| **Course Name:** | **Sensors in Augmented and Virtual Reality** | **Semester:** | **IV** |
| --- | --- | --- | --- |
| **Date of Performance:** |  | **Batch No:** |  |
| **Faculty Name:** | **Megha Sharma** | **Roll No:** | **16010121110** |
| **Faculty Sign & Date:** |  | **Grade/Marks:** |  |

**Experiment No:**

**Title: Interfacing of distance sensor with Arduino**

| **Aim and Objective of the Experiment:** |
| --- |
| **To learn how to interface HC-SR04 sensor with Arduino** |

| **COs to be achieved:** |
| --- |
| **CO1: Study basic sensors used in Augmented reality systems**  **CO2: Gain basic knowledge sensors in Virtual reality headsets**  **CO5: Interface sensors and actuators to AR and VR systems** |

| **Theory:** |
| --- |
| What is Ultrasound?  Ultrasound is high-pitched sound waves with frequencies higher than the audible limit of human hearing.  Ultrasonic Frequency Range Spectrum  Human ears can hear sound waves that vibrate in the range from about 20 times a second (a deep rumbling noise) to about 20,000 times a second (a high-pitched whistling). However, ultrasound has a frequency of over 20,000 Hz and is therefore inaudible to humans.  HC-SR04 Hardware Overview  At its core, the HC-SR04 Ultrasonic distance sensor consists of two [ultrasonic transducers](https://en.wikipedia.org/wiki/Ultrasonic_transducer). The one acts as a transmitter which converts electrical signal into 40 KHz ultrasonic sound pulses. The receiver listens for the transmitted pulses. If it receives them it produces an output pulse whose width can be used to determine the distance the pulse travelled. As simple as pie!  The sensor is small, easy to use in any robotics project and offers excellent non-contact range detection between 2 cm to 400 cm (that’s about an inch to 13 feet) with an accuracy of 3mm. Since it operates on 5 volts, it can be hooked directly to an Arduino or any other 5V logic microcontrollers.  Here are complete specifications:   | Operating Voltage | DC 5V | | --- | --- | | Operating Current | 15mA | | Operating Frequency | 40KHz | | Max Range | 4m | | Min Range | 2cm | | Ranging Accuracy | 3mm | | Measuring Angle | 15 degree | | Trigger Input Signal | 10µS TTL pulse | | Dimension | 45 x 20 x 15mm |   HC-SR04 Ultrasonic Sensor Pinout  Let’s take a look at its Pinout.  HC-SR04 Ultrasonic Distance Sensor Pinout  VCC is the power supply for HC-SR04 Ultrasonic distance sensor which we connect the 5V pin on the Arduino.  Trig (Trigger) pin is used to trigger the ultrasonic sound pulses.  Echo pin produces a pulse when the reflected signal is received. The length of the pulse is proportional to the time it took for the transmitted signal to be detected.  GND should be connected to the ground of Arduino.  How Does HC-SR04 Ultrasonic Distance Sensor Work?  It all starts, when a pulse of at least 10 µS (10 microseconds) in duration is applied to the Trigger pin. In response to that the sensor transmits a sonic burst of eight pulses at 40 KHz. This 8-pulse pattern makes the “ultrasonic signature” from the device unique, allowing the receiver to differentiate the transmitted pattern from the ambient ultrasonic noise.  The eight ultrasonic pulses travel through the air away from the transmitter. Meanwhile the Echo pin goes HIGH to start forming the beginning of the echo-back signal.  In case, If those pulses are not reflected back then the Echo signal will timeout after 38 mS (38 milliseconds) and return low. Thus a 38 mS pulse indicates no obstruction within the range of the sensor.  HC-SR04 Ultrasonic Sensor Working - Echo when no Obstacle  If those pulses are reflected back the Echo pin goes low as soon as the signal is received. This produces a pulse whose width varies between 150 µS to 25 mS, depending upon the time it took for the signal to be received.  HC-SR04 Ultrasonic Sensor Working - Echo reflected from Obstacle  The width of the received pulse is then used to calculate the distance to the reflected object. This can be worked out using simple distance-speed-time equation, we learned in High school. In case you forgot, an easy way to remember the distance, speed and time equations is to put the letters into a triangle.  Distance Speed Time Formula Triangle  Let’s take an example to make it more clear. Suppose we have an object in front of the sensor at an unknown distance and we received a pulse of width 500 µS on the Echo pin. Now let’s calculate how far the object from the sensor is. We will use the below equation.  Distance = Speed x Time  Here, we have the value of Time i.e. 500 µs and we know the speed. What speed do we have? The speed of sound, of course! Its 340 m/s. We have to convert the speed of sound into cm/µs in order to calculate the distance. A quick Google search for “speed of sound in centimeters per microsecond” will say that it is 0.034 cm/µs. You could do the math, but searching it is easier. Anyway, with that information, we can calculate the distance!  Distance = 0.034 cm/µs x 500 µs  But this is not done! Remember that the pulse indicates the time it took for the signal to be sent out and reflected back so to get the distance so, you’ll need to divide your result in half.  Distance = (0.034 cm/µs x 500 µs) / 2  Distance = 8.5 cm  So, now we know that the object is 8.5 centimeters away from the sensor.  Wiring – Connecting HC-SR04 to Arduino Uno  Now that we have a complete understanding of how HC-SR04 ultrasonic distance sensor works, we can begin hooking it up to our Arduino!  Connecting the HC-SR04 to the Arduino is pretty easy. Start by placing the sensor on to your breadboard. Connect VCC pin to the 5V pin on the Arduino and connect GND pin to the Ground pin on the Arduino.  When you’re done you should have something that looks similar to the illustration shown below.  Arduino Wiring Fritzing Normal Mode Connections with HC-SR04 Ultrasonic Sensor  Wiring HC-SR04 Ultrasonic Sensor to Arduino UNO – Normal Mode |

