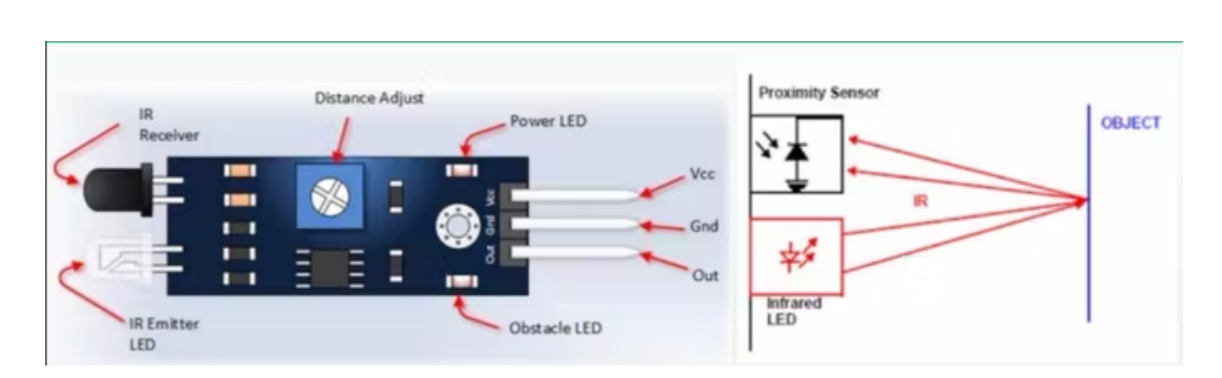
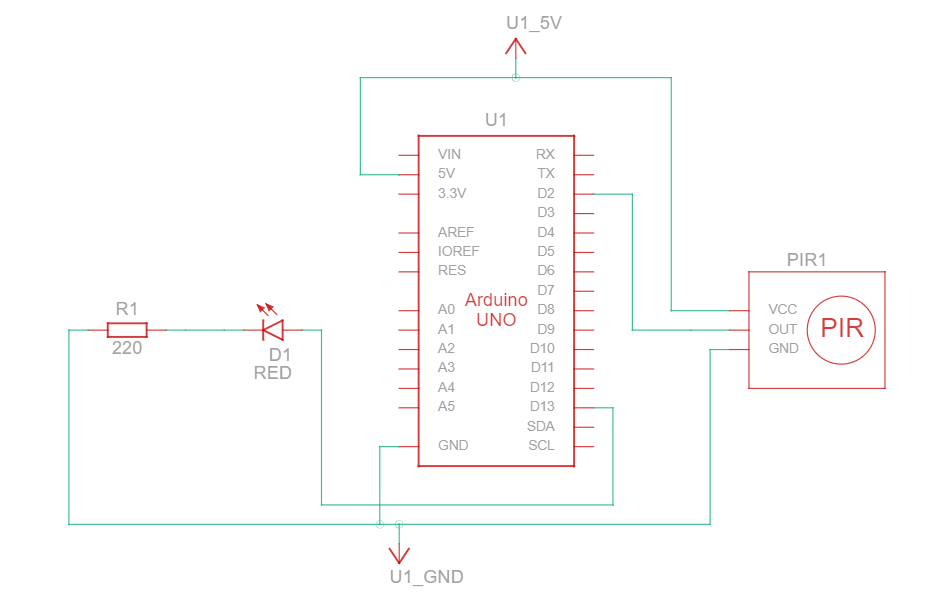
1. **INFRARED SENSOR:**

**THEORY:**

A passive infrared sensor is an electronic sensor that measures infrared light radiating from objects in its field of view. They are most often used in PIR-based motion detectors. IR sensors are commonly used in security alarms and automatic lighting applications.



