

Pulse Checker

Autor

Toma Elena-Bianca 331CB

Introducere

Pulse Checker este un dispozitiv de verificare a pulsului și poate fi folosit de oricine dorește să verifice care este starea lui în funcție de valoarea pulsului.

Descriere generală

- Dispozitivul preia prin intermediul unui keypad anumite date despre utilizator (ex: varstă,sex) și apoi măsoară pulsul cu ajutorul unui senzor.
- După măsurare, folosind datele adăugate anterior afișează un mesaj pe ecran din care să rezulte starea pacientului conform intervalului în care se situează valoarea măsurată împreună cu aceasta.
- În cazul în care valoarea este prea mare sau prea mică se va activa un buzzer care va emite un zgomot.
- Se va aprinde un led și în funcție de starea pacientului culoarea acestuia se va modifica.

Schema bloc:

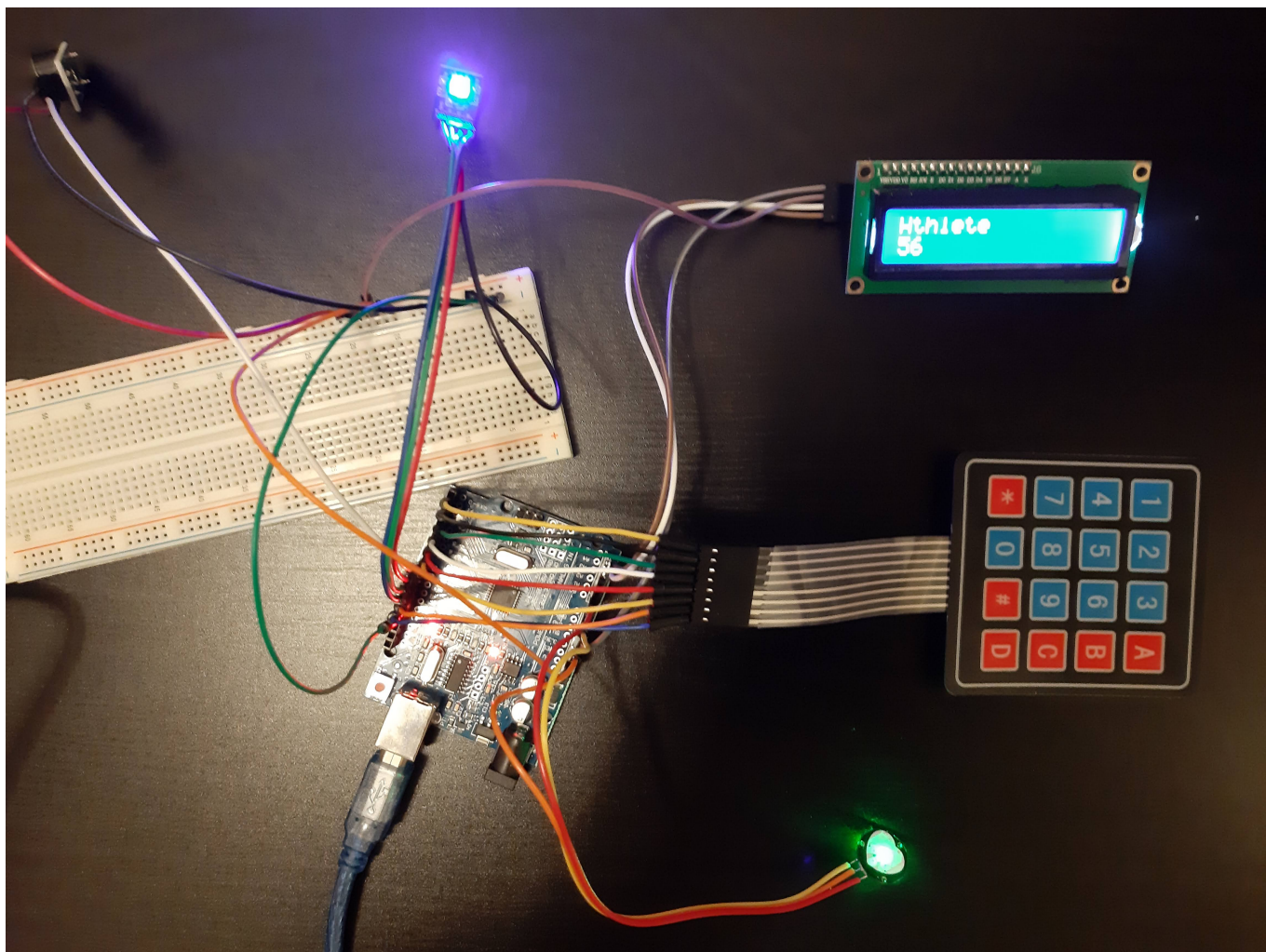


Hardware Design

Listă componente:

- Arduino Uno
- Senzor puls
- Buzzer
- Led
- Ecran LCD
- Keypad
- Buton
- Fire

Schema electrica



Software Design

Pentru dezvoltarea proiectului, am folosit urmatoarele biblioteci:

- [Wire.h](#) si [LiquidCrystal_I2C.h](#) pentru comunicarea cu ecranul LCD folosind protocolul I2C
- [Keypad.h](#) pentru a citi informatiile primite de la Keypad
- [PulseSensorPlayground.h](#) pentru a putea interpreta semnalele primite de la senzorul de puls.

