

# Steps to build our project

## **Steps to build Subsystem 1 (LCD Countdown timer clock):**

- Implement a countdown timer on the LCD screen which is configurable by 3 pushbuttons which also sends input to subsystem 2.

### **Materials**

1x Arduino  
1x Breadboard  
3x Pushbutton  
3x 10k Ohm resistor  
1x 220k ohm resistor  
1x LCD screen  
1x Potentiometer  
Wires

Preset:

- Arduino 5V to + on breadboard first row
- Arduino GND to – on breadboard first row

To wire your LCD screen to your board, connect the following pins:

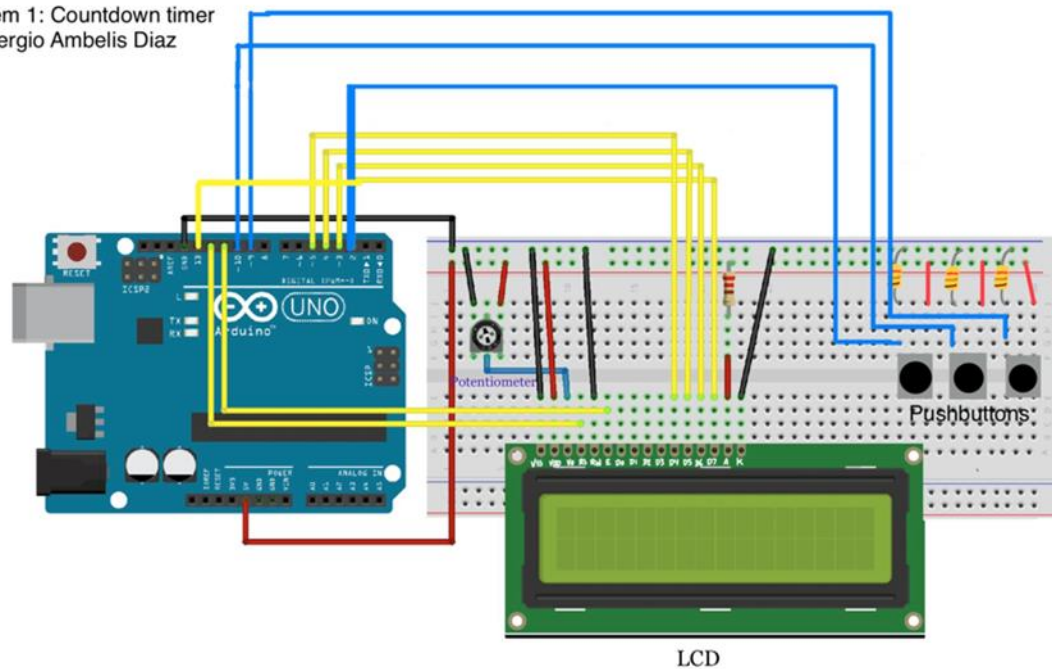
- LCD RS pin to digital pin 12
- LCD Enable pin to digital pin 11
- LCD D4 pin to digital pin 5
- LCD D5 pin to digital pin 4
- LCD D6 pin to digital pin 3
- LCD D7 pin to digital pin 2
- LCD R/W pin to GND
- LCD VSS pin to GND
- LCD VDD pin to 5V
- LCD LED+ A to 5V through a 220 ohm resistor
- LCD LED- K to GND
- LCD V0 to potentiometer (also top leg to ground and bottom leg to power)

To wire pushbutton:

- PUSHBUTTON(1) to pin 13 through a 10k ohm resistor to ground (bottom leg to power)
  - PUSHBUTTON(2) to pin 10 through a 10k ohm resistor to ground (bottom leg to power)
  - PUSHBUTTON(3) to pin 9 through a 10k ohm resistor to ground (bottom leg to power )
- 
- Button1 (top): start timer/ stop timer / reset if hold for 10 secs
  - Put in code a trigger to send to Arduino 2.
  - Button2 (middle) : increases timer by a minute
  - Button3 (bottom): decrements by a minute up to  $\geq 0$  (which sends a Tx byte at 0)

### Diagram Subsystem 1:

Subsystem 1: Countdown timer  
By: Sergio Ambelis Diaz



## Steps to build Subsystem 2 (Fan Alarm):

- Implement a buzzer and 6V motor that turn off with 2 push buttons and turn on when it receives a signal from subsystem 1 and subsystem 2 respectively.

