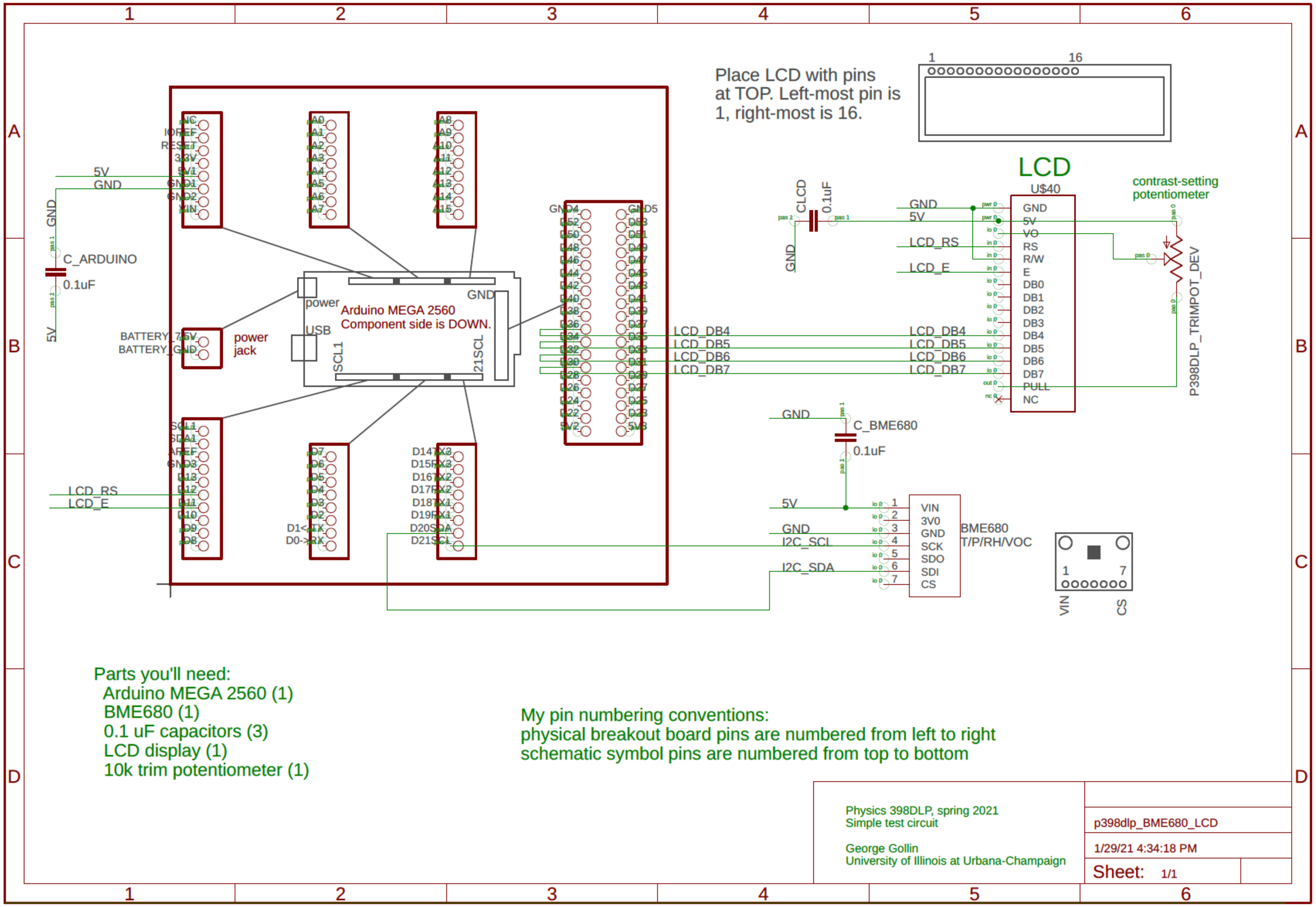
HOMEWORK DEBUNKING

Dr. Riccardo Longo
01/27/2023
Week 2

How do we read this?



Parts you'll need:
- Arduino MEGA 2560 (1)
- BME680 (1)
- 0.1 uF capacitors (3)
- LCD display (1)
- 10k trim potentiometer (1)

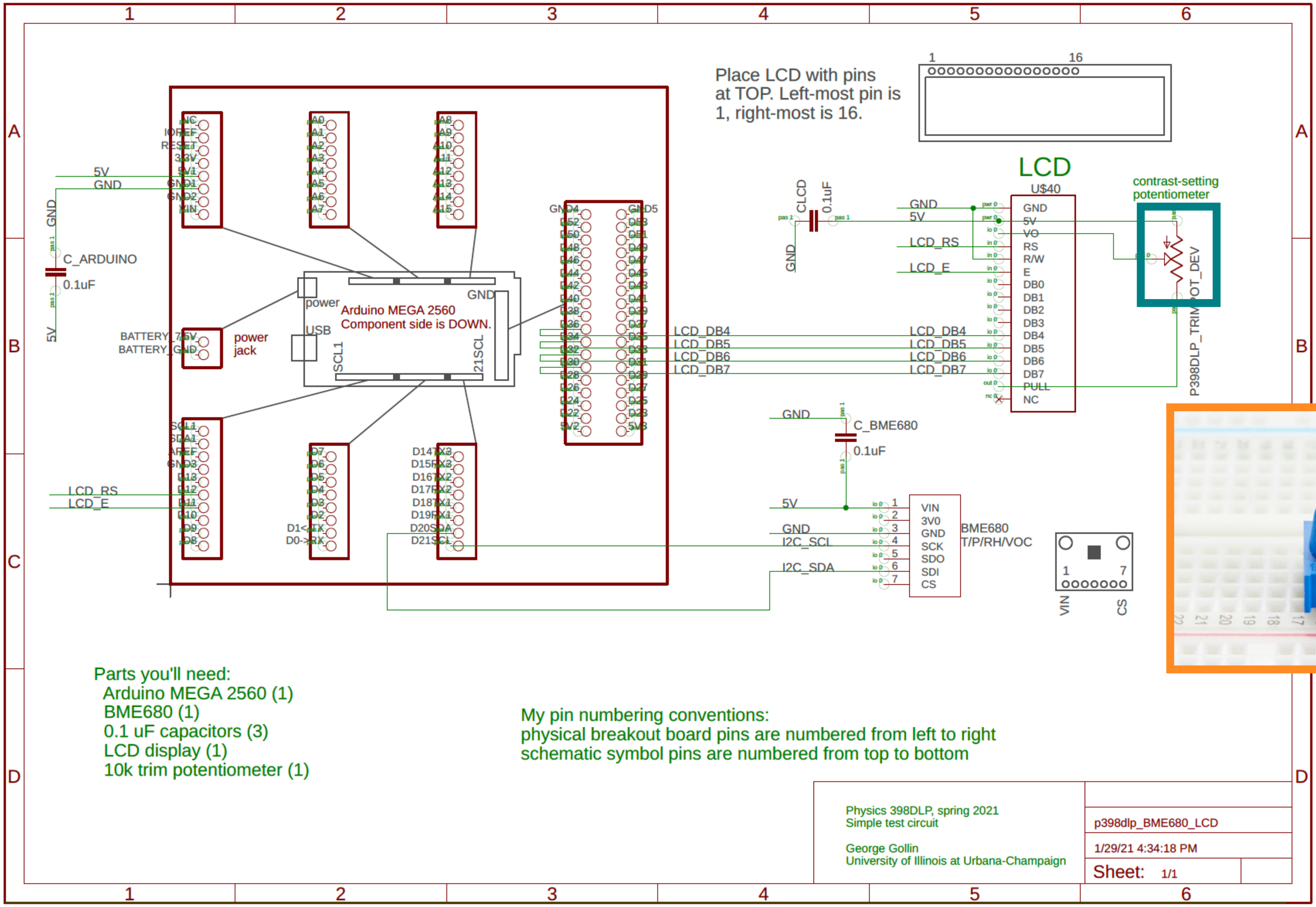
My pin numbering conventions:
physical breakout board pins are numbered from left to right
schematic symbol pins are numbered from top to bottom

