**Arduino Exercises**

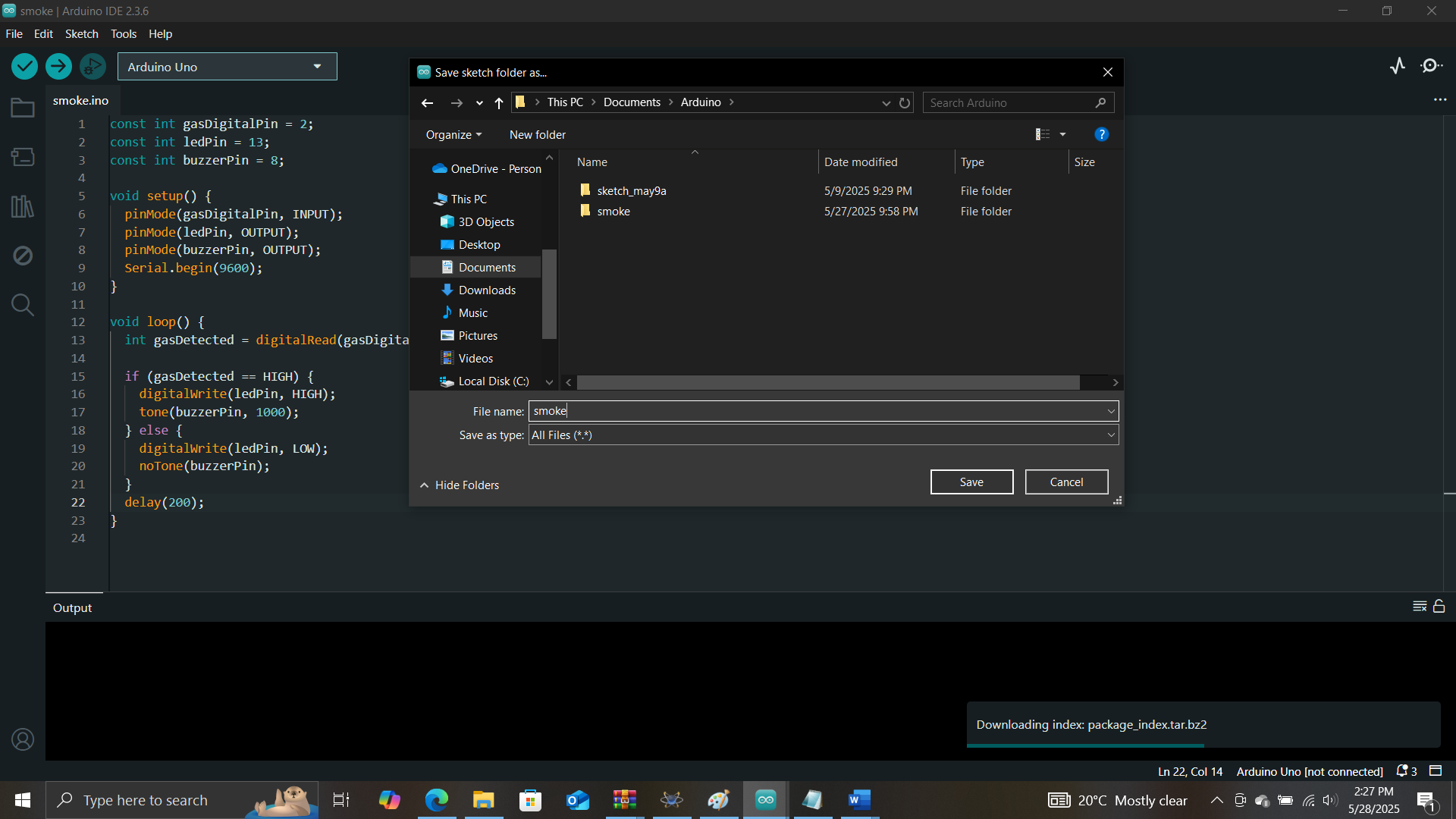
* For the exercises in this document, the images to be drawn on proteus are found in the parent folder as well as the code.
* All the sensors used in the exercises produce digital output that’s why they are connected to the digital pins on the right and not the analogue pins on the left (A pins)