**CIRCUIT DIAGRAM:**



**SOURCE CODE:**

int buttonState = 0;

void setup()

{

pinMode(2, INPUT);

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop()

{

// read the state of the pushbutton

buttonState = digitalRead(2);

// check if pushbutton is pressed. if it is, the

// button state is HIGH

if (buttonState == HIGH) {

digitalWrite(LED\_BUILTIN, HIGH);

} else {

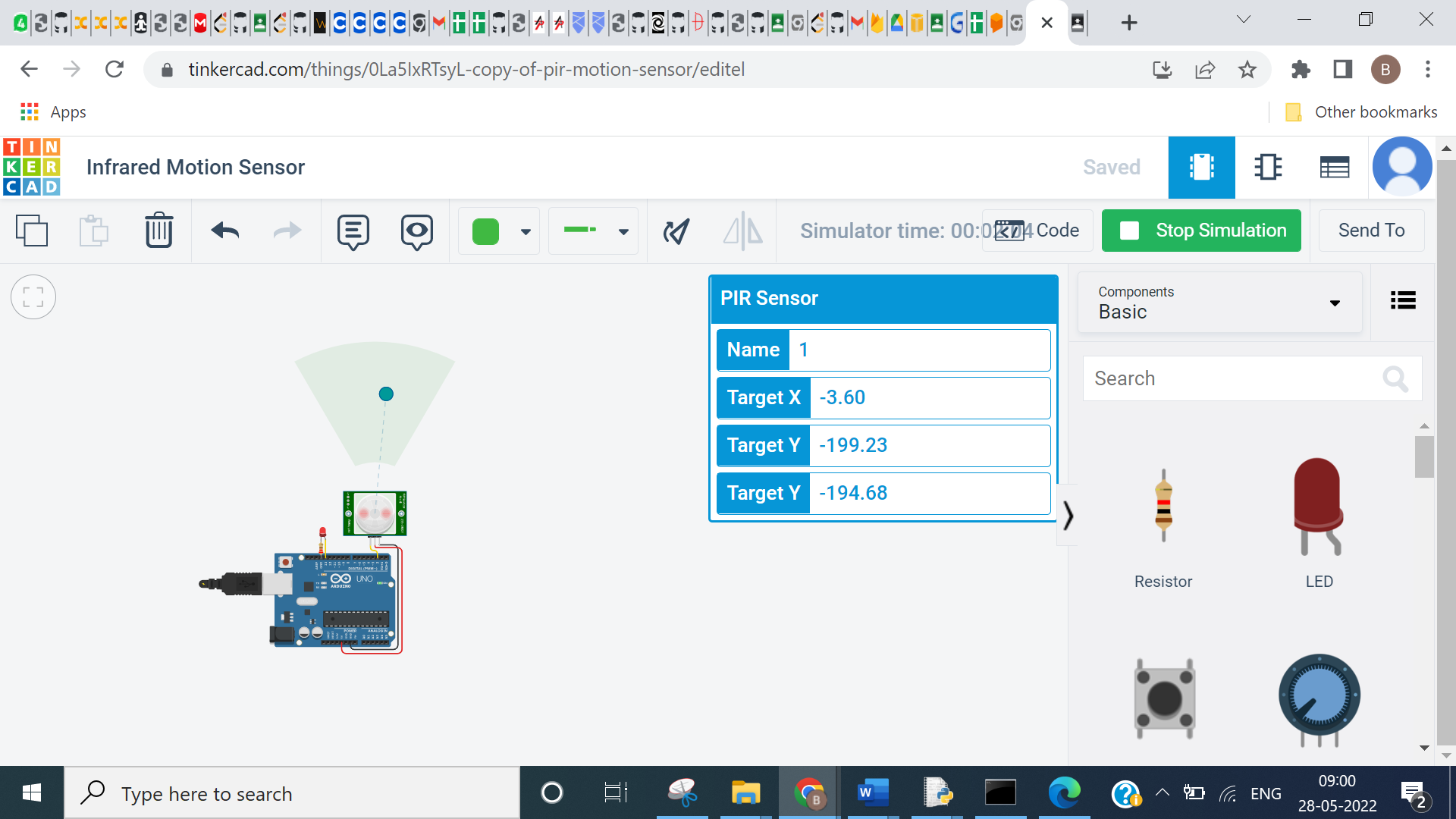
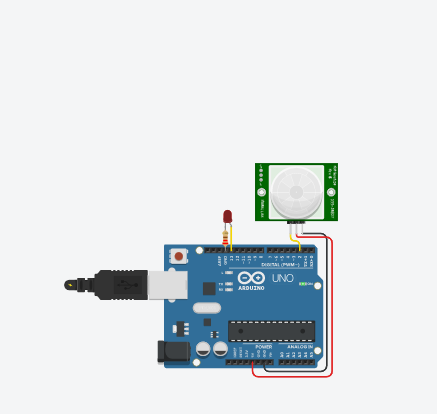
digitalWrite(LED\_BUILTIN, LOW);

