



ПРОЕКТ

Модул 8 - Разработка на софтуер

Тема: Scientific Calculator.

Изготвили:

Име: Катерина Александрова Димитрова

Име: Петър Павлов Бакларов

Име: Асен Иванов Царев

Име: Маттео Иванов

Клас: **XI клас**

Училище: **ПРОФИЛИРАНА ПРИРОДОМАТЕМАТИЧЕСКА
ГИМНАЗИЯ "Васил Левски" и Езикова гимназия „Иван Вазов“ гр.
Смолян**

Проверил: Красимира Юрукова



Съдържание

1. Описание на проекта	3
2. Блокова схема.....	3
3. Електрическа схема	5
4. Списък съставни части	6
5. Монтажна схема.....	10
6. Сорс код – описание на функционалността	11
7. Линк към GitHub, TinkerCad.....	19
8. Заключение	19

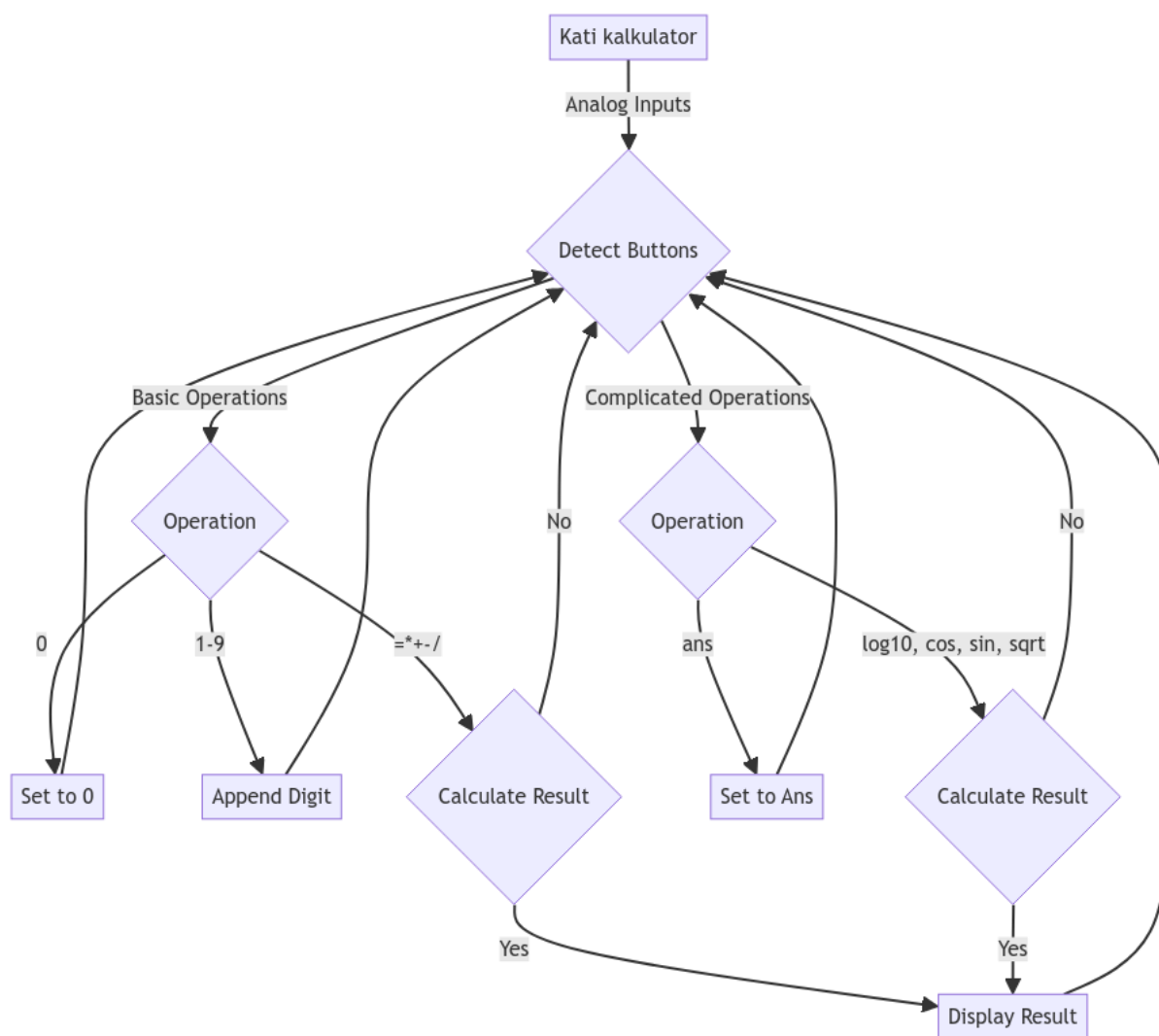


1. Описание на проекта

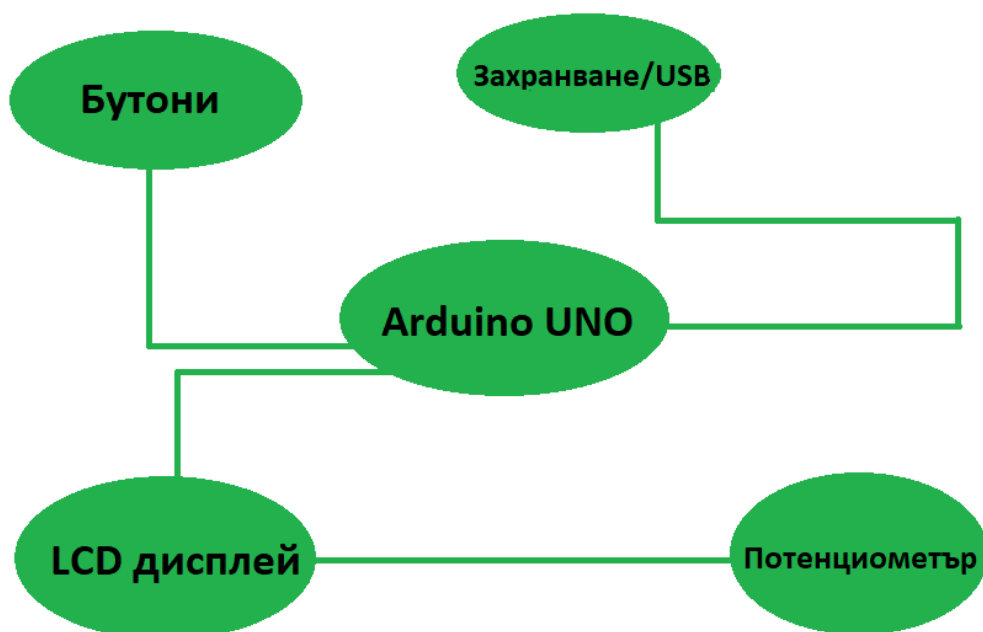
- Темата на проектната работа е научен калкулатор. Той представлява калкулатор, който използва Arduino платка и LCD дисплей. Разработената вградена система, позволява на потребителя да извършва различни математически операции и да види визуализираният се резултат на LCD екрана. Според това, кой от бутоните е натиснат от потребителя, програмата го разпознава като дадена математическа операция или число, и извършва съответните действия с въведените данни. Включени са различни операции като събиране, изваждане, умножение, деление, изчисляване на корен квадратен, тригонометричните функции – синус и косинус и повдигане на дадено число на избраната степен. Програмата включва една обработка на грешка и тя е проверка за деление на нула и извеждане на съобщения на LCD екрана.
- Проектът се осъществява с помощта на симулационната платформа TinkerCad. Създадена е както работеща симулация и функционира перфектно.
- Целта на проекта е да предостави удобен и лесен за използване калкулатор, който може да бъде изграден с достъпни електронни компоненти и да се използва за различни математически задачи.
- След като обмислихме идеята и разпределихме задачите, всеки член от екипа взе участие в разработването на всяка част от проекта. Създадохме симулация в TinkerCad. След успешното създаване на симулацията, преминахме към дебъгване на проекта и създаването на електрическа и блок схема и таблица с компоненти. Накрая, написахме документацията и подготвихме презентацията за нашия проект.

