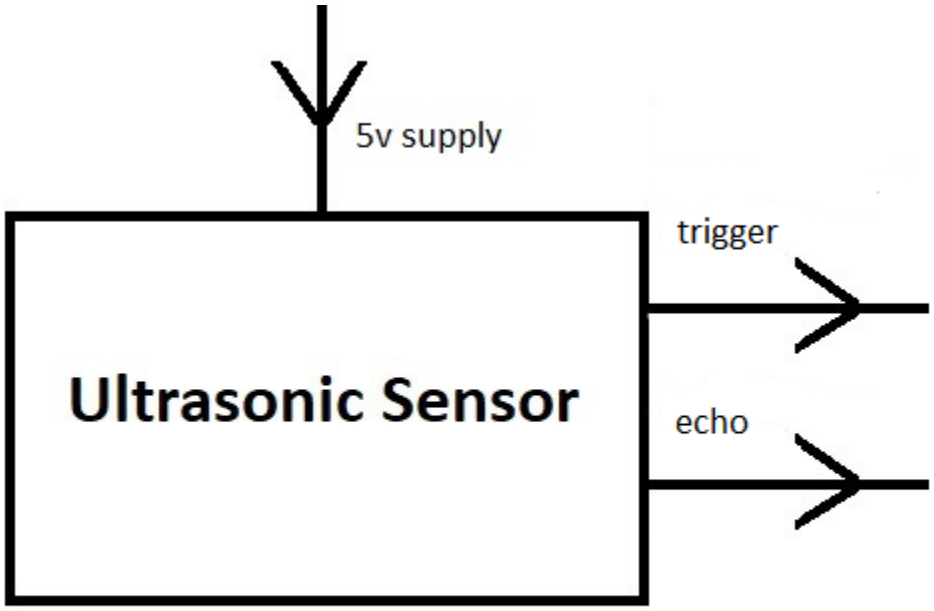
System Overview

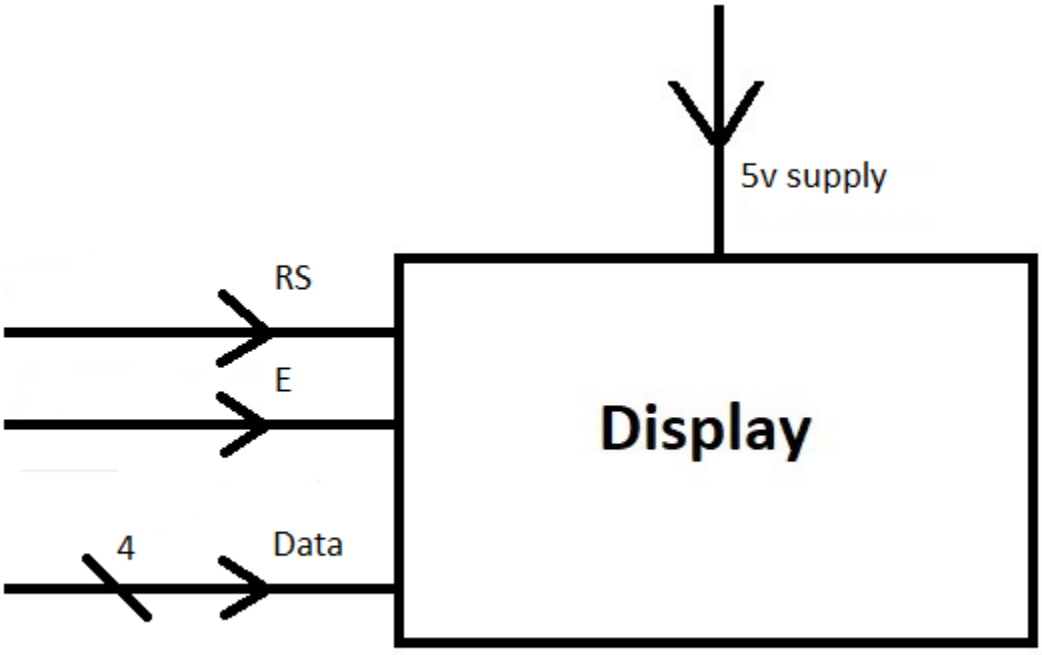
This device is a handheld rangefinder that is capable of accurately measuring and displaying the distance from the unit to a nearby object within 400 cm. The device measures the distance with an ultrasonic ranging module that emits pulses of high frequency waves and measuring the time it takes from the pulse being emitted to when it bounces off of the object being measured and returns to the ranging module. The Arduino then interprets this information and puts it into both standard and imperial units. This information is then displayed on the liquid crystal display giving a real time distance measurement.

