

Electronica Programabila

Introducere

Tema propusa:

Emoji Led. Să se afișeze pe o matrice de LED-uri minim 8 emoji. Utilizatorul va putea să aleagă oricare emoji folosind minim 3 butoane.

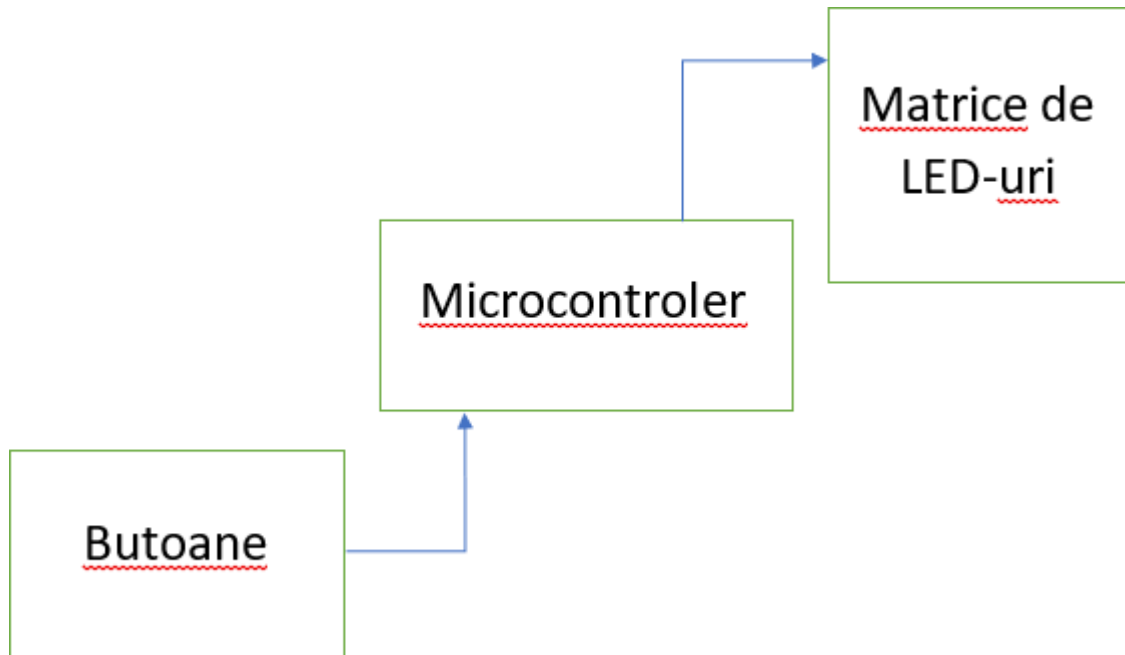
Solutie:

Pentru a putea afișa 8 emoji-uri pe o matrice de LED-uri, am ales să utilizăm 3 butoane cu reținere, acestea generând 8 combinații posibile prin cele două stări pe care le poate avea fiecare dintre ele: apasat și neapasat. În funcție de combinația aleasă (exemple: primele două butoane apasate și al treilea buton neapasat, toate butoanele apasate, toate butoanele neapasate), pe matricea de LED-uri va fi afișat unul dintre cele opt emoji-uri, programate în cu ajutorul programului AVR Studio și a paginii de catalog a matricei de LED-uri.

Intrucât simulatorul nu simulează în mod corect prezența tensiunii de pull-up folosind rezistențe interne, în ceea ce privește cele trei butoane, am ales să punem câte o rezistență de pull-up externă, explicită, de $10\text{K}\Omega$, conectată la sursa de 5V, pentru fiecare din cele trei butoane.

Partea teoretica

Schema bloc:



- **Microcontroler:**

Microcontrolerul folosit este Atmega164A.

Este un microcontroler pe 8 biti, de joasa putere, avand 32 de registre de uz general. Are o arhitectura tip RISC, opereaza la voltaje intre 1,8 si 5,5 volti. Are o viteza de ceas de pana la 20MHz, surse de intrerupere interne si externe. Prezinta un CAN la portul A, 2 timere si interfata seriala.

- **Butoane:**

Butoanele folosite sunt cu retinere, prezinta 6 pini, dintre care am folosit doar doi: primul, Normally Closed, este coenctat la sursa de 5V, iar al doilea, Common, la rezistenta de pull-up externa.

- Matricea de LED-uri:

Matricea de LED-uri folosita este o matrice bicoloră roșie și verde formată din 8x8 LED-uri, de obicei folosita pentru afișarea diferitelor jocuri de lumini sau numere/cifre. Am folosit doar una dintre cele doua culori disponibile pe aceasta, fiind suficient pentru a afisa diferitele emoji-uri, ce difera unele de celelalte doar prin forma, nu si prin culoare.

- Rezistente:

Rezistentele folosite au valoarea nominala de 10K Ω , sistemul de montare al acestora fiind THT.

