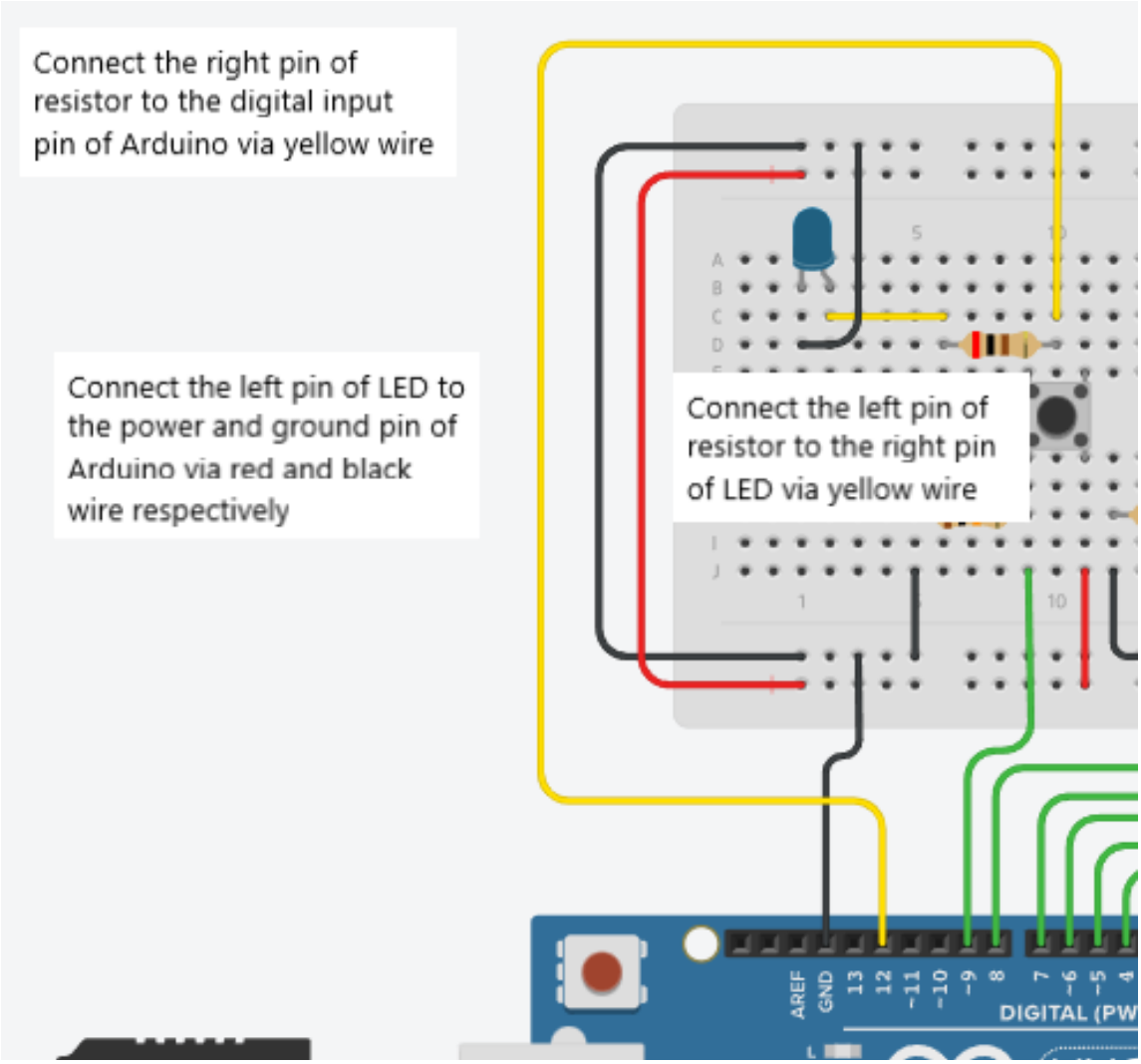
7. Connect the signal wire from the -9 pin of Arduino to the first resistor.
8. Repeat the process 7 for 7 times to the following resistors but they are connected to different digital input pins respectively as shown in the diagram.

Step 5: Connect Piezo to the Arduino



1. Select a piezo.
2. Select a resistor and connect it to the piezo.
3. Set the resistor to 100Ω.
4. Connect the resistor to the -1 pin of the Arduino via the signal wire.
5. Connect the piezo to the breadboard using ground wire.