Ideea principala a proiectului consta in preluarea unor informatii date de catre utilizator prin intermediul keypad-ului si in functie de cele primite trece in una dintre cele 6 stari interne ale programului. Primele 5 stari acumuleaza pe rand informatii despre utilizator, iar in ultima stare se asteapta citirea datelor de la senzorul de puls si se afiseaza informatiile corespunzatoare.

Prelucrarea semnalelor de la senzorul de puls nu se face in mod continuu, ci se realizeaza doar in momentul in care senzorul intalneste o schimbare (se detecteaza prima bataie a inimii).

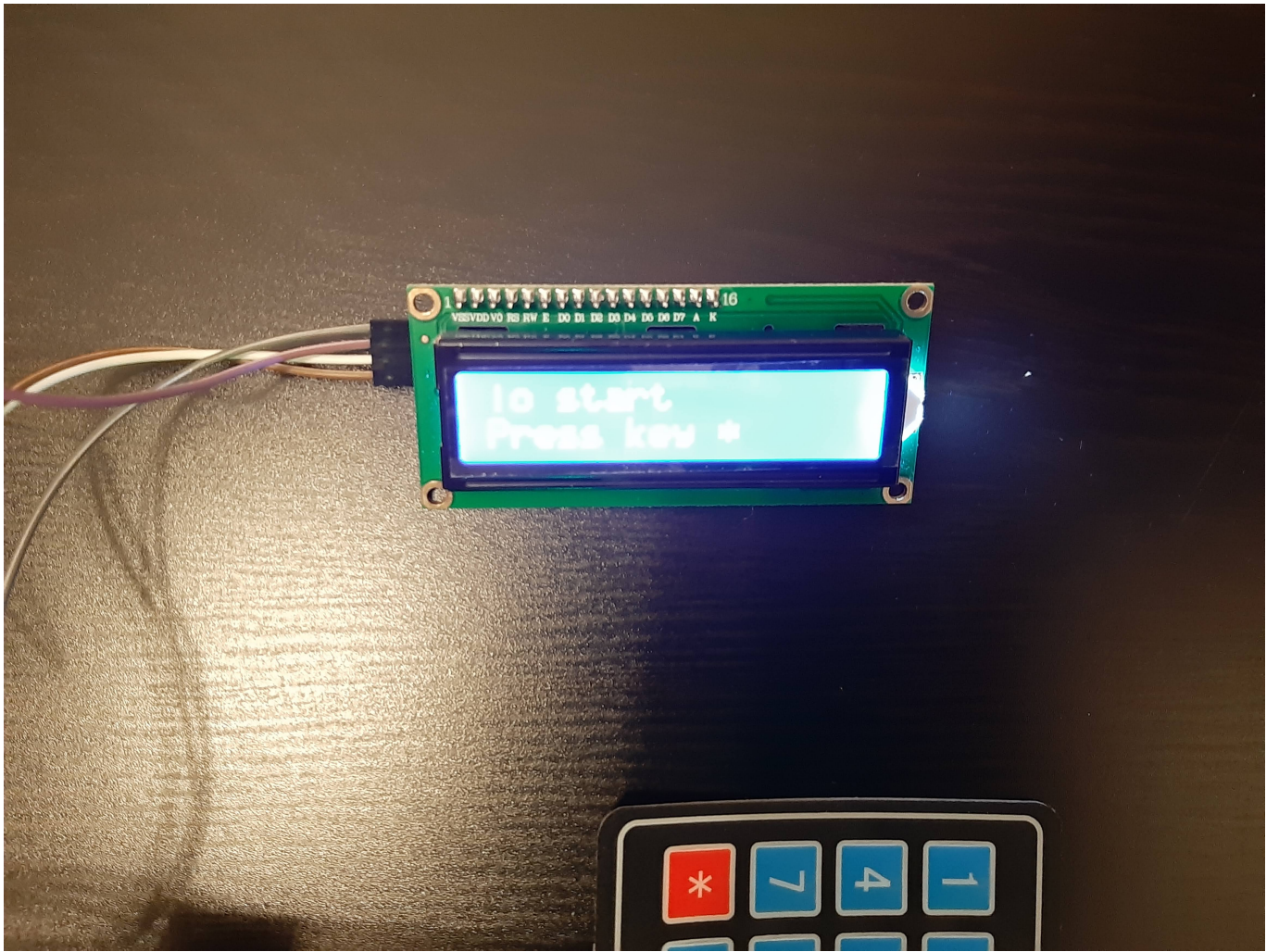
Pentru a determina starea pacientului se compara valoarea obtinuta de la senzor cu valorile din tabelul de mai jos, tinandu-se cont de varsta si genul acestuia. Pentru a realiza acest lucru am creat

cate o functie pentru fiecare coloana a tabelului (ex: barbat intre 26 si 35 de ani, femeie intre 56 si 65 de ani etc.) in care se stabileste pe ce linie se incadreaza valoarea citita, mai exact ce stare si culoare i se potriveste din tabel. Culoarea corespunzatoare starii respective este afisata cu ajutorului ledului RGB.

