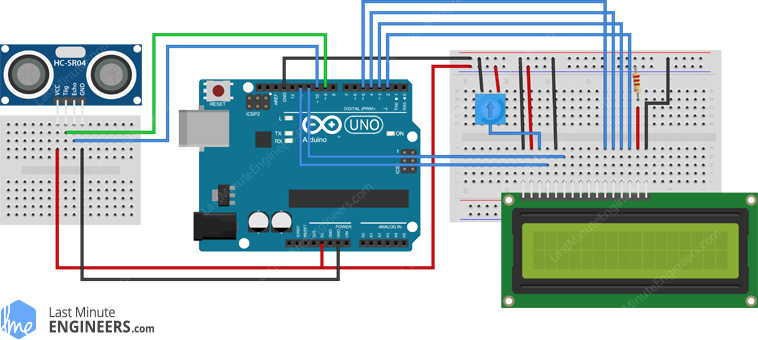
PROJECT

HC-SR04 Ultrasonic distance sensor Using ARDUINO UNO

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.

Circuit and Working



PIN CONNECTIONS

vss = ground

vdd = +5v

rw = ground

rs = 12pin ard

e/en = 11pin ard

d4 = 5 pin ard

d5 = 4 pin

d6 = 3 pin

d7 = 2 pin

k = ground

trig Pin = 9

echo Pin = 10

WORKING

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

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..

CODE

#include <LiquidCrystal.h>

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

const int trigPin = 9;

const int echoPin = 10;

long duration;

int distanceCm, distanceInch;

void setup() {

lcd.begin(16,2);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

}

void loop() {

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distanceCm= duration\*0.034/2;

distanceInch = duration\*0.0133/2;

lcd.setCursor(0,0); // Sets the location at which subsequent text written to the LCD will be displayed

lcd.print("Distance: "); // Prints string "Distance" on the LCD

lcd.print(distanceCm); // Prints the distance value from the sensor

lcd.print(" cm");

delay(10);

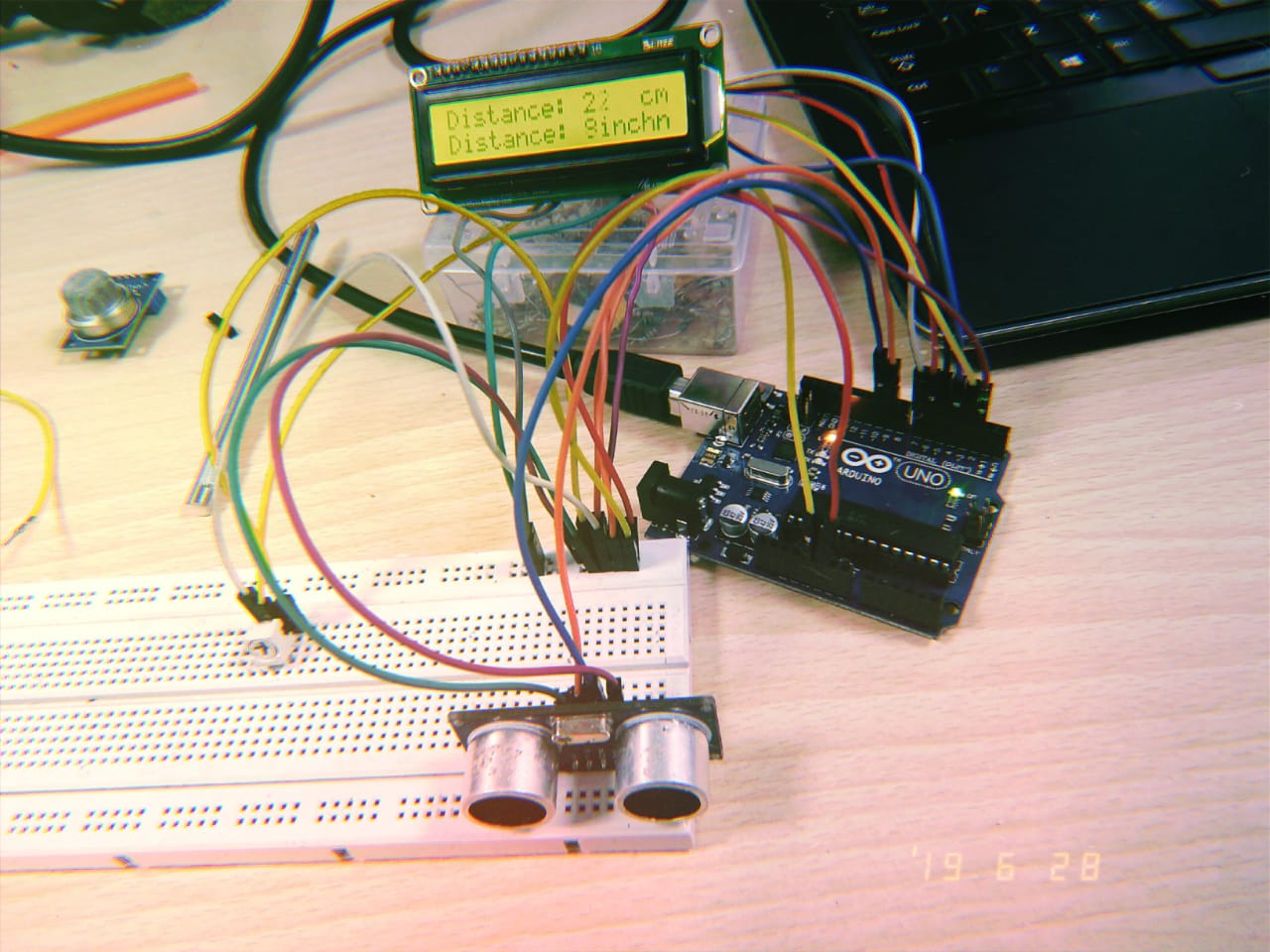
lcd.setCursor(0,1);

lcd.print("Distance: ");

lcd.print(distanceInch);

lcd.print("inch");

delay(10);

}