Physics 398DLP, spring 2021
Simple test circuit

George Gollin
University of Illinois at Urbana-Champaign

p398dlp_BME680_LCD
1/29/21 4:34:18 PM
Sheet: 1/1

No - it's not the same arrow ☝️



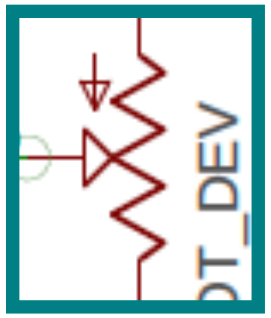
Parts you'll need:
- Arduino MEGA 2560 (1)
- BME680 (1)
- 0.1 uF capacitors (3)
- LCD display (1)
- 10k trim potentiometer (1)

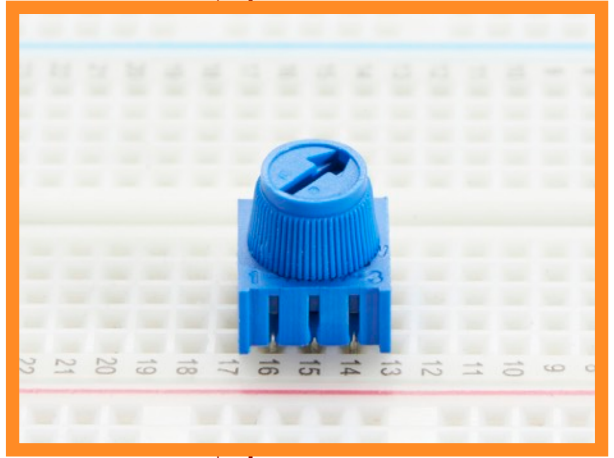
My pin numbering conventions:
physical breakout board pins are numbered from left to right
schematic symbol pins are numbered from top to bottom

Physics 398DLP, spring 2021
Simple test circuit

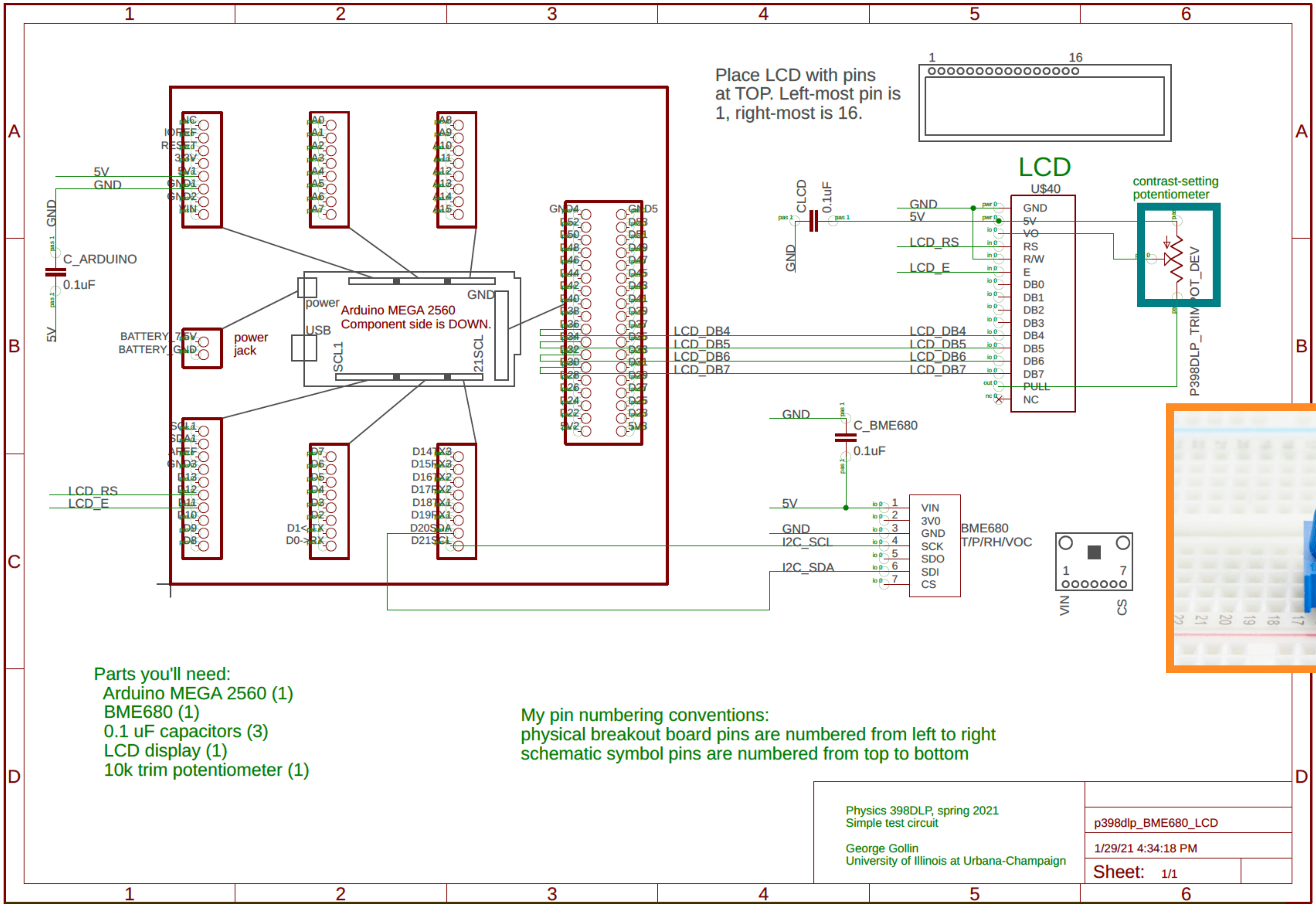
George Gollin
University of Illinois at Urbana-Champaign