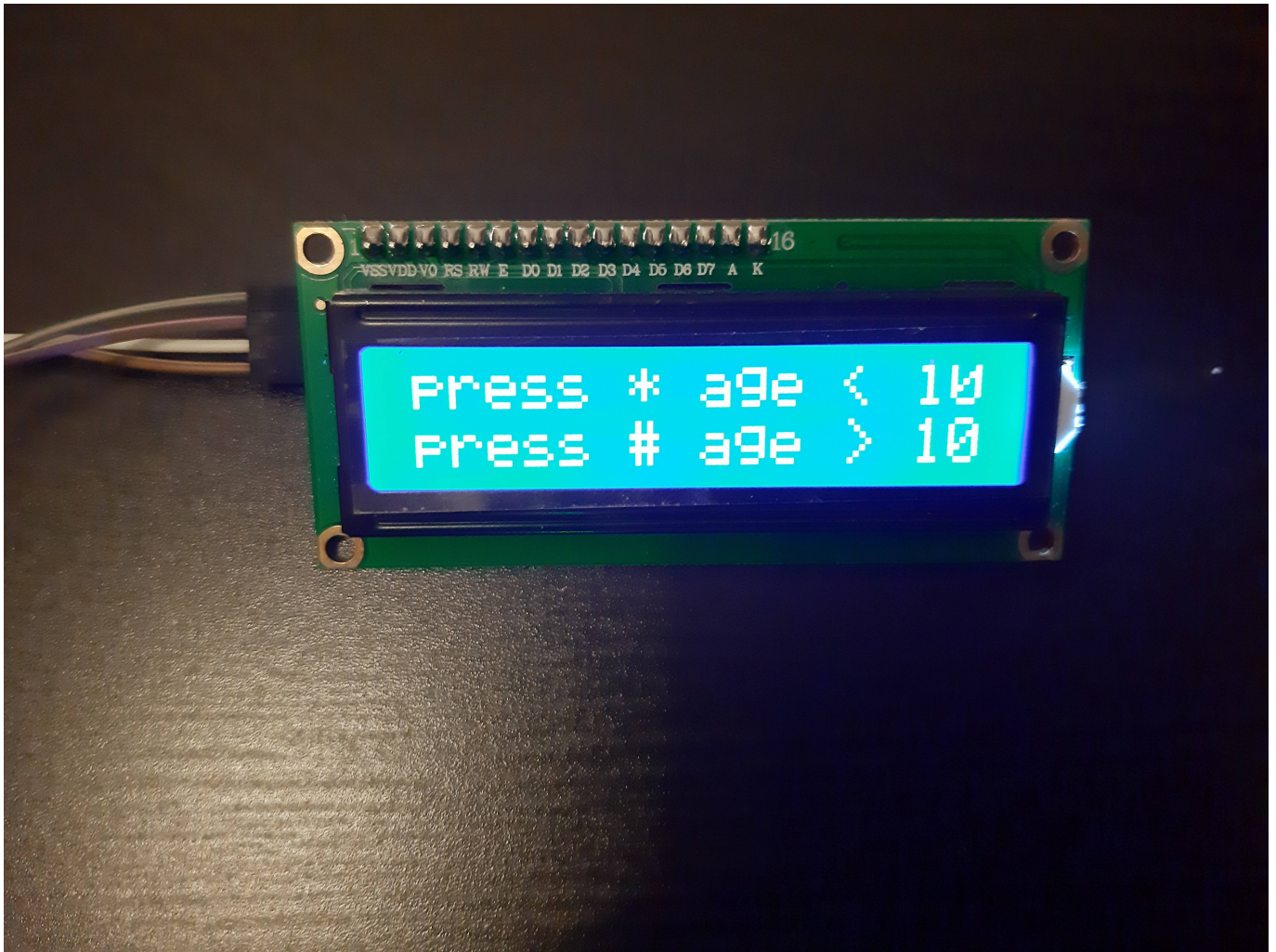
Resting Hear Rate Chart						
Men (beats per minute)						
Age	18 - 25	26 - 35	36 - 45	46 - 55	56 - 65	65 +
Athlete	49 - 55	49 - 54	50 - 56	50 - 57	51 - 56	50 - 55
Excellent	56 - 61	55 - 61	57 - 62	58 - 63	57 - 61	56 - 61
Great	62 - 65	62 - 65	63 - 66	64 - 67	62 - 67	62 - 65
Good	66 - 69	66 - 70	67 - 70	68 - 71	68 - 71	66 - 69
Average	70 - 73	71 - 74	71 - 75	72 - 76	72 - 75	70 - 73
Below Average	74 - 81	75 - 81	76 - 82	77 - 83	76 - 81	74 - 79
Poor	82 +	82 +	83 +	84 +	82 +	80 +
Women (beats per minute)						
Age	18 - 25	26 - 35	36 - 45	46 - 55	56 - 65	65 +
Athlete	54 - 60	54 - 59	54 - 59	54 - 60	54 - 59	54 - 59
Excellent	61 - 65	60 - 64	60 - 64	61 - 65	60 - 64	60 - 64
Great	66 - 69	65 - 68	65 - 69	66 - 69	65 - 68	65 - 68
Good	70 - 73	69 - 72	70 - 73	70 - 73	69 - 73	69 - 72
Average	74 - 78	73 - 76	74 - 78	74 - 77	74 - 77	73 - 76
Below Average	79 - 84	77 - 82	79 - 84	78 - 83	78 - 83	77 - 84
Poor	85 +	83 +	85 +	84 +	84 +	85 +

