

Документация на проект ArduinoPong

Изготвена от
Валери Тончев

Съдържание

- Описание на проекта
- Блокова схема
- Електрическа схема
- Списък съставни части
- Сорс код – описание на функционалността
- Заключение

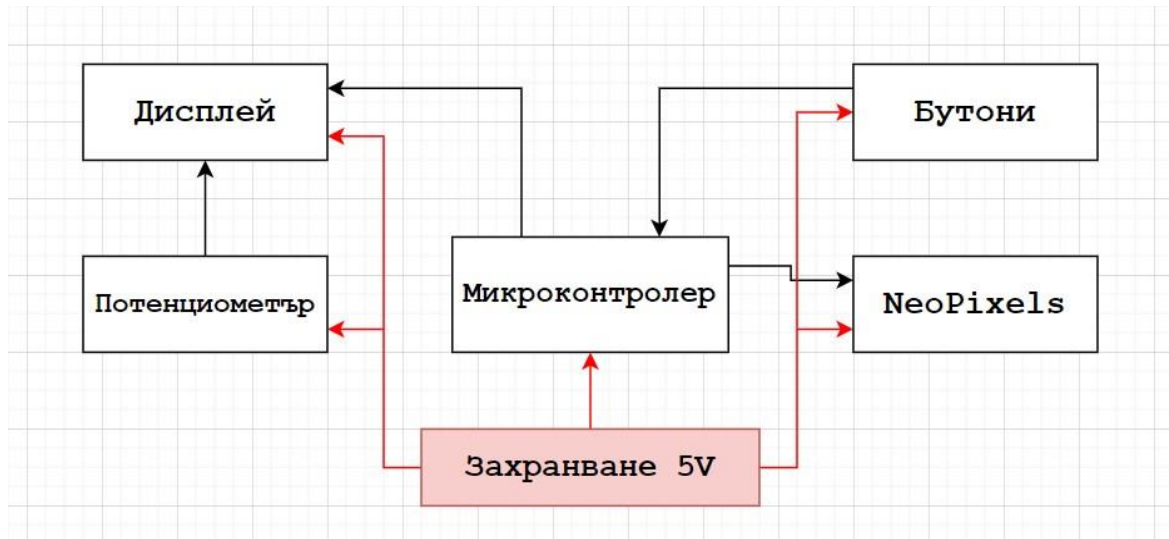
Описание на проекта

Проектът пресъздава аркадната пиксел игра „Pong“, в която двама играчи, използвайки платформи, отблъскват топче без да трябва да го изпускат. Точки получават при всяко успешно върнато топче. Печели този, който не изпусне топчето.

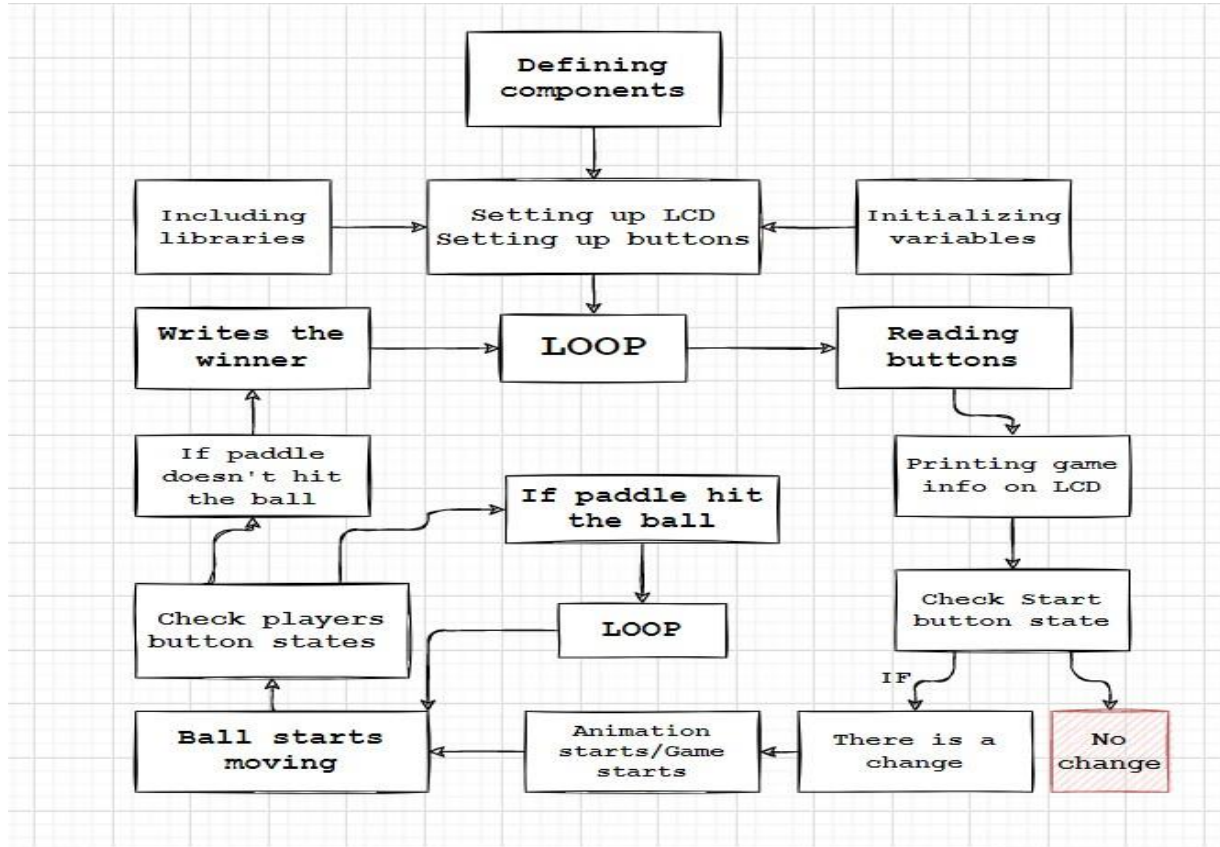
Анимацията на играта се визуализира на LCD. За да се стартира играта се натиска бутон „Start“, при което се появява игралното поле. Движението на двете платформи се случва чрез бутоните на двамата играча, като всеки играч има бутон за движение нагоре и надолу. В момент, в който играч спечели, NeoPixel светва в зелен цвят за победителя, а червен – за загубилия.

