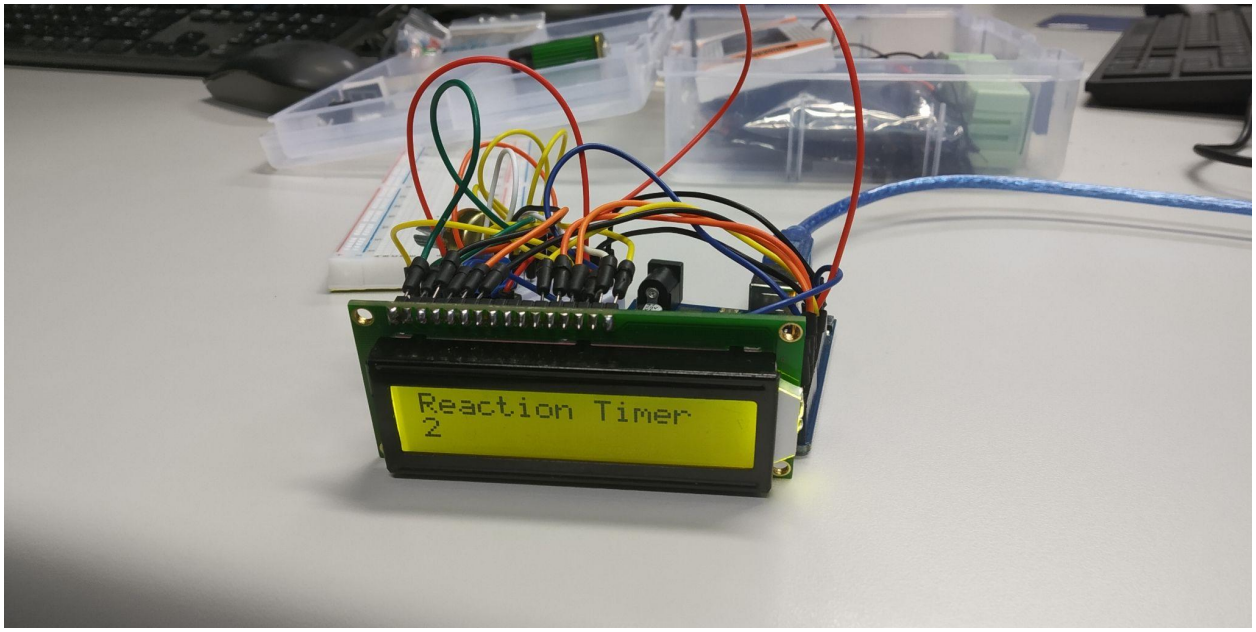
Lab 3- LCD and Reaction Timer

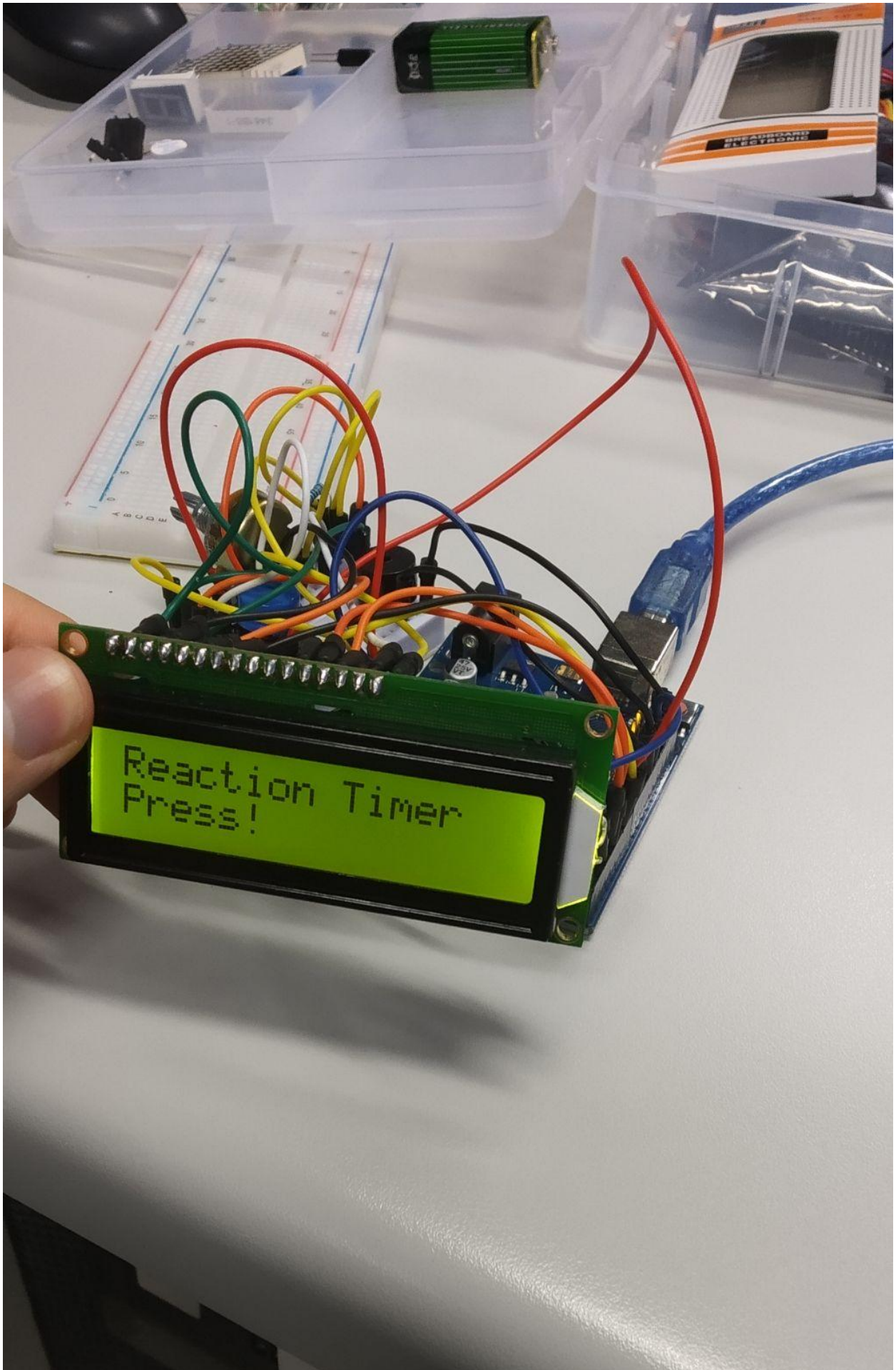
Alexander Stradnic - 119377263

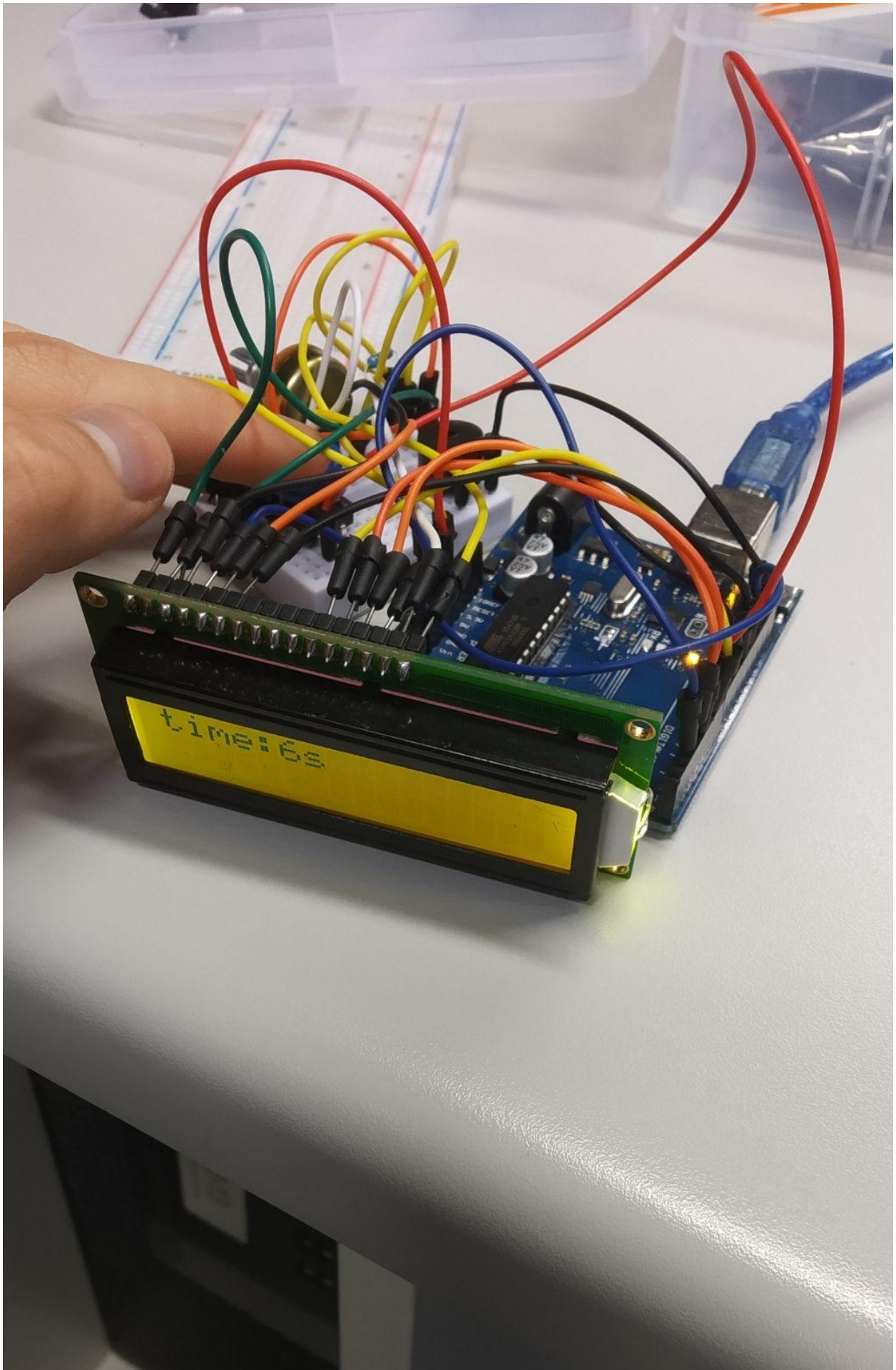
1. I set up the LCD as shown in the diagram.
2. I was then able to change the LCD suitably to show the HelloWorld example clearly.
3. Millis returns the amount of time since the Arduino program started.
4. random() generates a pseudo-random number, whereas randomSeed(n) arbitrarily chooses a part of a long sequence as a ‘random’ number, given a seed n. The sequence is always the same for the same n, however the part chosen will be different.
5. I was able to assemble a circuit that would start a timer, ending 3-10 seconds after starting. It would then play a buzzer until a button was pressed. Then it would print the time to press the button.