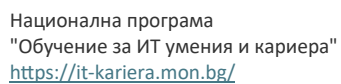
2. Блокова схема

- Представената блокова схема е визуализираната на блокова схема 1



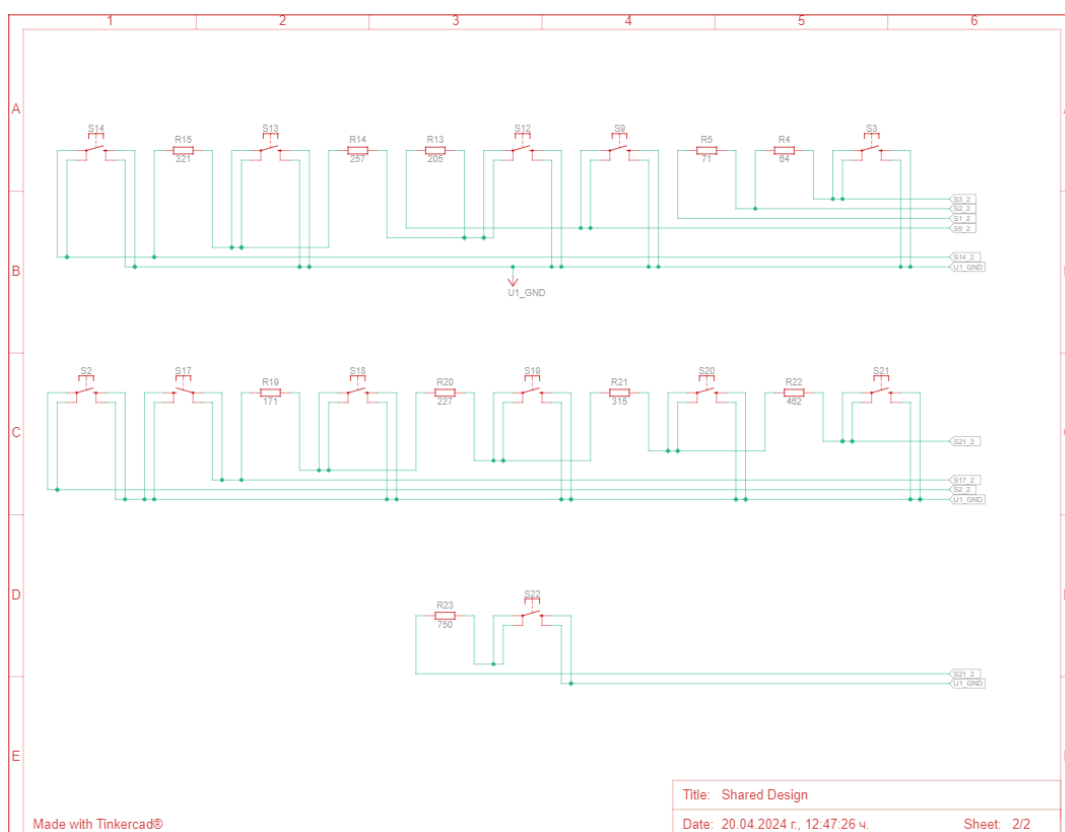
Блокова схема 1





- Представената електрическа схема е разделена на две части. Тя е визуализираната на фиг. 1 и фиг. 2