| **Stepwise-Procedure:** |
| --- |
| Login on TinkerCAD  Connect Circuit diagram  Type code in Arduino and run  #define echoPin 2 // attach pin D2 Arduino to pin Echo of HC-SR04  #define trigPin 3 //attach pin D3 Arduino to pin Trig of HC-SR04  // defines variables  long duration; // variable for the duration of sound wave travel  int distance; // variable for the distance measurement  void setup() {  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT  pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT  Serial.begin(9600); // // Serial Communication is starting with 9600 of baudrate speed  Serial.println("Ultrasonic Sensor HC-SR04 Test"); // print some text in Serial Monitor  Serial.println("with Arduino UNO R3");  }  void loop() {  // Clears the trigPin condition  digitalWrite(trigPin, LOW);  delayMicroseconds(2);  // Sets the trigPin HIGH (ACTIVE) for 10 microseconds  digitalWrite(trigPin, HIGH);  delayMicroseconds(10);  digitalWrite(trigPin, LOW);  // Reads the echoPin, returns the sound wave travel time in microseconds  duration = pulseIn(echoPin, HIGH);  // Calculating the distance  distance = duration \* 0.034 / 2; // Speed of sound wave divided by 2 (go and back)  // Displays the distance on the Serial Monitor  Serial.print("Distance: ");  Serial.print(distance);  Serial.println(" cm");  } |

| **Output Screen shots:** |
| --- |
|  |

| **Results:** |
| --- |
|  |

| **Post Lab Subjective/Objective type Questions:** |
| --- |
| 1. Explain different distance sensors used with Arduino   Ultrasonic Sensor:  Ultrasonic Sensor HC-SR04 is a sensor that can measure distance. It emits an ultrasound at 40 000 Hz (40kHz) which travels through the air and if there is an object or obstacle on its path It will bounce back to the module. Considering the travel time and the speed of the sound you can calculate the distance.  The configuration pin of HC-SR04 is VCC (1), TRIG (2), ECHO (3), and GND (4). The supply voltage of VCC is +5V and you can attach TRIG and ECHO pin to any Digital I/O in your Arduino Board.  IR Based Sensor:  The SHARP 2Y0A21 proximity sensor measures distance by shining a beam of infrared light and uses a phototransitor to measure the intensity of the light that bounces back. The effective distance measuring range for this proximity sensor is 10-80cm. If an object is closer than the shortest distance, it reports a significantly higher analog output and is inconsistent with expected in-range results. |

| **Conclusion:** |
| --- |
| In this experiment we learnt how to interface HC-SR04 sensor with Arduino and noted the reading from it. We found out distance using sonar sensor |

| **Signature of faculty in-charge with Date:** |
| --- |