p398dlp_BME680_LCD
1/29/21 4:34:18 PM
Sheet: 1/1



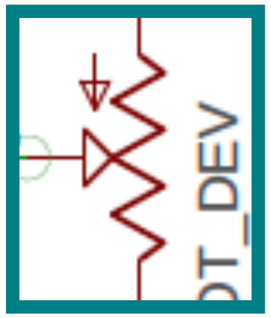


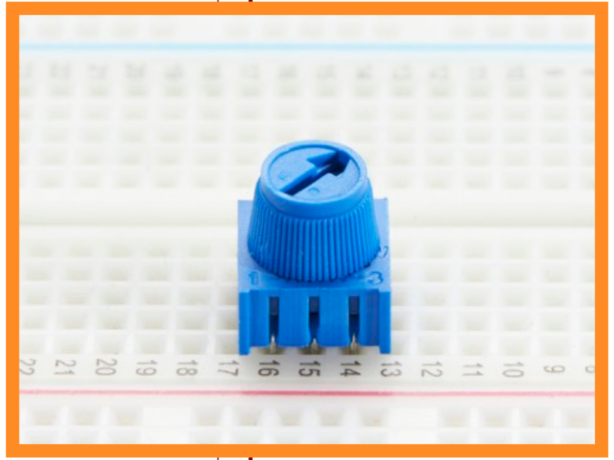
No - it’s not the same arrow ☝️



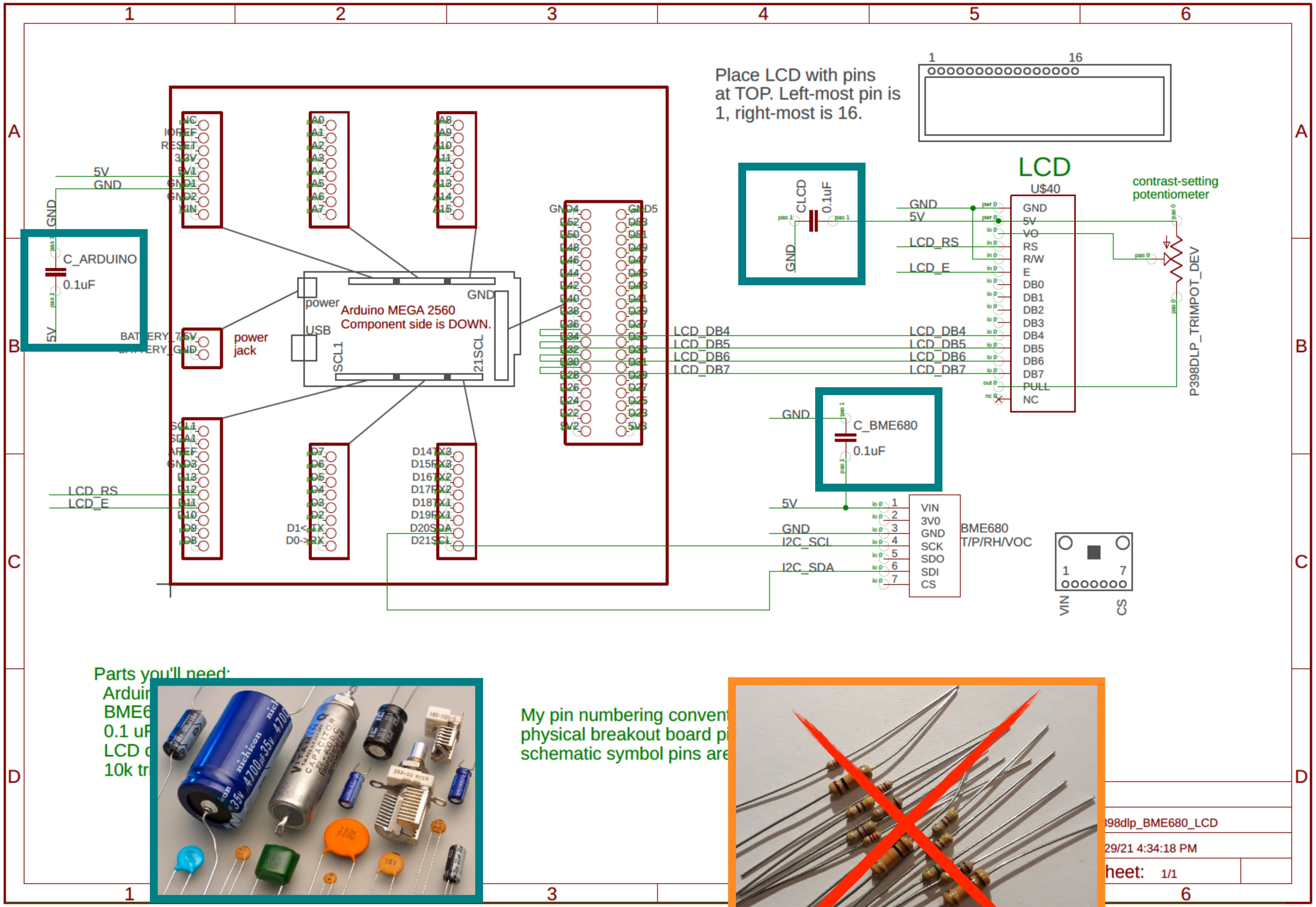
Place LCD with pins at TOP. Left-most pin is 1, right-most is 16.
LCD
U$40
contrast-setting potentiometer
Arduino MEGA 2560
Component side is DOWN.
power jack
C_ARDUINO
0.1uF
CLCD
0.1uF
C_BME680
0.1uF
BME680
T/P/RH/VOC
VIN 3V0 GND SCK SDO SDI CS
GND 5V VO RS R/W E DB0 DB1 DB2 DB3 DB4 DB5 DB6 DB7 PULL NC
LCD_RS
LCD_E
LCD_DB4
LCD_DB5
LCD_DB6
LCD_DB7
I2C_SCL
I2C_SDA
P398DLP_TRIMPOT_DEV
Parts you'll need:
Arduino MEGA 2560 (1)
BME680 (1)
0.1 uF capacitors (3)
LCD display (1)
10k trim potentiometer (1)
My pin numbering conventions:
physical breakout board pins are numbered from left to right
schematic symbol pins are numbered from top to bottom
Physics 398DLP, spring 2021
Simple test circuit
George Gollin
University of Illinois at Urbana-Champaign
p398dlp_BME680_LCD
1/29/21 4:34:18 PM
Sheet: 1/1

It’s a diode symbol!





Those are capacitors, not resistors

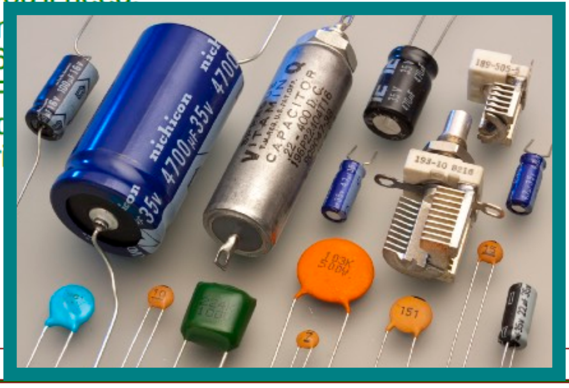


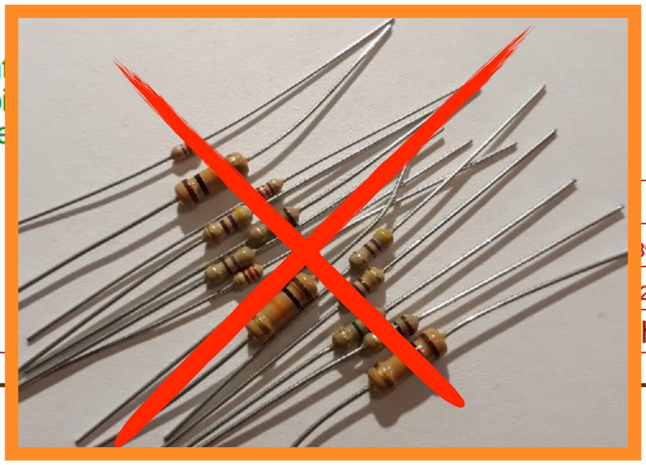
Place LCD with pins at TOP. Left-most pin is 1, right-most is 16.