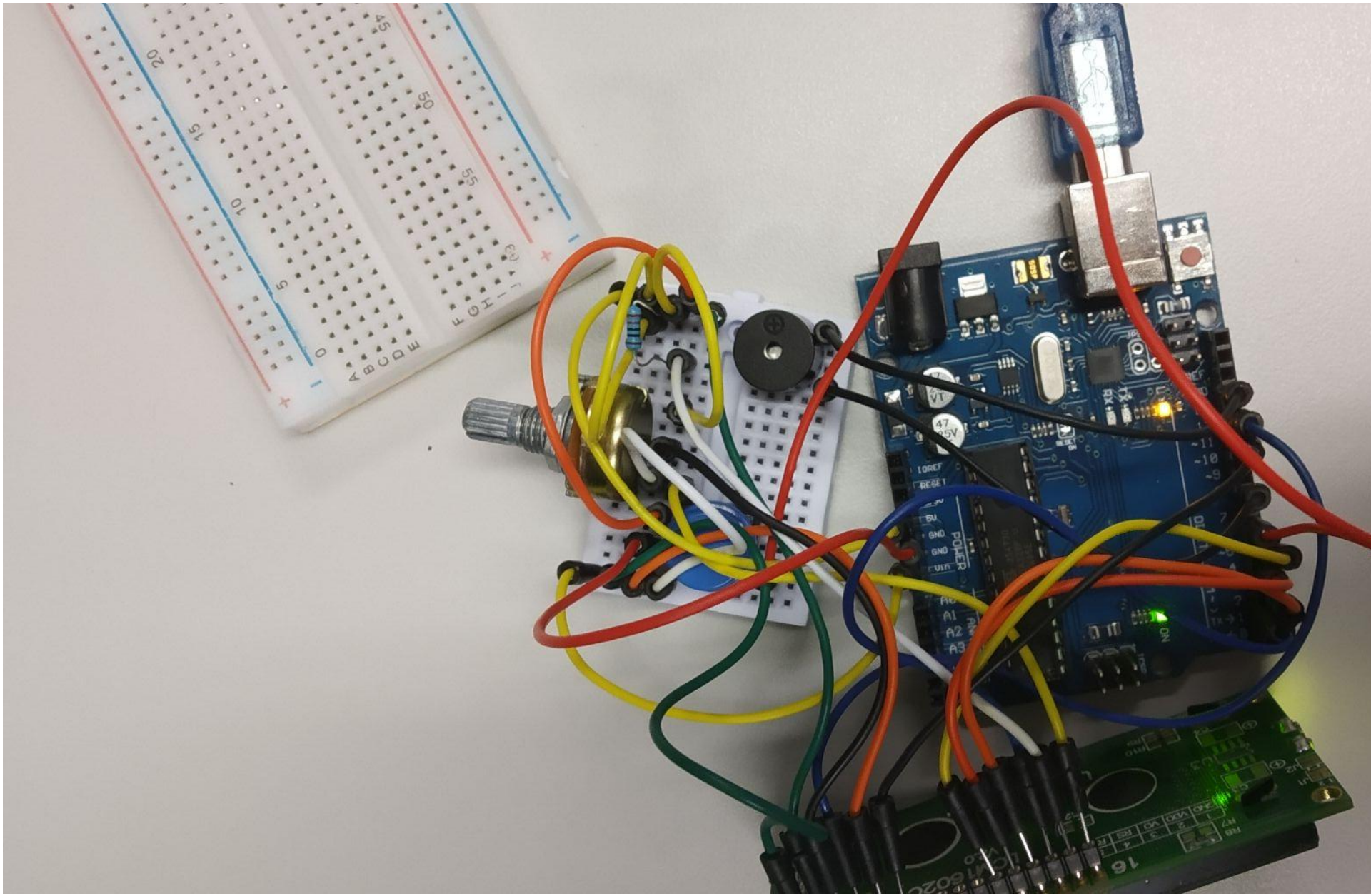
```
#include <LiquidCrystal.h>
#define PIN 8
#define BUTTON 7
// initialize the library by associating any needed LCD interface pin
// with the arduino pin number it is connected to
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
unsigned long t;
unsigned long duration;
boolean buzz;
int state;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
  lcd.begin(16, 2);
  buzz = true;
  lcd.print("Reaction Timer");
  duration = random(3000, 10000);
  pinMode(PIN, OUTPUT);
  pinMode(BUTTON, INPUT);
}
```

```
void loop() {
  t = millis();
  if(buzz){
    if(t > duration){
      digitalWrite(PIN, HIGH);
      delayMicroseconds(1800);
      digitalWrite(PIN, LOW);
      delayMicroseconds(1800);
      lcd.setCursor(0, 1);
      lcd.print("Press!");
      state = digitalRead(BUTTON);
      if(state == HIGH){
        buzz = false;
        lcd.setCursor(0,1);
        char s[20];
        sprintf(s, "%d", ((t-duration)/1000));
        lcd.clear();
        lcd.print("time:");
        lcd.print(s);
        lcd.print("s");
      }
    }
    else{
      lcd.setCursor(0, 1);
      lcd.print(t/1000);
    }
```


Reaction Timer
2


Reaction Timer
Press!


time:63


IOREF
RESET
5V
GND
GND
Vin
POWER
A0
A1
A2
A3
ON
RX
TX