

Lucrarea nr. 5

Utilizarea unei tastaturi

Obiective:

- Studenții vor crea o tastatură 3x3 din butoane push-up , pe care o vor conecta la portul Analog A0. Butoanelor vor fi conectate printr-o serie de divizoare de tensiune. Fiecare buton va avea o semnificație.

Hardware necesar:

- Arduino UNO R3,
- Keypad (3x4, 4x4, butoane push-up)
- Rezistențe;

1. Teorie (preluat de la [1])

În multe proiecte este nevoie de o tastatură numerică, keypad. Acestea sunt o matrice de butoane, precum tabla de șah, fiecare buton fiind identificat de o pereche de numere, linia respectiv coloana. Pentru o matrice 4x4 avem nevoie de 8 pini digitali, iar pentru o matrice 3x4 avem nevoie de 7 pini digitali. Dacă în proiectul nostru numărul de pini digitali disponibili este limitat, atunci putem conecta tastatura la 1 pin analogic prin intermediul unor rezistoare conectate adecvat, și care furnizează o altă tensiune la apăsarea fiecărui buton.

Principiul funcțional este dat de divizorul rezistiv de tensiune, prezentat în figura 1:

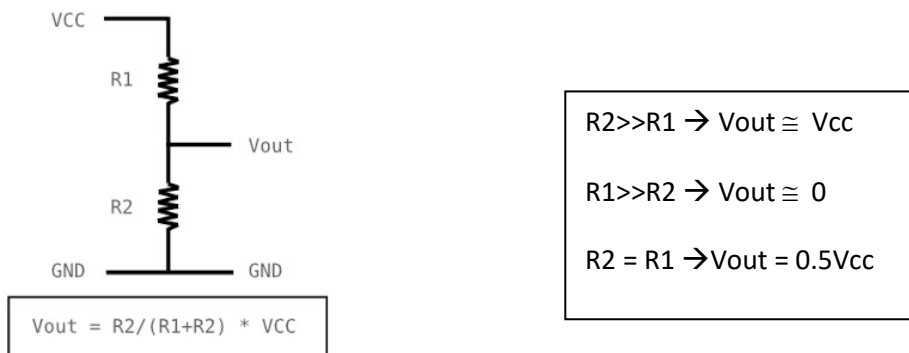


Figura 1: Divizorul rezistiv de tensiune.

Tensiunea de ieșire (V_{out}) este o anumită fracție a tensiunii de intrare (V_{CC}) care este definită prin raportul dintre cele două rezistențe R_1 și R_2 . Dacă R_2 este mult mai mare decât R_1 , atunci V_{out} va fi aproape de V_{CC} . Dacă R_2 este mult mai mic decât R_1 , V_{out} va fi aproape de zero. Și dacă $R_1 = R_2$, atunci $V_{out} = V_{CC} / 2$.

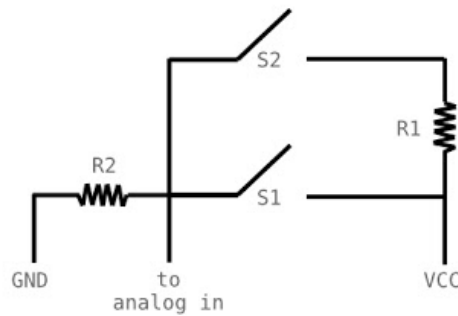
Exemplu1: Circuit cu 2 Butoane

Figura 2: Circuit cu două butoane

Pinul din mijloc este conectat la o intrare analogică Arduino.

Dacă :

`val = analogRead (A1);`

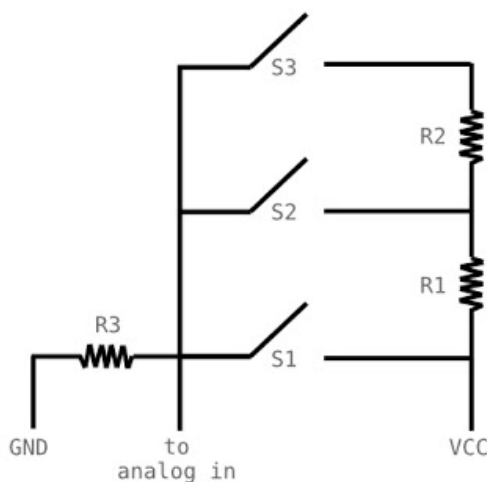
- S1 și S2 deschise, \rightarrow `val = 0`; (deoarece intrarea este conectată la GND, prin R2).
- S1 închis și S2 deschis, `val = 1023` (intrarea este conectată la Vcc);
- S2 închis și S1 deschis, `val = 512` (la intrare se aplică tensiunea $V_{cc} / 2$),

OBS: Acestea sunt valorile teoretice, în practică ele variază în timp și cu temperatura (rezistența electrică scade cu creșterea temperaturii).

