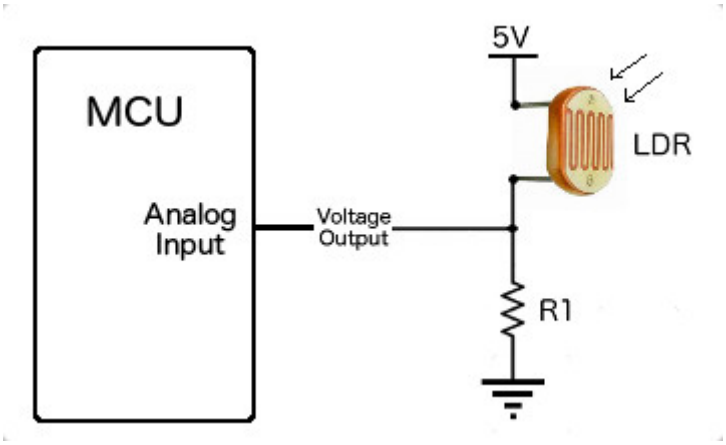
Multidimensional Sensing Techniques
Assignment 3 – Light Sensors
You can collect up to 19 points in this assignment
Deadline: 15.10.2021 at 23:55

The aim of this lab is to evaluate the precision of a light sensor and develop a simple low-level data fusion application across sensors (in a competitive configuration with 4 identical sensors). Additionally, you will need to implement a way to detect faulty sensor(s), so that the output of the fusion algorithm will take into account only the data from working sensors.

This assignment will be done using the (free) online modeling platform www.tinkercad.com.

An overview of the electric schematic of the sensing circuit is the following:



With R1= 1KΩ +/- 5%

In practice putting the LDR in darkness will increase the resistor value of the LDR.. When exposed to very intensive sun light the value of the LDR resistor will decrease. You can assume a linear calibration curve for the LDR resistor.

You will connect each sensing circuit to an analog input pin of the Arduino UNO. When reading the value from one analog input pin, you can use the analogRead() command. It converts the input voltage range, 0 to 5 volts, to a digital value between 0 and 1023. This is done by a circuit inside the microcontroller called an analog-to-digital converter or ADC.

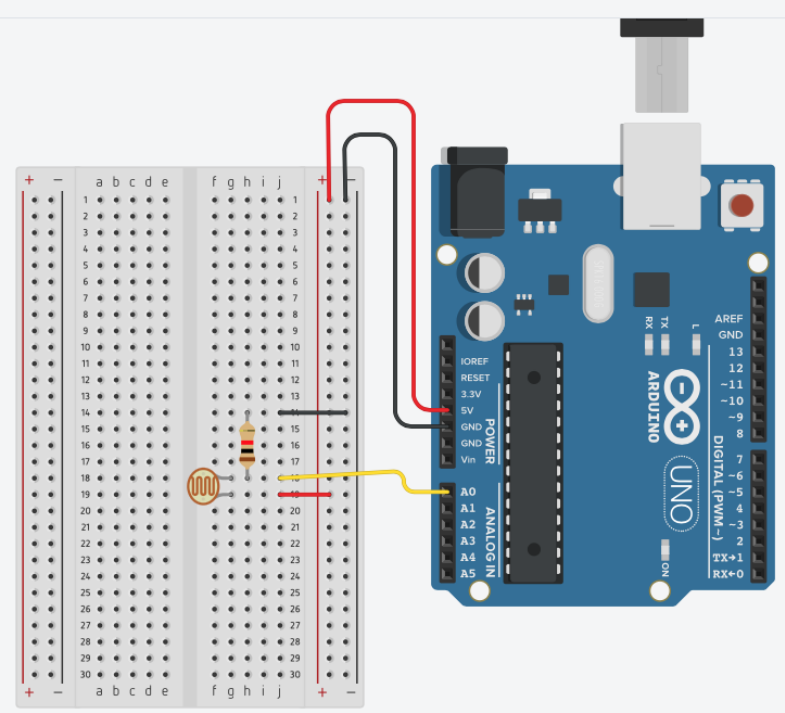
Tasks to be performed

1 (0 point) Create an account on www.tinkercad.com.