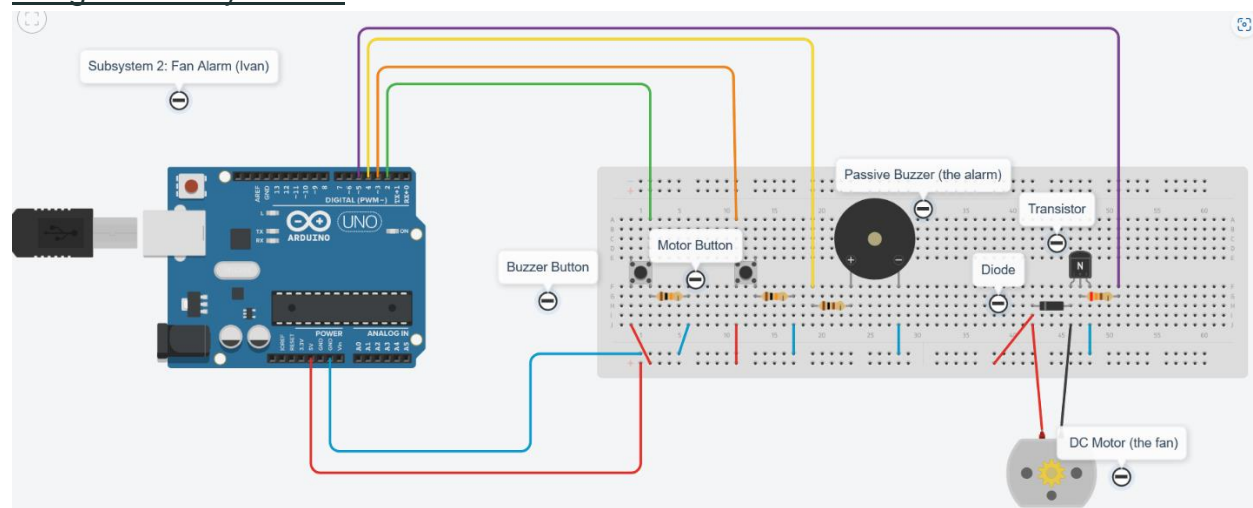
### Materials

1x Arduino  
1x Breadboard  
2x Pushbutton  
2x 10k Ohm resistor  
1x 100 Ohm resistor  
1x 220 Ohm resistor  
1x Passive Buzzer  
1x NPN Transistor  
1x Diode Rectifier  
1x Fan Blade  
1x 3-6V Motor  
Wires

### Subsystem 2 (Fan Alarm):

- The first steps to build the Fan Alarm subsystem are to set up the push buttons. We'll need 2 10k ohm resistors and 2 push buttons. Each button will have its own pins in the Arduino: pin 2 for the buzzer button and pin 3 for the motor button. Make sure to connect the buttons to the 5 volt pin, otherwise their inputs will not be detected.
- We will set up the passive buzzer, we'll need a 100 ohm resistor and a passive buzzer. Connect the positive end of the buzzer to the resistor and connect that to pin 4 of the Arduino. Then we need to connect the negative end of the buzzer to ground.
- We'll set up the DC motor, we need the diode, transistor, 220 ohm resistor, and of course the motor itself. The positive end of the motor should be connected to the 5 volt pin and a diode should connect the positive terminal with the negative.
- A transistor should connect to the negative connection of the diode, a connection to ground, and the 220 ohm resistor. The resistor should connect to pin 5 of the Arduino. The flat side of the transistor should face the diode, resistor and ground.