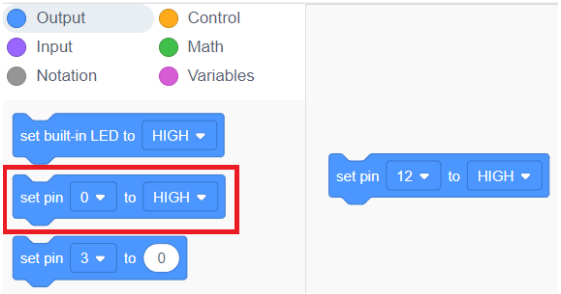
Step 6: Place and Connect LED to the Arduino



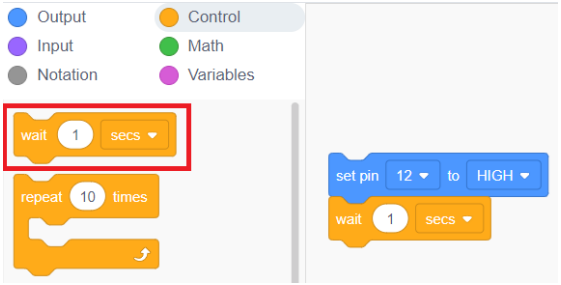
1. Select a LED and place it on the desired position of breadboard
2. Connect the resistor to the 12 pin of Arduino via signal wire.
3. Connect the resistor to the LED via signal wire
4. Connect the LED with power wire and ground wire by extending the wires from the breadboard.

Step 7: Coding

Code for LED

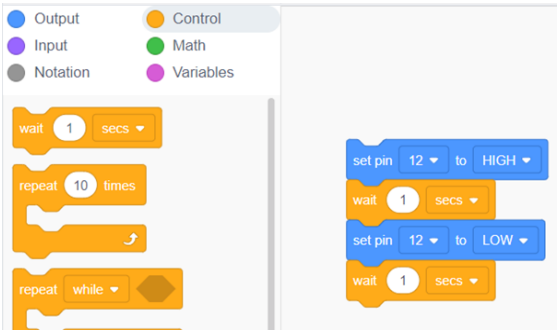


1. Select the “set pin” output block.
2. Set the pin number to 12.
3. Set it to HIGH output.



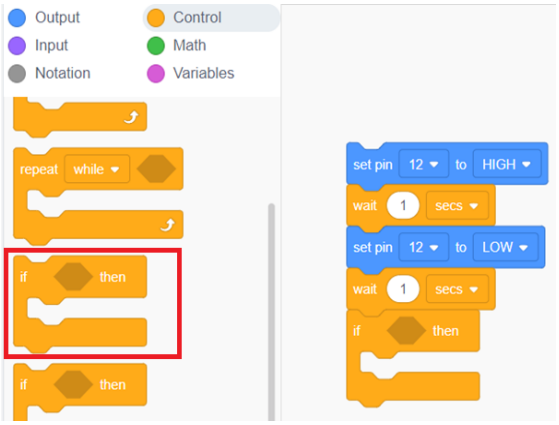
4. Select the “wait” control block.
5. Set it to 1 seconds.

6. Duplicate the set of blocks.

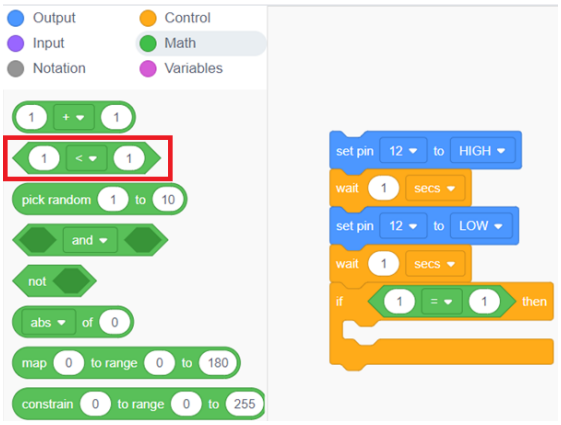


7. Place the new set below the previous block set.
8. Set the pin of the new set to LOW output

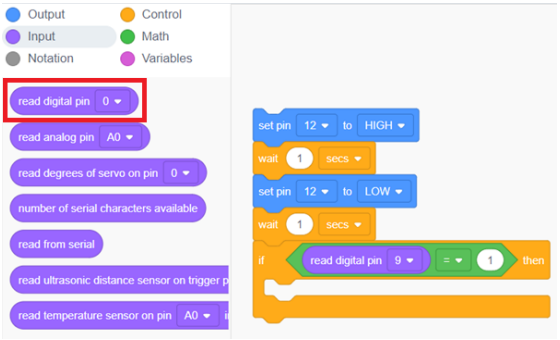
Code for Piezo



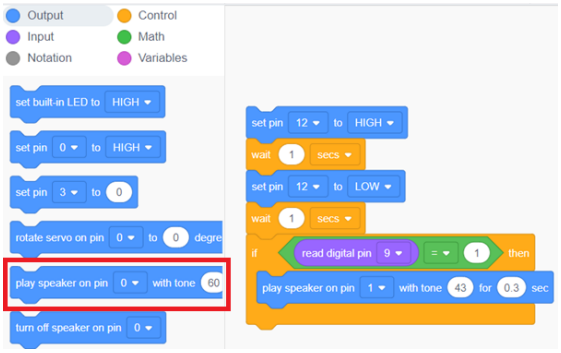
1. Select a “if and then” control block.
2. Place it below the code block for LED .