Проектът е с чисто развлекателна цел, имащ за идея да забавлява и да откъсне от напрежението в ежедневието.

Блокова схема

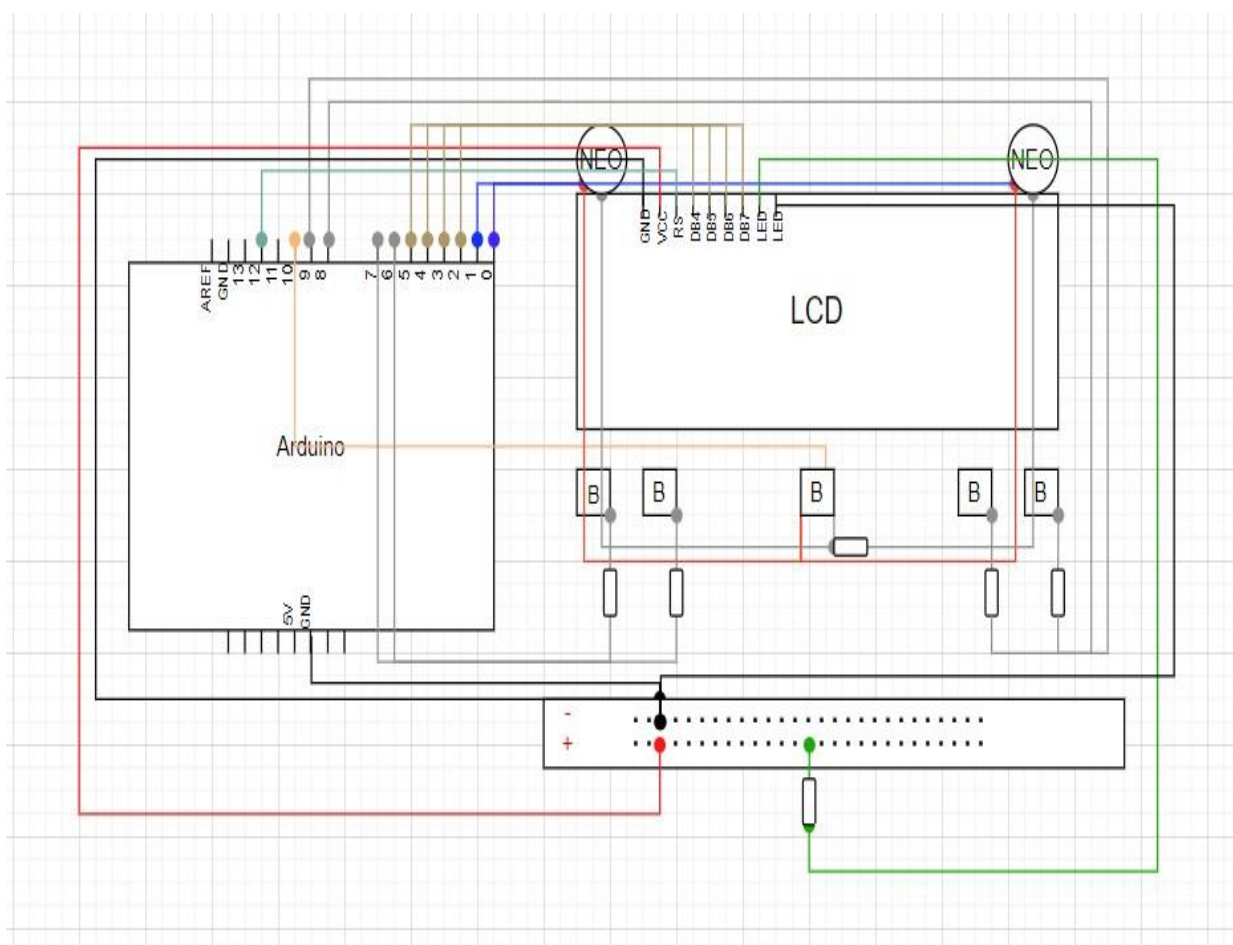


Блок схема на проекта.



Блок схема на кода.

Електрическа схема



Ел. схема на проекта.

Списък съставни части

Компонент	Име	Брой
Arduino Uno R3	U1	1
LCD 16 x 2	U2	1
220 Ω Resistor	R6	1
Pushbutton	S1, S2, S3, S4, S5	5
10 k Ω Resistor	R1, R2, R3, R4, R5	5
NeoPixel	LED1, LED2	2
10 k Ω Potentiometer	Rpot1	1

Сорс код

Описание на функционалността

В началото на кода въвеждам необходимата библиотека за използването на LCD и NeoPixel, както и дефинирам константните стойности и променливи, които ще ползвам:

```
1  #include <LiquidCrystal.h>
2  LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
3
4  #include <Adafruit_NeoPixel.h>
5
6  #define neoPixel1 0
7  #define neoPixel2 1
8  #define LEDs 1
9
10 Adafruit_NeoPixel strip1 = Adafruit_NeoPixel(LEDs, neoPixel1, NEO_GRB + NEO_KHZ800);
11 Adafruit_NeoPixel strip2 = Adafruit_NeoPixel(LEDs, neoPixel2, NEO_GRB + NEO_KHZ800);
12
13 #define start 13
14 #define P1U 6
15 #define P1D 7
16 #define P2U 8
17 #define P2D 9
18
19
20
21
22
23
24
25
26
27
28