Rezultate obtinute

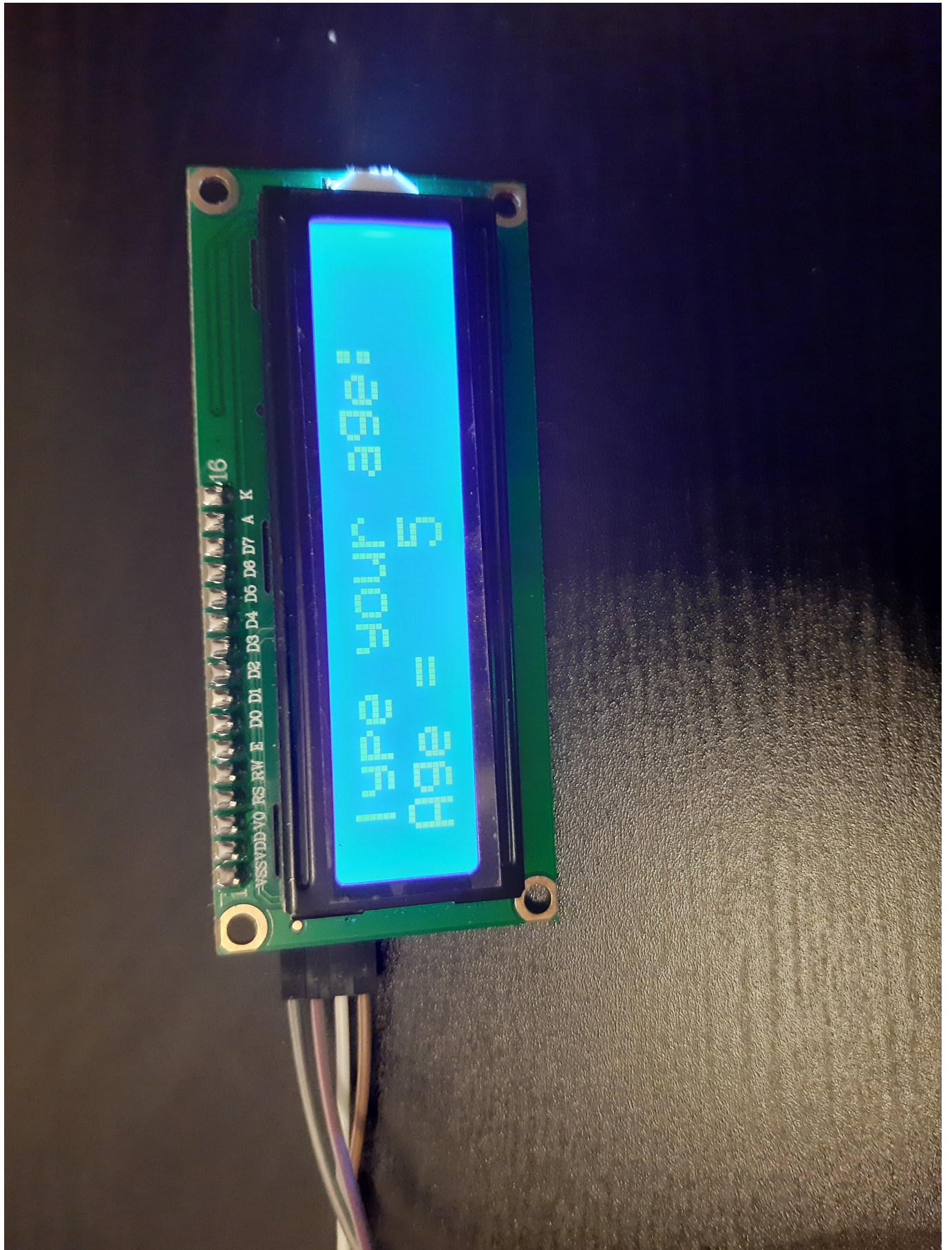
- Ecran de start



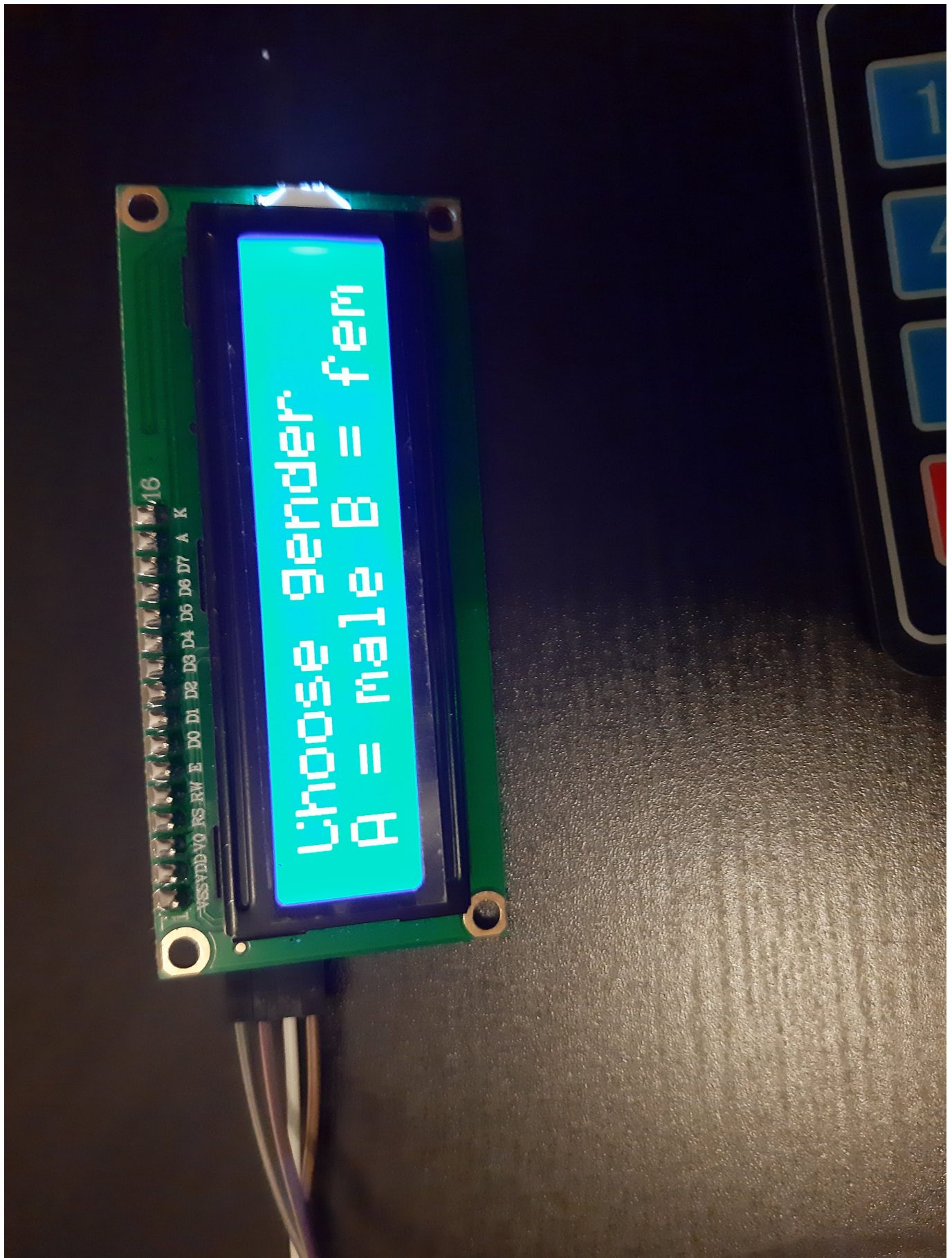