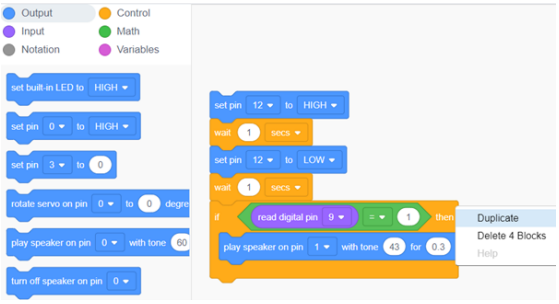
3. Select a “comparison” math block.
4. Place it inside the condition column of the “if and then” block.
5. Set the operator of the math block to “=”.



6. Select the “read digital pin” input block.
7. Place it inside the column before the operator of the math block.
8. Set the pin number to 9.

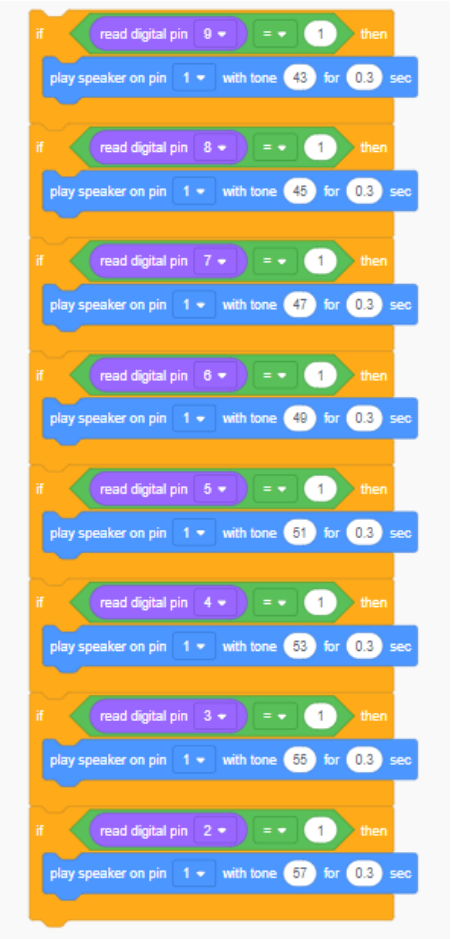


9. Select the “play speaker” output block.
10. Place it inside the “if and then” block.
11. Set the pin number to 1.
12. Set the speaker tone to 43 and last for 0.3 seconds.

