Note :

There are 2 possible examination assignment to choose between.

1. The Arduino data logger.
2. The One-Arm Arduino Bandit.

You are only asked to do one of them.

Your solution MUST show that you master the topics in C Programming mentioned through the course.

Your solution must consist of your code and a 1–2 A4 page description of :

- Which problem you solve
- Who you did it together with
- What you have planned to do
- What you ended up doing

And it must be uploaded on Wiseflow before June, 4th, 2020.

The Data logger with the Arduino

This exercise build upon your 2nd mandatory exercise.

Now it is about communication and the protocol you are using between the two computers.

You have to control the Arduino from your other computer (The PC)

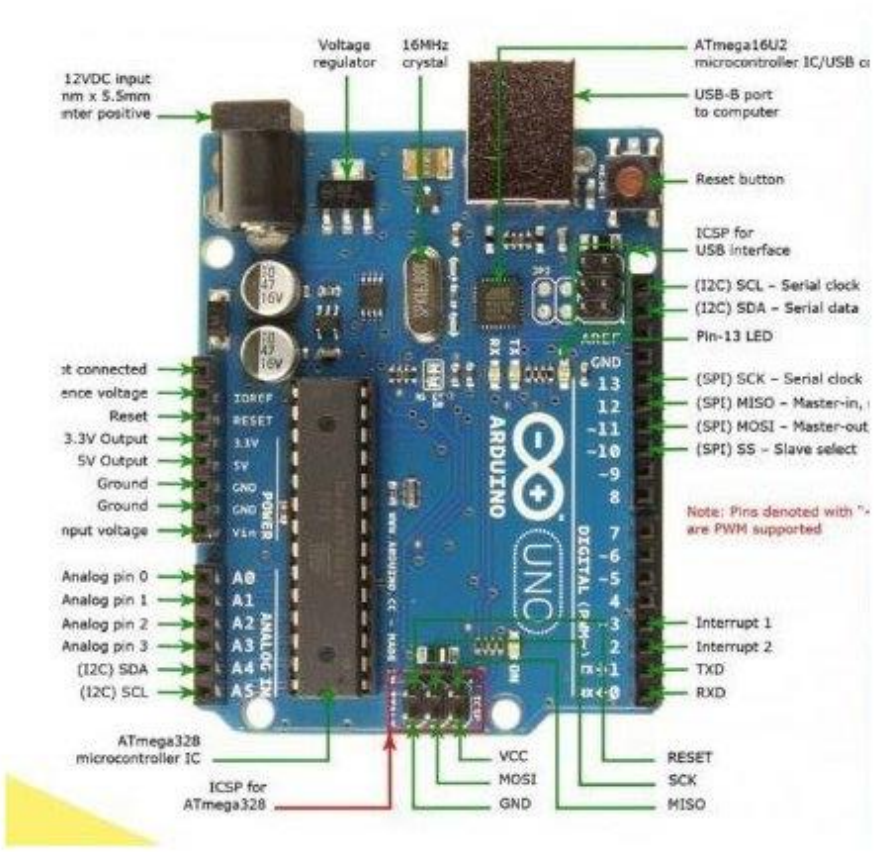
1. Start the logging process.
2. Stop the logging process.
3. Transmit all the logged data.
4. You can change between different sensors through a menu.
5. If you are simulating data, how many, and what numbers should it be between.

The program on the Arduino has to collect some data from a kind of a sensor, it could be light, sound, distance, speed or any simulated data. And on a request from the PC transmit the data to the PC. The program is now totally under control from the PC.

Remember to include the facilities C code offer, and of which we have talked about through the semester. This is an examination in embedded C code, not in any other programming language, all though you are welcome to have some assistance on the PC from another language.

The One-Arm Arduino Bandit

This exercise build upon your 2nd mandatory exercise, i.e. every fact mentioned there are still valid.



Your code must be running on your Arduino computer, and …..

1)

The results must at least be shown on the serial monitor.

If you have a display, it can be shown there.

2)

Make it possible through a menu to either run the bandit manually or do it a number of times decided by the user.

3)

The wheels must run with a fixed speed.

Beginning with the right one they must stop one by one due to a running interrupt AND a random number, which decides the length of the interrupt process.

Place the code around the interrupt in its own set of .c and .h files.

4)

Save the result in a structure and use an array of this structures to calculate the statistics of a given (high) number of runs.

Try to construct a payout scheme, so you as owner of the bandit earns around 10%? What is your best attempt ?

5)

Table 1 Binary codes for displayed values

Code	Name	Binary	Decimal
A	Apple	000	0
B	Bell	001	1
C	Cherry	010	2
G	Grape	011	3
L	Lemon	100	4
O	Orange	101	5
P	Plum	110	6
BAR	BAR	111	7

Your Arduino Bandit is now entering the digital age. Each of the possible values in the display is now represented by 3 bits as shown in Tabel 1.

So the end display is 9 bits, 3 for each wheel.

Recode your bandit so the winning comparison is done bitwise.

Can your answer to 4) be done more efficient ?