

# არდუინოს შემსწავლელი სახელმძღვანელო

20 STEAM პროექტი

## II ნაწილი

სსიპ მასწავლებელთა პროფესიული განვითარების  
ეროვნული ცენტრი

სახელმძღვანელო მოამზადა  
მაია ნაპაშვიძემ

კონსულტანტები  
კახა ჟღენტი  
ნუგზარ მოსულიშვილი

დიზაინი  
ბესიკ დანელია

# არღუიწოს ზემსწაპლელი სახელმძღვანელოს I და II ნაწილი ელექტრონული ვერსია, საჭირო ბიბლიოთეკები და პროექტების კოდები

ვებგვერდი:

<https://chkhirkedela2019.wixsite.com/mysite>

გვერდი ARDUINO UNO

ასევე, შეგიძლიათ, ვებგვერდის გვერდის ავლით, იხილოთ:

ბიბლიოთეკები:

<https://bit.ly/413fjht>

სახელმძღვანელოს I ნაწილის პროექტების კოდები:

<https://bit.ly/3NoeyYv>

სახელმძღვანელოს II ნაწილის პროექტების კოდები:

<https://bit.ly/4i5gGIY>



# არდუინოს დამაჯავიტი ბიბლიოთეკების ინსტალაცია

საზოგადოდ, ბიბლიოთეკების ორი სახე არსებობს: სტანდარტული და დამატებითი. Arduino IDE-ში არის სტანდარტული ბიბლიოთეკები, რომლებიც ხშირად გამოიყენება და შედის Arduino IDE-ს პროგრამულ უზრუნველყოფაში. ეს ბიბლიოთეკები შეიცავენ ფუნქციებს, ისინი სტანდარტულ ინტერფეისებთან სამუშაოდ გამოიყენება, მაგალითად, თხევადკრისტალური ეკრანი, სერვოდრავები და ა.შ. ინტერნეტში დიდი რაოდენობითაა დამატებითი ბიბლიოთეკები სხვადასხვა მოწყობილობისთვის, საკმაოდ მოხერხებული ფუნქციებით. ძირითადად, ეს ბიბლიოთეკები მოთავსებულია **Arduino Playground**, **Github** და **Google Code** რესურსებზე.

ბიბლიოთეკა პროგრამირების გარემოში შეგიძლიათ ნახოთ მენიუ Sketch-ში. მენიუ Sketch-ზე დაწკაპუნებისას ჩამოიშლება პატარა ფანჯარა, სადაც ნახავთ Include Library-ს, რომელიც გიჩვენებთ ხელმისაწვდომ ბიბლიოთეკებს. მაგალითად, ბიბლიოთეკა LiquidCrystal საშუალებას მოგცემთ იმუშაოთ LCD ეკრანთან.

Library არის კოდების კრებული, რომლებიც საჭიროებისამებრ შეიძლება ჩართული იყოს თქვენს sketch-ში, რომ გაფართოვდეს, გაძლიერდეს თქვენი პროექტის შესაძლებლობები. ამით დაზოგავთ დროს და შეძლებთ თქვენ მიერ ან სხვისი შექმნილი კოდის გამოყენებას სხვადასხვა პროექტებში.

ჩვენ მიერ შემოთავაზებული პროექტების შესაქმნელად დაგჭირდებათ შემდეგი ბიბლიოთეკების იმპორტირება: *RFID, Tone, Pitches, Keypad, Password, Ultrasonic, NewPing, IRRemote, LiquidCrystal\_12C, Wire* და *DHT*.

## ყუჩაღლება!

თქვენთვის სამუშაოს გამართივების მიზნით, გთავაზობთ უკვე ჩამოტვირთულ ბიბლიოთეკებს ჩვენს ვებგვერდზე: <https://chkhirkedela2019.wixsite.com/mysite>, გვერდი ARDUINO UNO.

ბიბლიოთეკების ჩამოტვირთვის შემდეგ საჭიროა მათი ინსტალაცია Arduino UNO 1.8.19 ვერსიაში.

## ბიბლიოთეკის ინსტალაცია ARDUINO UNO 1.8.19-ის ვერსიაში

დამატებითი ბიბლიოთეკების დასაყენებლად მნიშვნელოვანია საჭირო ბიბლიოთეკა შესაბამის საქალაქოდეში დავაყენოთ, წინააღმდეგ შემთხვევაში, ბიბლიოთეკას კომპილატორი ვერ იპოვის.