Electrical Specifications

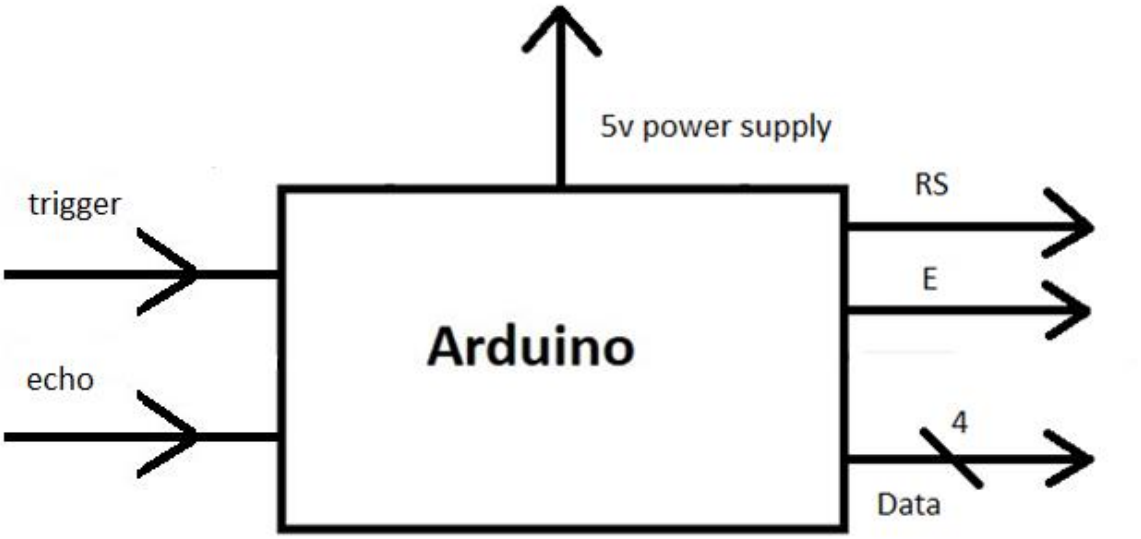
	Min Supply Voltage	Max Supply Voltage	Min Nominal Current	Max Nominal Current	Operating temperature
Handheld Rangefinder	7V DC	12V DC	30mA per pin	40mA per pin	-40C to +120C



Interface Name	Interface Type	Specifics
5v supply	DC Power	$V_{nominal}$ = 5V $I_{nominal}$ = 15mA
Trigger	Digital Signal	$pulse\ width_{min}$ = 10µs
Echo	Digital Signal	$measurement\ cycle_{min}$ = 60ms



Interface Name	Interface Type	Specifics
5v supply	DC Power	V_{max} = 5.2V $V_{nominal}$ = 5V V_{min} = 4.8V I_{max} = 2.5mA $I_{nominal}$ = 1.5 mA
RS	Digital Signal	$setup\ time_{min}$ = 80ns
E	Digital Signal	$cycle\ time_{min}$ = 500ns $pulse\ width_{min}$ = 300ns
Data	Digital Bus	$output\ delay_{min}$ = 80ns



Interface Name	Interface Type	Specifics
Trigger	Digital Signal	$pulse\ width_{min}$ = 10µs
Echo	Digital Signal	$measurement\ cycle_{min}$ =60ms
5v supply	DC Power	$V_{nominal}$ = 5V $I_{nominal}$ = 17mA I_{max} = 20 ma
RS	Digital Signal	$setup\ time_{min}$ = 80ns
E	Digital Signal	$cycle\ time_{min}$ = 500ns $pulse\ width_{min}$ = 300ns
Data	Digital Bus	$output\ delay_{min}$ = 80ns

User Guide

For power, the rangefinder can take a USB connection, or a 5.5 x 2.1 mm barrel jack connector. Once power is supplied to the rangefinder it will begin displaying the distance between it, and the nearest object in front of the sensor. From there simply hold the device and point the sensor towards something to measure the distance. When measuring be sure to hold the device steady and keep it as level as possible. If the device is held at too steep of an angle the measurements will be inaccurate or the system could throw an error, especially if the distance being measured is at the upper ends of the system's range limitations. Ideally measurement should be made with the object at 90 degrees relative to the rangefinder unit. The most accurate and stable ranges are between 5 to 300 cm, and while the device can measure up to 400 cm in ideal circumstances it is very sensitive to the movement of the device, as well as the angle it's being held at.