}

delay(10); // Delay a little bit to improve simulation performance

}

**OUTPUT:**



**CONCLUSION:**

From this practical, I have learned and implemented the infrared sensor with Arduino in tinkercad.

1. **ULTRASONIC SENSOR:**

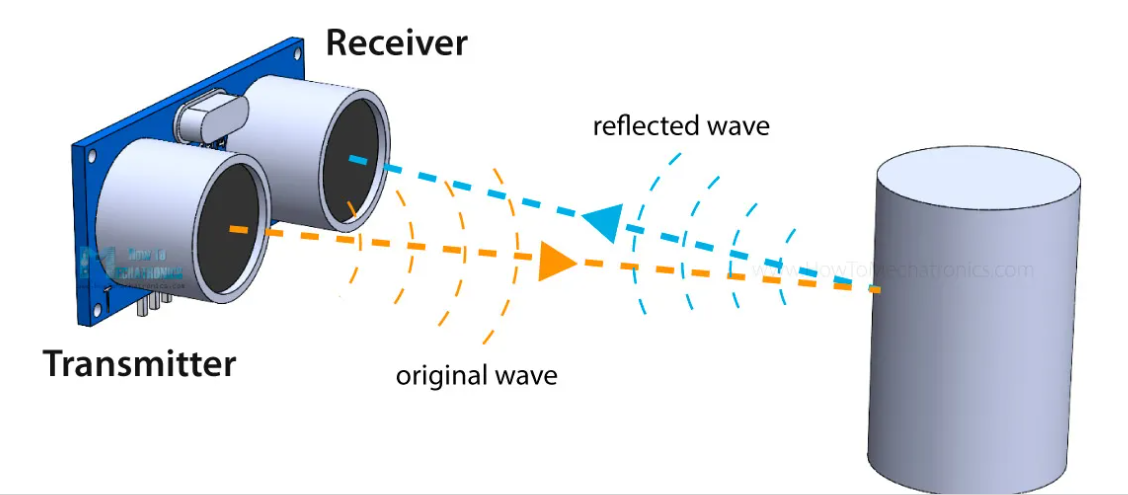
**THEORY:**

**Ultrasonic Distance Sensor:**

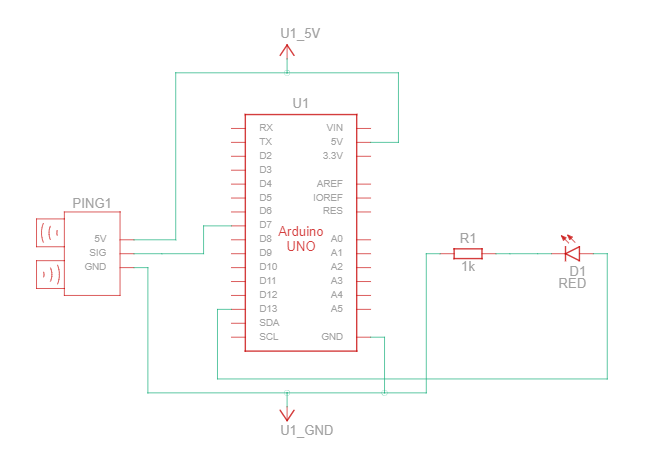
An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal.

An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object’s proximity.

High-frequency sound waves reflect from boundaries to produce distinct echo patterns.