BOM:

Nr. Crt.	Catalog	Producator	Nume Produs	Clasa	Cantitate	Descriere
1	Link	Microchip	8-bit Atmel Microcontroler	Microcontroler	1	16/32/64/128K Bytes In-System Programmable Flash
2	Link	Optimus Digital	Matrice de LED-uri		1	Matrice de LED-uri Bicoloră Roșie și Verde 8x8 3.75 mm
3	Link	Optimus Digital	Buton		3	Modul Buton cu Reținere
4	Link	MULTICOMP PRO	MF50 10K	Rezistor	3	Through Hole Resistor, 10 kohm

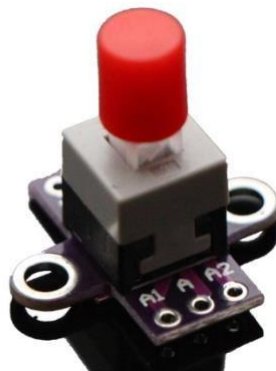
Descrierea componentelor hardware:

1. Atmel ATmega164A/164PA/324A/324PA/644A/644PA/1284/1284P este un CMOS de 8 biți cu putere redusa microcontroler bazat pe arhitectura RISC imbunatatita de AVR. Executand instructiuni intr-un singur ciclu de ceas, ATmega164A/164PA/324A/324PA/644A/644PA

/1284/1284P realizeaza randamente apropiindu-se de 1 MIPS permitand proiectantului sistemului sa optimizeze consumul de energie fata de viteza de procesare. Acesta este elementul care face legatura intre partea software si hardware.



2. Cele trei butoane cu retinere sunt portiunea din circuit ce functioneaza, impreuna, ca un dispozitiv de intrare, intrucat in functie de combinatia aleasa de utilizator, unul dintre cele 8 emoji-uri va fi afisat pe matricea de LED-uri.



3. Matricea de LED-uri se comporta ca un dispozitiv de iesire pentru circuitul proiectat de noi, deoarece in functie de pozitia LED-urilor ce se aprind dupa alegerea de catre utilizator a unei combinatii de butoane apasate/neapasate, se creeaza o forma ce reprezinta unul dintre emoji-uri.

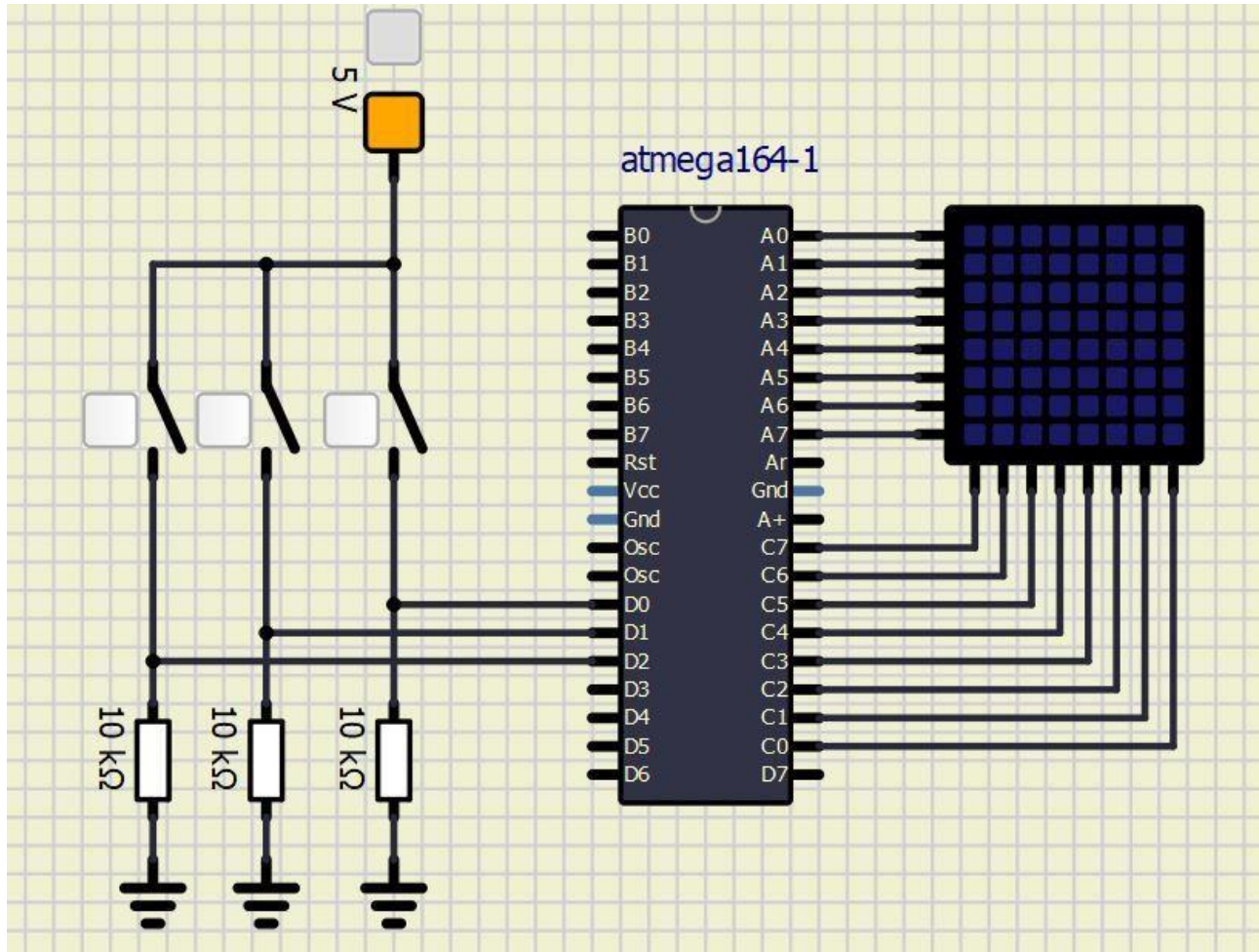


4. Rezistențele cu film metalic din seria MFR sunt fabricate utilizând un sistem de pulverizare sub vid pentru a depune mai multe straturi de aliaje metalice mixte și materiale pasive pe un substrat ceramic tratat cu atenție.



