Internet-of-Things
MCA 5036

By,

Dr. Vidya Rao
Assistant Professor,
Dept of DSCA, MIT, MAHE

Outline

- Arduino Uno
- Sensor
- Programming Construct
- Examples

Arduino Uno

Arduino:

- **Arduino is an open-source electronics platform based on easy-to-use hardware and software.**
- **Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online.**

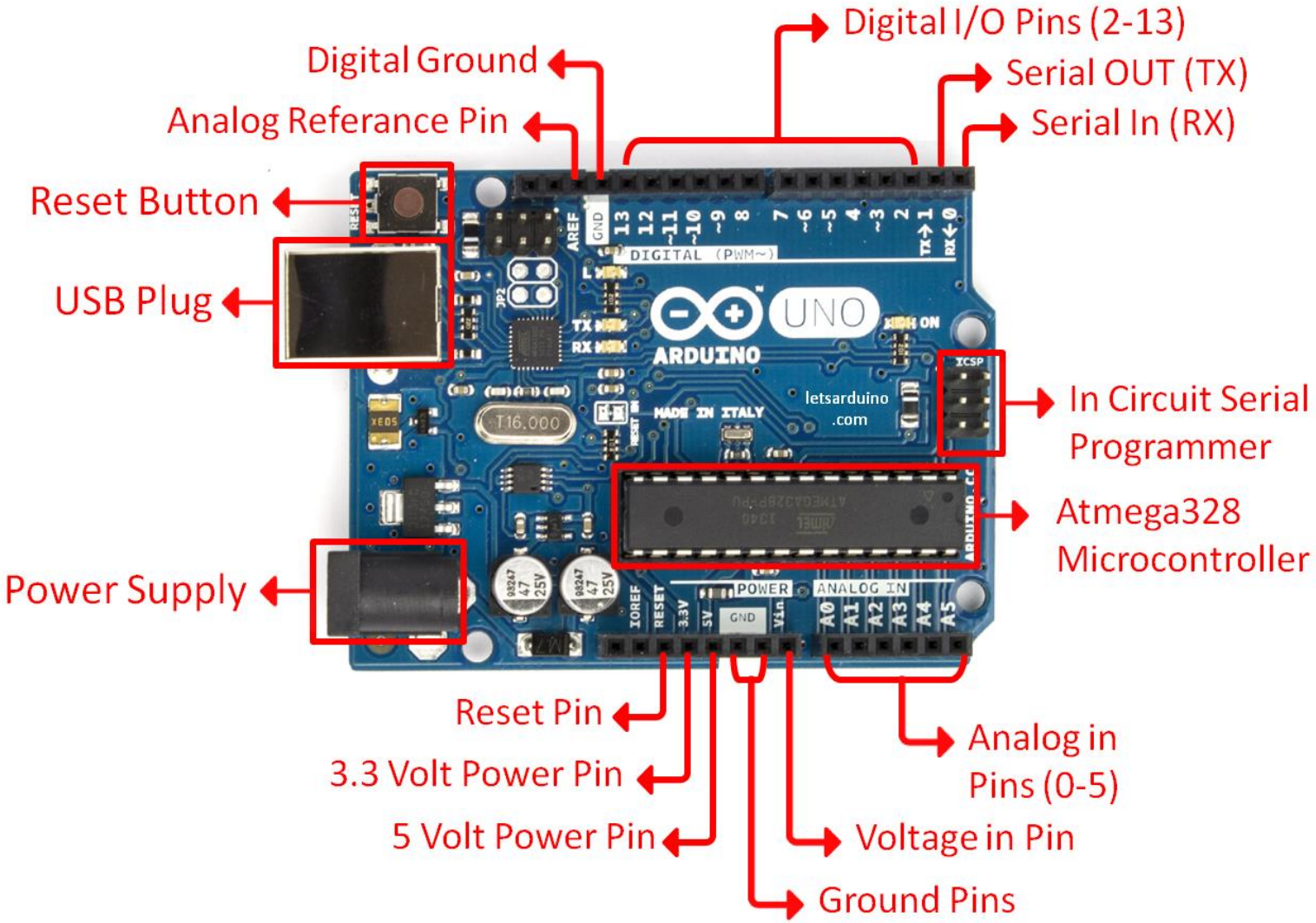
Arduino Uno (contd..)

Arduino UNO:

- Arduino Uno is a microcontroller board based on the ATmega328P.
- It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.
- "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases.

Arduino Uno

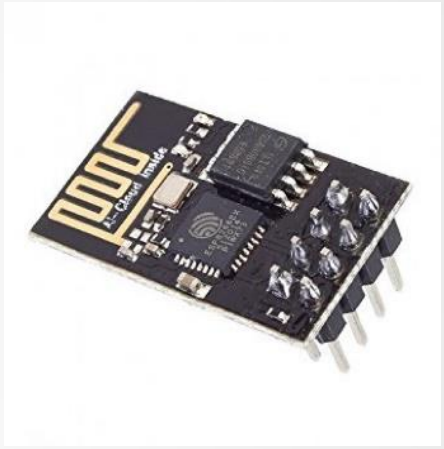
(contd..)