29 byte paddle1[16] = {0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0};
30 byte player11[8] = {0, 0, 0, 0, 0, 0, 1, 1};
31 byte player12[8] = {1, 1, 0, 0, 0, 0, 0, 0};
32
33 byte paddle2[16] = {0, 0, 0, 0, 0, 0, 16, 16, 16, 16, 0, 0, 0, 0, 0, 0};
34 byte player21[8] = {0, 0, 0, 0, 0, 0, 16, 16};
35 byte player22[8] = {16, 16, 0, 0, 0, 0, 0, 0};
36
37 byte ball[16] = {0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0};
38 byte ball1[8] = {0, 0, 0, 0, 0, 0, 0, 1};
39 byte ball2[8] = {0, 0, 0, 0, 0, 0, 0, 0};
40
41 int x = 0, bounce = 0, v1 = 0, v2 = 0, v3 = 0;
42 char direction = 'L';
43 int score1 = 0, score2 = 0;
44 boolean game = 0;
45 int a = 0;
```

Въвеждам необходимите input на пиновете и въвеждам текст, който да излезе на LCD при стартиране на програмата:

```
59 void setup() {
60
61     strip1.begin();
62     strip1.show();
63     strip2.begin();
64     strip2.show();
65
66     pinMode(P1U, INPUT);
67     pinMode(P1D, INPUT);
68     pinMode(P2U, INPUT);
69     pinMode(P2D, INPUT);
70     pinMode(start, INPUT);
71
72     lcd.begin(16, 2);
73     lcd.clear();
74     lcd.setCursor(4, 0);
75     lcd.print("PingPong");
76     delay(1500);
77 }
```