**Instructions**

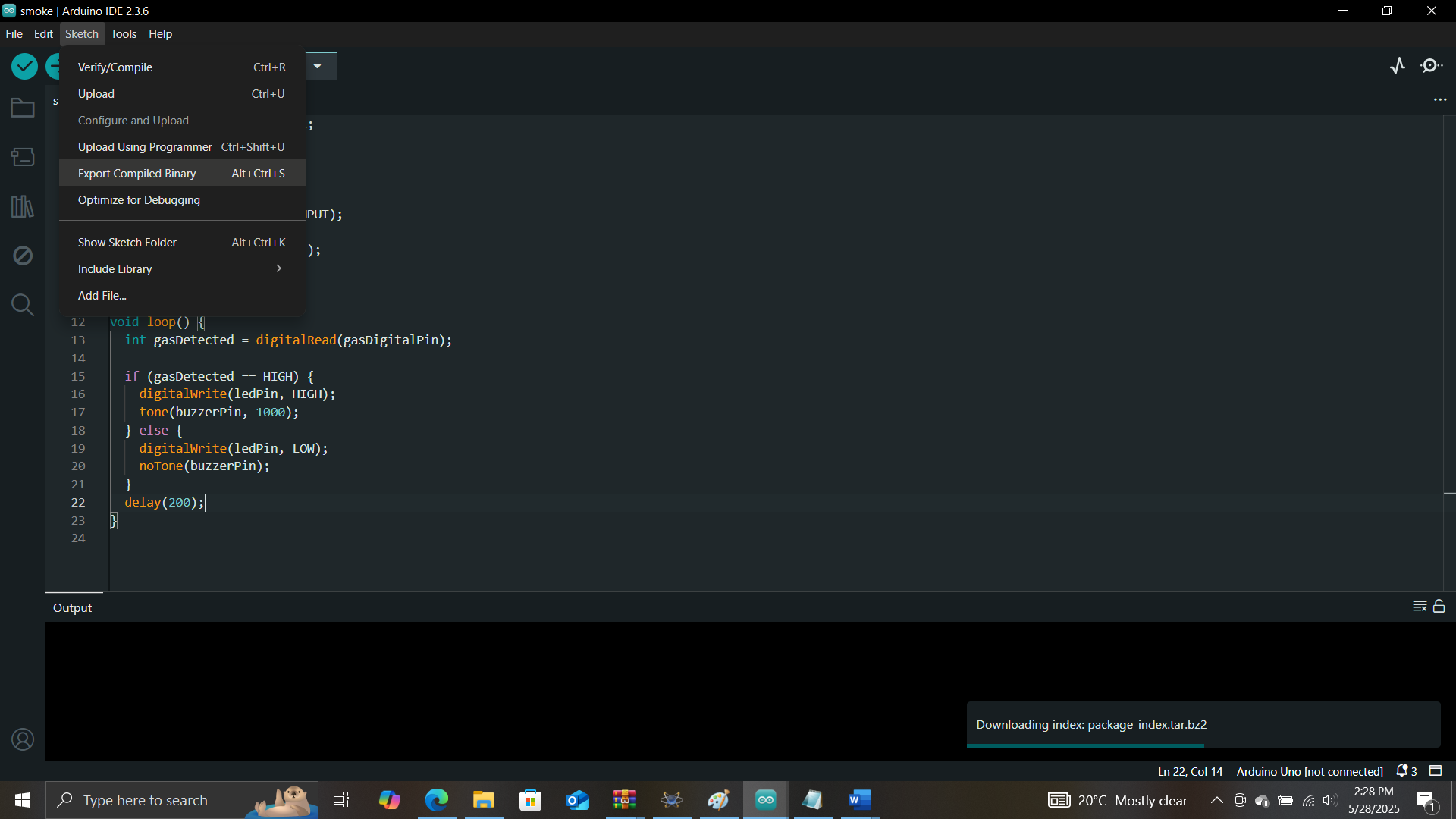
* Copy and paste all the files with .lib extension in the Arduino folder to drive C:/program files(x86)/Labcenter electronics/proteus/data/library
* If your proteus is 64bit, copy to C:/program files/Labcenter electronics/proteus/data/library
* By doing this, proteus will have access to Arduino and the sensors