- Alegere interval varsta



- Introducere varsta

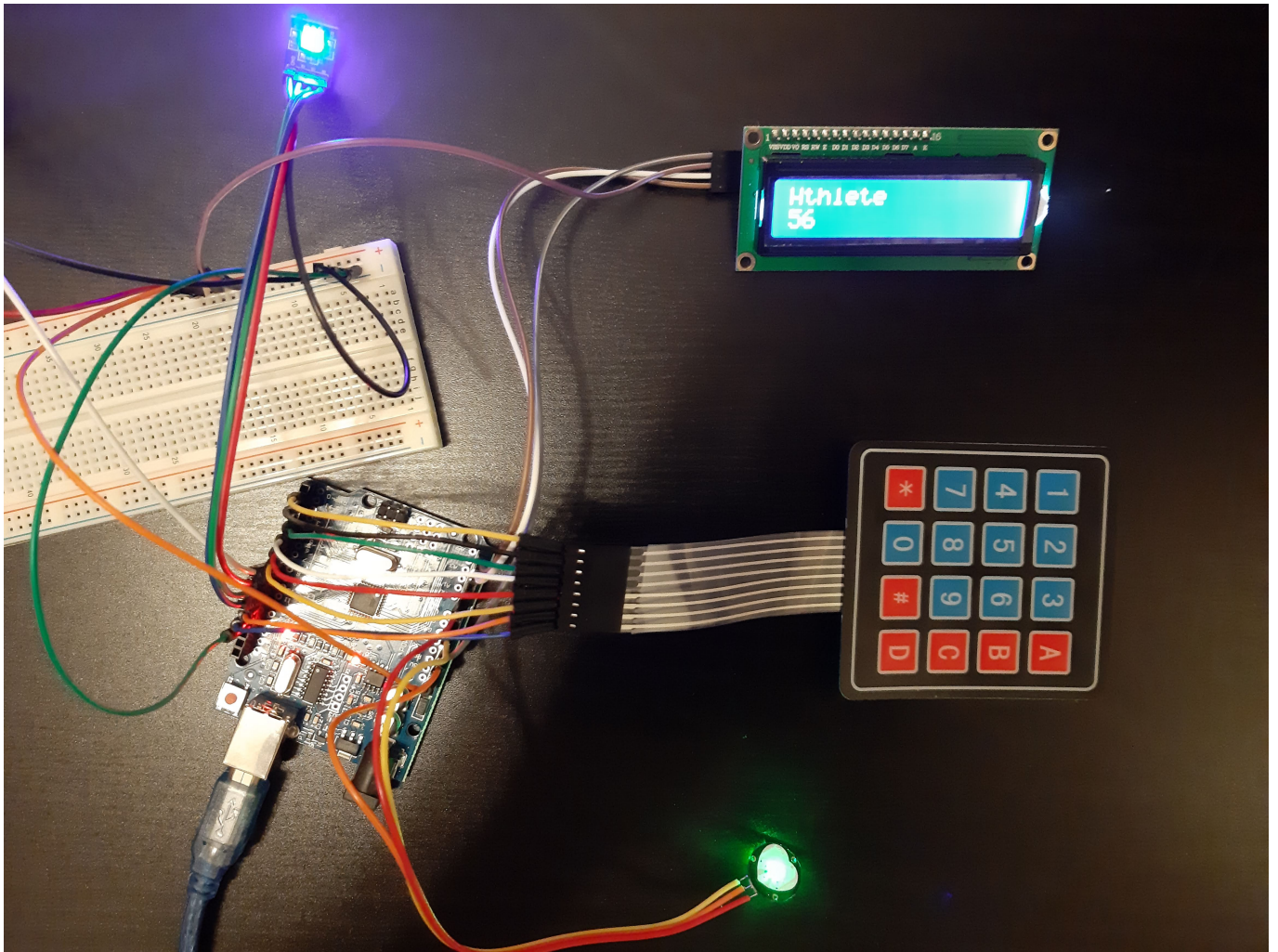