Този цикъл се изпълнява, докато не бъде натиснат бутона за стартиране, за да започне играта:

```
80 void loop() {
81
82     while (game == 0) {
83         for (a; a < 1; a++) {
84             for (int i = 0; i < 16; i++) {
85                 if (6 <= i && i <= 9) {
86                     paddle1[i] = 1;
87                     paddle2[i] = 16;
88                 }
89
90                 else {
91                     paddle1[i] = 0;
92                     paddle2[i] = 0;
93                 }
94
95                 if (i == 7) ball[i] = 1;
96                 else ball[i] = 0;
97             }
98             x = 8;
99             lcd.clear();
100             lcd.setCursor(3, 0);
101             lcd.print("Press Start");
102         }
99     }
```


Проверявам дали старт бутона е натиснат и ако е, започва отброяване до началото на играта:

```
105     if (digitalRead(start) == HIGH) {
106         lcd.clear();
107         lcd.setCursor(5, 0);
108         lcd.print("Player");
109         lcd.setCursor(0, 1);
110         lcd.print("1");
111         lcd.setCursor(15, 1);
112         lcd.print("2");
113         delay(2000);
114
115         for (int i = 3; i > 0; i--) {
116             lcd.clear();
117             lcd.setCursor(4, 0);
118             lcd.print("Ready...");
119             lcd.setCursor(8, 1);
120             lcd.print(i);
121             delay(1000);
122         }
123         a = 0;
124         game = 1;
125     }
126     delay(100);
127 }
```

Задавам движението на платформите на двамата играчи:

```
130     if (digitalRead(P1U) == HIGH) {
131         paddle1Up();
132         delay(50);
133     }
134
135     if (digitalRead(P1D) == HIGH) {
136         paddle1Down();
137         delay(50);
138     }
139
140     if (digitalRead(P2U) == HIGH) {
141         paddle2Up();
142         delay(50);
143     }
144
145     if (digitalRead(P2D) == HIGH) {
146         paddle2Down();
147         delay(50);
148     }
```

Разделям масива на платформите и топчето на 2 други масива:

```
156     for (int i = 0; i <= 7; i++) {
157         player11[i] = paddle1[i];
158     }
159
160     for (int i = 0, j = 8; i <= 7, j <= 15; i++, j++) {
161         player12[i] = paddle1[j];
162     }
163
164     for (int i = 0; i <= 7; i++) {
165         player21[i] = paddle2[i];
166     }
167
168     for (int i = 0, j = 8; i <= 7, j <= 15; i++, j++) {
169         player22[i] = paddle2[j];
170     }
171
172     for (int i = 0; i <= 7; i++) {
173         ball1[i] = ball[i];
174     }
175
176     for (int i = 0, j = 8; i <= 7, j <= 15; i++, j++) {
177         ball2[i] = ball[j];
178     }
```

Създавам персонализирани форми за двете платформи и топчето и ги принтирам, задавайки позициите им:

```
183     lcd.createChar(0, player11);
184     lcd.createChar(1, player12);
185     lcd.createChar(2, player21);
186     lcd.createChar(3, player22);
187     lcd.createChar(4, ball1);
188     lcd.createChar(5, ball2);
189
190
191     lcd.clear();
192     lcd.setCursor(0, 0);
193     lcd.print(score1);
194     lcd.setCursor(0, 1);
195     lcd.print(score2);
196
197     lcd.setCursor(3, 0);
198     lcd.write(byte(0));
199     lcd.setCursor(3, 1);
200     lcd.write(byte(1));
201     lcd.setCursor(15, 0);
202     lcd.write(byte(2));
203     lcd.setCursor(15, 1);
204     lcd.write(byte(3));
205     lcd.setCursor(x, 0);
206     lcd.write(byte(4));
207     lcd.setCursor(x, 1);
208     lcd.write(byte(5));
```

Проверявам дали топчето удря платформите, какво се случва ако я уцели и какво ако не я(само за единия играч):

```
211  if (x == 3) {
212      for (int b = 0; b < 16; b++) {
213          if (ball[b] != 0) {
214              v1 = ball[b];
215              v2 = paddle1[b];
216          }
217      }
218
219      if (v1 == v2) {
220          direction = 'R';
221          score1 += 1;
222      }
223
224      else {
225          lcd.clear();
226          lcd.setCursor(3, 0);
227          lcd.print("Game over");
228          lcd.setCursor(4, 1);
229          lcd.print("P2 won!");
```

Зеленият светодиод светва за победителя, а червеният за загубилия и нулирам резултата за нова игра:

```
233      for (int i = 0; i < 3; i++) {
234          strip1.setPixelColor(0, 255, 0, 0);
235          strip1.show();
236          strip2.setPixelColor(0, 0, 255, 0);
237          strip2.show();
238          delay(500);
239          strip1.setPixelColor(0, 0, 0, 0);
240          strip1.show();
241          strip2.setPixelColor(0, 0, 0, 0);
242          strip2.show();
243          delay(500);
244      }
245
246      score1 = score2 = 0;
247      delay(1000);
248      game = 0;
249  }
250 }
```

Същото се прави и за другия играч.