**How to run**

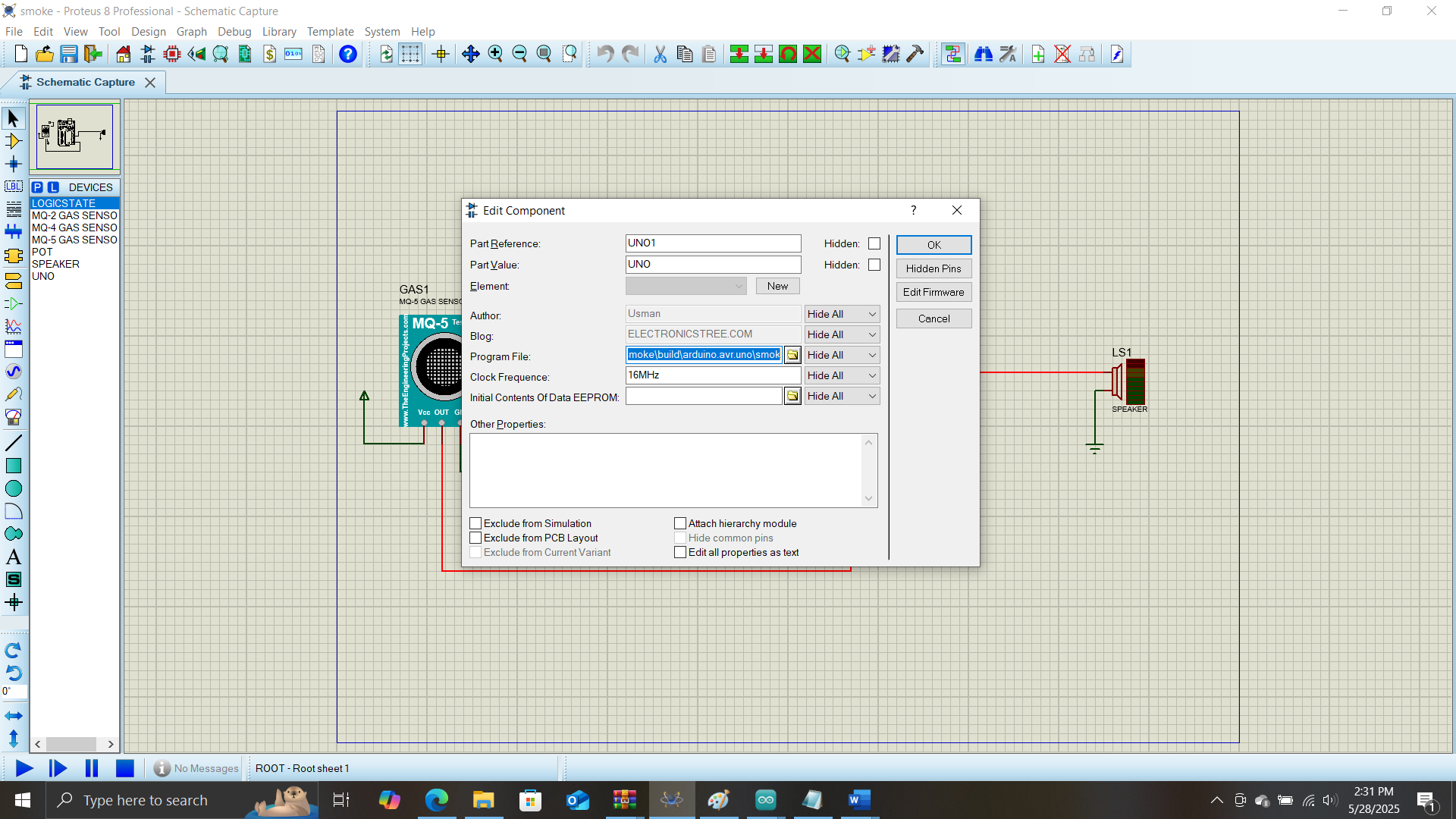
* The code for each exercise is found in the Arduino folder as text files. Open the text files, copy the code and paste in your Arduino ide.
* Click on file then Save. Choose a directory to save the file. In my case its under documents.



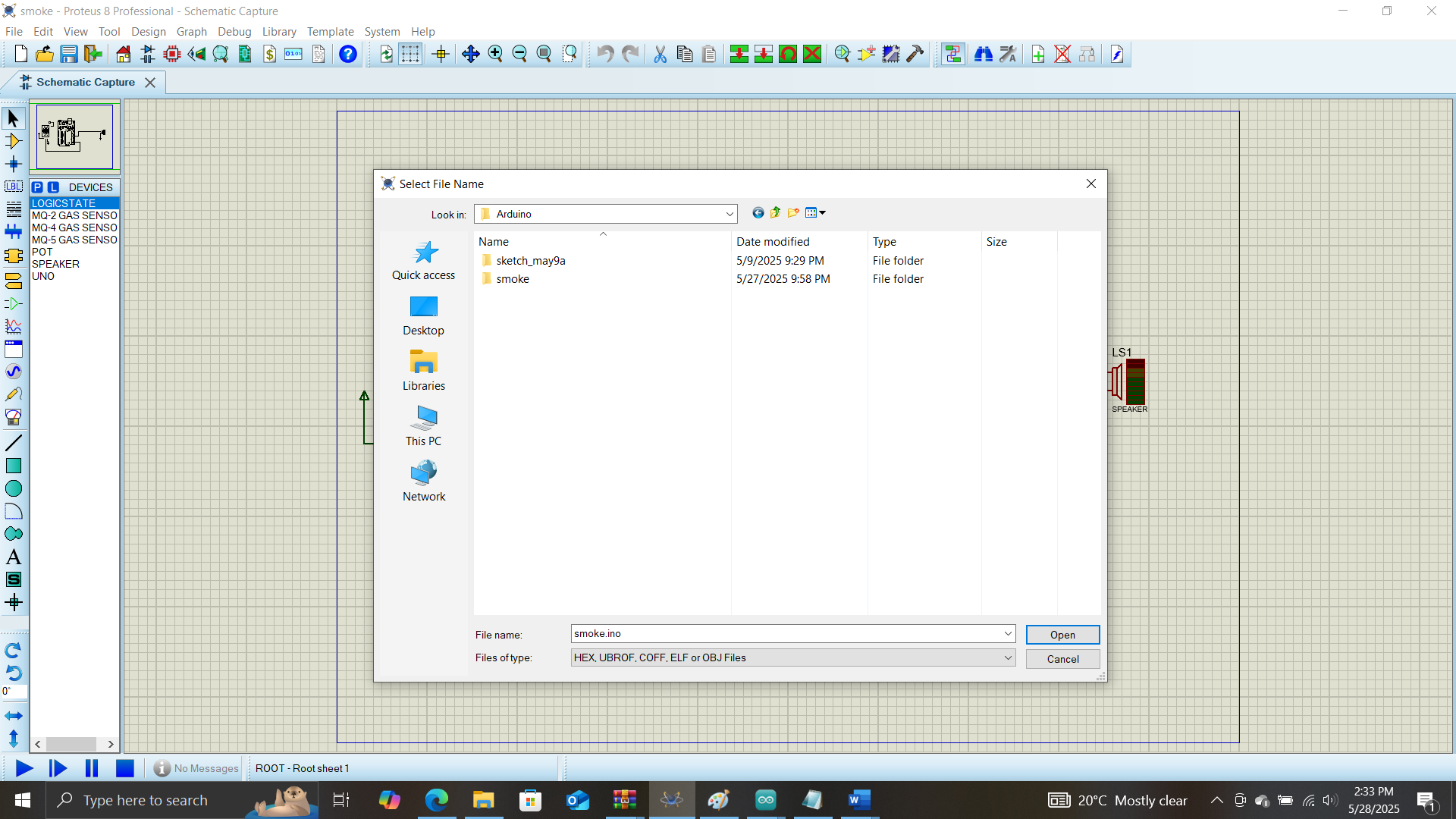
* Then click on ‘Sketch’ you will see ‘Export Compile Binary’. Click on it to generate a hex file.