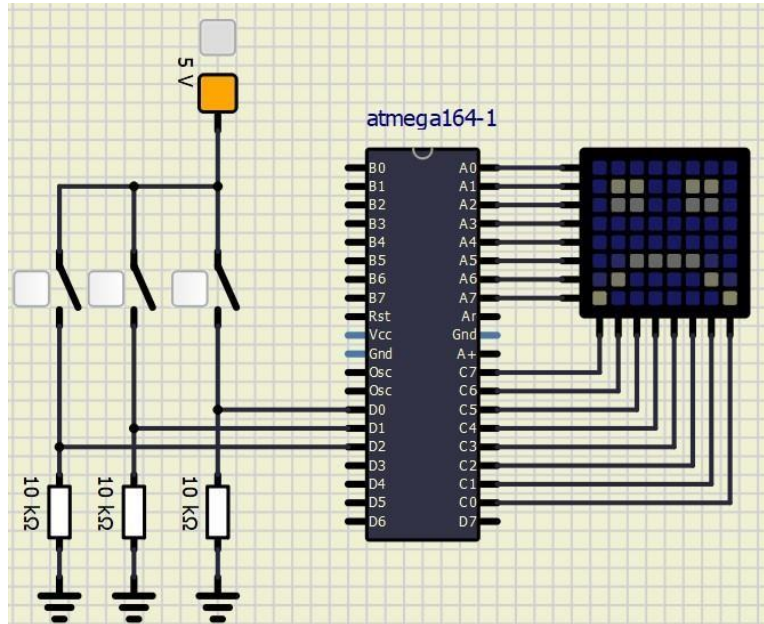
Partea experimentală

Schema simulare:

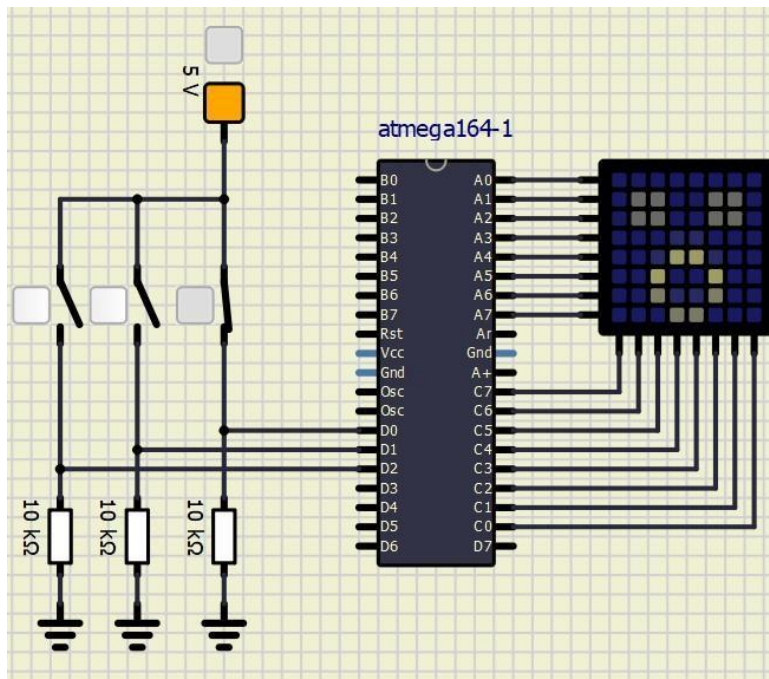


Cele 8 emoji-uri, in functie de combinatia de butoane apasate/neapasate, arata in felul urmator:

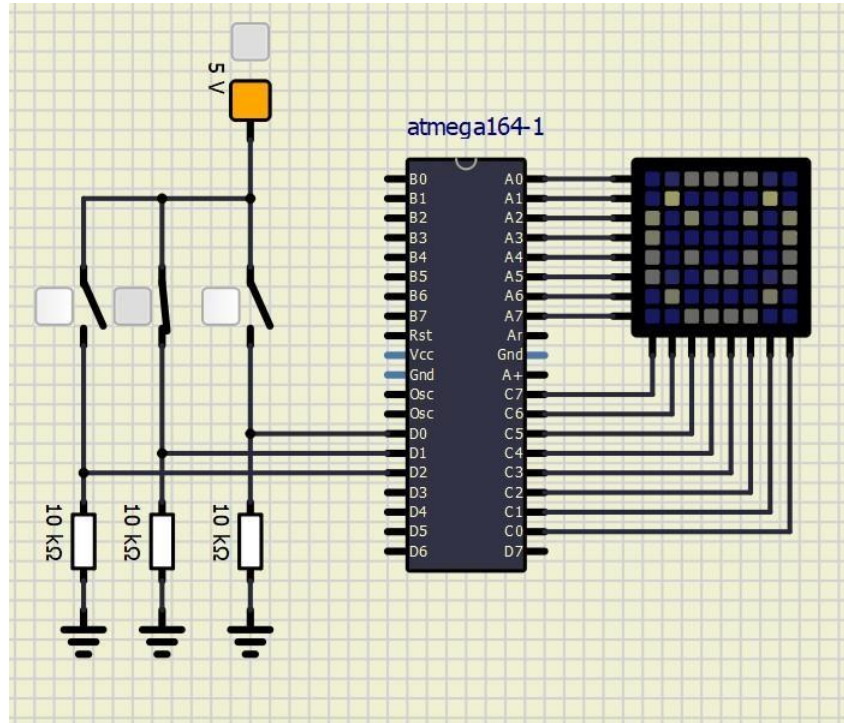
1) toate butoanele neapasate – emoji TRIST



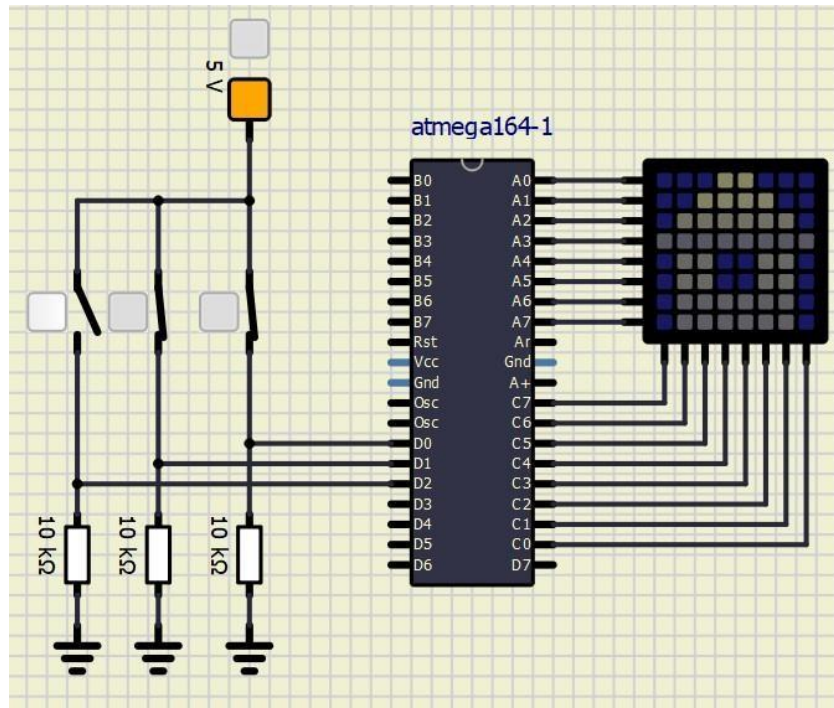
2) primele doua butoane neapasate, ultimul apasat – emoji MIRAT



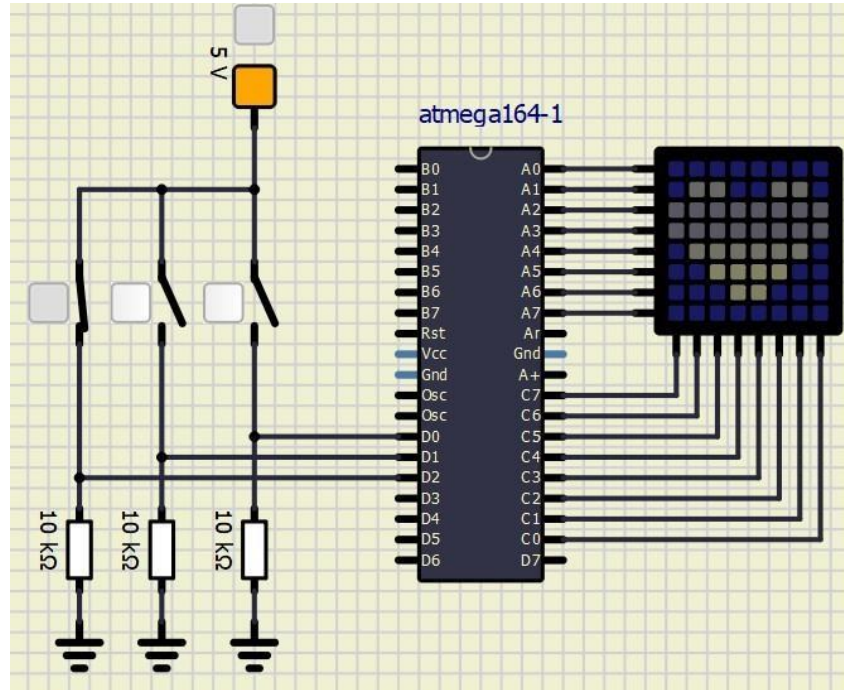
3) primul si al treilea buton neaprasate, al doilea apasat – emoji VESEL