Фигура 2

4. Списък съставни части

➤ Списък със съставни части

NAME	QUANTITY	COMPONENT
U1	1	Arduino Uno R3
U2	1	LCD 16 x 2
R1 R6 R17	3	1 k Ω Resistor
Rpot1	1	250 k Ω Potentiometer
S1 S2	22	Pushbutton



S3		
S4		
S5		
S6		
S7		
S8		
S10		
S11		
S9		
S12		
S13		
S14		
S15		
S16		
S17		
S18		
S19		
S20		
S21		
S22		
R2	1	51 Ω Resistor
R3	1	57 Ω Resistor
R5	1	71 Ω Resistor
R7	1	81 Ω Resistor
R8	1	90 Ω Resistor
R9	1	107 Ω Resistor

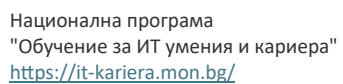


R10	1	121 Ω Resistor
R11	1	143 Ω Resistor
R5	1	71 Ω Resistor
R13	1	205 Ω Resistor
R14	1	257 Ω Resistor
R15	1	321 Ω Resistor
R16	1	424 Ω Resistor
R18	1	243 Ω Resistor
R19	1	171 Ω Resistor
R20	1	227 Ω Resistor
R4	1	64 Ω Resistor
R21	1	315 Ω Resistor
R16	1	424 Ω Resistor
R18	1	243 Ω Resistor

