1 Laboratory 1

Welcome to the first laboratory. Goal of this lab is to get you know with basics of Arduino and its basic concepts.

2 Theory

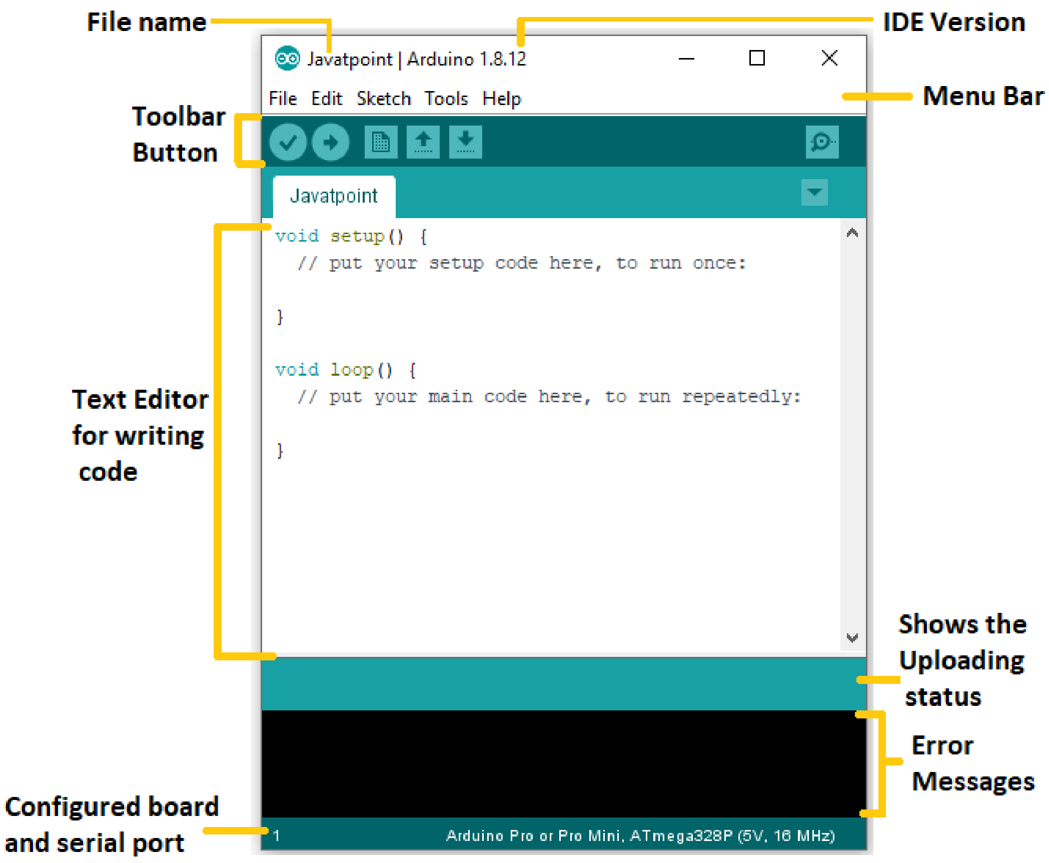
2.1 Arduino overview

Arduino in its essence is programmable microprocessor without operating system. It is great tool for staring in robotic because it is easy to use and official documentation is user-friendly. In this laboratory you will be using Arduino MEGA