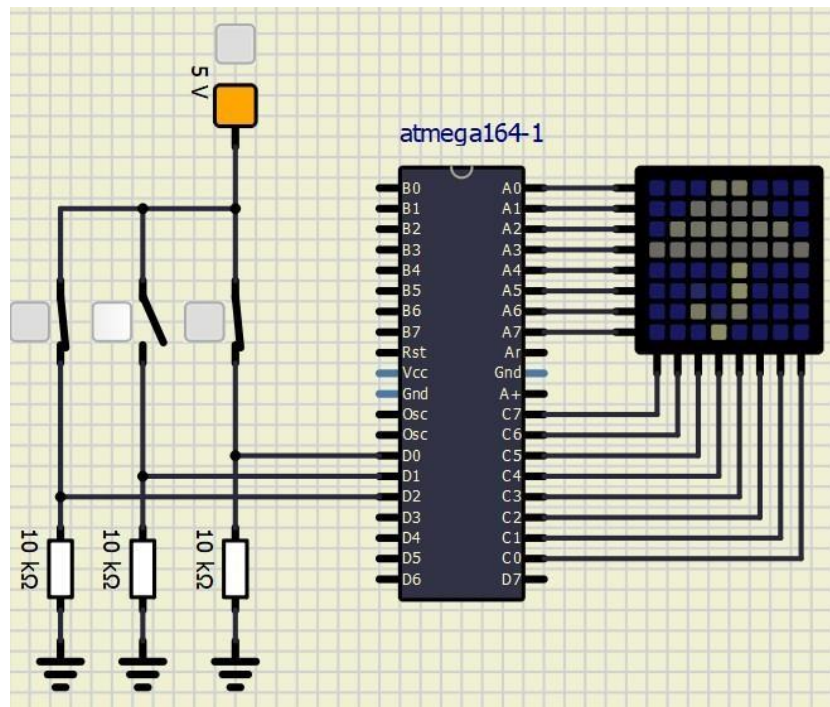
4) primul buton neaprasat, al doilea si al treilea apasate – emoji CASA



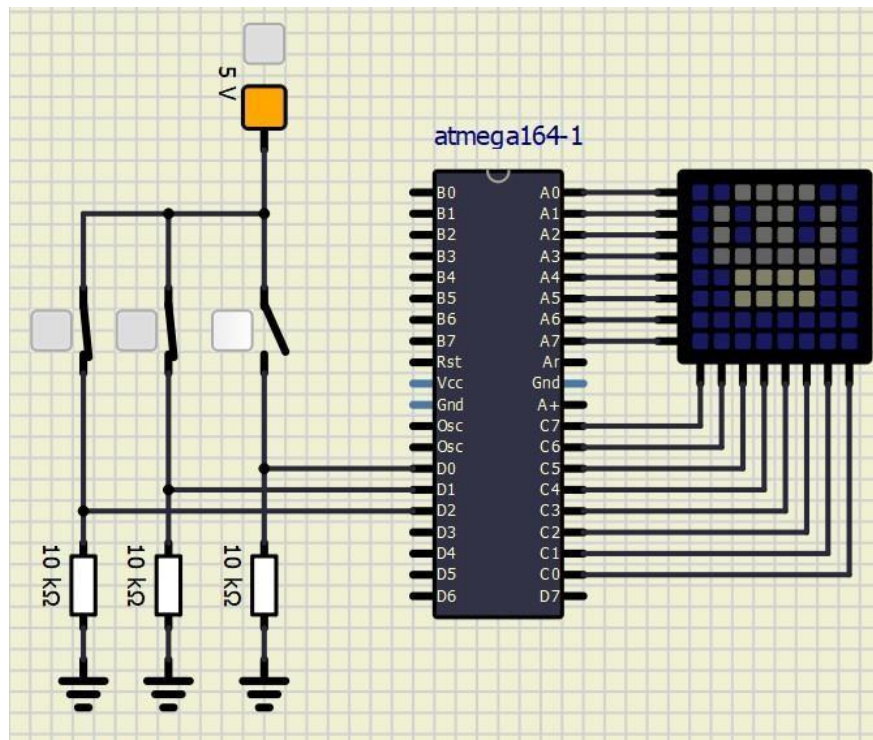
5) primul buton apasat, al doilea si al treilea neaprasate – emoji INIMA