Parts you'll need:
Ardui
BME6
0.1 uF
LCD d
10k tr

My pin numbering convent
physical breakout board pi
schematic symbol pins are

398dlp_BME680_LCD
29/21 4:34:18 PM
heet: 1/1

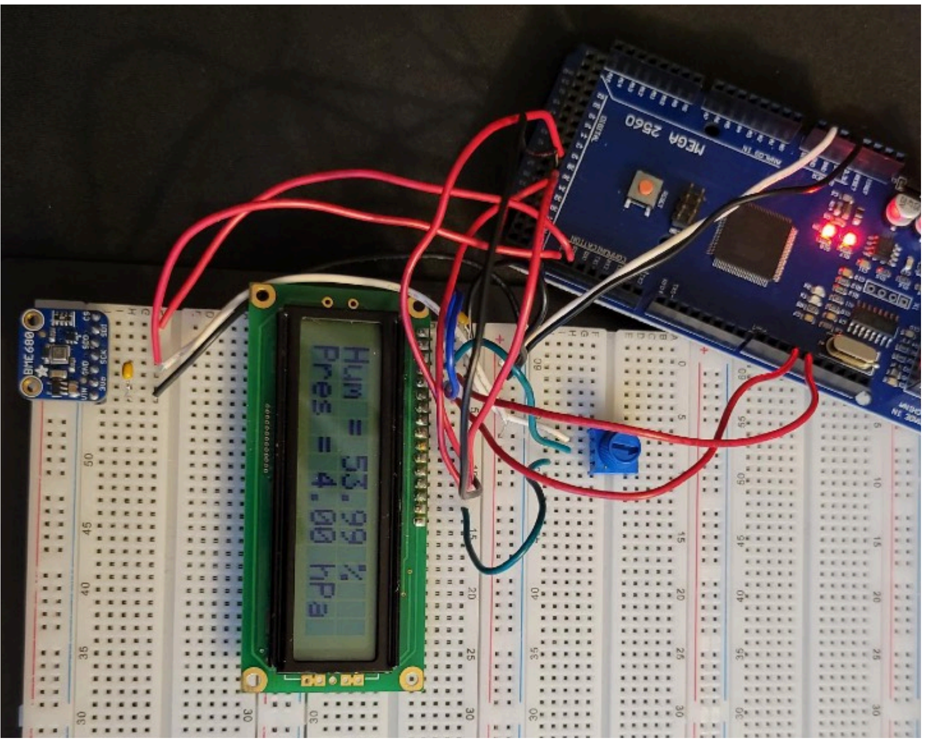
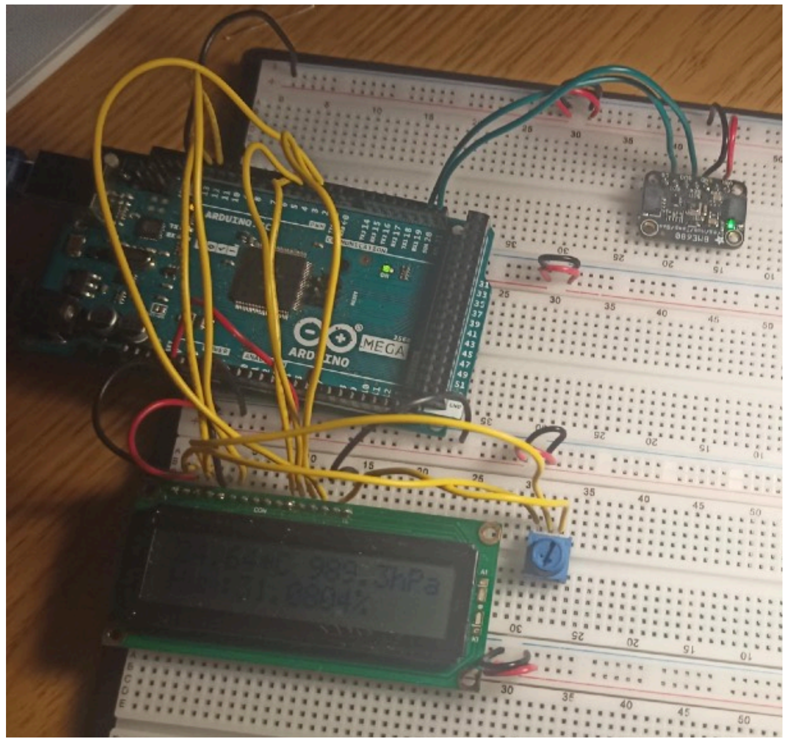
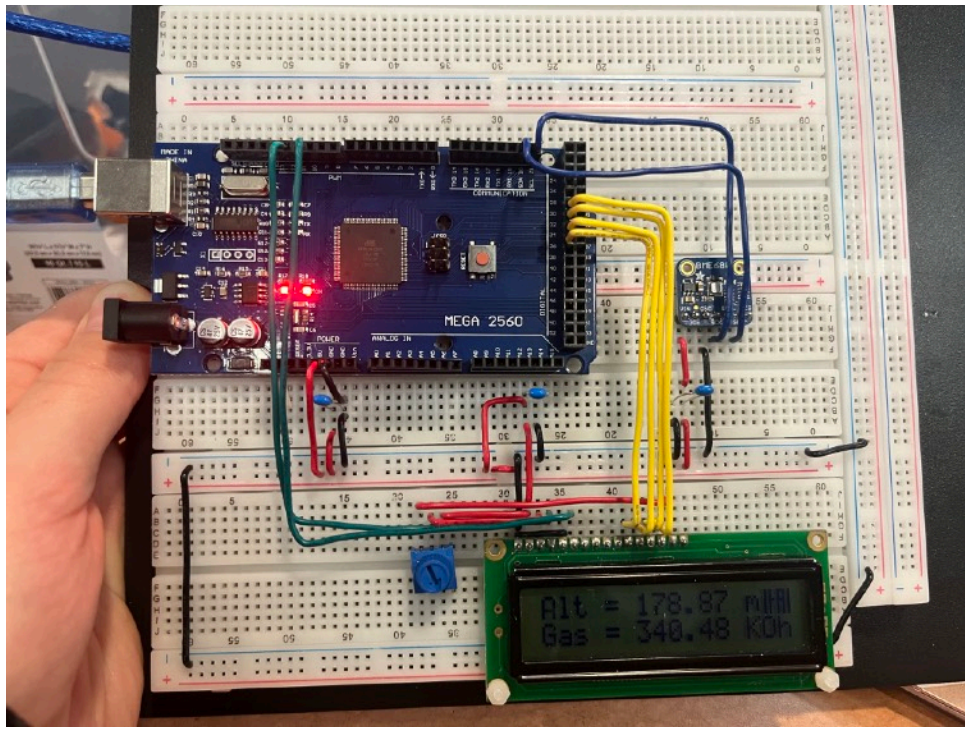




How do I wire my breadboard?