### Diagram Subsystem 2:



### **Steps to build Subsystem 3 (Temperature sensor and RGB led):**

Implement a RGB LED and temperature sensor so that RGB LED color will change based on the temperature

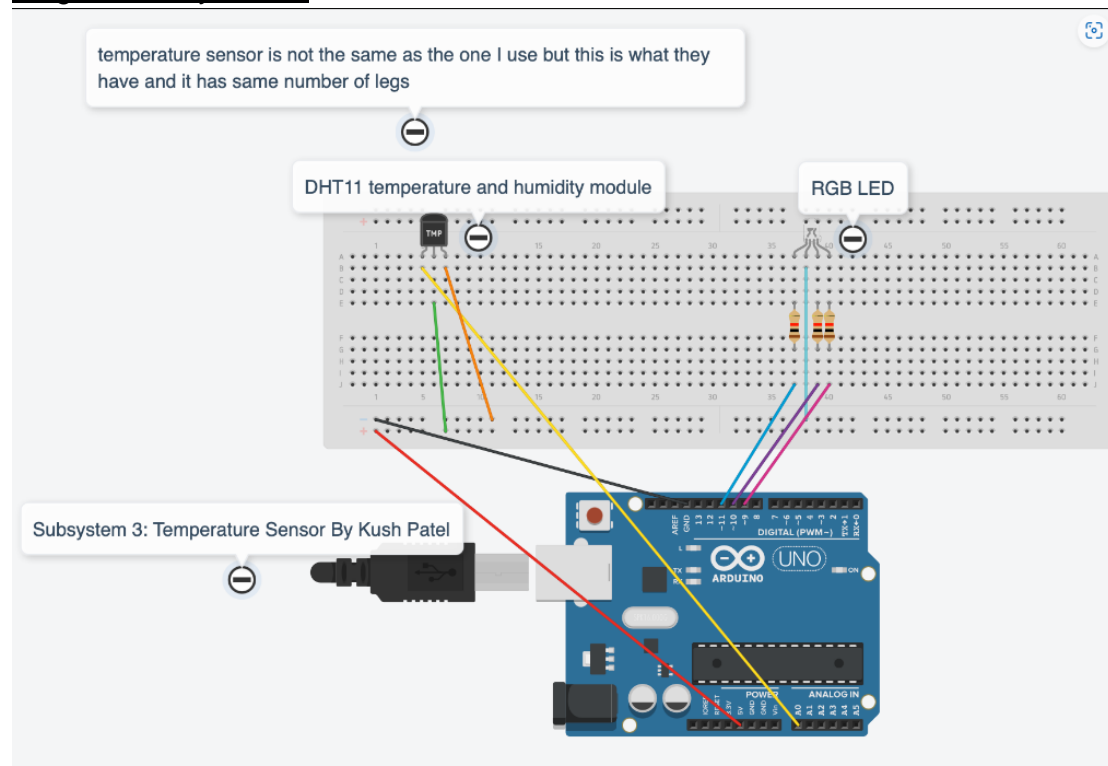
#### **Materials**

1x Arduino  
1x Breadboard  
1x RGB LED  
1x DHT11 (Temperature and humidity module)  
Wires  
3x 220k resistors

Subsystem 3: (Assume that all connections are done using breadboard jumper wires unless otherwise specified)

1. set up breadboard jumper wires from + and - lines on the breadboard to 5V and GND pins. Remember that + line connects to 5v and - line connects to GND
2. set up the temperature and humidity module(dht11) first. Stick the 3 pins(or legs) in the breadboard so there is one pin on a single horizontal line.
3. On the module there should be an S written on the side of one of the pins. The Pin closest to that S gets connected to the A0 pin using a breadboard jumper wire. The pin in the middle of the module gets connected to + line on the breadboard which is already connected to 5V. The pin farthest from the letter S on the module gets connected to the - line of the breadboard which is already connected to GND.
4. set up the RGB LED next. Stick the 4 legs in the breadboard so there is one leg on a single horizontal line.
5. The longest leg on the RGB LED gets connected to the - line that's connected to the GND. The other 3 legs represent the colors that light up so first set up one end of three 220k resistors, one on each line there is a RGB LED leg other than the longest leg.
6. Once the resistors are stuck in for the 3 shorter legs, connect the other ends of the resistor to another horizontal line with nothing on it and connect a breadboard jumper wire from that line to a pwd pin. I recommend using the digital pins 9, 10, and 11.