Hardware

USB Type A/B for Arduino

HC-05 Bluetooth

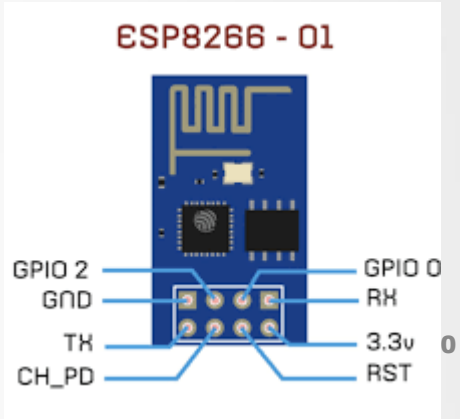


ESP-8266 WiFi

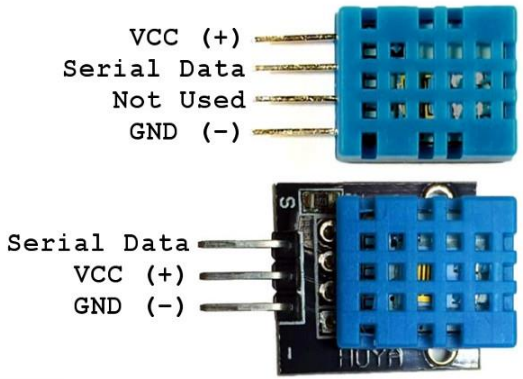
Xbee Zigbee Module

XBee Module

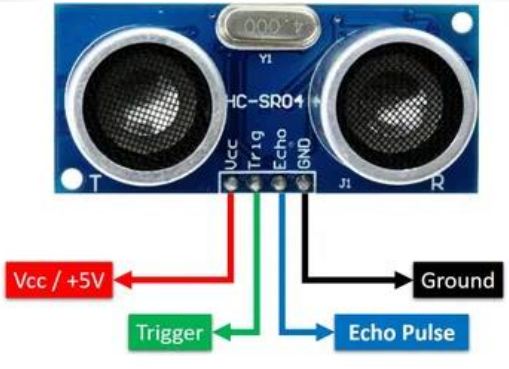
XBee Pin Configuration



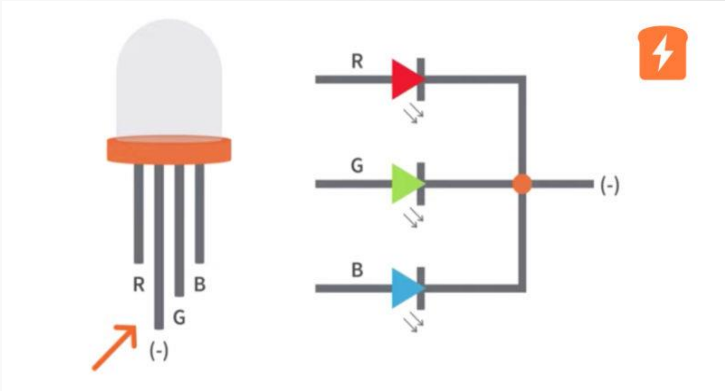
Hardware



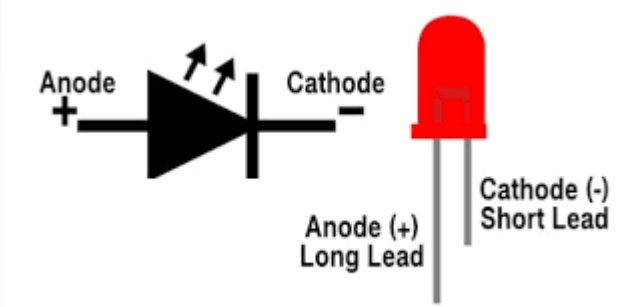
DHT-11/22 Sensor



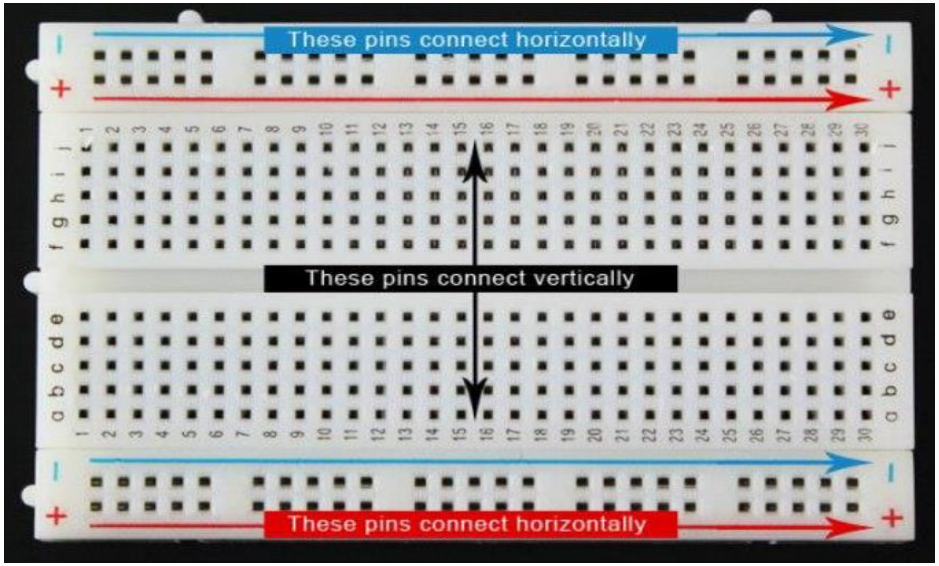
Ultra Sonic Sensor



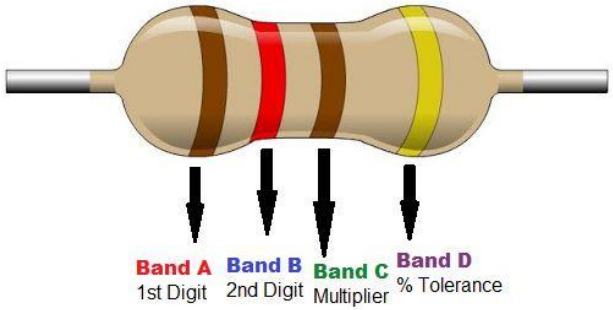
LED

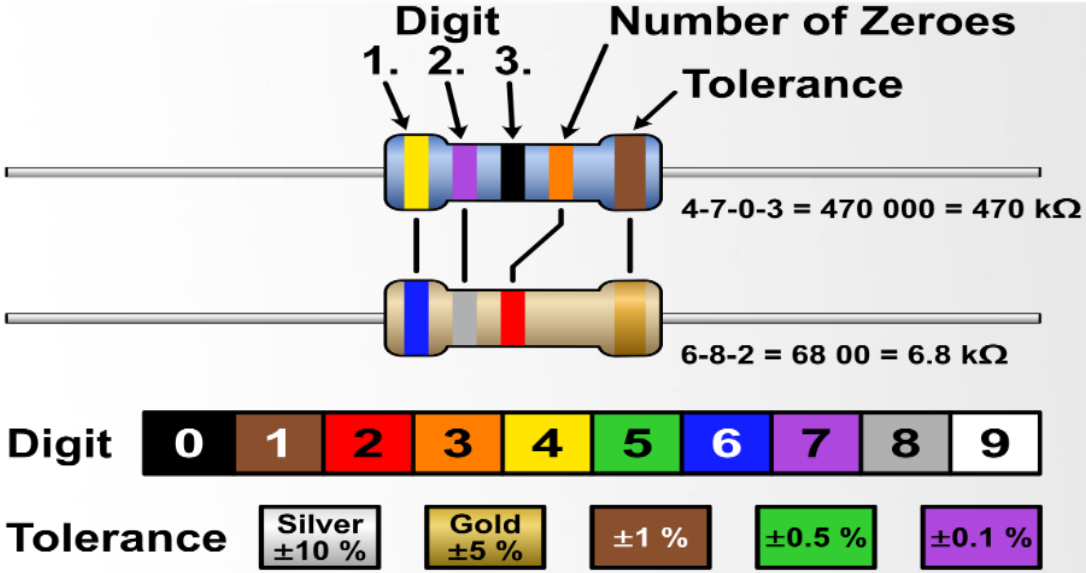