În program, vom verifica dacă :

- `val < 250` (nici un buton apăsat);
- `250 < val < 750` (S2 apăsat);
- `Val > 750` (S1 apăsat);

Acest circuit nu poate verifica dacă este apăsat doar S1 sau S1 și S2.

Exemplu 2: Circuit cu 3 butoane

$R1 = R2 = R3$ (1K ... 3K)

- `val = 0`, nici un buton apăsat;
- `val = 314`, S3 apăsat;
- `val = 518`, S2 apăsat;
- `val = 1023`, S1 apăsat.

Nu se poate identifica apăsarea a 2 butoane simultan.

Figura 3: Circuit cu 3 butoane.

Exemplu 3: Circuit pentru keypad:

Pentru o matrice unei tastaturi 3x4 se poate folosi schema din figura de mai jos, ce conține un set de rezistențe 1k și 3k. Frațiunile tensiunii de alimentare VCC, așa cum se vede la pinul de ieșire („analogic”), sunt listate în tabel. De asemenea, este posibil să adăugați butoane suplimentare, așa cum este indicat de piesa de sub linia punctată.

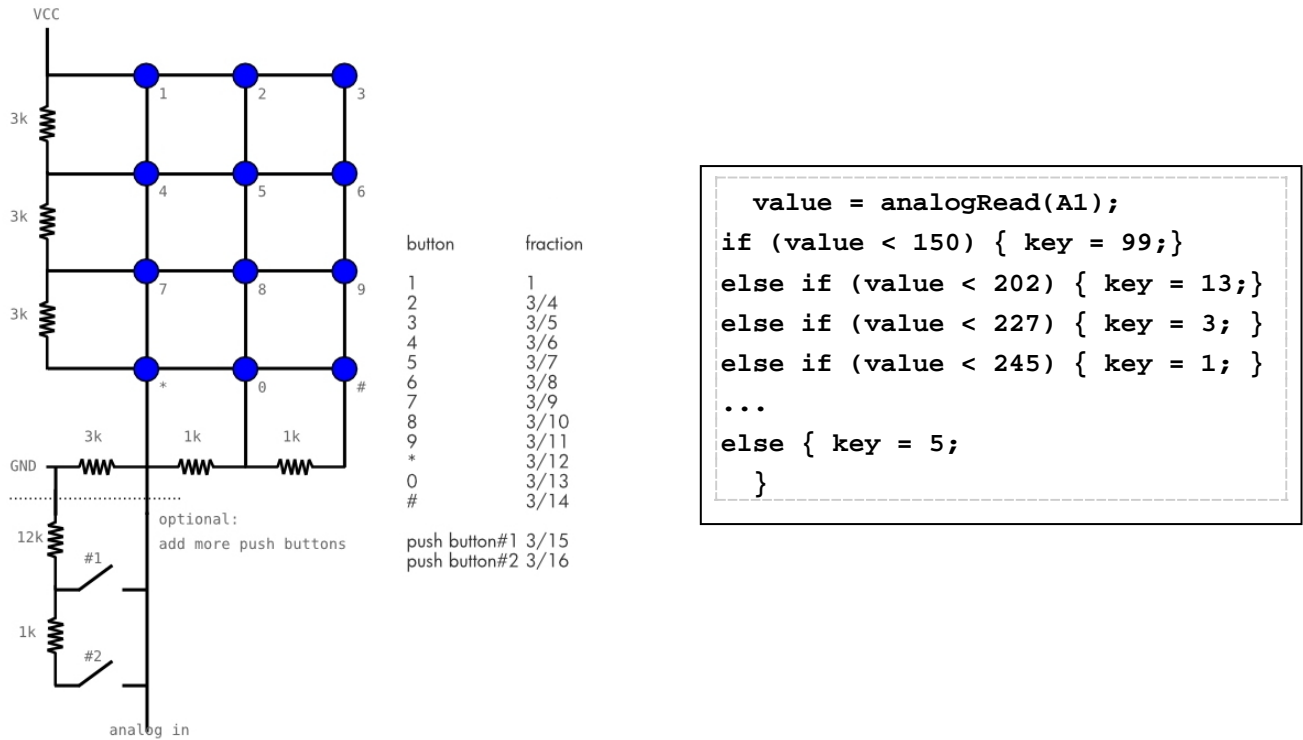


Figura 4: Keypad 3x4.