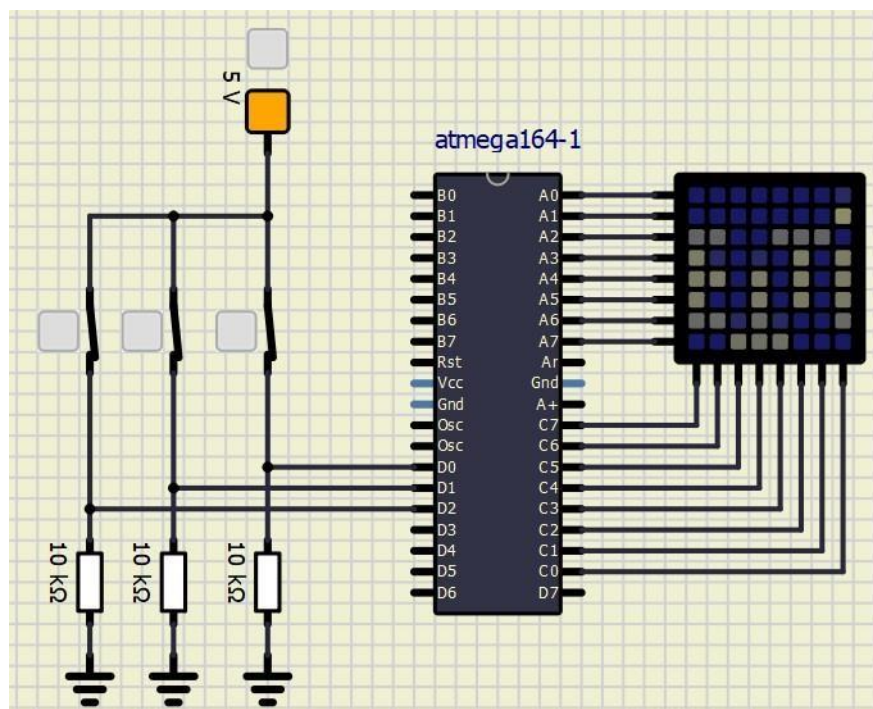
6) primul si al treilea buton apasate, al doilea neapasat – emoji UMBRELA

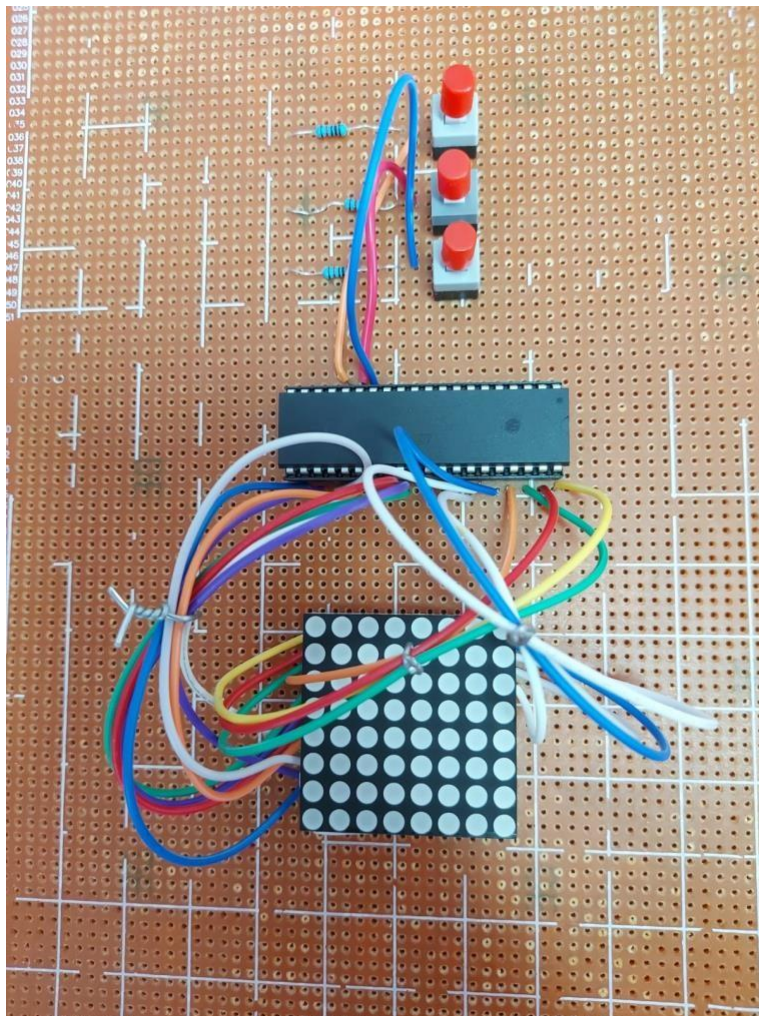
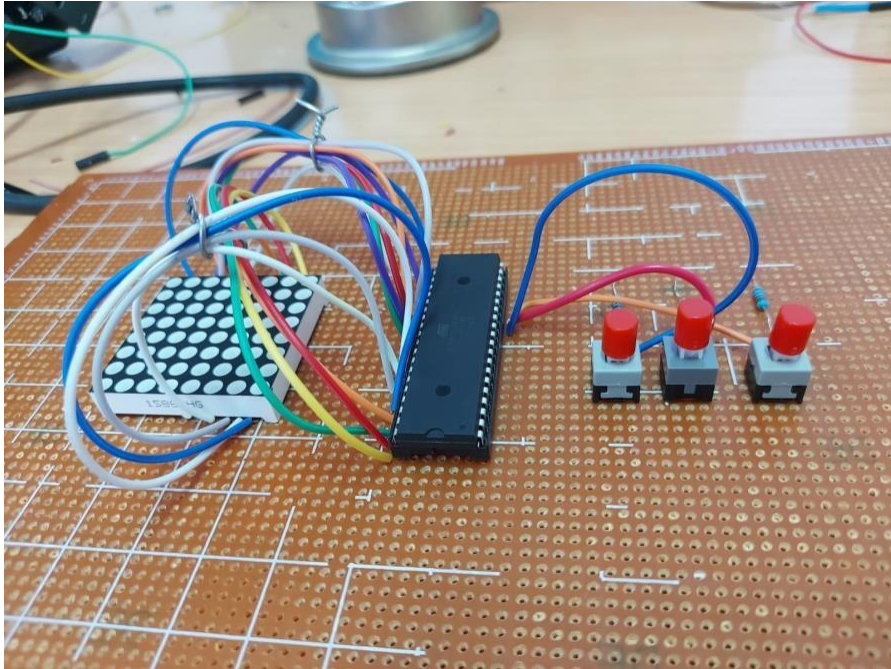


