**ASSIGNMENT 3**

|  |  |
| --- | --- |
| **SUBJECT** | IBM |
| **Team ID** | NM2023TMID14285 |
| **Team Leader** | LOKESH S |
| **Team member** | DEEPAKRAJ S |

**CIRCUIT DIGRAM:**



**CODING:**

#define echoPin \

* // attach pin D2 Arduino to pin Echo of HC-SR04 #define trigPin \
* // attach pin D3 Arduino to pin Trig of HC-SR04

\

long duration; // Variable to store time taken to the pulse

// to reach receiver

int distance; // Variable to store distance calculated using

// formula

void setup()

{

pinMode(trigPin,

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

// Serial Communication is starting with 9600 of

// baudrate speed

**Serial**.begin(9600);

// The text to be printed in serial monitor

**Serial**.println(

"Distance measurement using Arduino Uno."); delay(500);

}

void loop()

{

digitalWrite(trigPin, LOW);

delayMicroseconds(2); // wait for 2 ms to avoid

// collision in serial monitor

digitalWrite(

trigPin,

HIGH); // turn on the Trigger to generate pulse delayMicroseconds(

10); // keep the trigger "ON" for 10 ms to generate

// pulse for 10 ms.

digitalWrite(trigPin,

LOW); // Turn off the pulse trigger to stop

// pulse generation

// If pulse reached the receiver echoPin

// become high Then pulseIn() returns the

// time taken by the pulse to reach the

// receiver

duration = pulseIn(echoPin, HIGH); distance

= duration \* 0.0344 / 2; // Expression to calculate

// distance using time

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

**Serial**.print(

distance); // Print the output in serial monitor

**Serial**.println(" cms"); delay(100);

}

**WOKWI LINK:**

**IBM CLOUD IMAGE:**

Your wokwi link copy

Your IBM link copy