Pentru matrici de tastaturi 4x4 se pot construi mai multe scheme de conexiuni, două dintre ele fiind date mai jos.

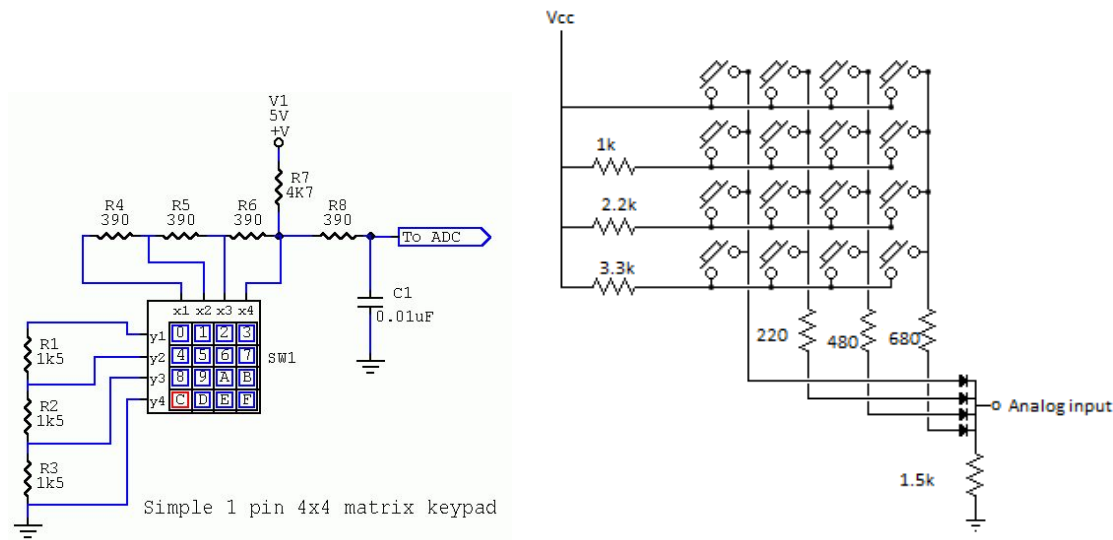


Figura 5: Scheme electrice pentru Keypad 4x4.

Mersul lucrării

Pasul 1:

Realizați în Proteus, schema de mai jos (evident, cu Arduino Uno și un terminal). Scrieți codul aferent și afișați pe serial tasta apăsată.

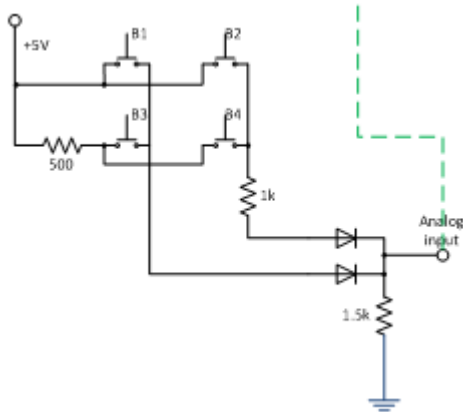


Figura 6: Schema electrică pentru Keypad 2x2.