Design Artifact Figures

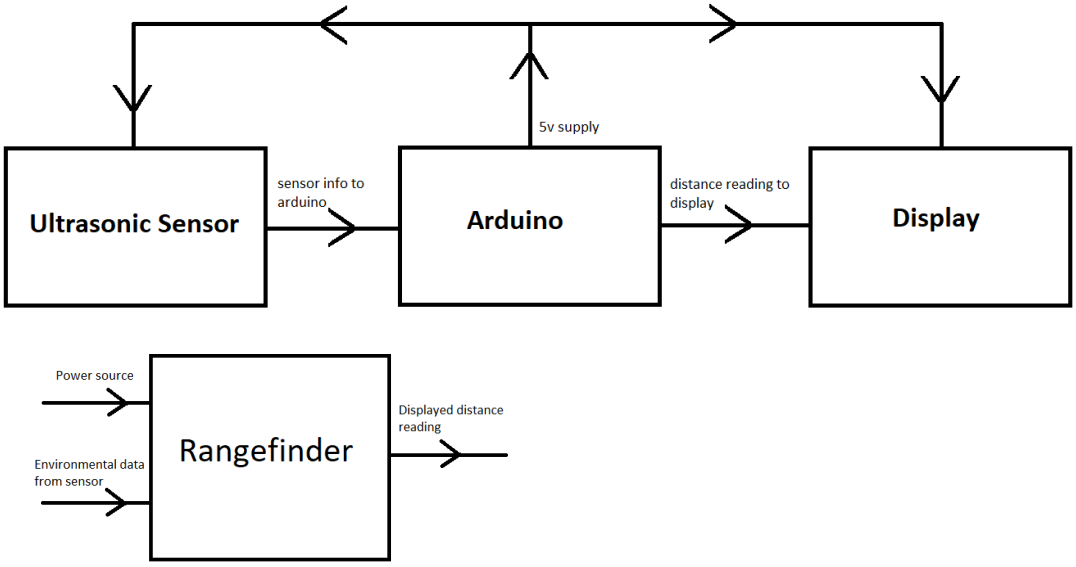


Figure 1, Block Diagram and Black Box Representation

The block diagram helps to boil the rangefinder down into its core three components, the sensor that records the information, the Arduino that processes the information, and the display that shows the information to the user. By looking at the system as a black box it only requires the two inputs, power and data from the sensors, in order to output the readable data to the display.

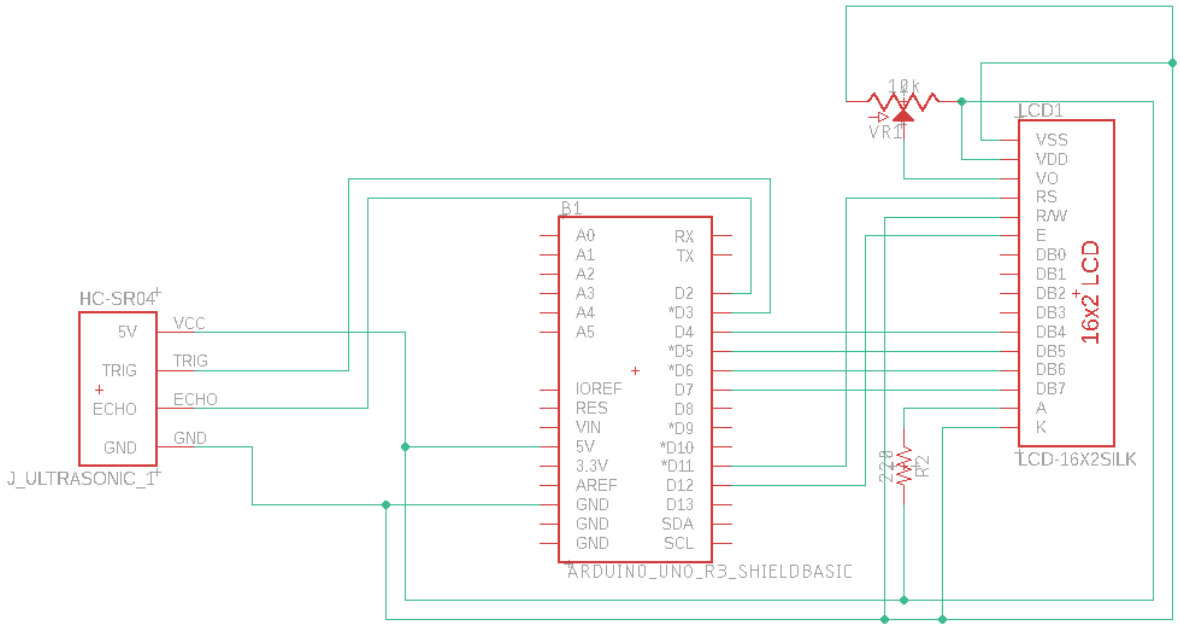


Figure 2, Schematic

The schematic shows two more components that were excluded from the block diagram from simplicities sake, the 220 Ω resistor and the 10k Ω potentiometer that both connect to the liquid crystal display. The 220 Ω resistor's purpose is to prevent the on/off LEDs from burning up due to unrestricted current. The middle leg of the potentiometer is routed to the V_0 pin on the display which controls the contrast. This allows the user to adjust the contrast to a level that is visible for them.