Bread Board

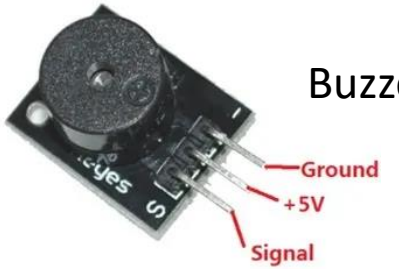


Hardware





Register and calculation



Buzzer

Connecting Cables

Hardware

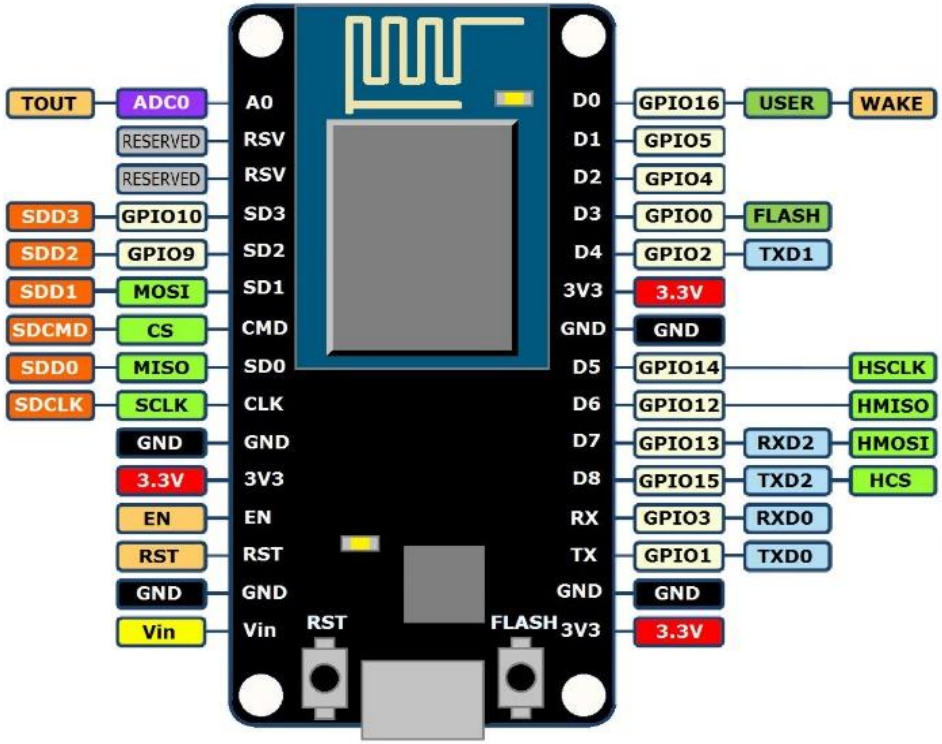
DC motor & Wheel

Servo motor

OLED display and LCD display

3x3 and 4x4 Key pad

Hardware



NodeMCU

Arduino Shields:

1. WiFi Shield
2. LCD Shield
3. GPS Logger Shield
4. MP3 Music Maker Shield
5. Ethernet Shield
6. Motor/Stepper/Servo Shield

Arduino IDE

Arduino programs run on two basic sections:

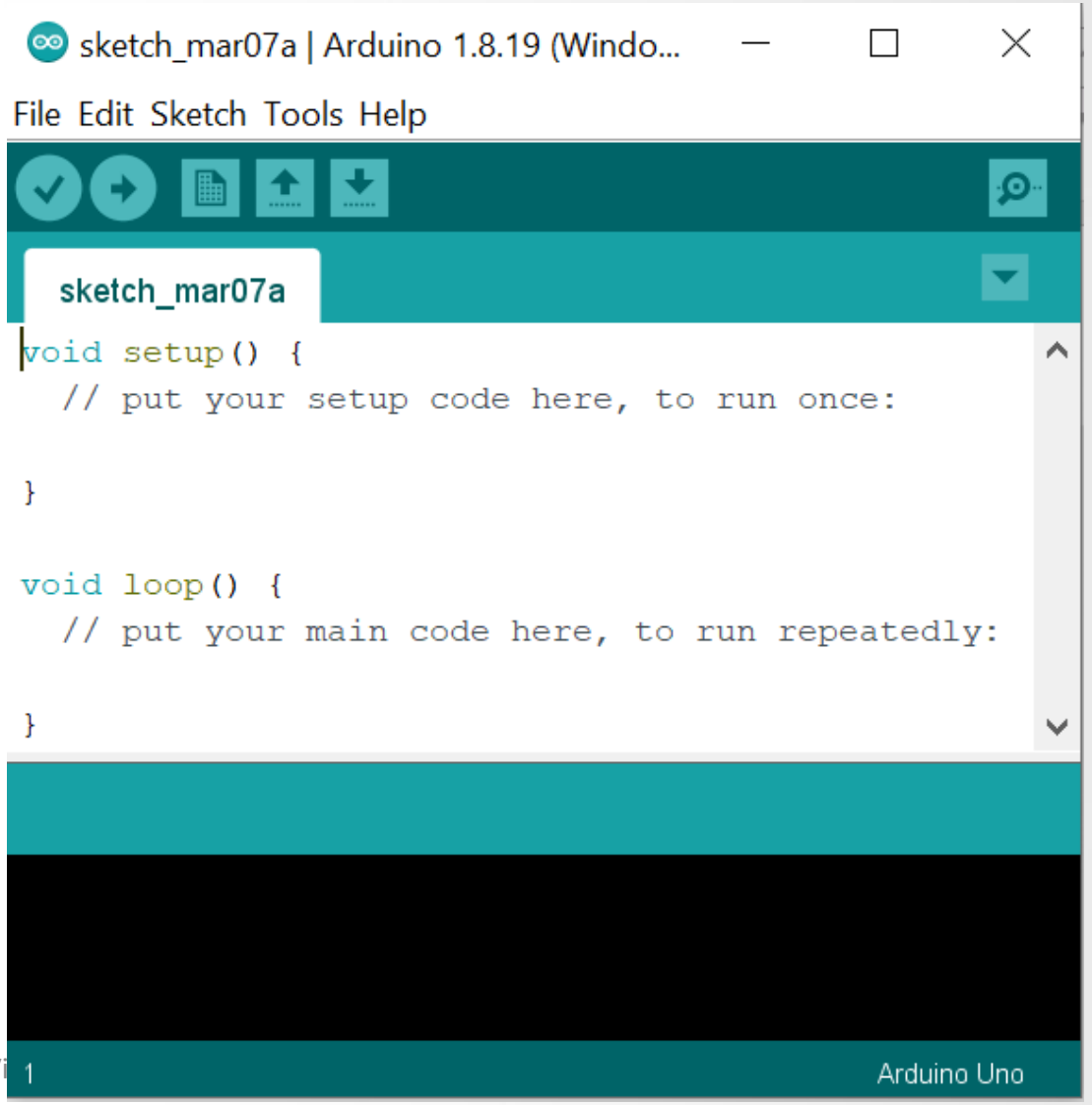
```
void setup() {

      //setup motors,
      sensors etc

}
void loop() {

      // get information
      from sensors
      // send commands to
      motors

}
```



Comments

Comments

```
// this is for single line comments

// it's good to put at the top and before anything 'tricky'
```

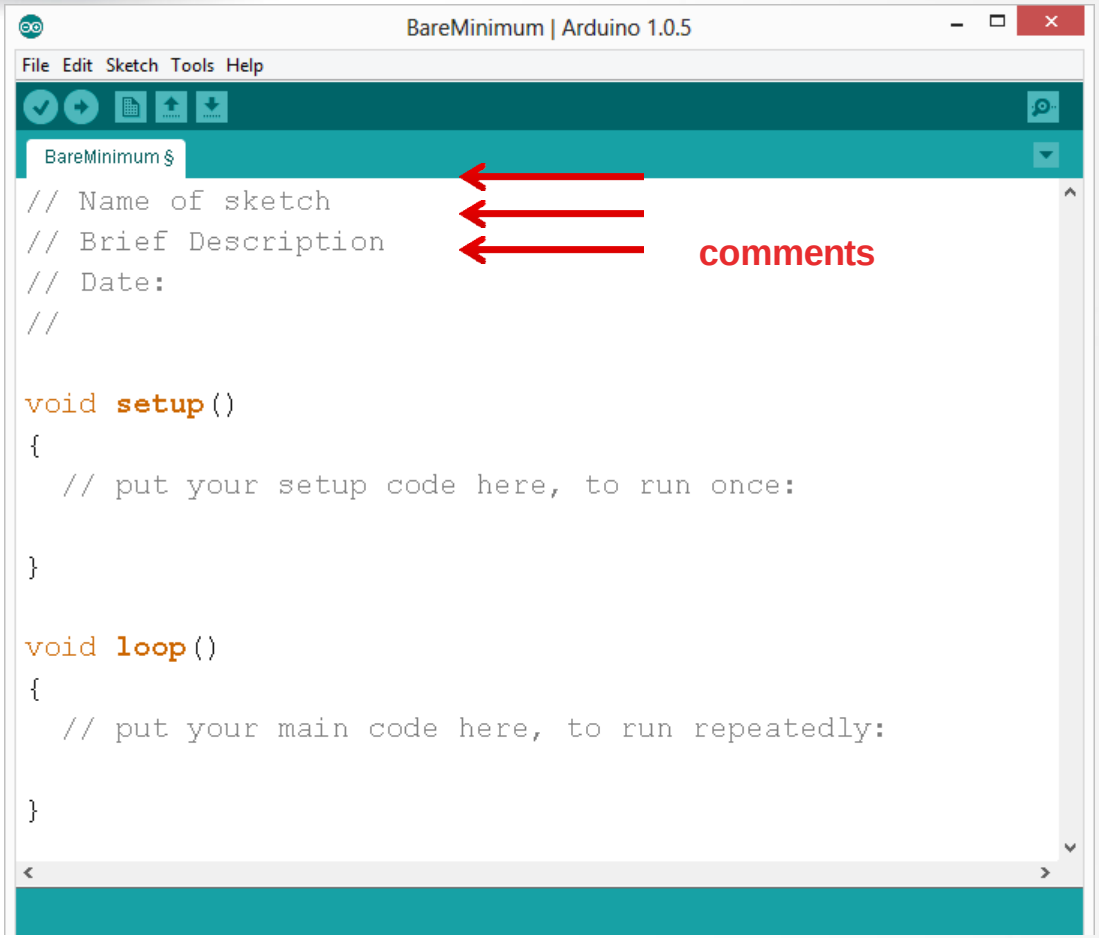
```
/* this is for multi-line comments

   Like this…

   And this….

*/
```

Comments


BareMinimum | Arduino 1.0.5
File Edit Sketch Tools Help
BareMinimum §
// Name of sketch
// Brief Description
// Date:
//
void setup()
{
// put your setup code here, to run once:
}
void loop()
{
// put your main code here, to run repeatedly:
}
comments