A few good practices:

1. Use an intuitive color code! **HV —> Red**, **Ground —> Black.**
2. Use the + and - rails for HV and Ground, respectively. Do not cluster you HV and ground on one of the internal pin arrays.
3. Use the other colors for communication lines and other tasks. Avoid using **Red** and **Black** for these purposes.
4. Try to keep your wires organized! Sloppy wiring will cause many headaches as the complexity of your projects increase.

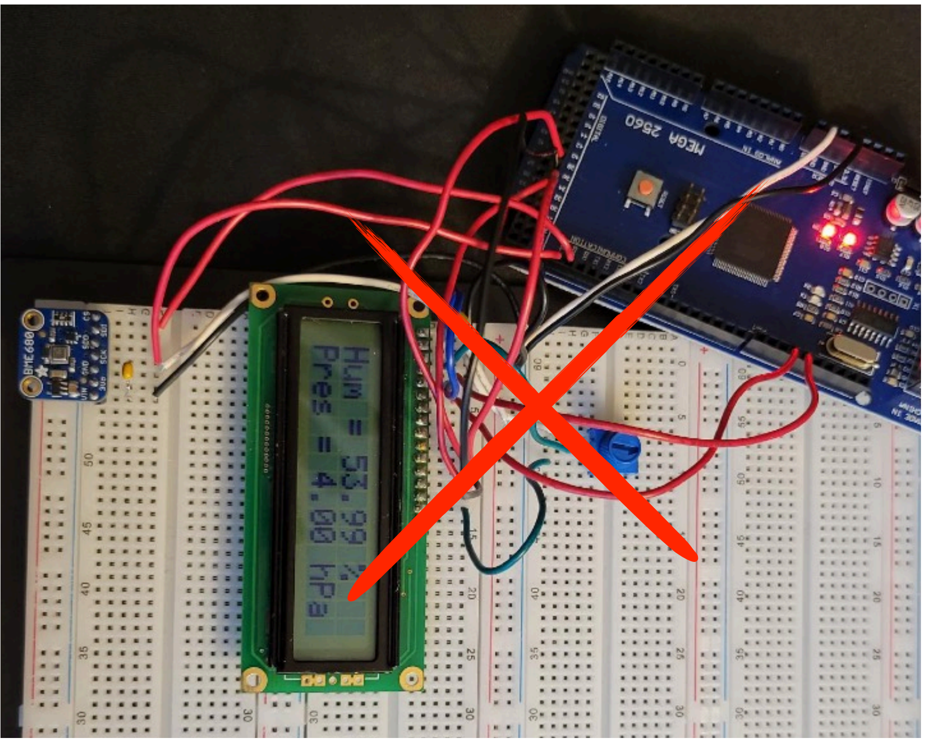
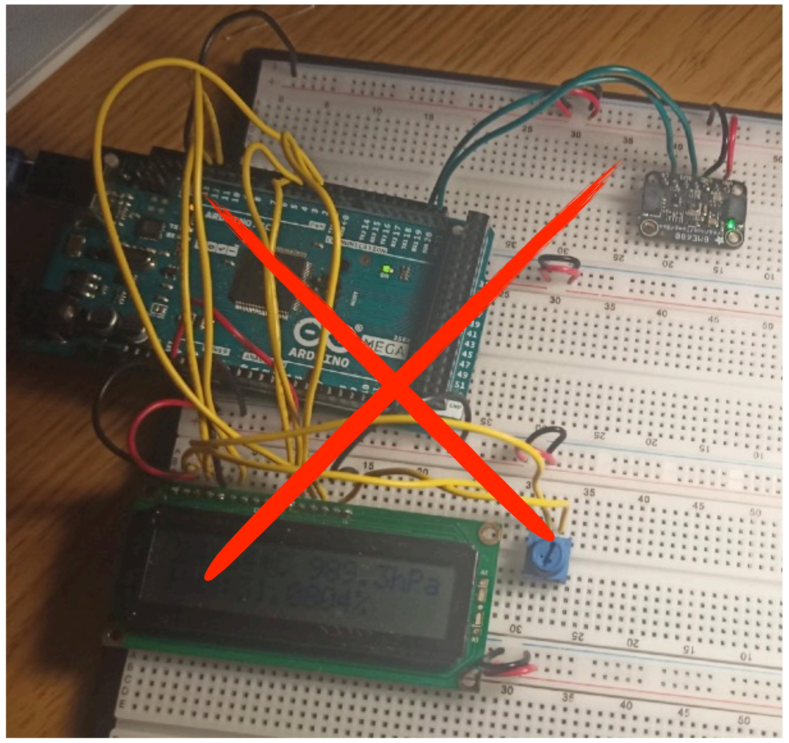
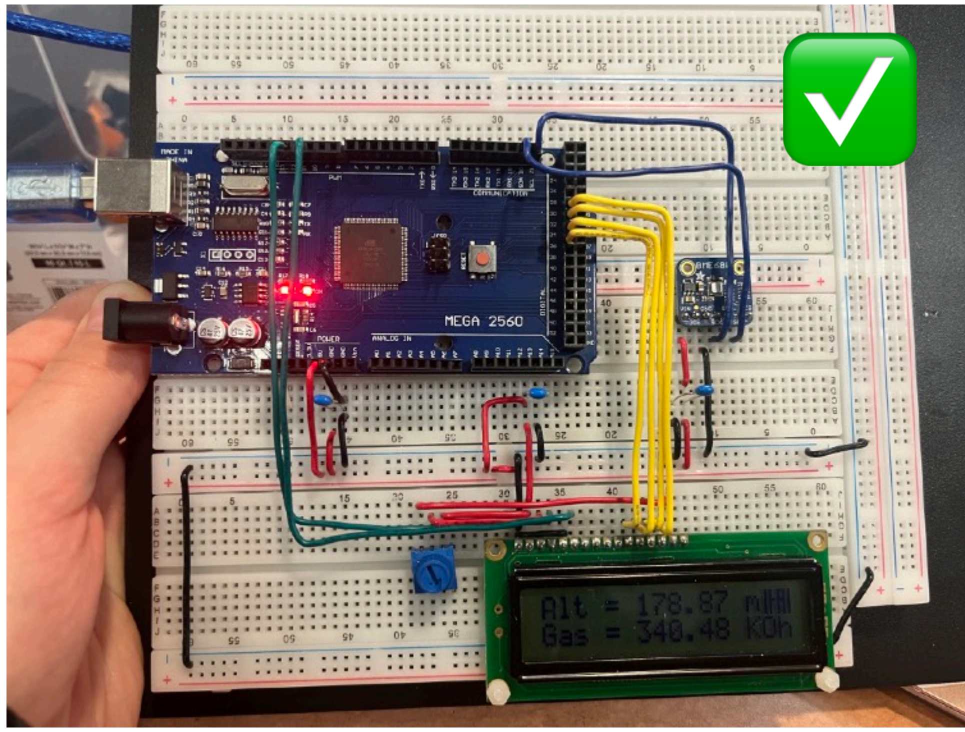




Random selection from HW turned in

How do I wire my breadboard?