**CIRCUIT DIAGRAM:**



**SOURCE CODE:**

const int pingPin = 7;

const int ledPin = 13;

void setup() {

// initialize serial communication:

Serial.begin(9600);

pinMode(ledPin, OUTPUT);

}

void loop() {

// establish variables for duration of the ping,

// and the distance result in inches and centimeters:

long duration, cm;

// The PING))) is triggered by a HIGH pulse of 2 or more microseconds.

// Give a short LOW pulse beforehand to ensure a clean HIGH pulse:

pinMode(pingPin, OUTPUT);

digitalWrite(pingPin, LOW);

delayMicroseconds(2);

digitalWrite(pingPin, HIGH);

delayMicroseconds(5);

digitalWrite(pingPin, LOW);

// The same pin is used to read the signal from the PING))): a HIGH

// pulse whose duration is the time (in microseconds) from the sending

// of the ping to the reception of its echo off of an object.

pinMode(pingPin, INPUT);

duration = pulseIn(pingPin, HIGH);

// convert the time into a distance

cm = microsecondsToCentimeters(duration);

// Print the distance

Serial.print("Distance: ");

Serial.print(cm);

Serial.print("cm");

Serial.println();

// Turn on the LED if the object is too close:

if(cm < 100) {

digitalWrite(ledPin, HIGH);

}

else {

digitalWrite(ledPin, LOW);

}

delay(100);

}

long microsecondsToCentimeters(long microseconds) {

// The speed of sound is 340 m/s or 29 microseconds per centimeter.

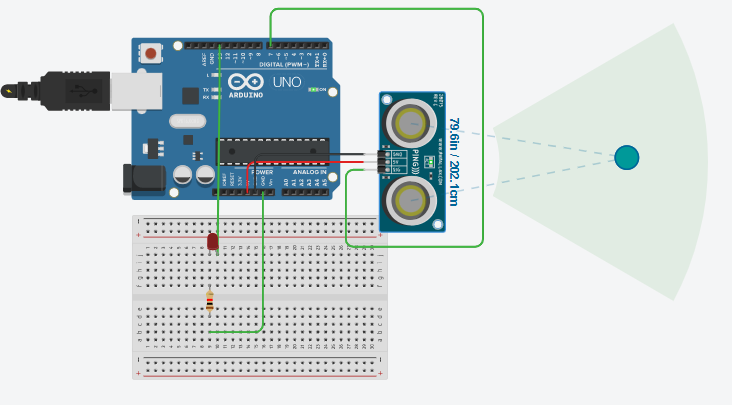
// The ping travels out and back, so to find the distance of the

// object we take half of the distance travelled.

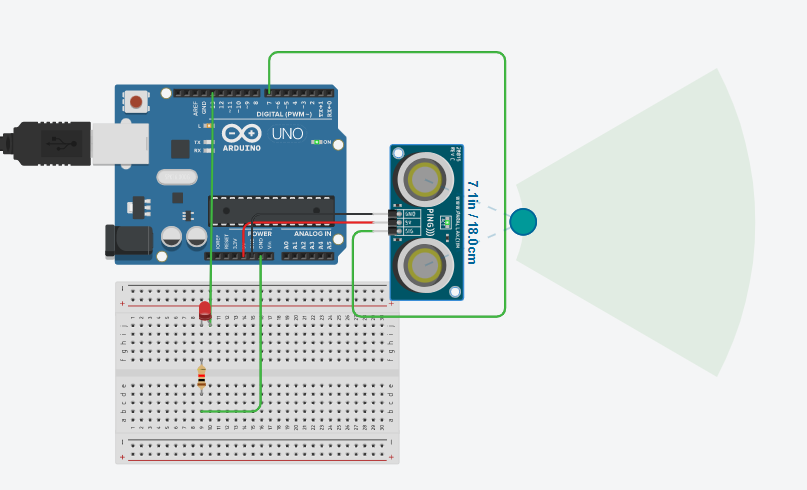
return microseconds / 29 / 2;

}

**OUTPUT:**



LED will glow if distance is less than 100 cm:



**CONCLUSION:**

From this practical, I have learned and implemented the ultrasonic sensor with Arduino in tinkercad.