Few commands to know…

```
pinMode(pin, INPUT/OUTPUT);

      ex: pinMode(13, OUTPUT);

digitalWrite(pin, HIGH/LOW);

      ex: digitalWrite(13, HIGH);

digiatlRead(pin, HIGH/LOW);

      ex: digitalRead(14, INPUT);

delay(time_ms);

      ex: delay(2500); // delay of 2.5 sec.

         NOTE: -> commands are CASE-sensitive
                  Camel casing
```

Few commands to know…

- `Serial.println(value);`
- `Serial.println(value,format);`
  - Prints the value to the Serial Monitor on your computer

```
Ex:Serial.println(analogValue);
   Serial.println(analogValue, DEC);
```

- `analogRead(pin);`
  - Reads a digital value (HIGH or LOW) on a pin set for input

```
Ex: val = analogRead(analogPin);
```

- `analogWrite(pin, value);`
  - Writes the digital value (HIGH or LOW) to a pin set for output

```
Ex: analogWrite(ledPin, val / 4);
```

SETUP()

- The setup section is used for assigning input and outputs (Examples: motors, LED's, sensors etc) to ports on the Arduino
- It also specifies whether the device is OUTPUT or INPUT
- To do this we use the command "pinMode"

```
void setup() {
    pinMode(9, OUTPUT);
}
```

port #

Input or Output

16

LOOP()

```
void loop() {

    digitalWrite(9, HIGH);
    delay(1000);
    digitalWrite(9, LOW);
    delay(1000);

}
```

Port # from setup

Turn the LED on or off

Wait for 1 second or 1000 milliseconds

DECLARING A VARIABLE

```
int val = 5;
```

Type

variable name

assignment
"becomes"

value

USING VARIABLES

```
int delayTime = 2000;
int greenLED = 9;
void setup() {

      pinMode(greenLED, OUTPUT);

}

void loop() {

      digitalWrite(greenLED, HIGH);
      delay(delayTime);
      digitalWrite(greenLED, LOW);
      delay(delayTime);

}
```

Declare delayTime Variable

Use delayTime

USING VARIABLES

```
int delayTime = 2000;
int greenLED = 9;

void setup() {
      pinMode(greenLED, OUTPUT);
}
void loop() {
      digitalWrite(greenLED, HIGH);
      delay(delayTime);
      digitalWrite(greenLED, LOW);
 delayTime = delayTime - 100;
      delay(delayTime);
}
```

subtract 100 from delayTime to gradually increase LED's blinking speed

CONDITIONS

- To make decisions in Arduino code we use an ‘if’ statement

- ‘If’ statements are based on a TRUE or FALSE question

```
if(true)
{
    “perform some action”
}
```

IF EXAMPLE

```
int counter = 0;

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

    if(counter < 10)
    {
        Serial.println(counter);
    }
    counter = counter + 1;

}
```

VALUE COMPARISONS

GREATER THAN	GREATER THAN OR EQUAL
a > b	a >= b
LESSER	**LESS THAN OR EQUAL**
a < b	a <= b
EQUAL	**NOT EQUAL**
a == b	a != b

SERIAL INPUT & OUTPUT

Transferring data from the computer to an Arduino is done using Serial Transmission

To setup Serial communication we use the following

```
void setup() {

    Serial.begin(9600);

}
```

24

26

WRITING TO THE CONSOLE

```
void setup() {

     Serial.begin(9600);
 Serial.println("Hello World!");

}

void loop() {}
```

25

IF - ELSE CONDITION

```
if( "answer is true")
{
    "perform some action"
}
else
{
    "perform some other action"
}
```

IF - ELSE EXAMPLE

```
int counter = 0;
void setup() {
    Serial.begin(9600);
}
void loop() {
    if(counter < 10)
    {
        Serial.println("less than 10");
    }
    else
    {
        Serial.println("greater than or equal to 10");
        Serial.end();
    }
    counter = counter + 1;}
```

IF - ELSE IF Condition

```
if( "answer is true")
{
    "perform some action"
}
else if( "answer is true")
{
    "perform some other action"
}
```

IF - ELSE IF Example

```
int counter = 0;  void
setup() {
     Serial.begin(9600);
}
void loop() {

     if(counter < 10)
     {
          Serial.println("less than 10");
     }
     else if (counter == 10)
     {
          Serial.println("equal to 10");
     }
     else
     {
          Serial.println("greater than 10");
          Serial.end();
     }
     counter = counter + 1;
}
```

BOOLEAN OPERATORS - AND

- **If we want all of the conditions to be true we need to use ‘AND’ logic (AND gate)**

- **We use the symbols &&**
  - **Example**

```
if ( val > 10 && val < 20)
```

30

32

BOOLEAN OPERATORS - OR

- **If we want either of the conditions to be true we need to use ‘OR’ logic (OR gate)**
- **We use the symbols ||**
- **Example**