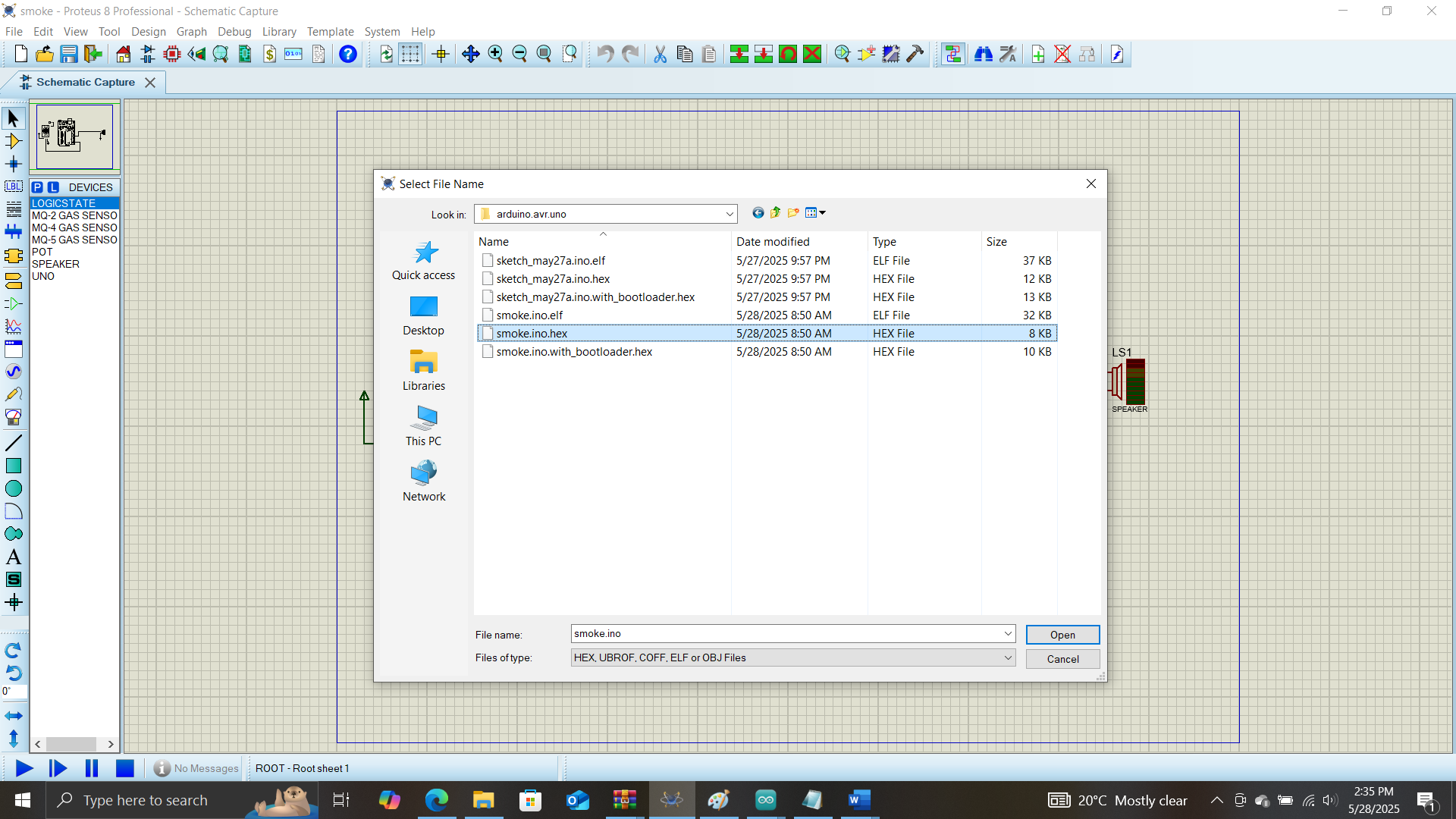
* Go to proteus and draw the circuit if you haven’t
* If you have, double click on the Arduino and try to add a hex file.