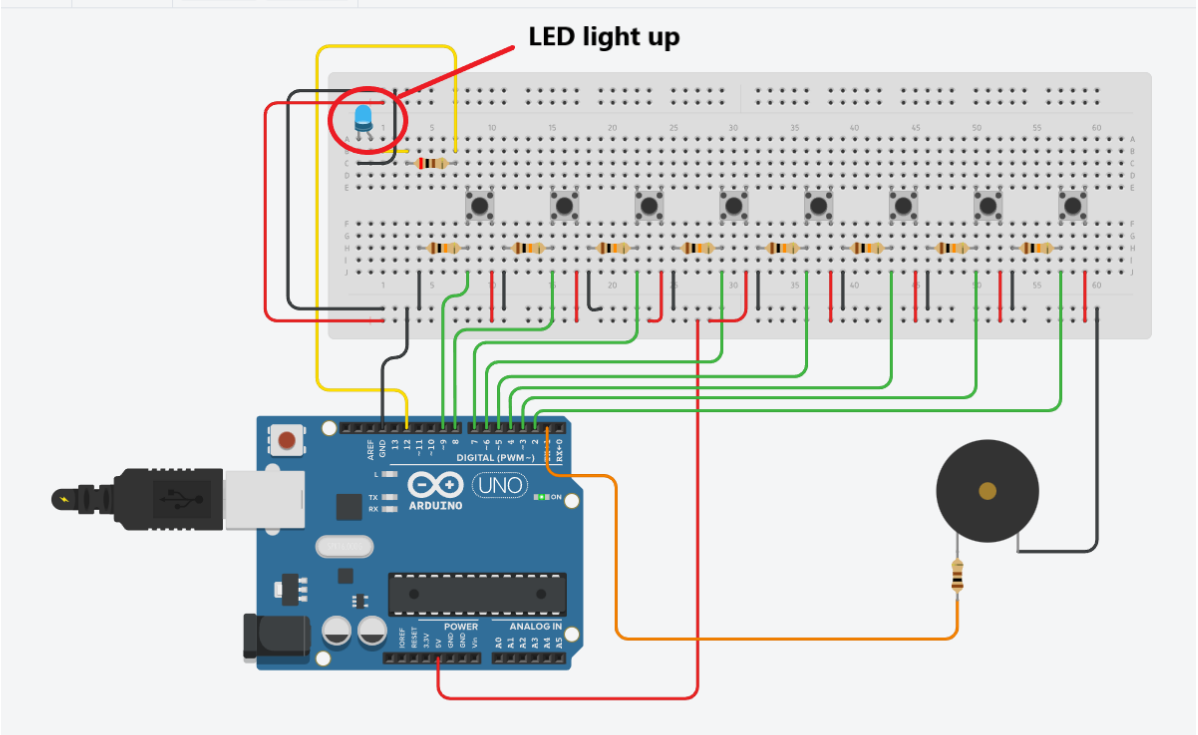


13. Duplicate the “if and then” block for 7 times.

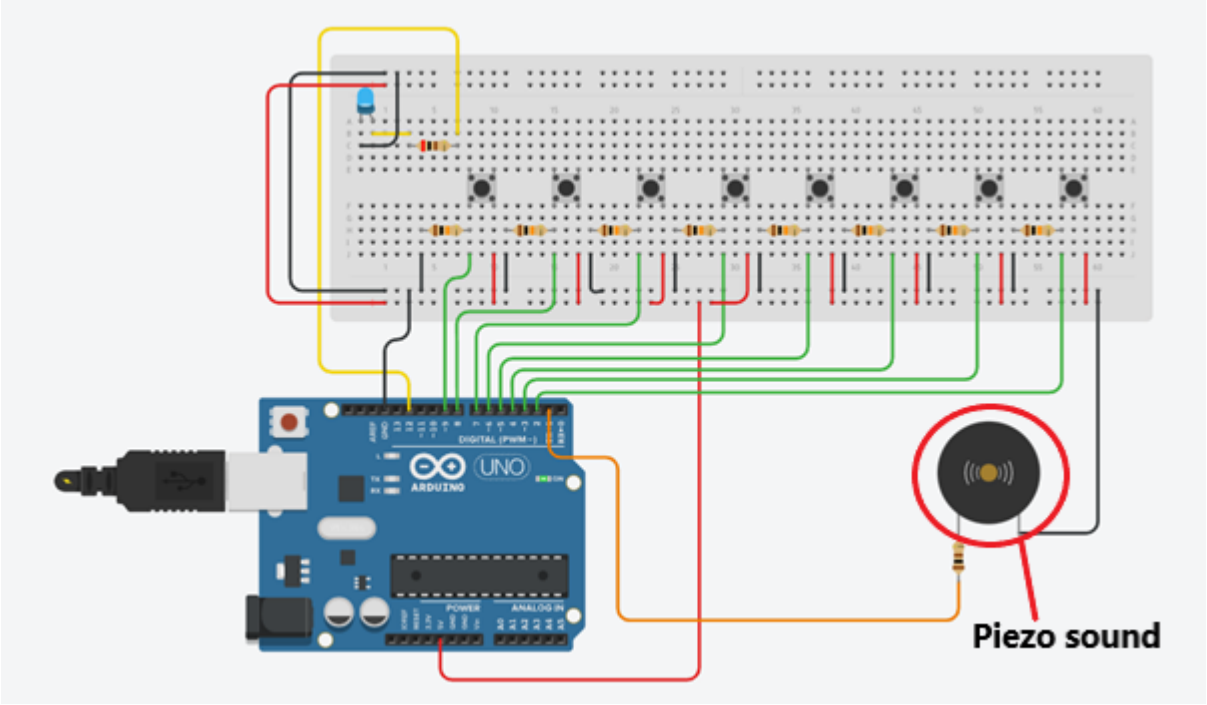
14. Set and change the pin number from 8 to 2 according to the sequence.
15. Set and change the speaker tone from 45 to 57 with the increment of 2 according to the sequence.



Result:



When the power is on, the LED will blink which shows that the piano is prepared to be used.



When you press the push button, the piezo will sound. Different push buttons will produce sound in different frequencies.