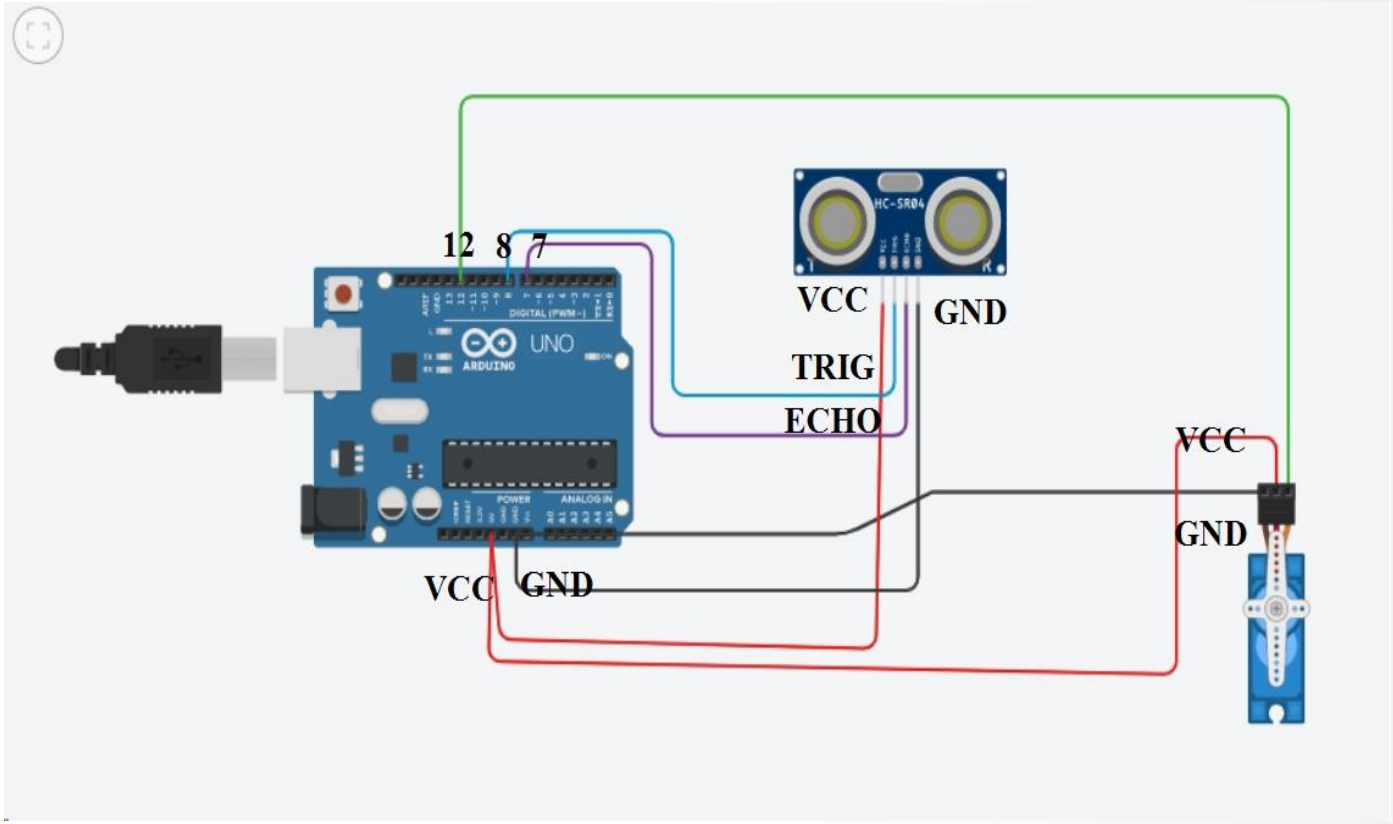
```
if ( val < 10 || val > 20)
```

BOOLEAN VARIABLES

```
boolean done = true;
   boolean done = false;
   void setup() {
   Serial.begin(9600);}
   void loop() {
     if(!done)  {
   Serial.println("HELLOWORLD");
        done = true;  }
}
```