- Introducere gen

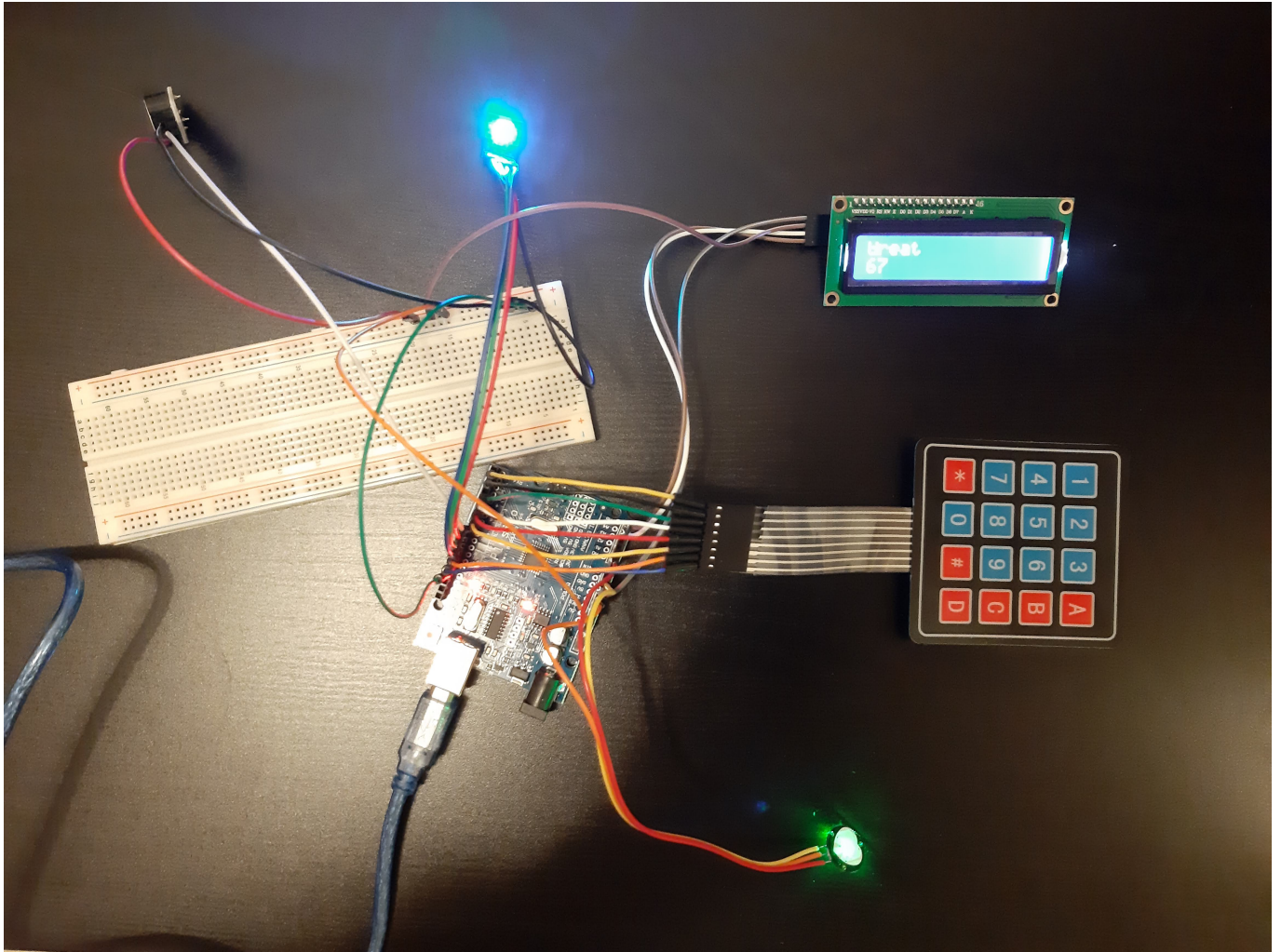


Exemple afisare stari in functie de