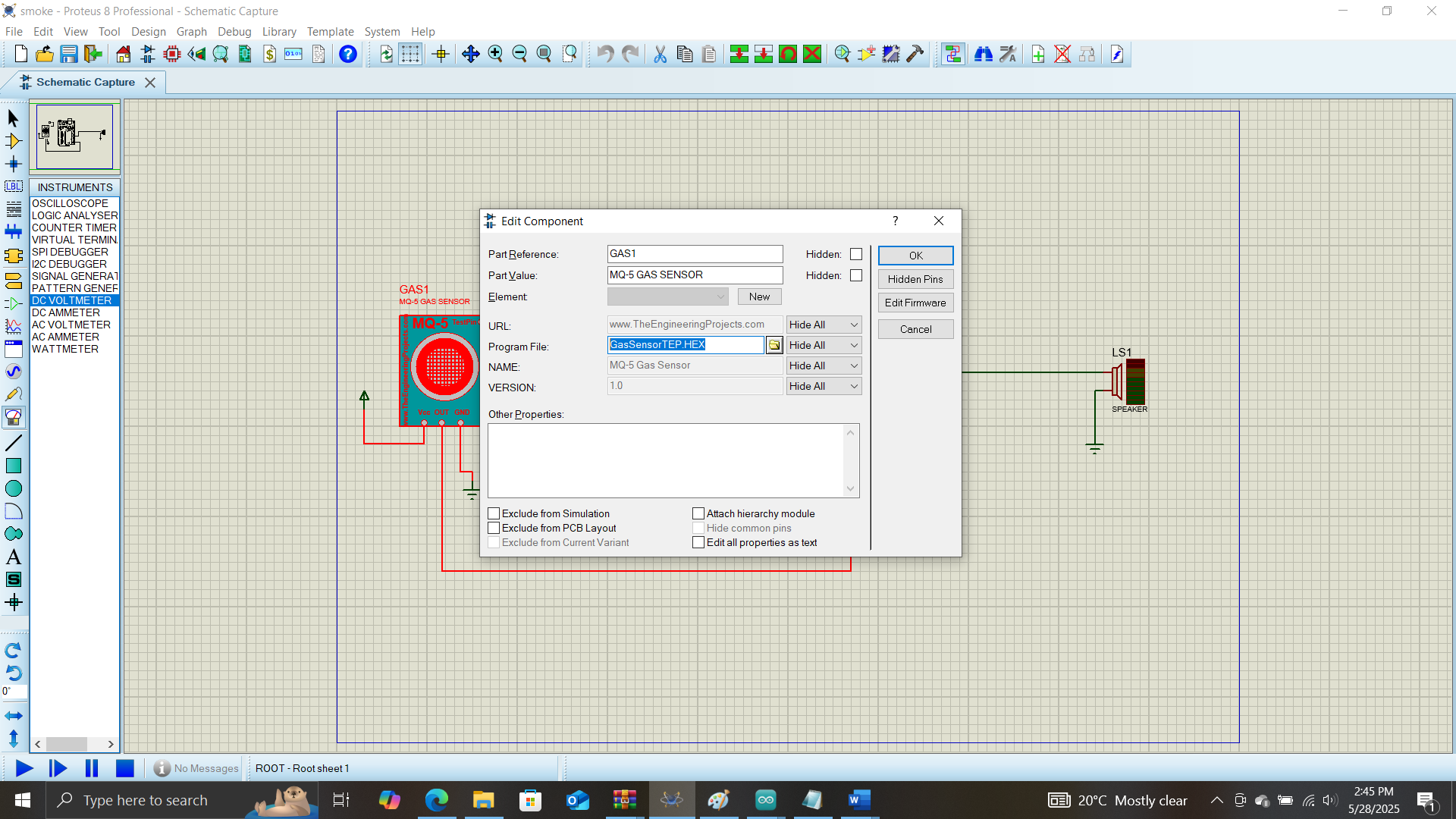
* Navigate to where you saved your code. In my case it as documents



