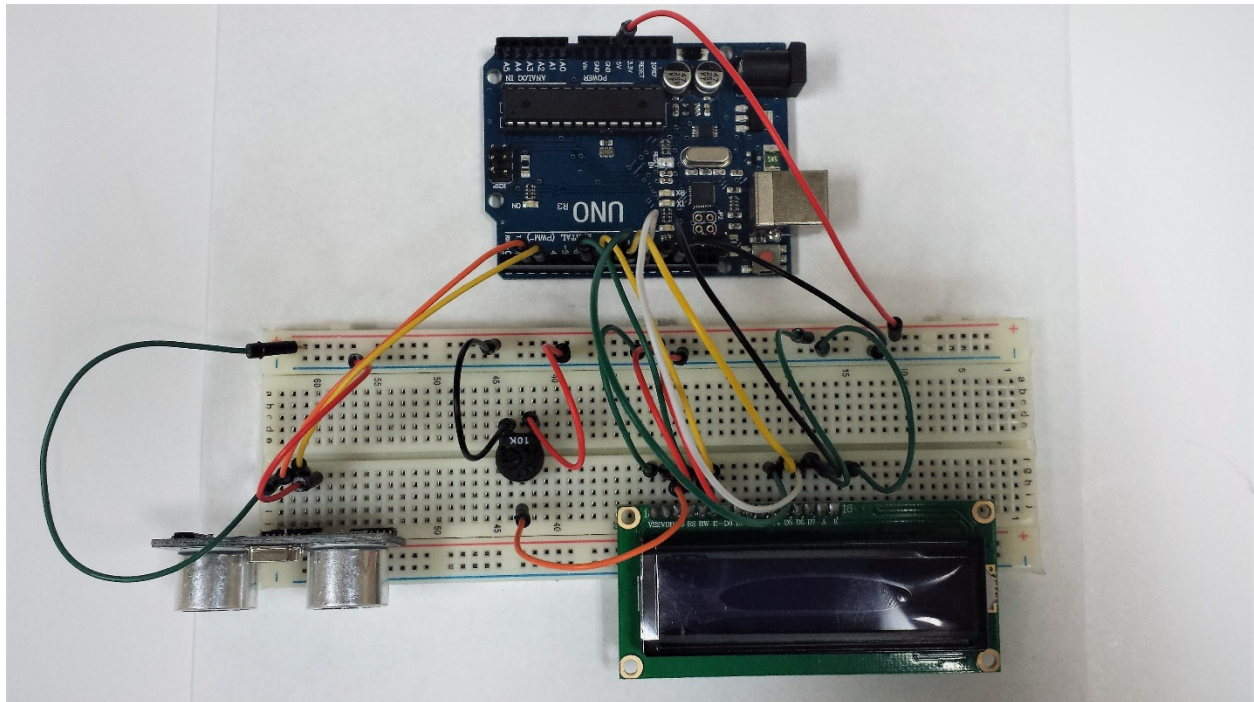


Measuring Distance with Ultrasonic Sensor, Arduino and LCD

Components:

1. Arduino Uno Board
2. Bread Board
3. 10k Potentiometer
4. Ultrasonic Sensor
5. LCD 1602 Display
6. Jumper Wires

SETTING UP:



STEPS:

1. From your Arduino, connect a jumper wire from the 5V Pin and Ground (GND) Pin to the Positive (+) and Negative (-) of the Breadboard.
2. **Ultrasonic Sensor:** Insert Your Ultrasonic Sensor onto the Bread Board. Insert a jumper wire from Vcc on the Ultrasonic Sensor to the Positive (+)

Rail of the Breadboard. Insert a jumper wire from Trig on the Ultrasonic Sensor to Pin 2 on the Arduino Board. Insert a jumper wire from Echo on the Ultrasonic Sensor to Pin 3 on the Arduino Board. Lastly, insert a jumper wire from Gnd on the Ultrasonic Sensor to the Negative (-) Rail of the Breadboard.