2 (0 point) Create a new circuit design (on the left menu, click on ‘Circuit’, then click ‘Create a new Circuit’)

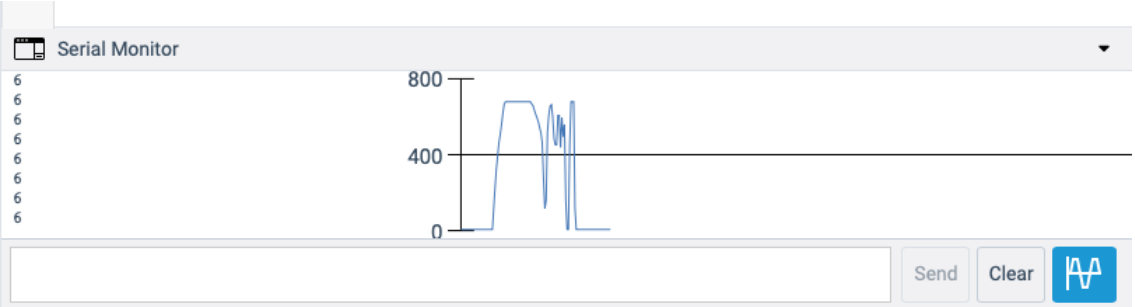
3 (1 points) Connect all sensors to your Arduino correctly (**double check your wiring that there are no short-circuits**)

Here is how the connection of one photoresistor sensor looks like (you need to extend this for 3 other sensors by connecting additional sensors to analog input pins [A1-A3]). Set the resistor values to 1kΩ.



4 (2 points) Write an Arduino program (called sketch, uses C/C++) that reads the analog values from the 4 sensors. Refer to the following page: https://maker.pro/arduino/tutorial/how-to-use-an-ldr-sensor-with-arduino and extend the code to read values from more than one sensor. You should use the “Text” code editor only. You can copy&past the content of the provided *Assignment3.ino* sketch as a starting point.

You can access the serial monitor by clicking on “Serial Monitor” bellow the source code windows:



5 (4 points) Calculate the average value and the variance of the readings obtained from the 4 sensors as they come. When running the simulation, you can dynamically change the amount of light the photoresistors get with the sliding bar next to each sensor.

Note: On the Arduino Uno (and other ATmega based boards) an int stores a 16-bit (2-byte) value. This yields a range of -32,768 to 32,767 for int variables. Long variables are extended size variables for number storage, and store 32 bits (4 bytes), from -2,147,483,648 to 2,147,483,647.

6 (4 points) Calculate the average value and the variance of each sensor separately over 100 readings (as a sliding window if possible)

7 (4 points) Based on the obtained average values and variances, write a function to detect when a sensor produces erroneous readings. Test this by having a large light exposure difference on one sensor compared to the 3 others. Write a function to detect which sensor produces erroneous readings (here we assume only one sensor can fail).

8 (4 points) Based on the sensing circuit and provided values for R1 (1KΩ) and the measured minimum and maximum values by the function *analogRead():*

- What is the minimal voltage value applied to the analog inputs of the Arduino?
- What is the maximal voltage value applied to the analog inputs of the Arduino?
- What is the minimal resistance value of the photoresistor?
- What is the maximal resistance value of the photoresistor?

Tips: remember the Ohm's law and check this page to know how to calculate voltage drops on resistors in series.

2. Reporting instructions

GitHub Classroom repository

You should upload all the implemented and used code for this lab on a GitHub Classroom repository. You can export your code in a .ino file, which is the file extension used for a sketch.

You should create your group repository from the following link:
https://classroom.github.com/a/uw1ciUUD

Report

The report document should contain:

- Your full name and student number at ÅA
- A direct link to your source code on GitHub.
- Each page of your submission should have a page number
- **Pay attention to the readability of your report!**

You should write a report documenting the work performed during the exercise. The report should contain information on:

- What you did
- How you did it
- Why you did it this way

- The results of your work
- How your solution could be improved/generalized

You are encouraged to provide pictures, graphs, screenshots, diagrams etc. Anything that help understanding your solutions should be included in your report. If you use content (pictures, graphs, etc.) from other sources, remember to properly cite and provide a reference to the used external source(s).
At the end of the report you should also provide a reflection on what you learned during this exercise. This section could provide answers to the following questions:

- What did you learn?
- Did anything surprise you?
- Did you find anything challenging? Why?
- Did you find anything satisfying? Why?

The expected size for this report is 3-5 pages of content.
Name you report file Assignment3_YourName.pdf and upload your report in PDF format on moodle before the deadline.