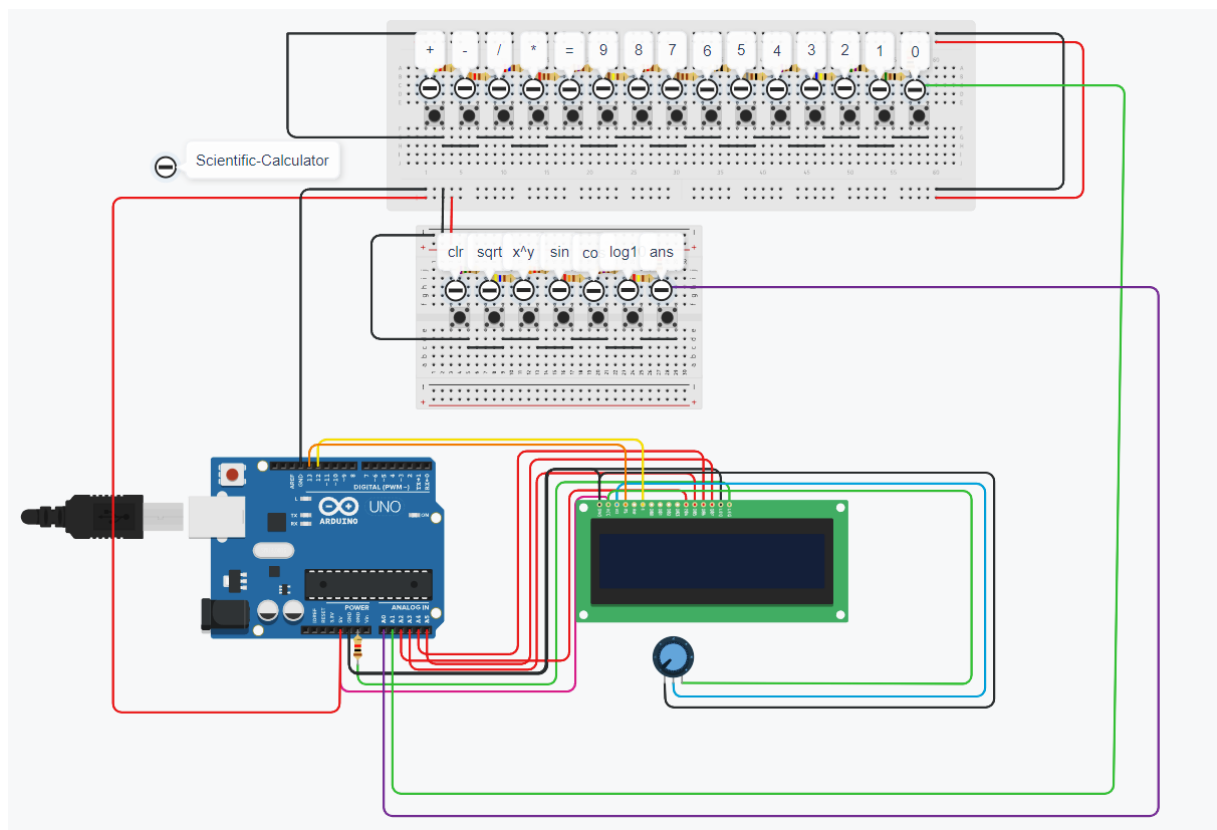


	A	B	C	D	E	F	G
1	Name	Quantity	Component		Name	Quantity	Component
2	U1	1	Arduino Uno R3		R2	1	51 Ω Resistor
3	U2	1	LCD 16 x 2		R3	1	57 Ω Resistor
4					R5	1	71 Ω Resistor
5	R1 R6 R17	3	1 kΩ Resistor		R7	1	81 Ω Resistor
6					R8	1	90 Ω Resistor
7	Rpot1	1	250 kΩ Potentiometer		R9	1	107 Ω Resistor
8					R10	1	121 Ω Resistor
9	S1 S2 S3 S4 S5 S6 S7 S8 S10 S11 S9 S12 S13 S14 S15 S16 S17 S18 S19 S20 S21 S22	22	Pushbutton		R11	1	143 Ω Resistor
10					R12	1	172 Ω Resistor
11					R13	1	205 Ω Resistor
12					R14	1	257 Ω Resistor
13					R15	1	321 Ω Resistor
14					R16	1	424 Ω Resistor
15					R18	1	243 Ω Resistor
16					R19	1	171 Ω Resistor
17					R20	1	227 Ω Resistor
18					R4	1	64 Ω Resistor
19					R21	1	315 Ω Resistor
20					R22	1	462 Ω Resistor
21					R23	1	750 Ω Resistor
22							
23							
24							
25							
26							
27							
28							
29							