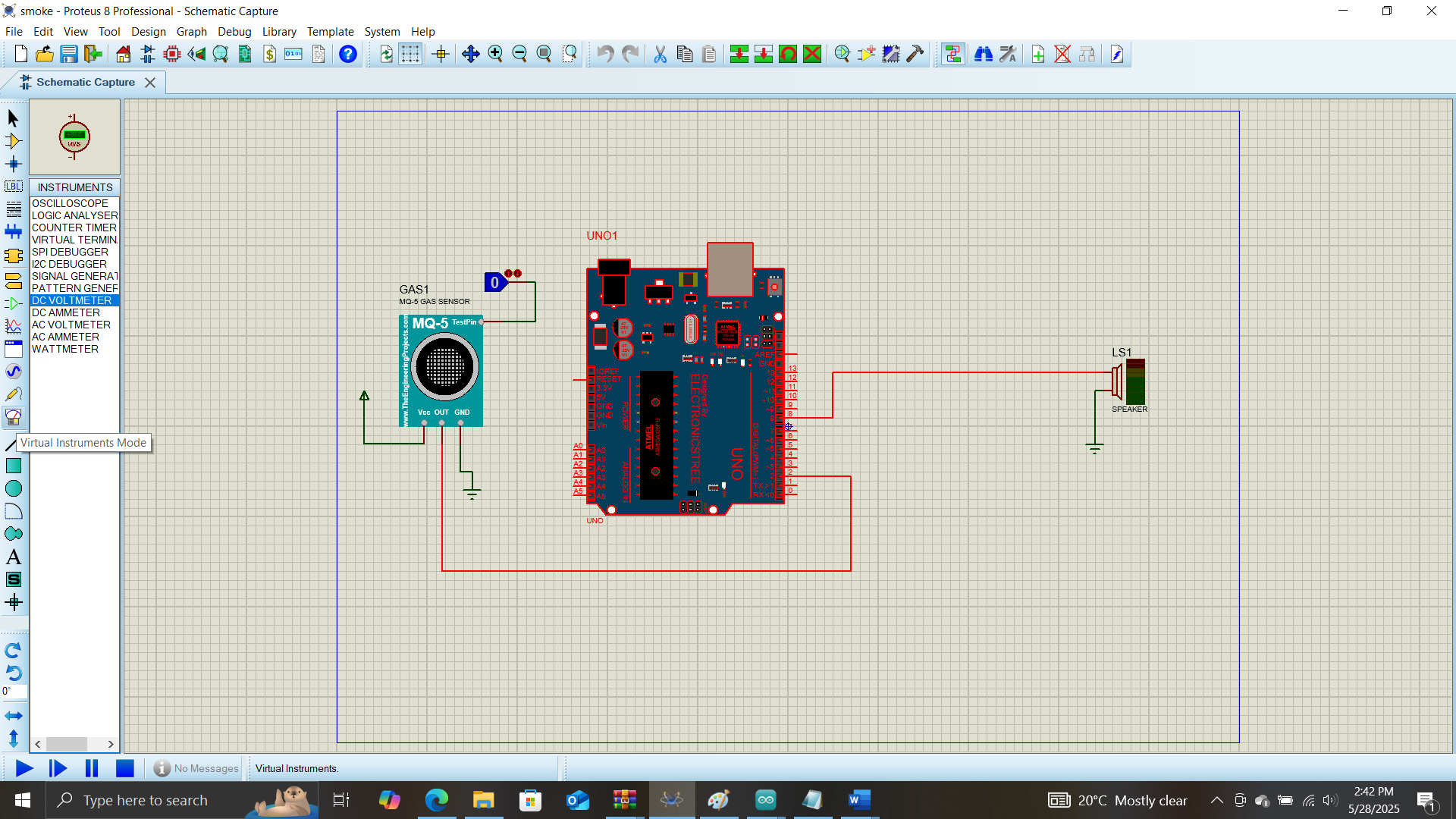
* I saved it as ‘smoke’. So I will double click on ‘smoke’ then to build, arduino.avr.uno, smoke.ino.hex as on the image below



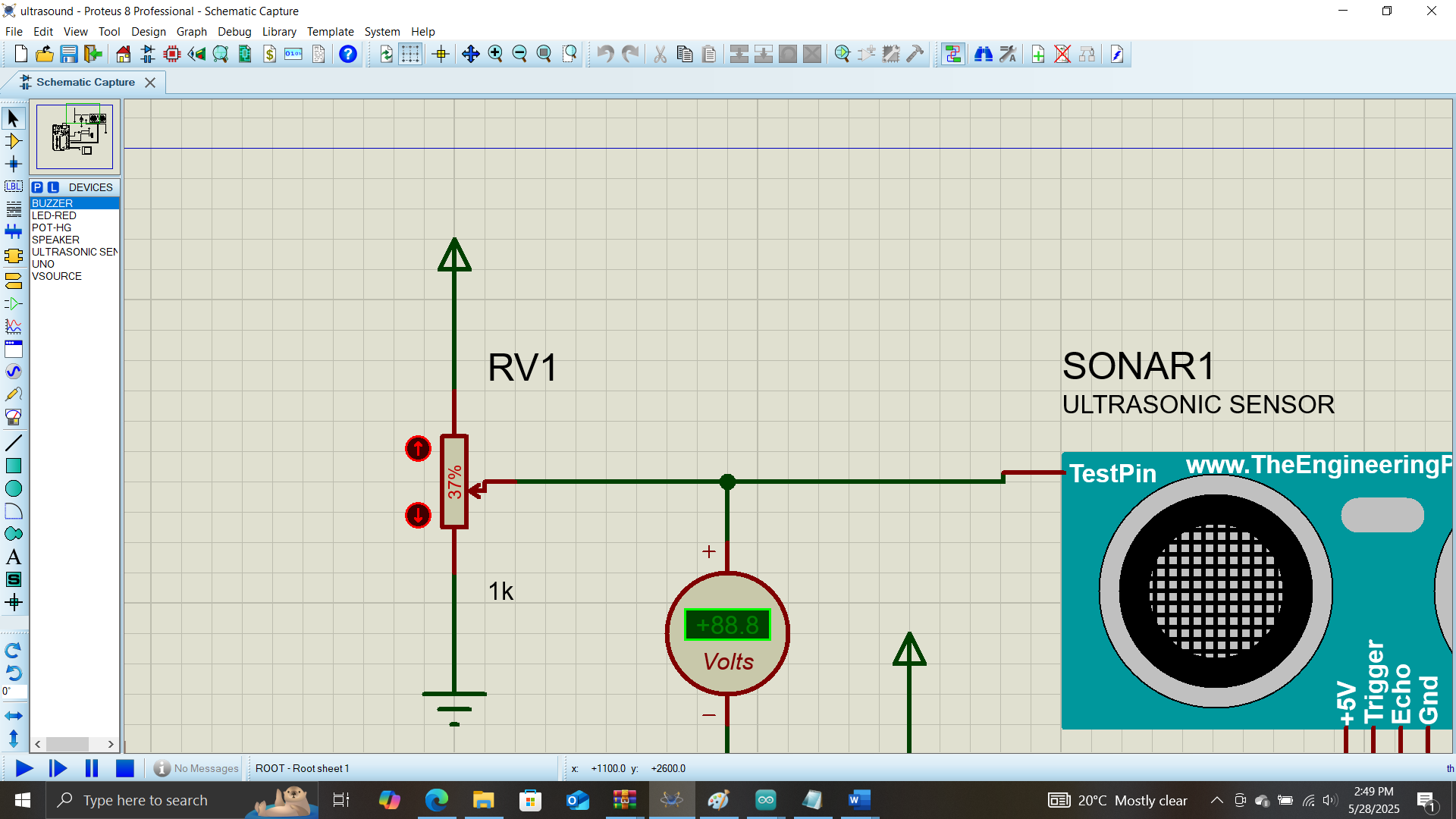
* Then click on ‘open’ to import the hex file into the arduino
* For the sensors on the other hand, in the folder of this document you will see hex files for both the ultrasound and gas sensor.
* For each sensor, double click on it and import its respective hex files. If not the sensors wont work.