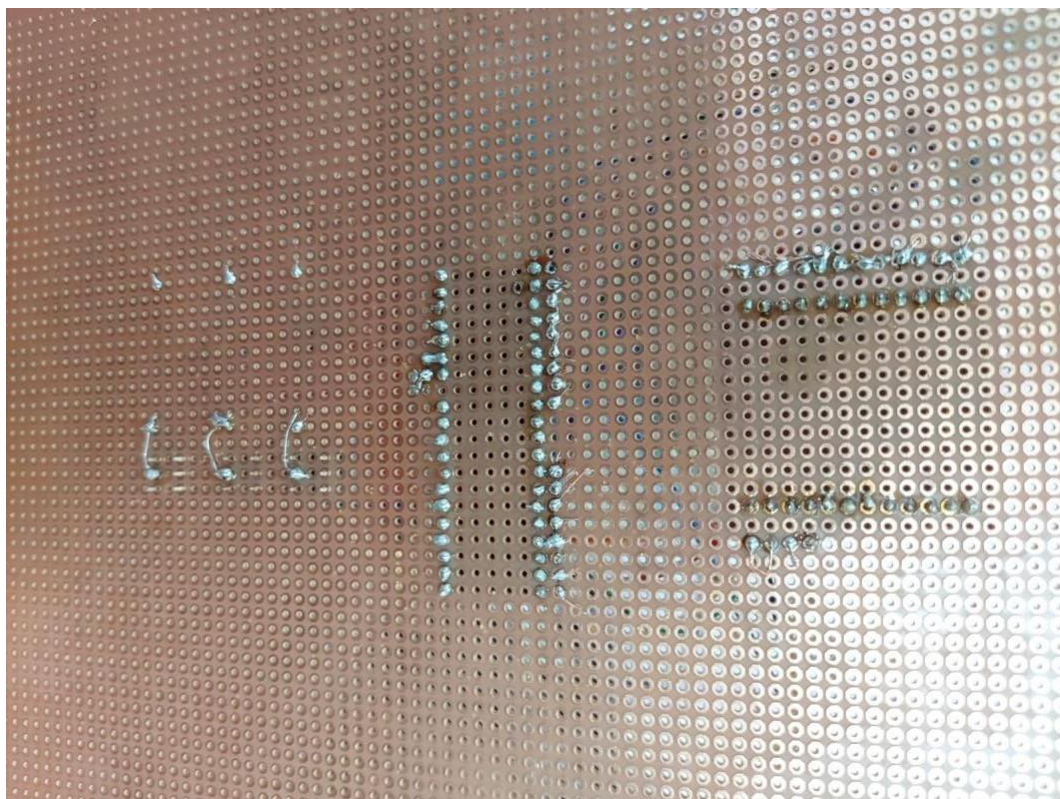
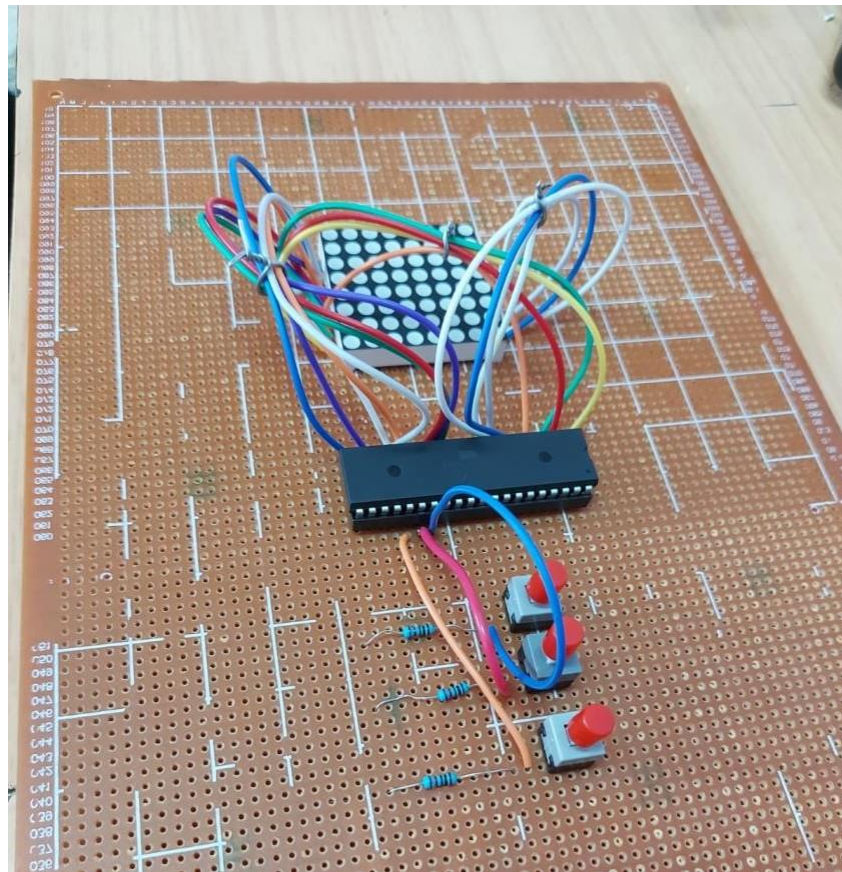
7) primul si al doilea buton apasate, al treilea neapasat – emoji SKULL