Съставни части 1



➤ **Монтажна схема 1**



Монтажна схема 1



6. Сорс код – описание на функционалността

- Сорс код:
- Инициализира комуникация и дисплей LiquidCrystal. Непрекъснато чете аналогови входове от двата пина (A0 и A1). Ако се открие някакъв вход (стойността не е равна на 1023), се извиква функцията `detectButtons()`. Ако е готово за изчисление (`calculatable` е `true`), се извиква функцията `calculateResult()`. Извежда резултата на LCD. Изчиства LCD дисплей след забавяне от 500 милисекунди.
- `detectButtons()` – Чете аналогови стойности от A0 и A1. В зависимост от стойността от A1, извършва различни действия. Ако е натиснат бутон за число (0-9), добавя цифрата към текущия резултат. Ако е натиснат бутон "=", задава втория аргумент за изчисление и задава `calculatable` на `true`. Ако е натиснат бутон за аритметична операция (+, -, *, /), задава операцията, запазва текущия резултат като първи аргумент и нулира резултата. В зависимост от стойността от A0, извършва допълнителни операции като извикване на последния отговор, извършване на логаритмични функции, тригонометрични функции.
- `calculateResult()` – Изчислява резултата въз основа на запазената операция и аргументи. Обработва аритметични операции (+, -, *, /) и различни математически функции (`log10`, `sin`, `cos`, `sqrt`). Извежда съобщение за грешка на LCD ако се опитва да се раздели на нула.
- `displayResult()` – Извежда текущото състояние на калкулатора на LCD. Показва първия аргумент, операцията и втория аргумент (ако има такъв). Ако е готово за изчисление (`calculatable` е `true`), извежда резултата.



```
1  #include <LiquidCrystal.h>
2  #include <math.h>
3
4  LiquidCrystal lcd(13, 12, A2, A3, A4, A5);
5
6  unsigned long int firstArg = 0, secondArg = 0;
7  double result = 0;
8  double ans = 0;
9  bool calculatable = false;
10 String operation = "";
11
12  void setup()
13  {
14      Serial.begin(9600);
15      lcd.begin(16, 2);
16      lcd.print("Scientific-Calc");
17      delay(1000);
18      lcd.clear();
19  }
20
21  void loop() {
22      int valueA0 = analogRead(A0);
23      int valueA1 = analogRead(A1);
24      if (valueA0 != 1023 || valueA1 != 1023) {
25          detectButtons();
26      }
27      if (calculatable) {
28          calculateResult();
29      }
30      displayResult();
31      delay(500);
32      lcd.clear();
```



```
33     }
34
35     void detectButtons() {
36         int valueA0 = analogRead(A0);
37         int valueA1 = analogRead(A1);
38         switch (valueA1) { //switch on the larger keypad (basic operations)
39             case 0: // number 0
40                 {
41                     if (result == 0) {
42                         result = 0;
43                     } else {
44                         result *= 10;
45                     }
46                 }
47                 break;
48             case 50: // number 1
49                 {
50                     if (result == 0) {
51                         result = 1;
52                     } else {
53                         result = result * 10 + 1;
54                     }
55                 }
56                 break;
57             case 100: // number 2
58                 {
59                     if (result == 0) {
60                         result = 2;
61                     } else {
62                         result = result * 10 + 2;
63                     }
64                 }
65                 break;
66             case 150: // number 3
67                 {
68                     if (result == 0) {
69                         result = 3;
70                     } else {
71                         result = result * 10 + 3;
72                     }
73                 }
```



```
74         break;
75         case 200: // number 4
76         {
77             if (result == 0) {
78                 result = 4;
79             } else {
80                 result = result * 10 + 4;
81             }
82         }
83         break;
84         case 250: // number 5
85         {
86             if (result == 0) {
87                 result = 5;
88             } else {
89                 result = result * 10 + 5;
90             }
91         }
92         break;
93         case 300: // number 6
94         {
95             if (result == 0) {
96                 result = 6;
97             } else {
98                 result = result * 10 + 6;
99             }
100        }
101        break;
102        case 350: // number 7
103        {
104            if (result == 0) {
105                result = 7;
106            } else {
107                result = result * 10 + 7;
108            }
109        }
110        break;
111        case 400: // number 8
112        {
113            if (result == 0) {
114                result = 8;
```



```
115         } else {
116             result = result * 10 + 8;
117         }
118     }
119     break;
120     case 450: // number 9
121     {
122         if (result == 0) {
123             result = 9;
124         } else {
125             result = result * 10 + 9;
126         }
127     }
128     break;
129     case 500: //=
130     {
131         secondArg = result;
132         calculatable = true;
133     }
134     break;
135     case 550: //multiplication
136     {
137         operation = "*";
138         firstArg = result;
139         result = 0;
140     }
141     break;
142     case 600: //division
143     {
144         operation = "/";
145         firstArg = result;
146         result = 0;
147     }
148     break;
149     case 650: //subtraction
150     {
151         operation = "-";
152         firstArg = result;
153         result = 0;
154     }
155     break;
```



```
156     case 700: //addition
157     {
158         operation = "+";
159         firstArg = result;
160         result = 0;
161     }
162     break;
163 }
164
165 switch (valueA0) { //switch on smaller keypad (complicated operations)
166     case 0: //ans
167         result = ans;
168         break;
169     case 200: //log10
170         operation = "log10";
171         break;
172     case 300: //cos
173         operation = "cos";
174         break;
175     case 400: //sin
176         operation = "sin";
177         break;
178     case 500: //x^y
179         operation = "^";
180         firstArg = result;
181         result = 0;
182         break;
183     case 600: //sqrt
184         operation = "sqrt";
185         break;
186     case 700: //clr
187         operation = "";
188         firstArg = 0;
189         secondArg = 0;
190         ans = result;
191         result = 0;
192         calculatable = false;
193         break;
194 }
195
196 delay(500);
```




```
197     }
198
199   void calculateResult() {
200       result = 0;
201       if (operation.equals("+")) {
202           result = firstArg + secondArg;
203       }
204       if (operation.equals("-")) {
205           result = firstArg - secondArg;
206       }
207       if (operation.equals("*")) {
208           result = firstArg * secondArg;
209       }
210       if (operation.equals("/")) {
211           if (secondArg == 0) {
212               lcd.setCursor(0, 1);
213               lcd.print("Can't divide by 0");
214           }
215           result = (float)firstArg / secondArg;
216       }
217       if (operation.equals("log10")) {
218           result = log10(secondArg);
219       }
220       if (operation.equals("sin")) {
221           result = sin(secondArg);
222       }
223       if (operation.equals("cos")) {
224           result = cos(secondArg);
225       }
226       if (operation.equals("^")) {
227           result = pow(firstArg, secondArg);
228       }
229       if (operation.equals("sqrt")) {
230           result = sqrt(secondArg);
231       }
232       //resultText = new String(result);
233   }
234
235   void displayResult() {
236       lcd.setCursor(0, 0);
237       if (firstArg != 0) {
238           lcd.print(firstArg);
```




```
239     }
240     else {
241         lcd.print(result, 0);
242     }
243     if (!operation.equals("")) {
244         lcd.print(operation);
245     }
246     else {
247         lcd.print(" ");
248     }
249     if (secondArg != 0) {
250         lcd.print(secondArg);
251     }
252     else if (firstArg != 0) {
253         lcd.print(result, 0);
254     }
255     if (calculatable) {
256         if (firstArg == 0) {
257             lcd.clear();
258             lcd.print(operation);
259             lcd.print(secondArg);
260         }
261         lcd.print(" = ");
262         lcd.print(result, 2);
263     }
264 }
```



