Arduino Testing List of Commands

Goals:

- Print “Hello World” to the LCD Monitor
- Read from the temperature sensor and print results
- Read from the ultrasonic sensor and print results to the LCD

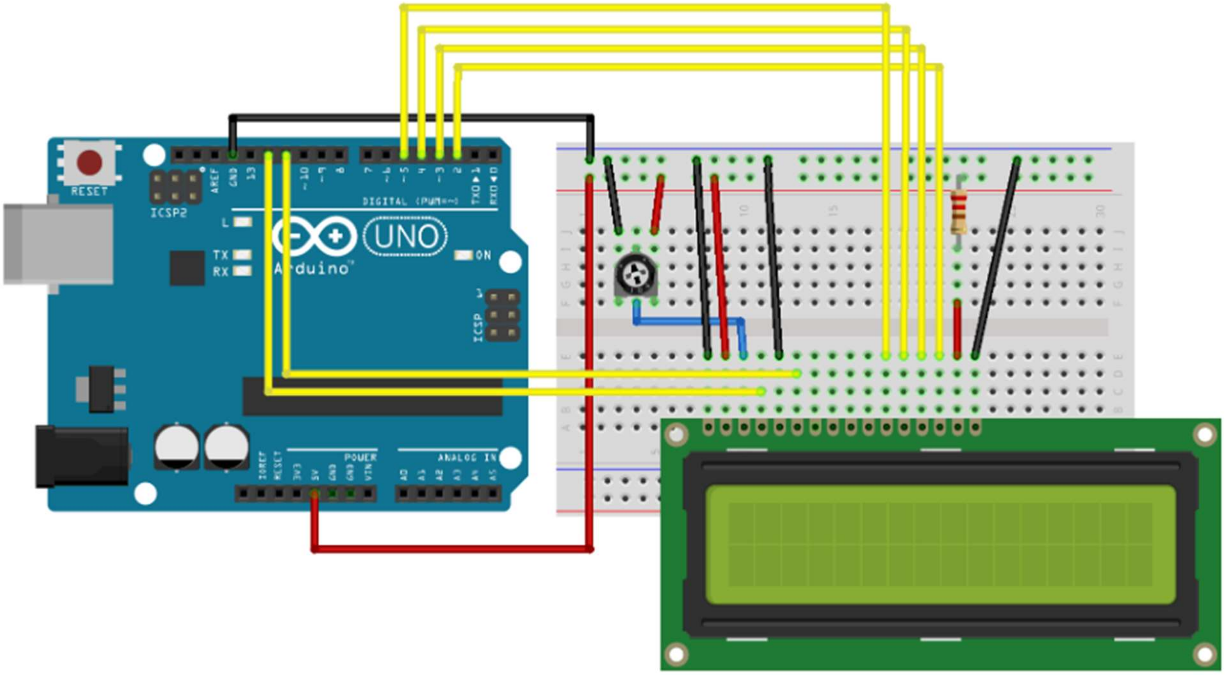
Overall Commands:

delay(*number of seconds in milliseconds*);