* For the ultrasound exercise, the virtual terminal device and dc voltmeter can be gotten in proteus from ‘virtual instruments mode’ below the ‘injection sign’



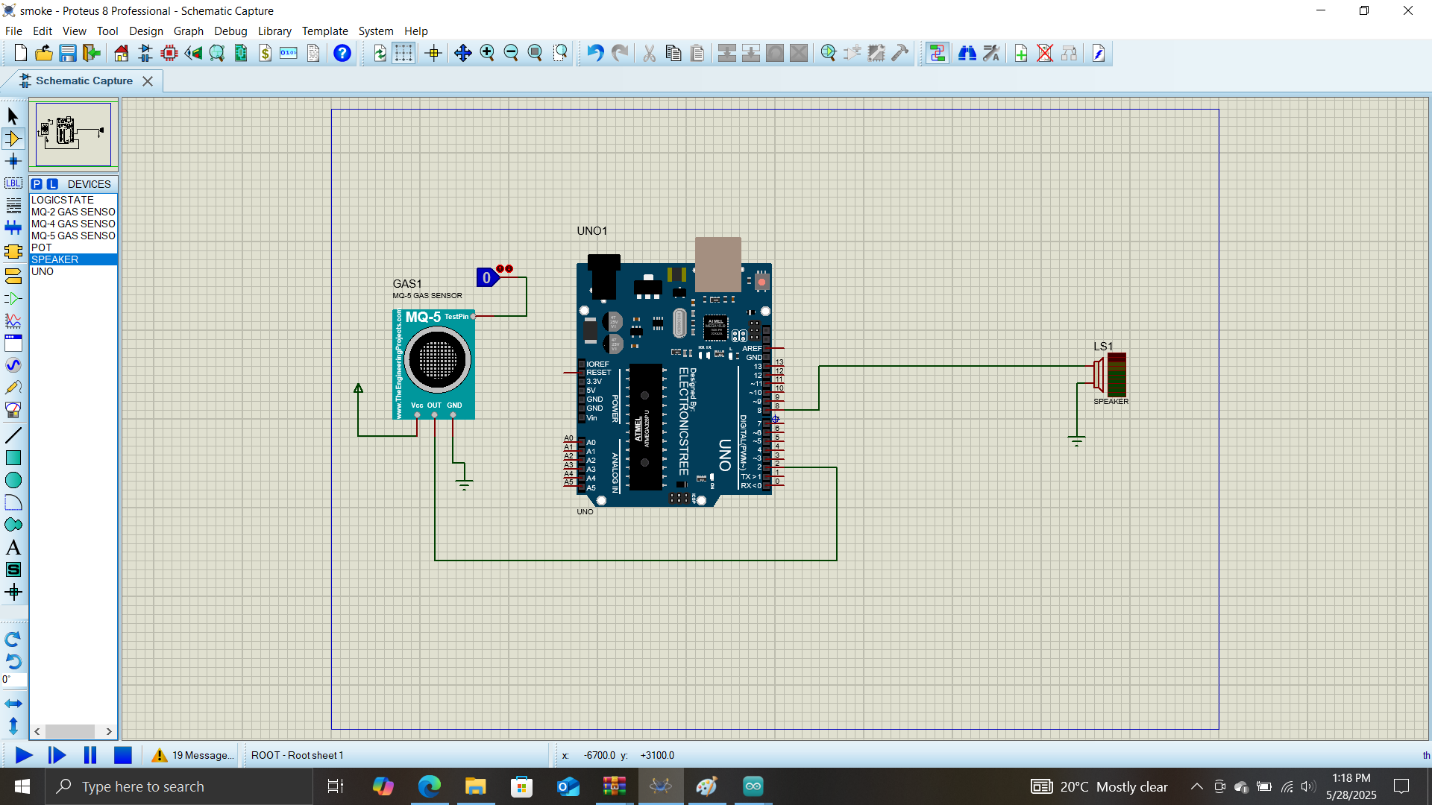
* When you are done setting up everything, you run the simulation
* For the ultrasound exercise, after simulating you can increase or reduce the distance of the imaginary object by clicking on the 2 circles of the ‘POT-HG’ device below.