TODO jemne opisat arduino GPIO PINS a ked tak graf z doku

2.2 Programming Arduino

Arduino is programmed using its own programming language which is subset of c/c++. For programming you will be using Arduino IDE.



```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure 1: Arduino IDE (TODO spravit vlastny image)

The code structure has 2 parts - void setup() and void loop()

void setup() code in this function will be run only once and its used for initializing variables, setting modes for pins, initializing libraries, etc.

void loop() code in this function will be run repeatedly. Here is where you will write your main code - like reading data from sensors, setting pins output values or controlling motors.

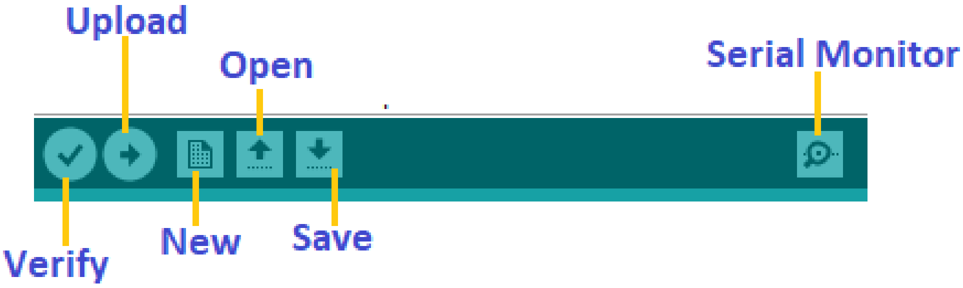


Figure 2: Arduino IDE toolbar (TODO spravit vlastny image)

For uploading your code you will be using

verify compiles and links your code but does not upload it.

upload compiles and uploads your code to the Arduino board

Serial monitor used for communication with Aruduino. Useful for reading data from arduino or sending it custom messages.

2.3 Prototyping on breadboard

When working with Arduino you want to have capability of making of making electronics circuits. The most convinient way is to use breadboard. Its main advatage is that it does not need soldering on order to make electrical connection.

2.3.1 How does this breadboard work?

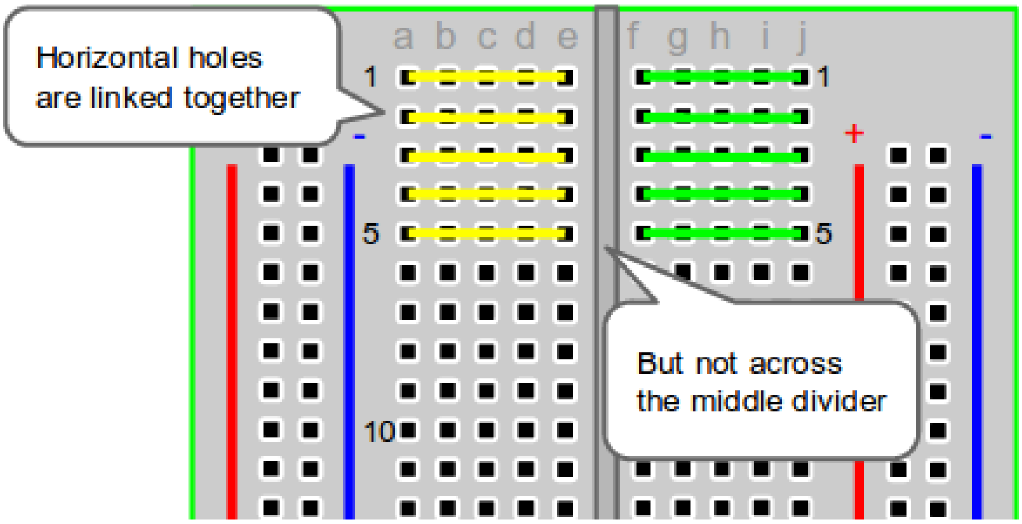

Horizontal holes are linked together
a b c d e
f g h i j
1
5
10
1
5
+
-
-
But not across the middle divider

Figure 3: Breadboard (TODO spravit vlastny image)

Breadboards usually have horizontally and vertically connected holes. Horizontally connected are in the middle and vertically connected holes are on sides. Vertical ones usually function as your powerline. As it runs trough whole breadboard you can power and of your electrical parts in main part - which are horizontal holes. This is the place where you assemble your circuit.

TODO mozno vysvetlit ako a kde najst COM (pretoze to celkom dost casto blbne)

TODO mozno opisat co je LED/ rezistor a pod

3 Exercises

IDEAS: spravit cheat sheet kde by boli vsetky funkcie ktore sa postupne beru (ako napr pin mode a tak)

TODO: nazvy suborov, cesty pomocou bold

3.1 Blinking led

Lets make first simple Arduino exercise and check if Arduino works.

1. Open file **Exercise1.ino** with Arduino IDE using **File -> Open...**
2. Upload your code by clicking on Upload button (TODO IMG HERE) or pressing Ctrl+U
3. If you have error during uploading that says An error occurred while uploading the sketch you need to check configuration of your Arduino IDE. Go to Tools -> Board and check if it set to Arduino Mega. After that try different COM port with Tools -> Port. Try every one and see which one works (Port number depends on where you plug your Arduino into PC)
4. LED on Arduino board should be blinking in one second intervals.

3.2 Blinking on breadboard

One blinking led on Arduino is fine, but imagine what to do when you want to blink LED that is not on Arduino board.

1. Open file **Exercise2.ino** with Arduino IDE using **File -> Open...**
2. Take a look at **Scheme1.png** and create circuit as it is shown on image
3. Upload your code by clicking on Upload button (TODO IMG HERE) or pressing Ctrl+U
4. LED on breadboard should be blinking in one second intervals.

Now that LED is blinking experimentate with the duration of blinking. Change the timing as it is described below. What do you observe?

	TIME_ON	TIME_OFF
1.	100	100
2.	30	30
3.	1	1
4.	1	10
5.	1	20

Focus mainly on the last 3 lines. What do you observe? What is the LED doing?

3.3 PWM

What you did in the previous section is called PWM - Pulse Width Modulation TODO OPISAT PWM?

1. Open file **Exercise3.ino** with Arduino IDE using **File -> Open...**
2. Reuse LED circuit from previous exercise
3. Upload your code by clicking on Upload button (TODO IMG HERE) or pressing Ctrl+U
4. LED should be on, but dimmed
5. Try and modify constant **DUTY_CYCLE_PERCENTAGE**s value into 1,2,5,10,100

3.4 Challange 1

Make the LED oscilate between fully dimmed and fully bright using PWM.

1. Open file **Challange1.ino** with Arduino IDE using **File -> Open...**
2. Reuse LED circuit from previous exercise
3. Write your code
4. Upload your code by clicking on Upload button (TODO IMG HERE) or pressing Ctrl+U
5. Bonus: Slow down the brightening/dimming of the LED. For example make it 5 second cycle.

3.5 Challange 1

Make program and code in a way when pressing button led is OFF.

3.6 Getting input into arduino

Controlling stuff with Arduino is only half of the knowledge. Other half is how to know read data from real world.

Make new circuit by given schematic and copy code from FILEX.

TODO maybe tuna vysvetlit serial communication na debbugung, opisat debounce

3.7 Getting analog input

There are 2 types of inputs - binary - which might indicate if light is on/off. Other one is analog input which can indicate continuus signal - for example distance from obstacle, level of water in tank, speed...

Make new circuit by given schematic and copy code from FILEX.

3.8 Challange 2

Make circuit with potentiometer that dims or brightens LED.

3.9 Analog write

In last exercise you made your own PWM but Arduino offers library

Make new circuit by given schematic and copy code from FILEX.

ukazka priebehu na osciloscope

mozno nejkay kondenzator sem zakomponovat aby z toho bolo analog voltage - ukazat ze takto sa da generovat

4 Appendix

todo arduino mega scheme