3. **LCD 1602 AND Potentiometer:** Insert LCD 1602 and the Potentiometer onto the Breadboard. Insert a jumper wire from VSS on the LCD to Negative (-) Rail of the Breadboard. Insert a jumper wire from Pin VDD on the LCD to the Positive (+) Rail of the Breadboard. Insert a jumper wire from Pin VO on the LCD to the middle Pin on the Potentiometer. For the remaining Pins on the Potentiometer: Insert two jumper wires from the outside Pins on the Potentiometer to the Positive (+) and Negative (-) Rails on the Breadboard. For the remaining Pins on the LCD: Insert a jumper wire from Pin RS on the LCD to Pin 7 on the Arduino. Insert a jumper wire from the Pin RW on the LCD to the Negative Rail on the Breadboard. Insert a jumper wire from Pin E on the LCD to Pin 8 on the Arduino Board. Insert a jumper wire from Pin D4 on the LCD to Pin 9 on the Arduino Board. Insert a jumper wire from Pin D5 on the LCD to Pin 10 on the Arduino. Insert a jumper wire from Pin D6 on the LCD to Pin 11 on the Arduino. Insert a jumper wire from Pin D7 on the LCD to Pin 12 on the Arduino Board. Insert a jumper wire from Pin A on the LCD to the Positive (+) Rail on the Breadboard. Insert a jumper wire from Pin K on the LCD to the Negative (-) Rail of the Breadboard.
4. **Program:** Upload the program below onto your Arduino Board.

If all wires are connected correctly, the LCD will turn ON and should show some numbers. By putting an obstruction in front of the Ultrasonic Sensor at different Distances, The Ultrasonic Sensor will accurately vary its reading.

Program:

```
#include <LiquidCrystal.h> //Load Liquid Crystal Library

LiquidCrystal LCD(7, 8, 9, 10, 11, 12); //Create Liquid Crystal Object called LCD

int trigPin=2; //Sensor Trip pin connected to Arduino pin 2

int echoPin=3; //Sensor Echo pin connected to Arduino pin 3

int myCounter=0; //declare your variable myCounter and set to 0
```

```

float pingTime; //time for ping to travel from sensor to target and return
float targetDistance; //Distance to Target in inches
float speedOfSound=776.5; //Speed of sound in miles per hour when temp is 77 degrees.

void setup() {

Serial.begin(9600);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);

LCD.begin(16,2); //Tell Arduino to start your 16 column 2 row LCD
LCD.setCursor(0,0); //Set LCD cursor to upper left corner, column 0, row 0
LCD.print("Target Distance:"); //Print Message on First Row
}

void loop() {

digitalWrite(trigPin, LOW); //Set trigger pin low
delayMicroseconds(2000); //Let signal settle
digitalWrite(trigPin, HIGH); //Set trigPin high
delayMicroseconds(15); //Delay in high state
digitalWrite(trigPin, LOW); //ping has now been sent
delayMicroseconds(10); //Delay in high state

pingTime = pulseIn(echoPin, HIGH); //pingTime is presented in microceonds
pingTime=pingTime/1000000; //convert pingTime to seconds by dividing by 1000000 (microseconds in a second)
pingTime=pingTime/3600; //convert pingtime to hourse by dividing by 3600 (seconds in an hour)
targetDistance= speedOfSound * pingTime; //This will be in miles, since speed of sound was miles per hour
targetDistance=targetDistance/2; //Remember ping travels to target and back from target, so you must divide by
2 for actual target distance.

targetDistance= targetDistance*63360; //Convert miles to inches by multipling by 63360 (inches per mile)

```

```
LCD.setCursor(0,1); //Set cursor to first column of second row
LCD.print("      "); //Print blanks to clear the row
LCD.setCursor(0,1); //Set Cursor again to first column of second row
LCD.print(targetDistance); //Print measured distance
LCD.print(" inches"); //Print your units.
delay(250); //pause to let things settle

}
```