



Distance Gun

Instructor Guide

Level 1

1. Assemble parts 1 - Unknown

Student Hint: Use the build legend on the back of the map

Instructor guide: Connect the pieces together using the build legend on the back of the map. Be sure to connect the power pins of the LCD and Ultrasonic sensor to power rails on the breadboard (+ and -). For further explanation please see the build video for this kit located in your instructor portal.

IMPORTANT: THE SDA PIN ON THE LCD SHOULD BE CONNECTED TO A4 AND THE SCL PIN ON THE LCD SHOULD BE CONNECTED TO PIN A5.

Notes:

2. Upload Test Code: Ultrasonic Sensor

Instructor guide: Once the test code is uploaded it will print whether the sensor wiring is connected properly or not in the serial monitor. If the sensor is not wired properly then have the student rewire the ultrasonic sensor.

How to Perform the Test: The instructor will upload to the code through the USB cable. Place an object at a respectable distance away from the sensor and open the serial monitor. To open the serial monitor click the magnifying glass in the upper right hand corner of the Arduino IDE.

IMPORTANT: THIS CODE SHOULD BE UPLOADED WITH A COMPUTER THAT ALREADY HAS THE `ULTRADISTSENSOR` LIBRARY INSTALLED ON IT>

CODE Copy and paste or type the following into the Arduino IDE and upload to the Arduino Board. You may also visit: www.BuildARobotWorkshop.com/-----

```
#include <UltraDistSensor.h> // Ultrasonic sensor library

UltraDistSensor scanner;
float reading;

void setup() {
    Serial.begin(9600);
    scanner.attach(9, 10);
    Serial.println("Test Code Uploaded");
    Serial.println("Place and object in front of the sensor!");
}
```

```
void loop() {  
  
    reading = scanner.distanceInInch();  
  
    if(reading == 0.0){  
        Serial.println("Sensor not connected properly!");  
    } else {  
        Serial.println("Sensor connected successfully!");  
    }  
  
    delay(2000);  
  
}
```

Notes:

3. Calibrate LCD:

Instructor guide: Even after supplying power to the LCD, its screen will most likely need adjusting to work properly. If the screen lights up and two rows of rectangles appear upon the screen then the contrast of the display has been calibrated properly.

How to Calibrate: On the I2C module rotate the blue box (potentiometer) on the back with a small screwdriver. Continue to rotate until the contrast of the LCD is calibrated properly. This calibration can be done at any time.

Notes:

4. Find LCD Address:

Instructor guide: Each LCD has its own unique address that is used to communicate with it. Messages cannot be printed onto the LCD without the proper address. To find the address of the LCD a special code file is needed. The code will print the address in the serial monitor. If everything is connected properly the address will be printed in the serial monitor. The instructor will upload to the code through the USB cable. Open the serial monitor in the upper right hand corner of the Arduino IDE.

IMPORTANT: HAVE THE STUDENT RECORD THE ADDRESS FOR FUTURE USE

CODE Download the code from www.BuildARobotWorkshop.com/----- and upload it to the Arduino IDE>

Notes:

5. Download the I2C Library:

Student Hint: Go to www.BuildARobotWorkshop.com/----- to download the zip of the library. To use this library go to :

Sketch → Include Library → Add .ZIP Library → (Click on the Downloaded Library)

Instructor guide: This library will be used to communicate with the LCD monitor. Without this library code cannot be written for the LCD. Builders will use the library in conjunction with the Wire.h library which is pre-installed on the Arduino IDE.

Notes:

6. Explanation External Code Libraries

Instructor guide: Point out the `#include <LiquidCrystal_I2C.h>` section on their coding sheet is an external library that is used to communicate with the LCD. Explain the following:

- External code libraries are commonly used in coding so that you can limit the amount of code you write.
- This allows us to use code that is already written somewhere else that is designed to do or help us do something that we are going to include in our code. For example using servos.
- In order to use an external code library we need to include that library in our code and that is why we have the `#include <LiquidCrystal_I2C.h>` section in our code.

Notes:

7. Fill out the Level 1 Worksheet for the Distance Gun:

Student Hint: Print two different messages on different rows of the LCD

Instructor guide: Have the builder fill out the worksheet for Level 1. This worksheet focuses on teaching the builders to use the LCD to print different messages and explore its functions.

NOTE: THE ACTIVITY VIDEO CAN BE FOUND AT ⇒ www.BuildARobotWorkshop.com/-----

CODE This is a sample solution to the challenge above. You may also visit:
www.BuildARobotWorkshop.com/-----

```
#include <LiquidCrystal_I2C.h> // LCD Display library
#include <Wire.h> // LCD Communication library

LiquidCrystal_I2C lcd(0x3F, 20, 4);

void setup() {

    lcd.init();
    lcd.backlight();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Message 1");

}
void loop() {

    lcd.setCursor(0, 1);
    lcd.print("Message 2");
    lcd.scrollDisplayRight();

    delay(500);

}
```

Notes:

ULTRASONIC SENSOR WIRING

SENSOR MOUNTED TO PART X

Orange Wire: Connect to digital pin 8 on the Robot Brain Board

Red Wire: Connect to 5V. socket on Robot Brain Board

Brown Wire: Connect to GND socket on Robot Brain Board

LCD DISPLAY WIRING

DISPLAY MOUNTED TO PART 8

Orange Wire: Connect to digital pin 9 on the Robot Brain Board

Red Wire: Connect to 5V. socket on Robot Brain Board

Brown Wire: Connect to GND socket on Robot Brain Board

Notes:

POSSIBLE PROBLEMS AND ANSWERS

Problem: LCD display does not show text

Possible Solutions:

1. The ground and VCC pins are not connected to the (-) and (+) on the breadboard.
2. The SDA pin is not connected to A4 and the SCL pin is not connected to A5 on the Arduino Uno.
3. No batteries in battery pack(s).
4. Battery packs not turned on.
5. LCD was not initialized in the code by using `lcd.init()` or `lcd.backlight()`.