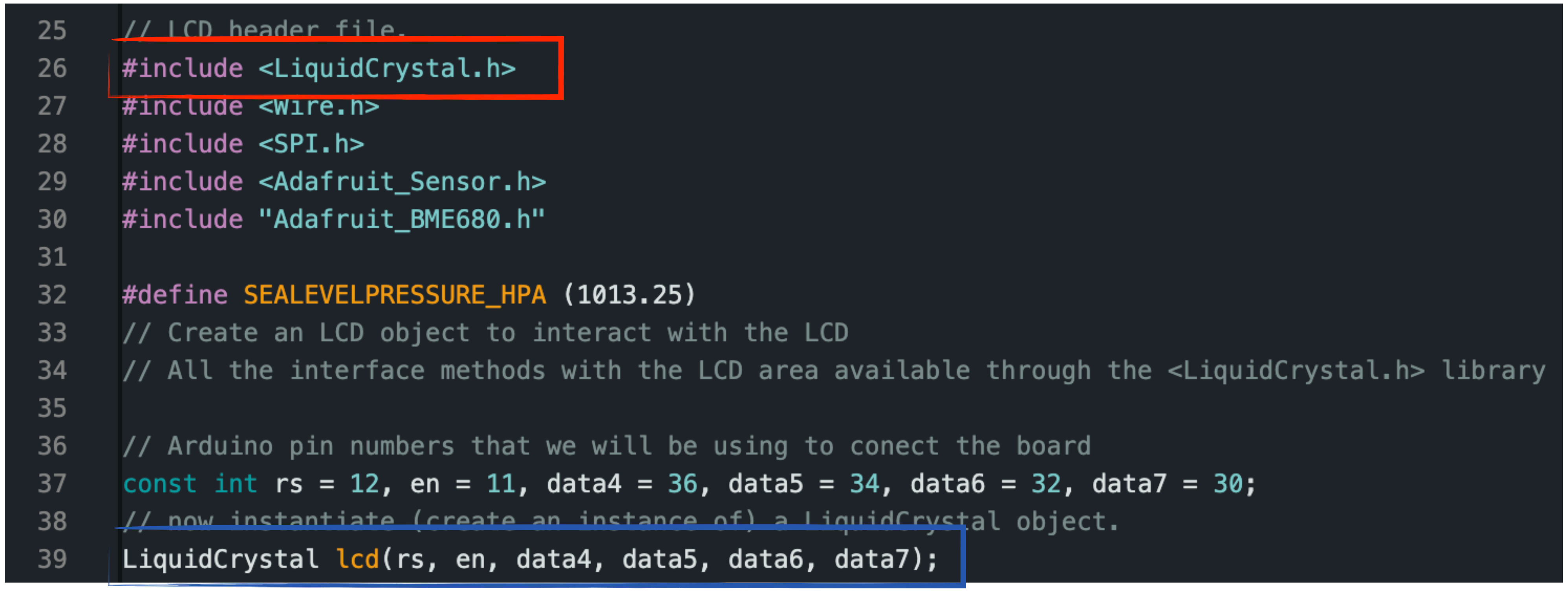
A few good practices:

1. Use intuitive color code! **HV —> Red**, **Ground** —> **Black**
2. Use the + and - rails for HV and Ground, respectively. Do not cluster you HV and ground on one of the internal pin arrays
3. Use the other colors for communication lines and other tasks. Avoid using **Red** and **Black** for these purposes.
4. Try to keep your wires organized, otherwise adding new parts will naturally increase the number of your headaches

Random selection from HW turned in

I have my circuit… how do I drive it?



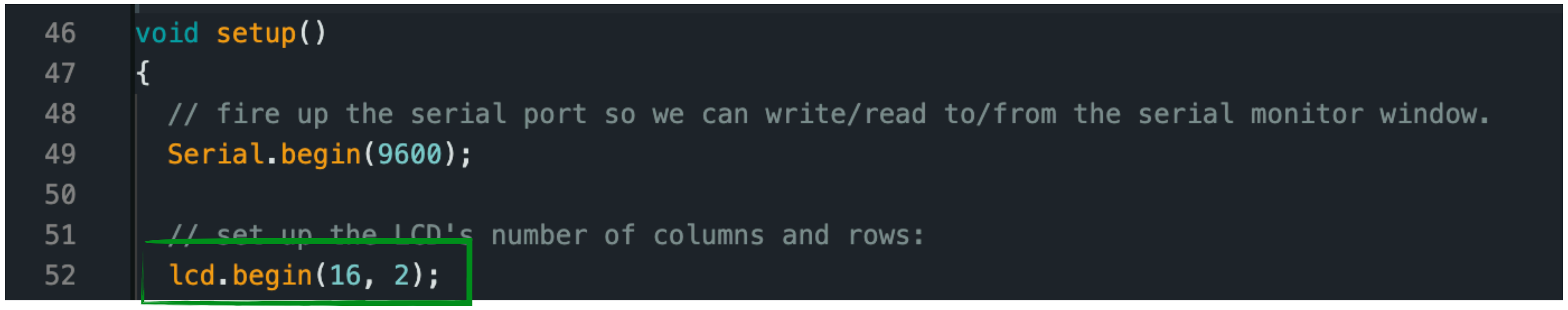
```
// LCD header file.
#include <LiquidCrystal.h>
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include "Adafruit_BME680.h"

#define SEALEVELPRESSURE_HPA (1013.25)
// Create an LCD object to interact with the LCD
// All the interface methods with the LCD area available through the <LiquidCrystal.h> library

// Arduino pin numbers that we will be using to conect the board
const int rs = 12, en = 11, data4 = 36, data5 = 34, data6 = 32, data7 = 30;
// now instantiate (create an instance of) a LiquidCrystal object.
LiquidCrystal lcd(rs, en, data4, data5, data6, data7);
```

Include relevant libraries needed to interface with the sensor

'Construct' an object from the class that we will be using to interface with the sensor

?

How do I know what the arguments of a specific constructor are?
Arduino IDE has a built-in helper that, once you write the name of a given class instance followed by '(', shows you what the arguments are. Alternatively, Google is always a great help ('LiquidCrystal class reference' should give you a lot of input)