**Exercises**

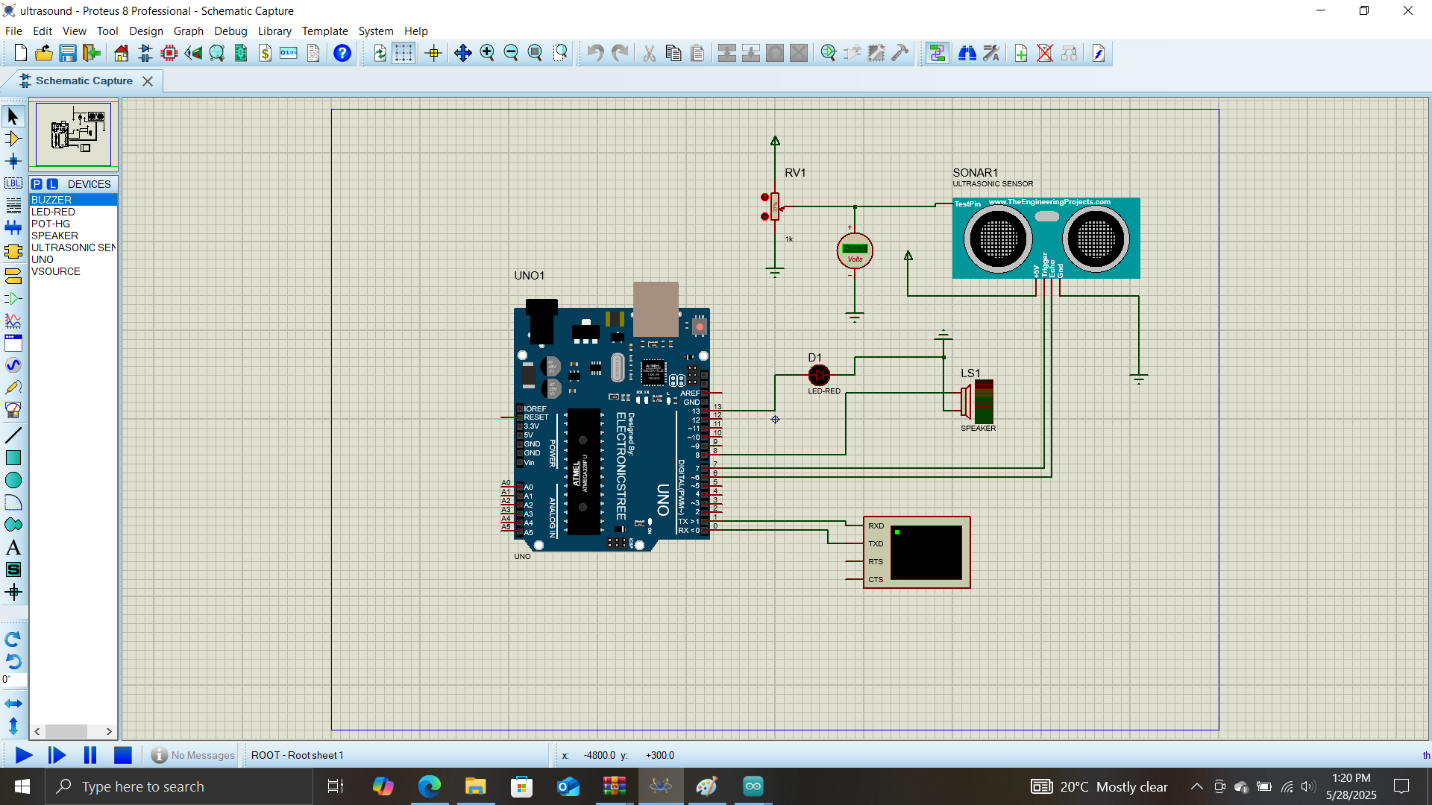
1. The purpose of this first exercise for security like securing a hospital from fire outbreak by making use of a smoke sensor.

* The goal is that when the smoke sensor detects smoke (by changing the logic state from 0 to 1), current is going to flow to pin 2 of the Arduino and our code will force the speaker (buzzer) connected to pin 8 to play a sound
* The code for this is in the parent folder



1. In this exercise, we are using the ultrasound sensor as a motion sensor to be able to detect in an object is in range or not. If the object is in range, we turn on an LED and play a sound. This can be used in home security so that when someone gets into your house uninvited, your security lights can automatically come on and an alarm rings to notify you of a stranger.

* The goal here is if someone gets within a 4m (400cm) range of the sensor the led at pin 13 should come on while the speaker at pin 8 should play a sound
* For the ultrasound sensor, the trig pin is used to shoot out sound wave and the echo pin Is used to receive the sound wave from the trig pin that has bounced on an object (like a human) then reflected back to the sensor



1. Write code below to turn on the led on pin 13