- Резисторите се използват, за да се свържат бутоните към един пин.

3 Band 4 Band 5 Band 6 Band



Resistance
10 Ω ±5%

Minimum
9.5 Ω

Maximum
10.5 Ω

	Band 1 1 st	Band 2 2 nd	Band 3 Multiplier	Band 4 Tolerance
Black		0	x10 ⁰	
Brown	1	1	x10 ¹	±1%
Red	2	2	x10 ²	±2%
Orange	3	3	x10 ³	±0.05%
Yellow	4	4	x10 ⁴	±0.02%
Green	5	5	x10 ⁵	±0.5%
Blue	6	6	x10 ⁶	±0.25%
Violet	7	7	x10 ⁷	±0.1%
Grey	8	8	x10 ⁸	±0.01%
White	9	9	x10 ⁹	
Gold			x10 ⁻¹	±5%
Silver			x10 ⁻²	±10%
Pink			x10 ⁻³	

Standard: IEC 60062:2016

7. Линк към GitHub, TinkerCad

- TinkerCad Link
- <https://www.tinkercad.com/things/7LO7A2wiD7R-copy-of-project-module-8/editel?sharecode=GNh6WFG-WGh9ctIyL7fgSFLxO-9XOmSvVyPHAaL6WXg>
- GitHub
- <https://github.com/cathy-09/Project-module-8>

8. Заключение

- Успяхме да сътворим проект с Arduino платка симулативно. В следствие на тази работа ще можем да работим с Arduino, а и ще сме по-добре запознати с това как работят компютърните и електрическите вградени системи.
- Използвани технологии – TinkerCad, GitHub, Microsoft Word, Microsoft PowerPoint
- Използвани сайтове – Youtube, <https://resistorcolorcodecalc.com/>