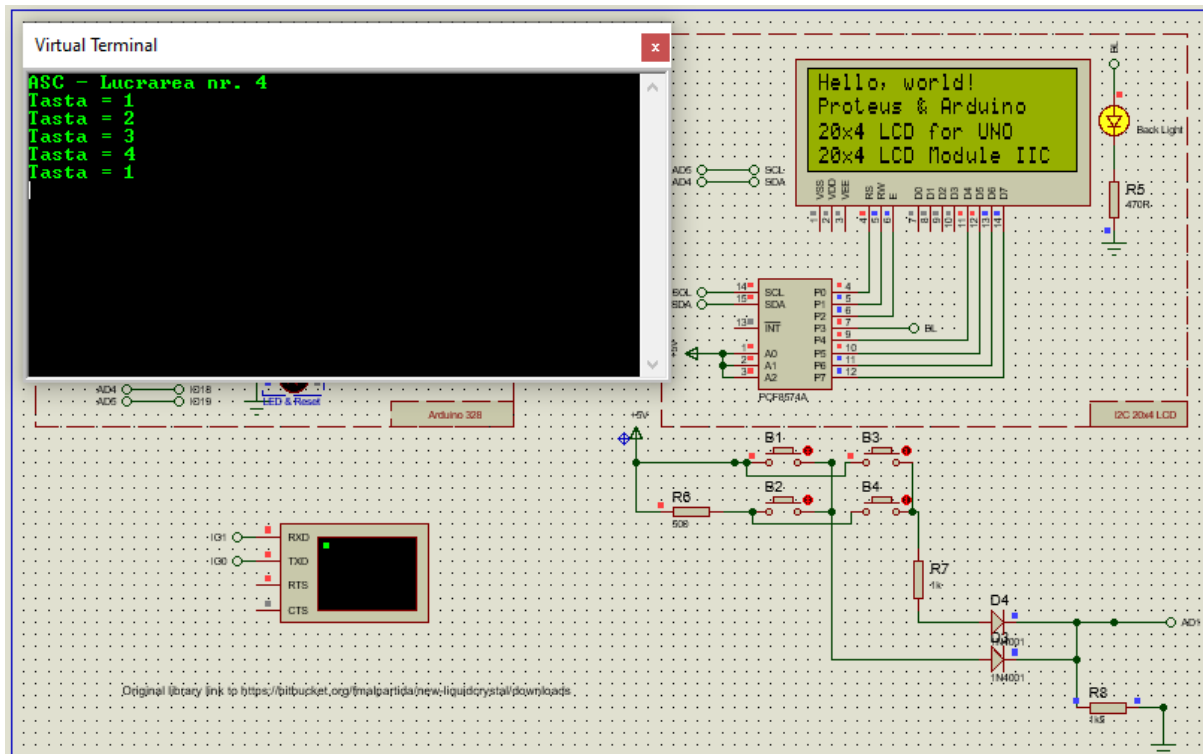


Figura 7: Schema realizată în Proteus cu keypad 2x2.

```
#define WAIT_TO_START 1 // Wait for key pressed in function keyPress
const int keyPin = A1;
int value = 0;
void setup()
{
  Serial.begin(9600);
  Serial.println("ASC - Lucrarea nr. 4");
}
```

```
int keyPress()
{
#if WAIT_TO_START
  while (analogRead(A1)<25);
  int key=99;
  int val = analogRead(keyPin);
  delay(100);
  if(val > 25)
  { if(val<500&&val>400){ key = 4;}
    else if(val<600&&val>500) {key = 3;}
    else if(val<700&&val>600) {key = 2;}
    else key = 1;
  }
#endif //WAIT_TO_START
return key;
}

void loop()
{
  value = keyPress();
  Serial.print("Tasta = ");
  Serial.println(value);
  delay(2);
}
```

Dacă nu apăsăm nici o tastă, funcția va întoarce, la infinit valoarea implicită pentru key.

Dacă nu apăsăm nici o tastă, A1 este pus la GND, deci citește o valoare apropiată de zero. Și atâta timp cât este 0, vom relua bucla while în așteptarea apăsării unei taste.

Pasul 2: Extindeți aplicația pentru o tastatură 4x4

Pasul 5:

Combinați cele două exemple de mai sus, astfel încât:

1. pe display-ul LCD i2c din laboratorul anterior, să se afișeze tasta apăsată;
2. adăugați un LED roșu și unul verde. Creați un program tip cifru/seif. Dacă codul pin (1234) este introdus, se stinge ledul roșu și se aprinde ledul verde. Pe LCD se va afișa un mesaj adecvat. Codul pin nu va fi afișat pe LCD și nici nu va fi transmis pe serial.

Bibliografie:

1. <http://markus-wobisch.blogspot.com/2019/01/saving-arduino-inputs-analog-readout-of.html>
2. <https://www.electroschematics.com/arduino-with-keypad/>
3. https://profs.info.uaic.ro/~arduino/index.php/Laboratorul_5