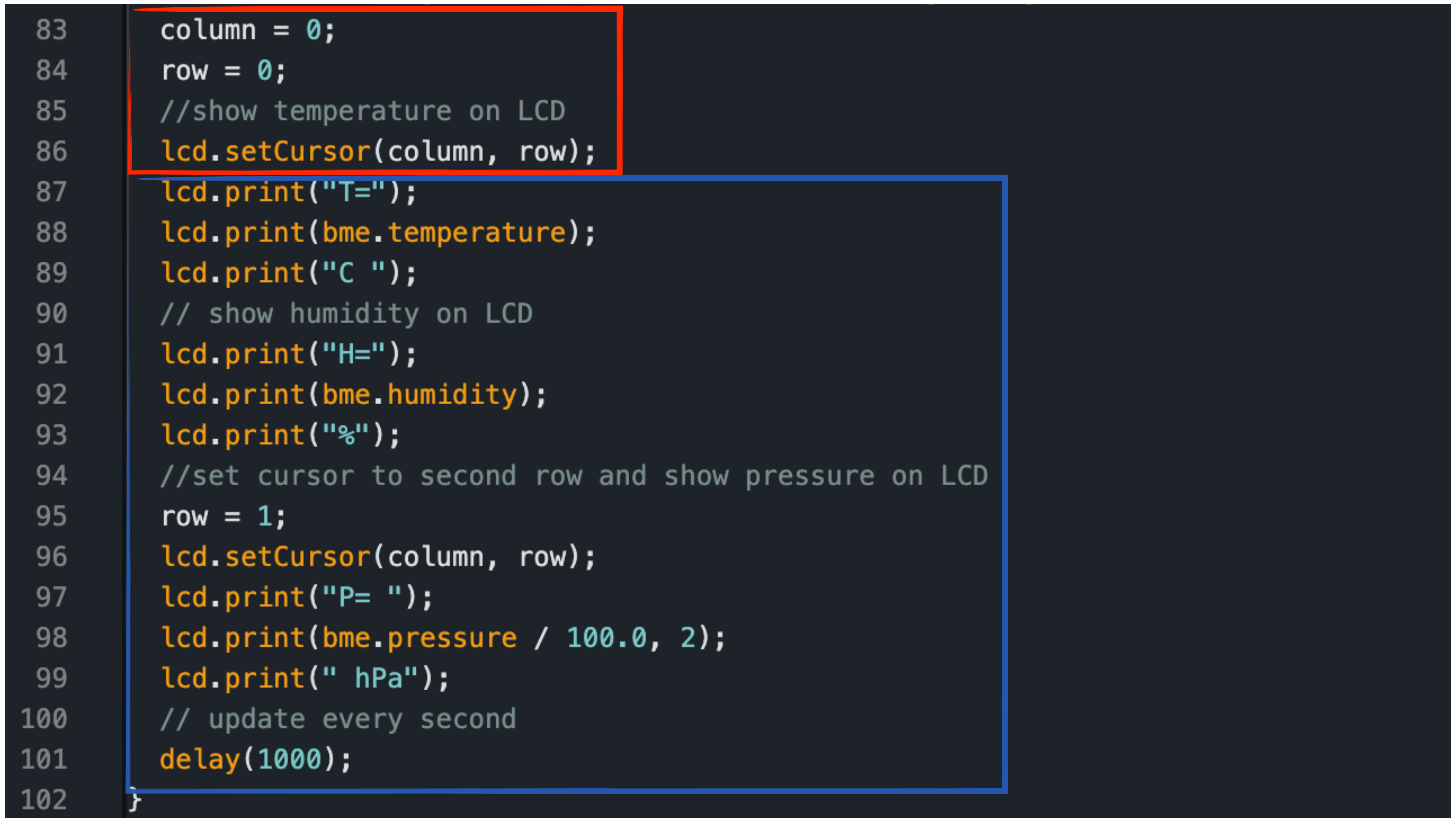


```
void setup()
{
  // fire up the serial port so we can write/read to/from the serial monitor window.
  Serial.begin(9600);

  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
```

Setup the sensor before using it in the loop

I have my circuit… how do I drive it?

Use the sensor in the *loop*() function



```
    column = 0;
    row = 0;
    //show temperature on LCD
    lcd.setCursor(column, row);
    lcd.print("T=");
    lcd.print(bme.temperature);
    lcd.print("C ");
    // show humidity on LCD
    lcd.print("H=");
    lcd.print(bme.humidity);
    lcd.print("%");
    //set cursor to second row and show pressure on LCD
    row = 1;
    lcd.setCursor(column, row);
    lcd.print("P= ");
    lcd.print(bme.pressure / 100.0, 2);
    lcd.print(" hPa");
    // update every second
    delay(1000);
}
```

Set the cursor to start at 0,0 (top left in the LCD map)

Start your printing!

?

Where can I learn how to operate the sensor?
This is where a large amount of examples available on the course webpage (and, in general, online) are invaluable. There are a multitude of examples for each sensors that can help you in exploring the capability of your device. To troubleshoot, ask yourself basic questions like "is this code running on my same circuit setup, or do I need to adapt it somehow?"

Not all the material on the course webpage is designed to run for your purposes without modification!
It is part of the learning process to assemble working code.

SHORT TALKS FOR NEXT WEEK

Dr. Riccardo Longo
01/27/2023
Week 2

Short group talks next week

- TCA9548A I2C Multiplexer
- P1890 Mini Metal Speaker
- W12934-A MicroSD card breakout board
- LSM9oS1 9 DOF
- Mic Amp MAX 4466
- Mini 2-wire Volt Meter
- **MCP4725 DAC** -> up for next week. Volunteers?
- ~~**BME680**~~ —> done by group 5 during Week 2
- Mono 2.5W Amp PAM8302A
- **Ultimate GPS** -> up for next week. Volunteers?
- MLX90614 3V
- ~~**DS3231 Precision RTC**~~ —> done by group 3 during Week 2
- DPS310 Pressure

TODAY'S HANDS ON SESSION

Dr. Riccardo Longo
01/27/2023
Week 2

Project-wise goals

- Once you will be working on your breadboards (tasks described in the next slide), we will stop by each group to hear an "informative report" about the status of your project
  - We will have informative reports from each group during class in each week from now on - to check-in with you on the work we discussed during the mid-week meeting w/ me and the TAs
- By the end of the class today we need to have from you:
  - A sketch of your project plan, including task sharing
    - Remember that your circuit should be fully functional on a breadboard by the time of your presentation on Week 5 (at the latest).
  - A first list of hardware components needed for your project
    - What (and how many) sensors and boards do you need?
    - Do you need access to 3D printing resources?
    - Do you need other hardware to establish your setup?
- Answer to these questions should be e-mailed to me and the TAs by the end of the class, after the informal report discussion