PCB Information

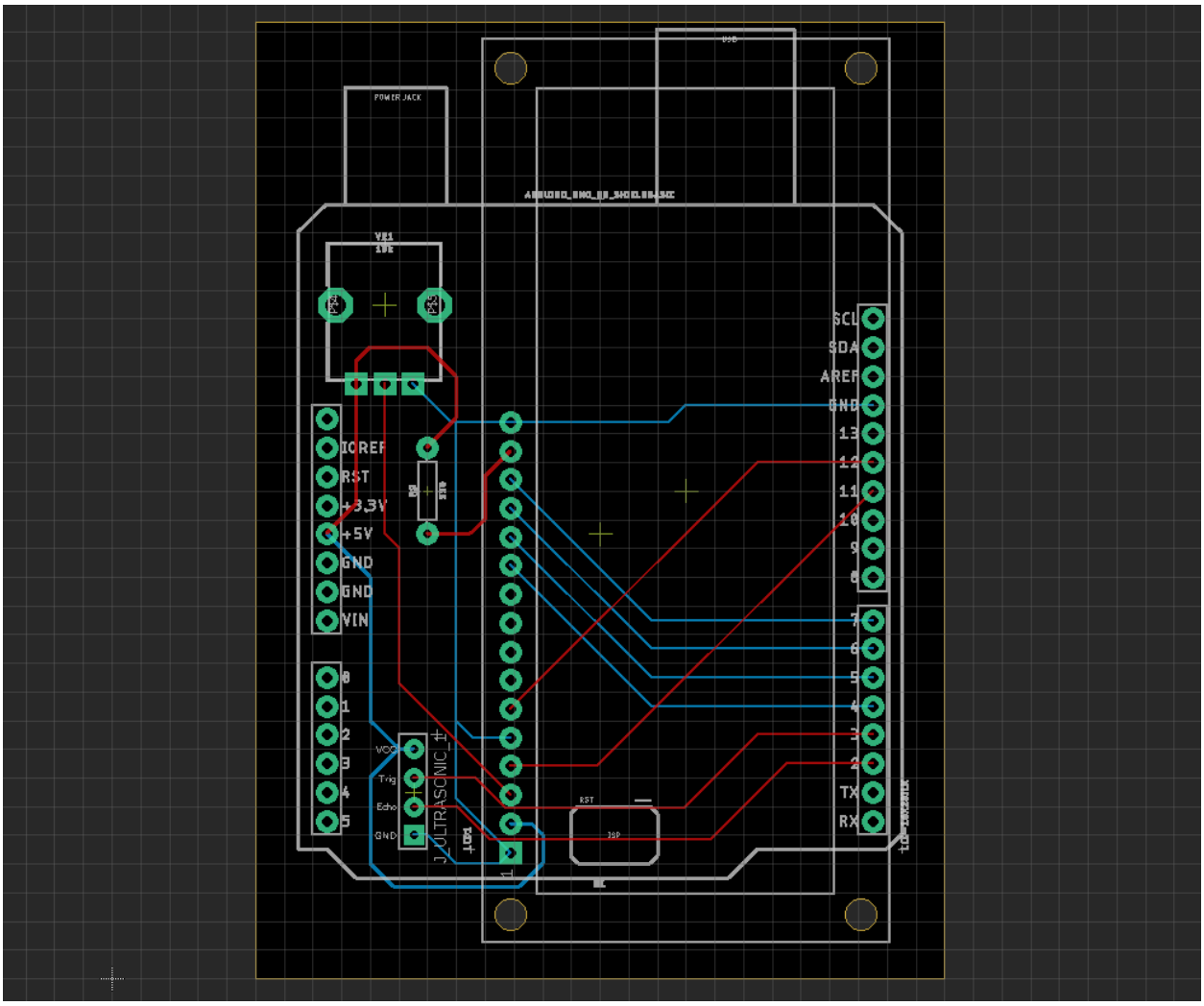


Figure 3, PCB design file

Dimensions: 60.94 mm x 84.76 mm or 2.4 in x 3.3 in

The PCB was designed to sit on top of the Arduino for a slim profile. The width of the signal traces is 0.152 mm or 0.006 in, and the width of the traces coming off of the 5V pin are 0.254 mm or 0.01 in. The power traces were made slightly wider to account for the higher current that would be flowing through them. In this layout the ultrasonic sensor will be pointing to the left, this puts the display right underneath it allowing the user to hold it in their hand and point it while still letting them read the display at the same time.

Part Information

	Amount	**Cost**
HC SR04 Ultrasonic Proximity Sensor	1	$3.99
ADM1602K Liquid Crystal Display	1	$15.00
220 Ω Resistor	1	$0.75
10k Ω Potentiometer	1	$1.02
Arduino Uno	1	$22.95