Project-1 Ultra Sonic sensor and Servo motor

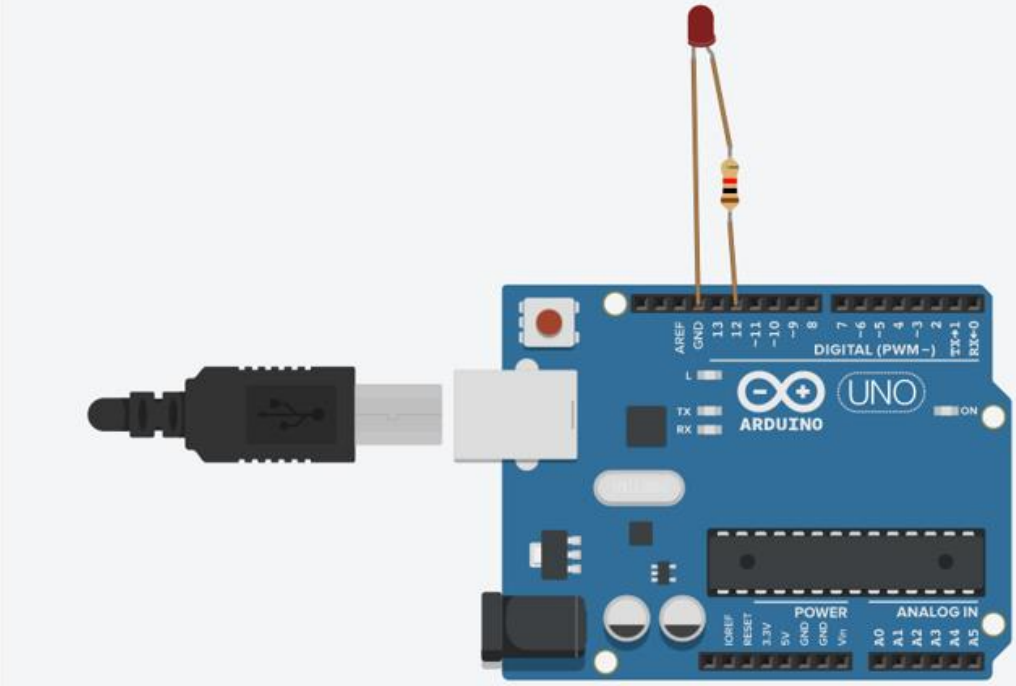
Ultra Sonic Sensor---live demo



Open Arduino IDE

Project 2- LED Blink

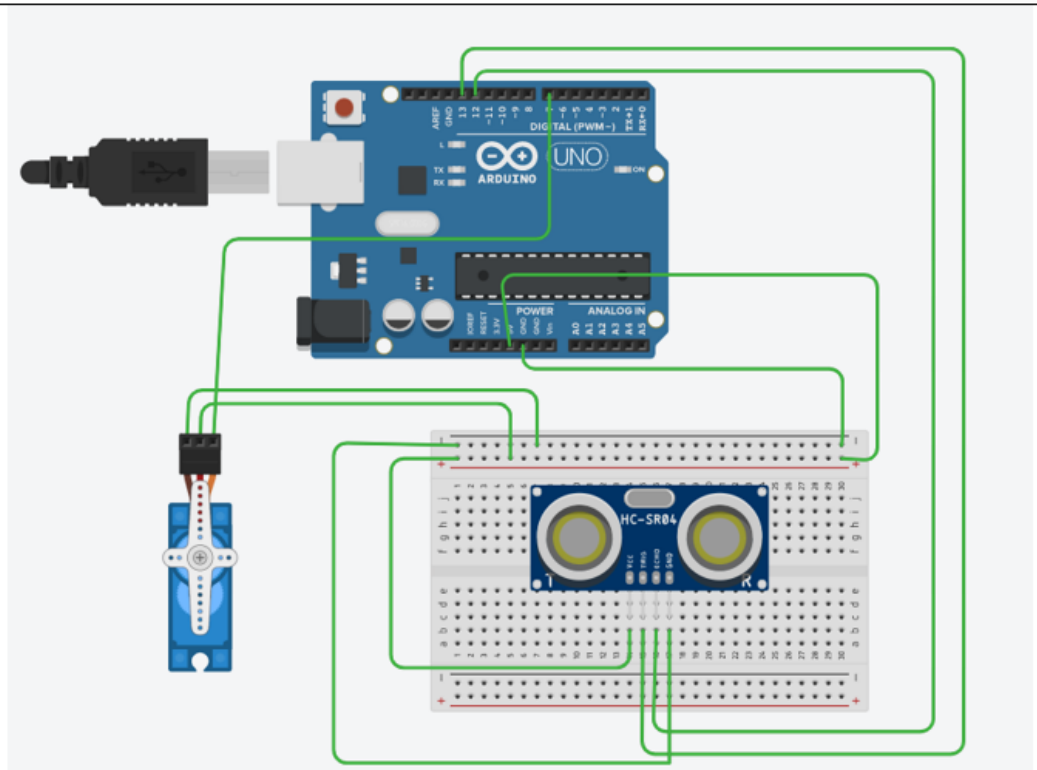
- **Consider the following circuit and following code: Create a new Tinkercad project with the circuit in the figure and type the provided code. Make sure circuit works by verifying that the LED is blinking when you run the simulation.**
- **Study the code and make sure you understand its functionality.**
- **The resistor used is 220 Ohms.**



LED	Arduino
+ve (anode)	GND
-ve (cathode)	Register front
Register	**Arduino**
End of register	Pin 12

Project 3 – Ultra Sonic Sensor and Servo Motor

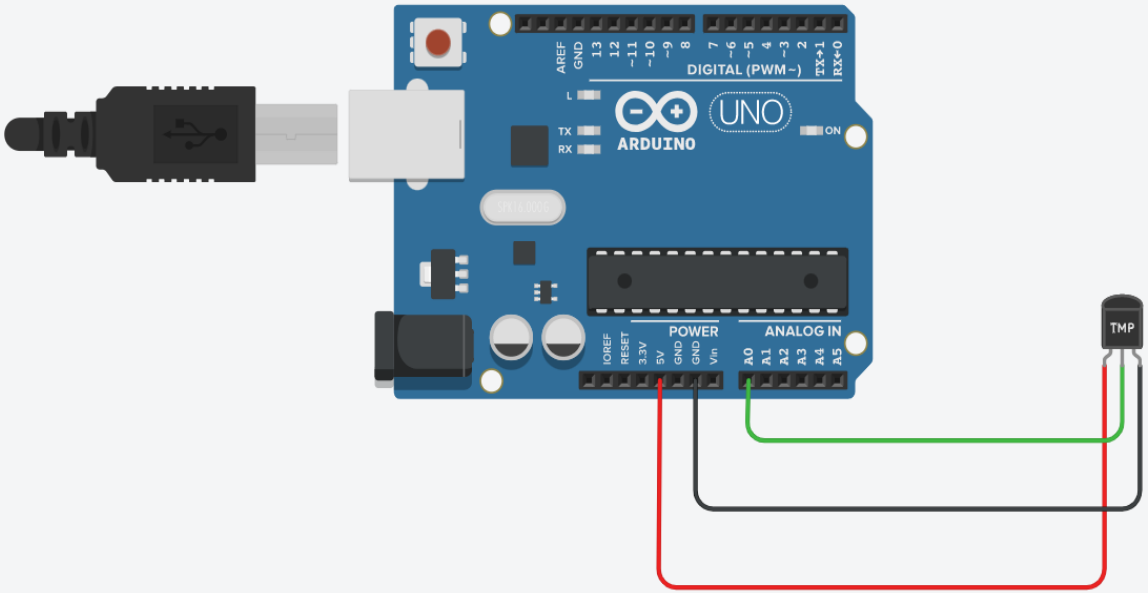
- Consider the following circuit and the following code.
- Create a Tinkercad project with the circuit in the figure and type the provided code. Make sure the servo more rotates when the distance sensor measures an object close by.
- Study the code and make sure you understand its functionality.



Ultra Sonic	Arduino
VCC	VCC
GND	GND
Trig	13
Echo	12
Servo	**Arduino**
VCC	VCC
GND	GND
Signal	7