### Diagram Subsystem 3:



### I2C Communication Setup:

Implement communication between arduinos using I2C by connecting 2 analog ports and ground of each subsystem on a mini breadboard

#### Materials

1x Mini breadboard

Wires

#### Connection:

Analog In for each arduino/subsystem:

A5 (Analog In, wires shown as color red) to mini breadboard

- Connects to other Arduino A5 analog pins by having them on the same line on the mini breadboard.

A4 (Analog In, wires shown as color green) to mini breadboard

- Connects to other Arduino A4 analog pins by having them on the same line on the mini breadboard.

Power:

GND (Ground, wires shown as color black) to mini breadboard

- Connects to other GND pins by having them on the same line on the mini breadboard.

Diagram Of Project with communication:

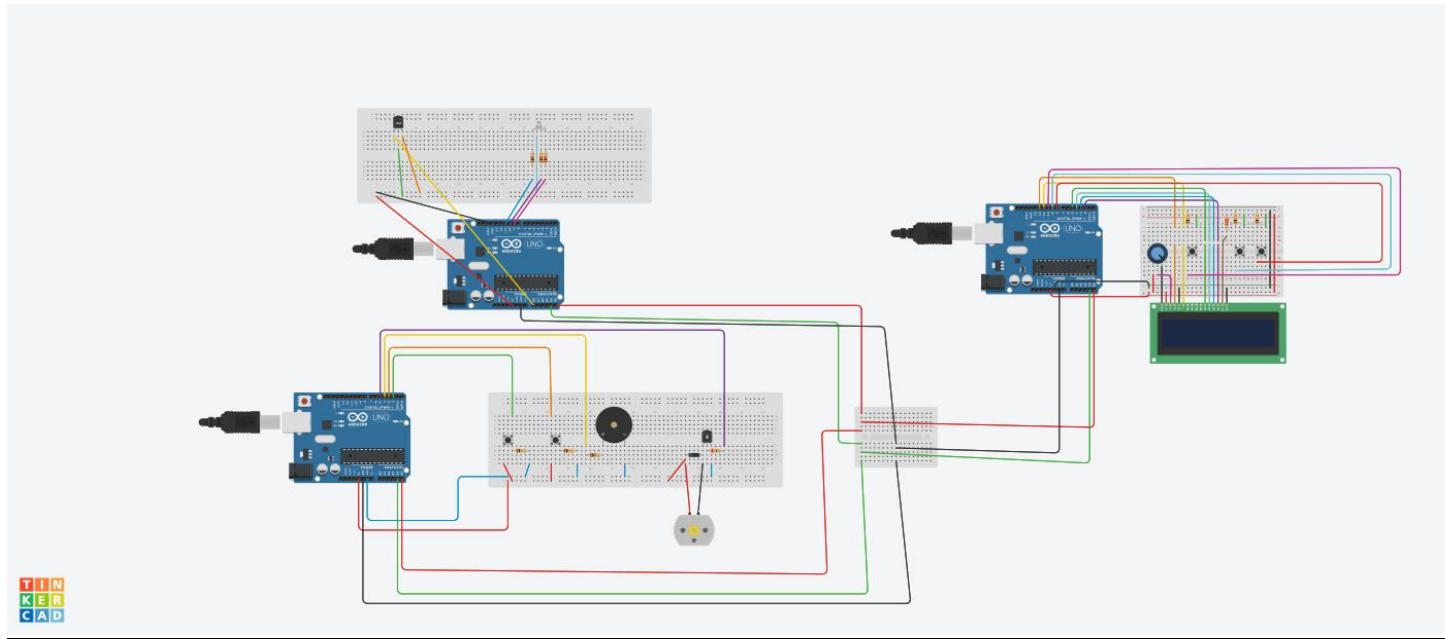


Diagram of small breadboard for communication