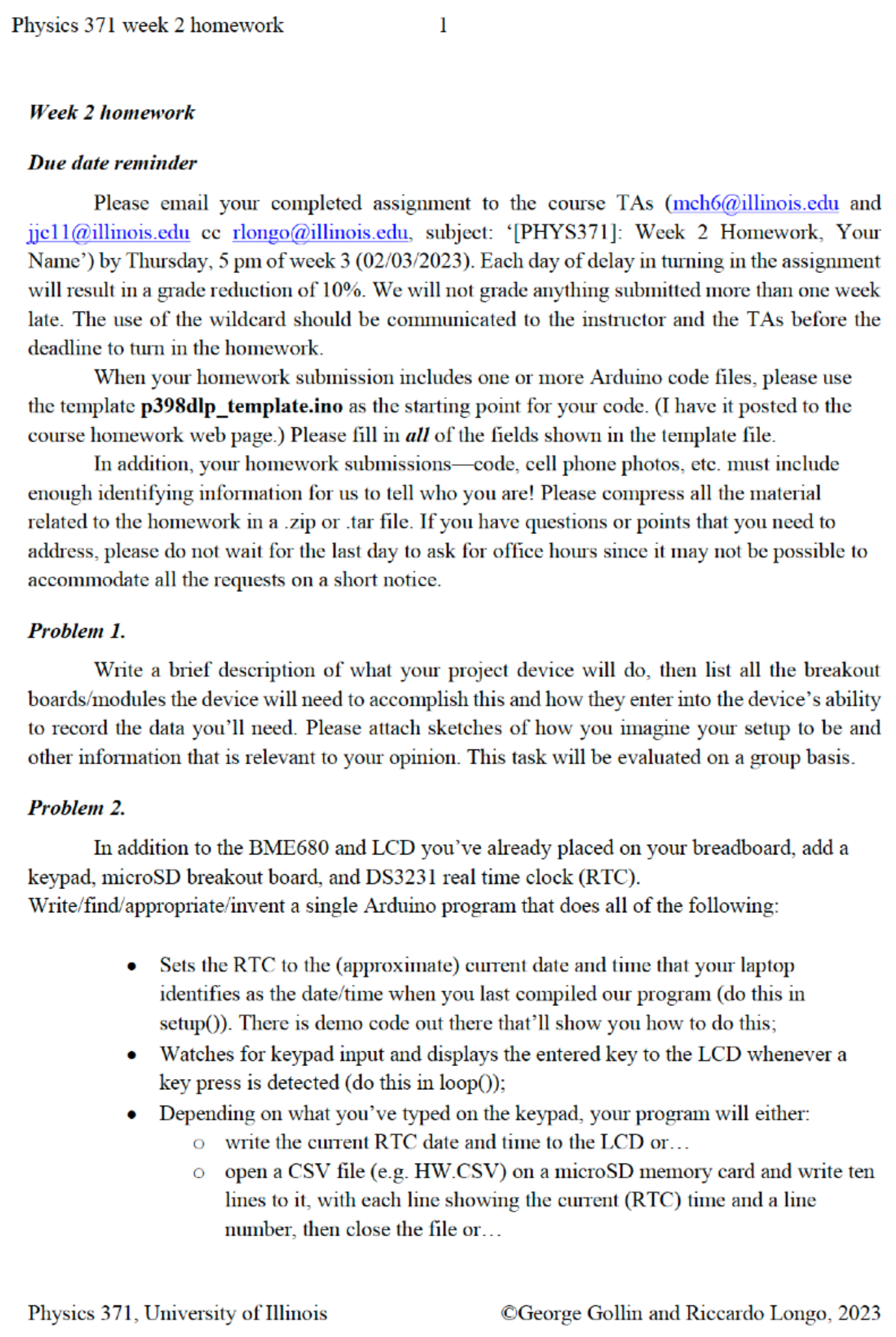
Milestone 1d... 27 to final grade!

Time for new sensors

- Once you have turned in your Homework, it's time to move forward and add some more sensors to your breadboard
- In your kit box, you can find:
  - A **DS3231 Precision RTC** (we just learned about it from Group 3!)
  - A **Keypad**
  - A **W12934-A MicroSD card breakout board**
- Let's start adding them to the circuit, and write a code that carries out the following goals:
  - Set the RTC to the (approximate) current date and time that your laptop identifies as the date/time when you last compiled our program (do this in the *setup*()).
  - Watch for keypad input and displays the entered key to the LCD whenever a key press is detected (do this in the *loop*());
  - Depending on what you've typed on the keypad, your program will either:
    - write the current RTC date and time to the LCD or…
    - open a CSV file (e.g. HW.CSV) on a microSD memory card and write ten lines to it, with each line showing the current (RTC) time and a line number, then close the file or…
    - write a message to the LCD, then fall into an infinite loop so that nothing else of interest happens

Milestone 1c… 26 to final grade!

Homework assignment



Physics 371 week 2 homework 1

Week 2 homework

Due date reminder

Please email your completed assignment to the course TAs (mch6@illinois.edu and jjc11@illinois.edu cc rlongo@illinois.edu, subject: '[PHYS371]: Week 2 Homework, Your Name') by Thursday, 5 pm of week 3 (02/03/2023). Each day of delay in turning in the assignment will result in a grade reduction of 10%. We will not grade anything submitted more than one week late. The use of the wildcard should be communicated to the instructor and the TAs before the deadline to turn in the homework.

When your homework submission includes one or more Arduino code files, please use the template **p398dlp_template.ino** as the starting point for your code. (I have it posted to the course homework web page.) Please fill in ***all*** of the fields shown in the template file.

In addition, your homework submissions—code, cell phone photos, etc. must include enough identifying information for us to tell who you are! Please compress all the material related to the homework in a .zip or .tar file. If you have questions or points that you need to address, please do not wait for the last day to ask for office hours since it may not be possible to accommodate all the requests on a short notice.

Problem 1.

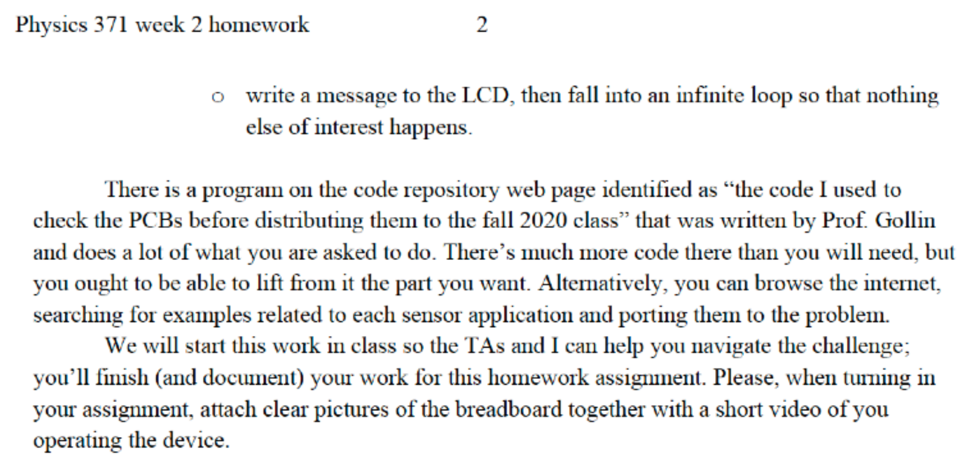
Write a brief description of what your project device will do, then list all the breakout boards/modules the device will need to accomplish this and how they enter into the device's ability to record the data you'll need. Please attach sketches of how you imagine your setup to be and other information that is relevant to your opinion. This task will be evaluated on a group basis.

Problem 2.

In addition to the BME680 and LCD you've already placed on your breadboard, add a keypad, microSD breakout board, and DS3231 real time clock (RTC).
Write/find/appropriate/invent a single Arduino program that does all of the following:

- Sets the RTC to the (approximate) current date and time that your laptop identifies as the date/time when you last compiled our program (do this in setup()). There is demo code out there that'll show you how to do this;
- Watches for keypad input and displays the entered key to the LCD whenever a key press is detected (do this in loop());
- Depending on what you've typed on the keypad, your program will either:
  - write the current RTC date and time to the LCD or…
  - open a CSV file (e.g. HW.CSV) on a microSD memory card and write ten lines to it, with each line showing the current (RTC) time and a line number, then close the file or…

Physics 371, University of Illinois ©George Gollin and Riccardo Longo, 2023



Physics 371 week 2 homework 2

  - write a message to the LCD, then fall into an infinite loop so that nothing else of interest happens.

There is a program on the code repository web page identified as "the code I used to check the PCBs before distributing them to the fall 2020 class" that was written by Prof. Gollin and does a lot of what you are asked to do. There's much more code there than you will need, but you ought to be able to lift from it the part you want. Alternatively, you can browse the internet, searching for examples related to each sensor application and porting them to the problem.

We will start this work in class so the TAs and I can help you navigate the challenge; you'll finish (and document) your work for this homework assignment. Please, when turning in your assignment, attach clear pictures of the breadboard together with a short video of you operating the device.

https://courses.physics.illinois.edu/phys371/sp2023/homeworks.asp

Questions?