valoarea masurata

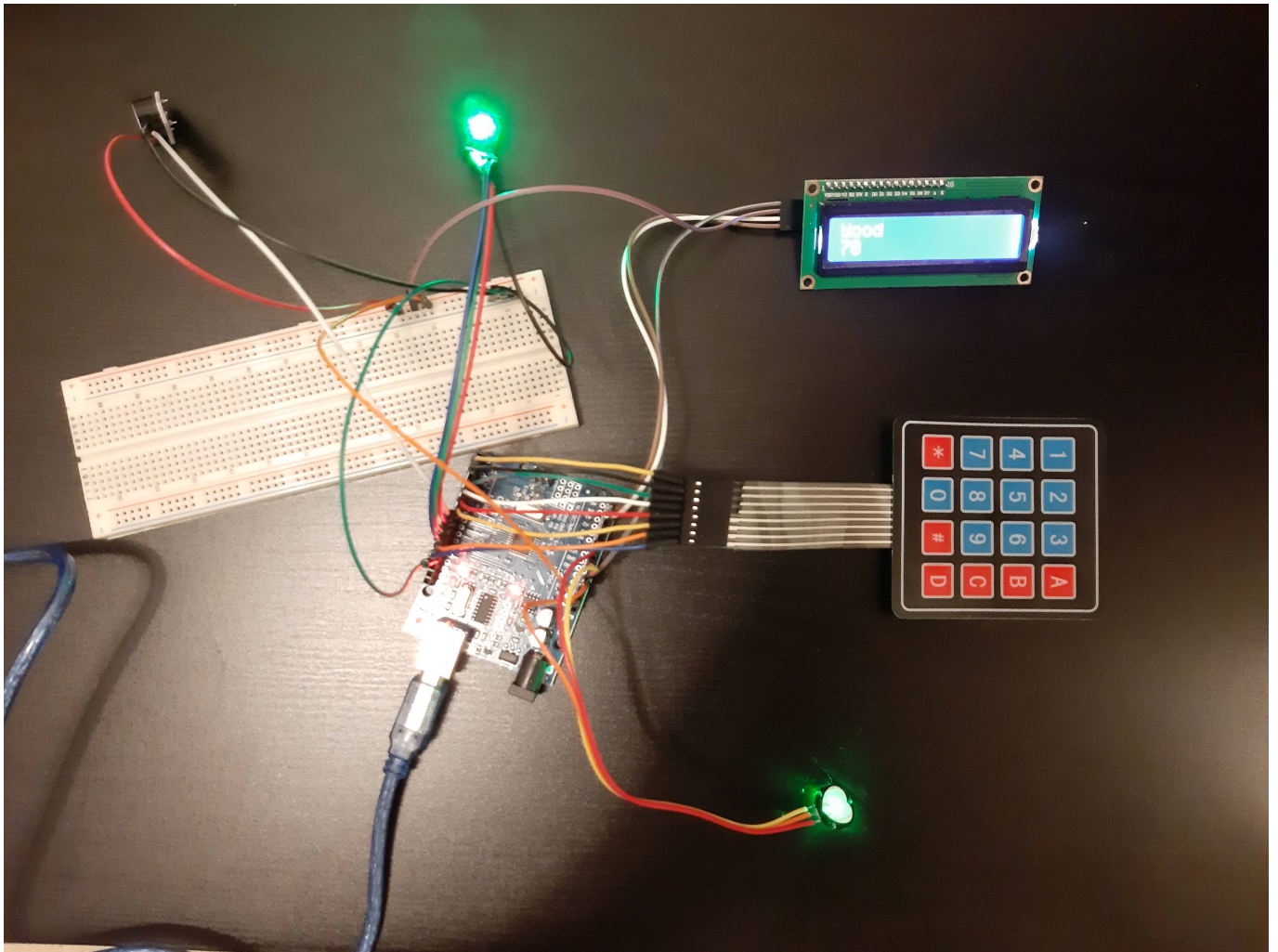
- *Athlete* - valoarea ideala a pulsului