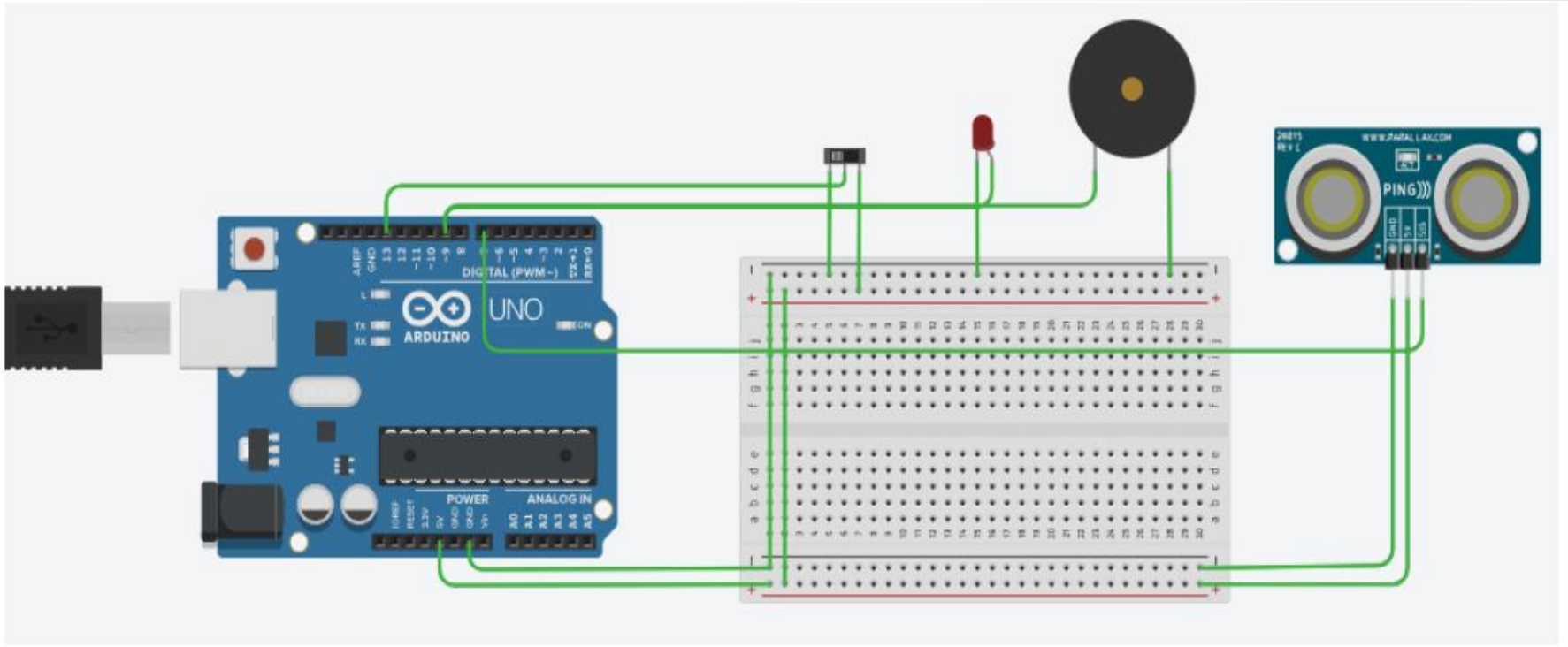
Project 4 – Temperature sensor

- Consider the following circuit and the following code. Create a Tinkercad project with the circuit in the figure and type the provided code.
- Make sure the temperature data is printed on the serial monitor for every 3 seconds delay.
- The signal pin of the temperature sensor is connected to A0 of Arduino.
- Study the code and make sure you understand its functionality.



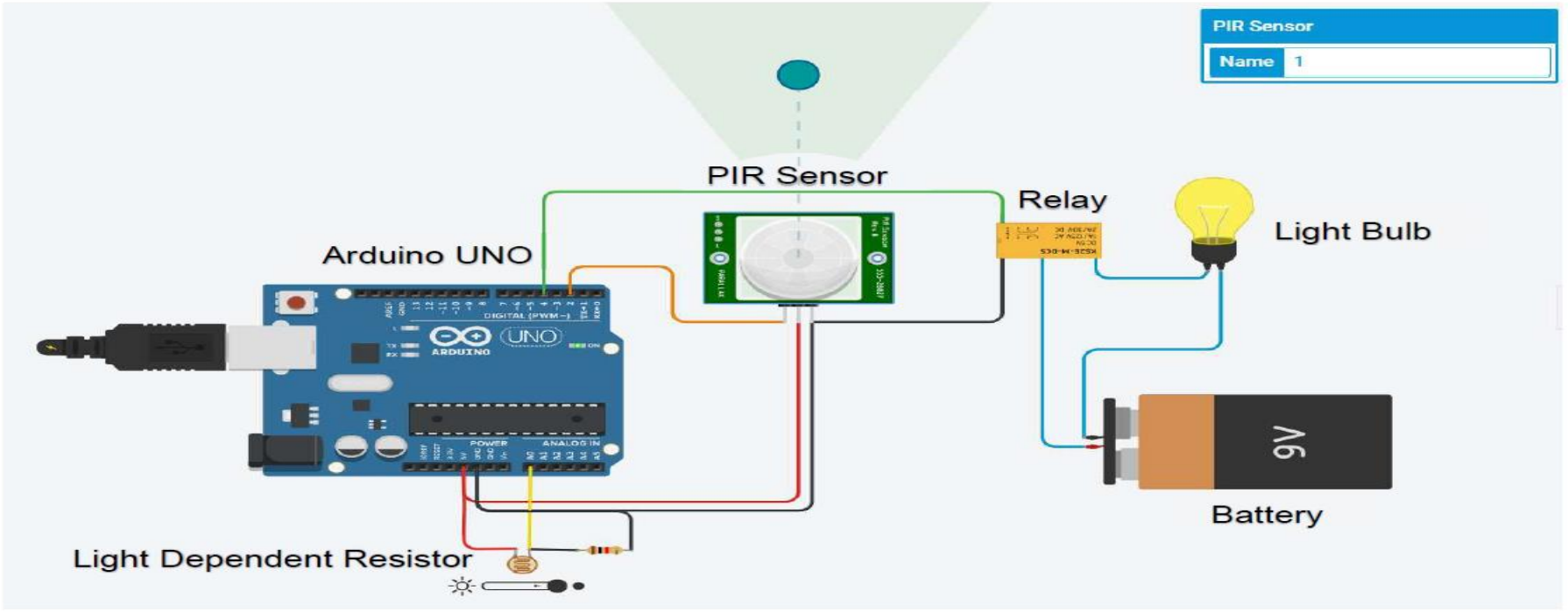
Question 1— Ultra Sonic Sensor with Speaker and LED

Develop an application with the following circuit of Ultra Sonic Sensor, Speaker, LED and Transistor. Make sure the speaker is beeping when the distance sensor measures an object close by. Make documentation of each hardware component you are using.



Question -2: Smart lighting

Develop a circuit shown below representing a smart lighting system using TinkerCad. Action: Slide the point on bar towards Day or Night on the LDR, it detects the brightness intensity. The Raw values can be seen on Serial Monitor. Any value below 500 means it is dark outside and any value above 500 means it is bright. Based on this value, the bulb should switch ON ay night and switch OFF during day. Also, if the blue dot of PIR sensor is clicked and dragged to create a physical movement in front of the sensor, then the PIR sensor should turn red and blub should ON.



IoT and M2M: SDN and NFV for IoT