8) toate butoanele apasate – emoji ETTI







Rezultate si concluzii

Codul a fost implementat cu succes, insa, datorita simplitatii acesteia, folosirea microcontrolerului Atmega164A nu reprezinta cel mai bun exemplu de utilizare eficienta a resurselor, intrucat acesta ar fi putut suporta un cod mai complex (ce ocupa mai multa memorie), pentru aplicatia prezentata putand fi utilizat un microcontroler cu performante mai scazute decat acesta.

Bibiografie

1. <https://www.farnell.com/datasheets/2860633.pdf>
2. https://www.optimusdigital.ro/ro/butoane-i-comutatoare/5622-modul-buton-cu-reinere-nelipit.html?search_query=buton&results=218
3. https://www.optimusdigital.ro/ro/optoelectronice-matrice-de-led-uri/1005-matrice-de-led-uri-bicolore-roie-i-verde-8x8-375-mm.html?search_query=matrice&results=73
4. <https://www.avrfreaks.net/forum>
5. <http://ham.elcom.pub.ro/proiect2/doc.htm>

Anexa:

```
#include
```

```
#<mega164a.h>
```

```
include <delay.h>
```

```
char smile[8] = {0xFF, 0x99, 0x99, 0xFF, 0x81, 0xBD, 0x81, 0xFF};
```

```
char face[8] = {0xc3, 0xbd, 0x5a, 0x7e, 0x5a, 0x66, 0xbd, 0xc3};
```

```
char heart[8] = {0xff, 0x99, 0x0, 0x0, 0x81, 0xc3, 0xe7, 0xff};
```

```
char pacman[8] = {0xc7, 0x83, 0x31, 0x33, 0x7, 0x3, 0x81, 0xc3};
```

```
char surprised[8] = {0xff, 0x99, 0x99, 0xff, 0xe7, 0xdb, 0xdb,
```

```
0xe7}; char umbrella[8] = {0xe7, 0xc3, 0x81, 0x0, 0xf7, 0xf7, 0xd7,
```

```
0xef}; char house[8] = {0xe7, 0xc3, 0x81, 0x0, 0x99, 0x99, 0x81,
```

```
0x81}; char etti[8] = {0xff, 0xfe, 0x31, 0x7a, 0x2a, 0x6a, 0x2e,
```

```
0xc7};
```

```
char skull[8] = {0xc3, 0xa5, 0xa5, 0x81, 0xc3, 0xc3, 0xff,
```

```
0xff}; char sad[8] = {0xff, 0x99, 0x99, 0xff, 0xff, 0xc3, 0xbd,
```

```
0x7e};
```

```
void main(void)
```

```
{int i;
```

```
// DDRA - linii matrice LED - setarea tuturor pinilor ca
```

```
outputDDRA = 0xFF;
```

```
// DDRC - coloane matrice LED - setarea tuturor pinilor ca
```

```
outputDDRC = 0xFF;
```

```
// DDRD - butoane - setarea tuturor pinilor ca
```

```
inputDDRD = 0x0;
```



```

while (1) {
    // Pornirea LED-urilor se face pe o singura linie la
    // un moment de timp dat, insa, datorita vitezei,
    // ochiul uman va vedea intreaga imagine
    // PORTA - linii matrice LED-uri (active pe 1)
    // PORTC - coloane matrice LED-uri (active pe 0)
    // Parcurge matricea de LED-uri rand cu rand
    for (i = 0; i < 8; i++) {
        // Citire stare butoane (buton1 - dreapta, buton3 - stanga)
        char button1 = PIND & (1 << 0);
        char button2 = PIND & (1
        << 1); char button3 = PIND
        & (1 << 2);

        // 111 - Toate butoanele apasate
        if (button1 && button2 &&
            button3) PORTC = etti[i];
        else if (button1 && button2 &&
            !button3) PORTC = house[i];
        else if (button1 && !button2 &&
            button3) PORTC = umbrella[i];
        else if (button1 && !button2 &&
            !button3) PORTC = surprised[i];
        else if (!button1 && button2 &&

```

```
button3) PORTC = skull[i];
```

```
else if (!button1 && button2 && !button3)
```

```
    PORTC = face[i];  
else if (!button1 && !button2 &&  
    button3) PORTC = heart[i];  
// 000 - Niciun buton apasat  
else if (!button1 && !button2 &&  
    !button3) PORTC = sad[i];  
// PORTA - se shifteaza cu 1 (se inmulteste cu 2) pentru  
//    a prelucra urmatoarea linie  
PORTA = (1 << i);  
delay_ms(10);  
}  
}  
}
```