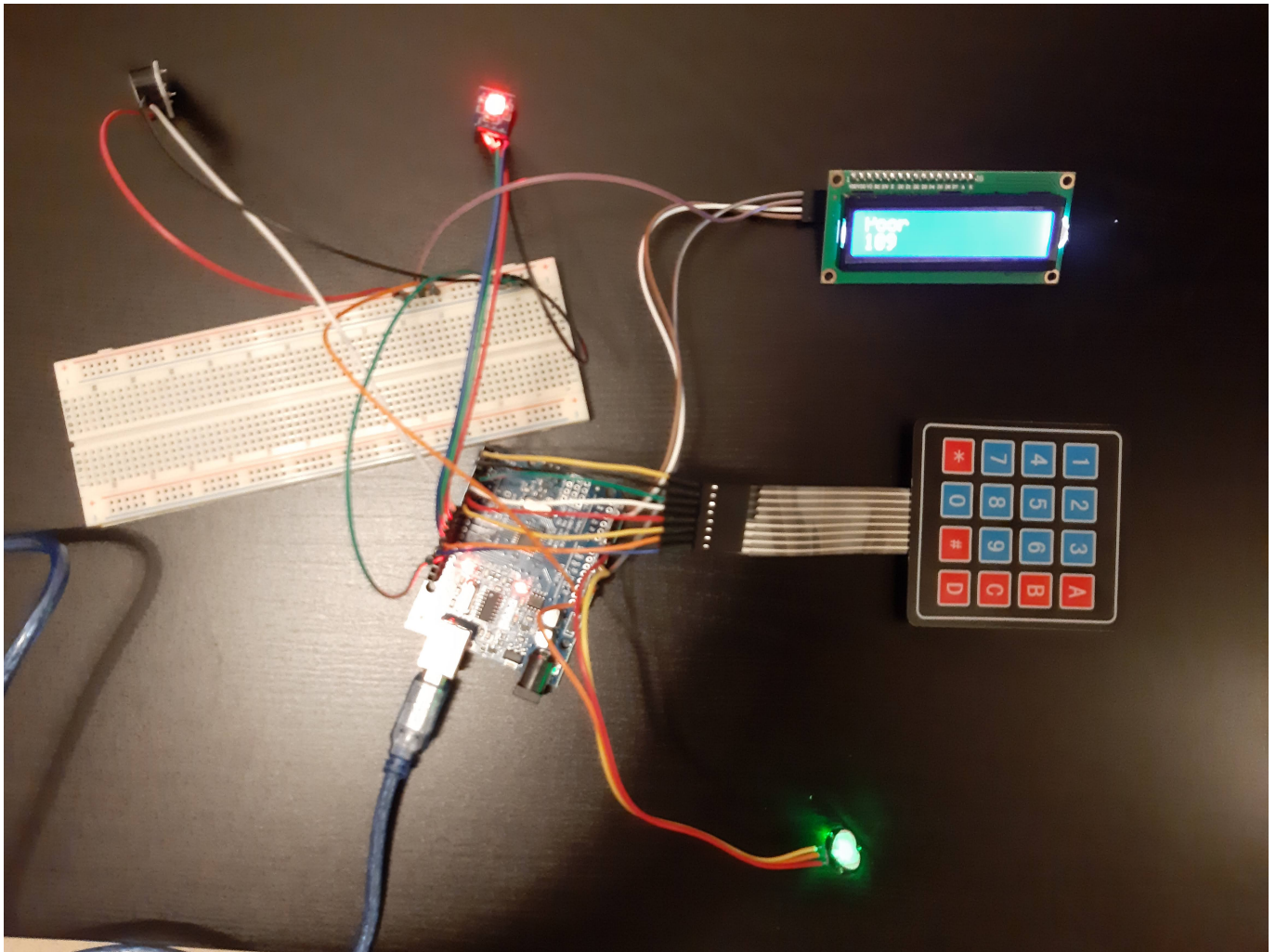
- Great - valoare foarte buna a pulsului



- *Good* - valoare buna a pulsului



- *Poor* - valoare nepotrivita a pulsului (prea mare)



DEMO-ul pentru proiect se gaseste [aici](#)

Concluzii

Intreaga implementare a proiectului a fost interesanta si cred ca m-a ajutat sa inteleg mai bine cum sa scriu cod pentru Arduino. Una dintre problemele intampinate pe parcurs a fost faptul ca precizia senzorului nu este foarte buna, avand destul de des erori. Folosind un senzor mai bun, acest proiect poate fi de ajutor multor oameni, deoarece le poate spune starea lor din punct de vedere a valorii pulsului prin afisarea pe ecran sau prin culoarea ledului si totodata ii poate avertiza prin semnale sonore in cazul in care valorile pulsului nu se afla in parametrii normali.

Download

Arhiva proiectului poate fi descarcata de [aici](#).

Jurnal

- 25 aprilie - Alegere tema proiect
- 25 aprilie - Creare pagina de wiki
- 29 aprilie - Comandare componente
- 10 mai - testare componente individuale
- 17-24 mai - implementare proiect
- 31 mai - scriere documentatie finala pentru proiect

Bibliografie/Resurse

PDF

- Pentru a intelege modul de functionare a componentelor individuale m-am folosit de exemplele puse de cei de la [Optimus Digital](#) si de exemplele din Arduino IDE si din cadrul bibliotecilor amintite mai sus.
- [Valori puls pentru fiecare varsta](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/alazar/pulse-checker>



Last update: **2021/05/31 17:14**