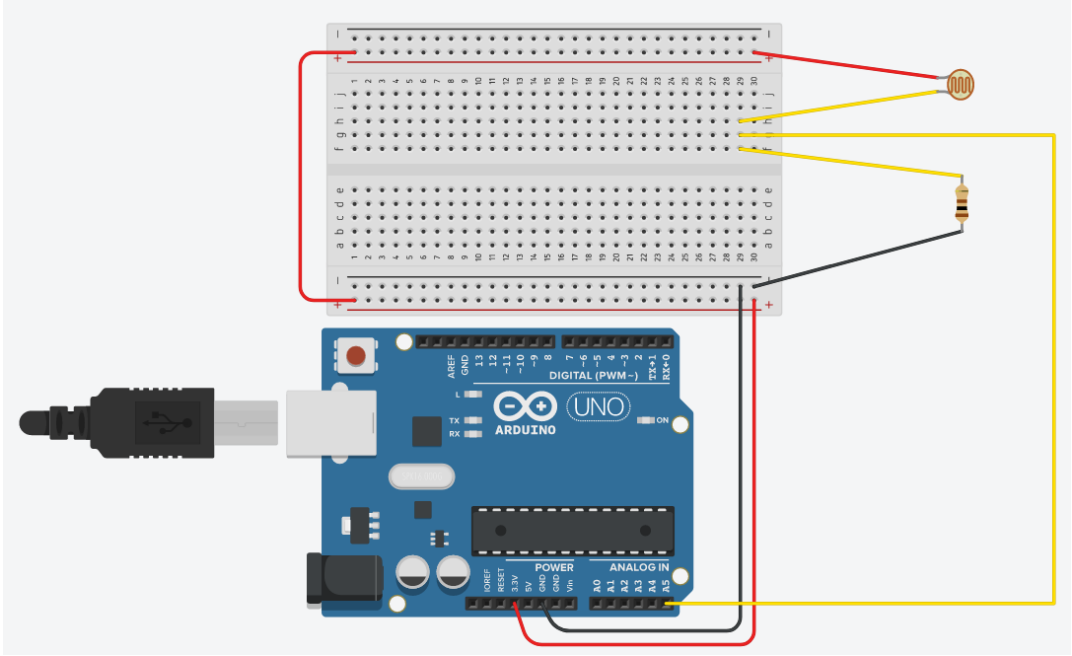
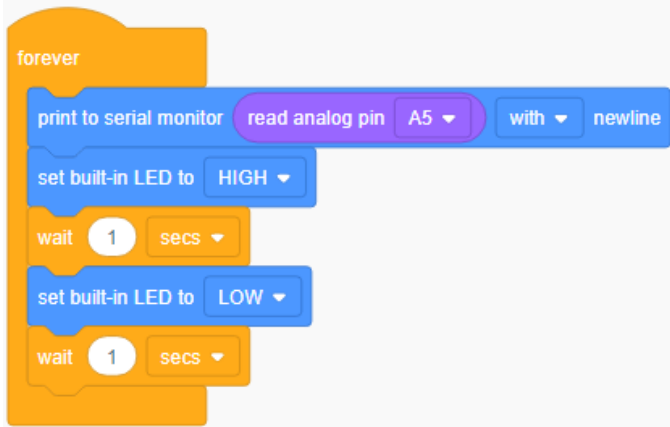


# Пупинов изазов – Децембар 2022

<b>Изазов 1</b>	Видео игра	<b>Област:</b> Програмирање софтвера
Потребно је направити помоћу <i>Processing</i> библиотеке „видео игру“ у којој се возило креће по земљи и може да гађа летилице. За сваку погођену летилицу добија се један поен. Летилице могу да гађају возило које губи један живот при сваком поготку (на почетку има N живота). Направити бројач за оборене летилице који ће бројати поготке све док не возило не изгуби све животе.		
<b>Изазов 2</b>	Фото отпорник	<b>Област:</b> Електротехника/Електроника
Направити <i>Arduino</i> шему који врши мерење податке са сензора фото-резистора.		
Дизајн је приказан на слици:		
		
Улаз A5 је замена за улаз за мерење напона у претходном примеру. То значи конфигурација <i>Arduino</i> уређаја може да мери интензитет светлости коју региструје фото-сензор преко пина A5.		
<b>Изазов 3</b>	Слање података серијским портом	<b>Област:</b> Програмирање хардвера
Потребно је написати програм за <i>Arduino</i> микроконтролер који шаље вредности које мери на фото сензору преко серијског порта.		
<b>Решење</b>		
Визуелни и програмски код којим се имплементира оваква функционалност су приказани у примеру:		
<i>Arduino Scratch</i> решење		<i>Arduino C</i> решење



```
void setup()
{
  pinMode(A5, INPUT);
  Serial.begin(9600);
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
{
  Serial.println(analogRead(A5));
  digitalWrite(LED_BUILTIN, HIGH);
  delay(1000); // Сачекај 1 секунд
  digitalWrite(LED_BUILTIN, LOW);
  delay(1000); // Сачекај 1 секунд
}
```

#### Изазов 4

Приказивање резултата мерења

**Област:** Програмирање софтвера

Направити *Processing* програм који ће преузети промене са сензора и приказати као правоугаоник који мења величину и боју на основу интензитета светлости.

#### Решење

Потребно је укључити библиотеке за рад са *Arduino* уређајем и декларисати серијски порт путем ког ће *Arduino* уређај слати податке:

```
import processing.serial.*;
```

```
Serial arduinoPort; // Серијски порт путем кога Arduino уређај шаље податке
```

Да бисмо нацртали вредности које се мере на *Arduino* уређају, потребно је да декларишемо једну променљиву која представља величину коју смо прочитали:

```
float value = 0; // висина правоугаоника који ће бити нацртан
```

Прво је потребно иницијализовати апликацију у функцији *setup()*:

```
void setup () {
  size(400, 300);
  background(0);
  stroke(155);
  arduinoPort = new Serial(this, Serial.list()[0], 9600);
```

// Не позивај *serialEvent()* све док не добијеш карактер који представља нову линију:

```
arduinoPort.bufferUntil('\n');
```

```
}
```

Потребно је написати функцију која ће реаговати на поруке које *Arduino* уређај шаље путем серијског порта.

```
void serialEvent (Serial arduinoPort) {
```

// читај поруку коју шаље *Arduino* до првог новог реда:

```
String data = arduinoPort.readStringUntil('\n');
```

```
if (data != null) {  
  // претвори текст у број и мапирај га на величину прозора:  
  value = float(data);  
  value = map(value, 0, 1023, 0, height);  
}  
}
```

Функција *draw()* из *Processing* библиотеке треба да нацрта правоугаоник на са једним теменом у доњем левом углу (0,height), ширине 100 и висине која одговара прочитаној величини.

```
void draw () {  
  // Мењај боју:  
  fill(value/5, 255 – value/5, 255):  
  // Нацртај правоугаоник:  
  rect(0, height, 100, value);  
}
```

<b>Изазов 5</b>	Програмирање LED штапа	<b>Област:</b> Програмирање софтвера
-----------------	------------------------	--------------------------------------

Направити *Processing* програм који ће преузети промене са сензора и приказивати их на штапу.

<b>Изазов 6</b>	Анимација – градијенти	<b>Област:</b> Програмирање
-----------------	------------------------	-----------------------------

Направити *Processing* програм који ће приказати градијент између две боје (црвена, плава). Искористити интензитет светлости да се центар градијента помера горе-доле у зависности од јачине светлости. Искористити координате миша да се одреди интензитет боје (лево слабо, десно јаче).