Serial.print();   // prints what is in the quotation marks

Serial.println();   // creates a new line

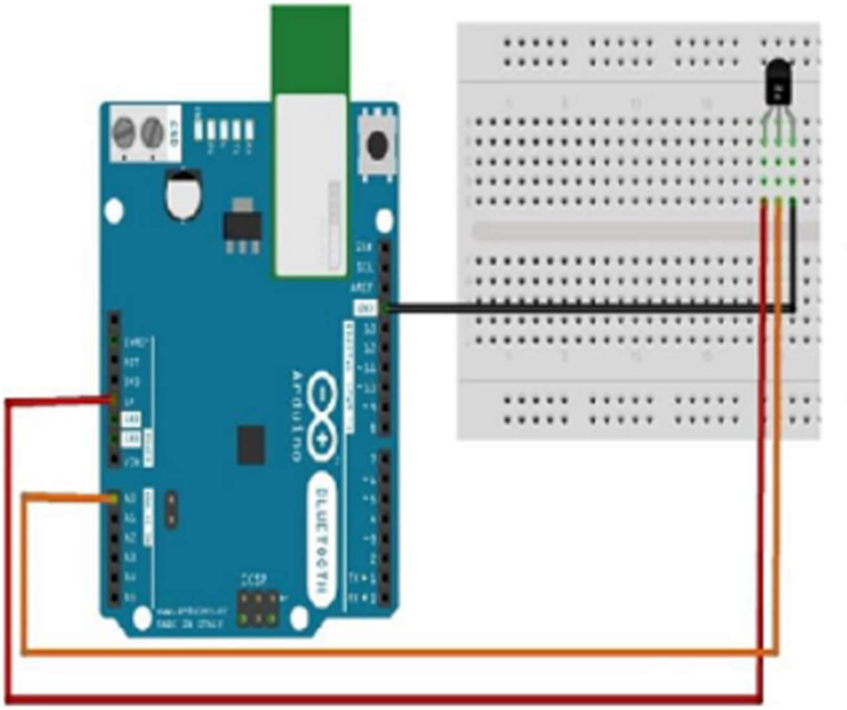
LCD Monitor



- Libraries

  - #include <Wire.h> // Library for I2C communication
  - #include <LiquidCrystal_I2C.h> // Library for LCD
- Under the libraries enter the command:
  - LiquidCrystal_I2C lcd = LiquidCrystal_I2C(0x27, 16, 2);
- Setup
  - lcd.init(); //initializes the LCD
  - lcd.backlight();
- Loop commands
  - lcd.setCursor(x,y); //set the cursor to an x and y coordinate starting with (0,0) in the first column and first row
  - lcd.print(“*example”*); //prints to the LCD screen starting where the cursor is set
- Notes:
  - All of these commands are in chronological order
  - Use code from the previous Arduino Lab and modify it to the LCD
  - Do not unplug the LCD-display the output from the temperature and distance sensors on to the LCD

Temperature Sensor



Temperature Code

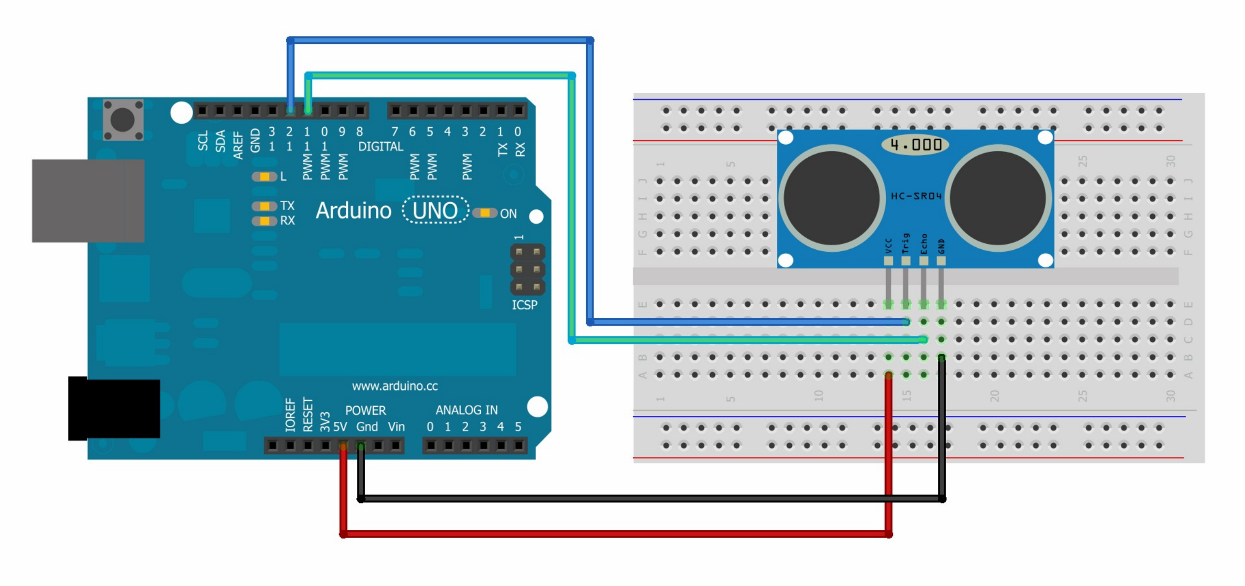
- Setup

  - Serial.begin(9600); //Sets the speed of communication
- Loop commands
  - temp = analogRead(*“pin variable/number”*); // reads from the temperature sensor
  - temp = temp * .48828125; // translates what is read to temperature in Celsius
- Notes:
  - This creates the temperature in Celsius
  - Use the print and println statements at the top to print
  - The temperature sensor uses an analog pin

Ultrasonic Sensor

How this works:

The ultrasonic sensor sends a sound out of one side (trigPin) which bounces back and is received as input in the other side (echoPin). The speed of sound must be incorporated to calculate the distance from an object.



- Setup
  - pinMode(trigPin, OUTPUT);
  - pinMode(echoPin, INPUT);
- Loop Commands
  - digitalWrite(trigPin, HIGH); // starts sending sound
  - delayMicroseconds(10);
  - digitalWrite(trigPin, LOW); // turns the sound off

    - pingTime = pulseIn(echoPin, HIGH);   //takes the input
    - pingTime=pingTime/1000000;
    - pingTime=pingTime/3600;
    - distance= speedOfSound * pingTime;
    - distance=distance/2;
    - distance= distance*63360;   //distance calculations
- **Print the distance on to the LCD**
- Notes:
    - Declare both the trig and echo pins in the beginning of the code
    - Distance is a variable. Use the variable and print it to the LCD as if it was a word or number.