Ако посоката е вляво, топчето се премества в тази посока:

```
298     if (direction == 'L') {
299         for (int b = 0; b < 16; b++)
300             if (ball[b] == 16) x -= 1;
301
302         if (bounce == 0)
303             ballLeftDown();
304         else
305             ballLeftUp();
306     }
```

(Същото правя и ако е вдясно)

Функционалност на платформите:

```
321 void paddle1Up() {
322     if (paddle1[0] != 1) {
323         int temp = paddle1[0], i;
324         for (i = 0; i < 16; i++)
325             paddle1[i] = paddle1[i + 1];
326         paddle1[i] = temp;
327     }
328 }
329
330 void paddle1Down() {
331     if (paddle1[15] != 1) {
332         int temp = paddle1[15], i;
333         for (i = 15; i > 0; i--)
334             paddle1[i] = paddle1[i - 1];
335         paddle1[i] = temp;
336     }
337 }
338
339 void paddle2Up() {
340     if (paddle2[0] != 16) {
341         int temp = paddle2[0], i;
342         for (i = 0; i < 16; i++)
343             paddle2[i] = paddle2[i + 1];
344         paddle2[i] = temp;
345     }
346 }
347
348 void paddle2Down() {
349     if (paddle2[15] != 16) {
350         int temp = paddle2[15], i;
351         for (i = 15; i > 0; i--)
352             paddle2[i] = paddle2[i - 1];
353         paddle2[i] = temp;
354     }
355 }
```

Функционалност на топчето:

```
362 void ballLeftDown() {
363
364     if (ball[15] == 0) {
365         int temp = ball[15], i;
366         for (i = 15; i > 0; i--)
367             ball[i] = ball[i - 1];
368         ball[i] = temp;
369         ballLeft();
370     }
371     else {
372         bounce = 1;
373     }
374 }
375
376 void ballLeftUp() {
377
378     if (ball[0] == 0) {
379         int temp = ball[0], i;
380         for (i = 0; i < 15; i++)
381             ball[i] = ball[i + 1];
382         ball[i] = temp;
383         ballLeft();
384     }
385     else {
386         bounce = 0;
387     }
388 }
389
390 void ballRightDown() {
391
392     if (ball[15] == 0) {
393         int temp = ball[15], i;
394         for (i = 15; i > 0; i--)
395             ball[i] = ball[i - 1];
396         ball[i] = temp;
397         ballRight();
398     }
399
400     else {
401         bounce = 1;
402     }
403 }
404
405 void ballRightUp() {
406
407     if (ball[0] == 0) {
408         int temp = ball[0], i;
409         for (i = 0; i < 15; i++)
410             ball[i] = ball[i + 1];
411         ball[i] = temp;
412         ballRight();
413     }
414     else {
415         bounce = 0;
416     }
417 }
418
419 void ballRight() {
420     for (int b = 0; b < 16; b++) {
421         if (ball[b] != 0) {
422             if (ball[b] == 16) ball[b] = 8;
423             else if (ball[b] == 8) ball[b] = 4;
424             else if (ball[b] == 4) ball[b] = 2;
425             else if (ball[b] == 2) ball[b] = 1;
426             else if (ball[b] == 1) ball[b] = 16;
427         }
428     }
429 }
430
431 void ballLeft() {
432     for (int b = 0; b < 16; b++) {
433         if (ball[b] != 0) {
434             if (ball[b] == 1) ball[b] = 2;
435             else if (ball[b] == 2) ball[b] = 4;
436             else if (ball[b] == 4) ball[b] = 8;
437             else if (ball[b] == 8) ball[b] = 16;
```

Заклучение

Проектът функционира и е достъпен в Tinkercad. Хората могат да играят играта, която цел да им доставя удоволствие и да ги забавлява е постигната. Следващият етап е да бъде разработен физически вариант на проекта, разполагащ с по-голям дисплей и алтернативни на бутоните контроли.

Проектът може да бъде изпробван на:

<https://www.tinkercad.com/things/dT7TTFiiEGX-arduinopong>