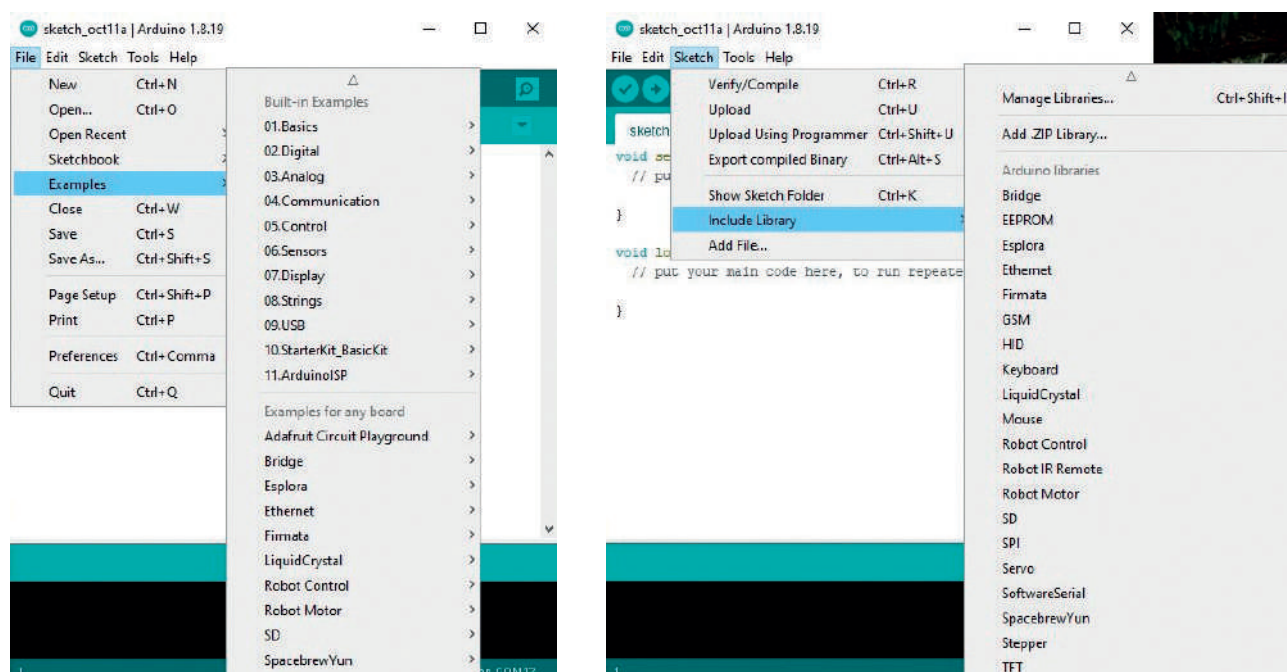
Windows-ის ოპერაციული სისტემისთვის ეს საქალაქდება Arduino და ის მოთავსებულია Documents საქალაქდეში. სტანდარტული ბიბლიოთეკები Arduino IDE-ს ინსტალაციის დროს თავსდება მისა-მართზე: C:\Program Files\Arduino\libraries. საქალაქდის სახელი იგივეა MAC სისტემისთვის, Linux-ში საქალაქდეს ეწოდება Scetchbook და მოთავსებულია /home/-ში.

როგორც წესი, ჩამოტვირთული ბიბლიოთეკა არის დაარქივებული (Zip) ფაილი. დამატებითი ბიბლიოთეკის დასაყენებლად არსებობს ორი გზა:

1. ბიბლიოთეკის გადმოწერის შემდეგ უნდა გაიაროთ შემდეგი ეტაპები:

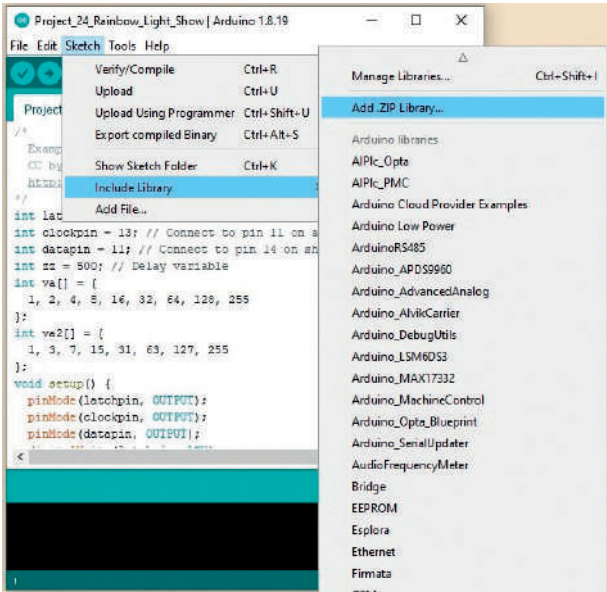
- ა. ბიბლიოთეკის დაარქივებული ფაილი განარქივით
- ბ. განარქივებული საქალაქდე მოათავსეთ:
  - ა) Documents\Arduino\Libraries – Windows-ში
  - ბ) Documents/Arduino/libraries – OS X-ში
  - გ) sketchbook/libraries – Linux-ში

ზოგიერთ შემთხვევაში, გადმოწერილი საქალაქდის სახელწოდება ტირებს შეიცავს, Arduino IDE ვერ ცნობს საქალაქდეებს, რომელთა სახელწოდებაშიც ტირებია. ასე რომ, ტირეს ნაცვლად შეგ-ვიძლია ჩავსვათ ქვედა ტირე. გავხსნათ Arduino IDE და შევამოწმოთ, გაჩნდა თუ არა ახლად დამა-ტებული ბიბლიოთეკა ბიბლიოთეკების სიაში. ეს ხდება მენიუს პუნქტიდან **File->Examples ან Sketch/Include Library** (იხ. სურათი 1). ჩავტვირთოთ მაგალითი და გავუშვათ შემოწმებაზე.



სურათი 1. სად ვეძებთ ახლად დამატებულ ბიბლიოთეკას

- 2. ბიბლიოთეკის გადმოწერის შემდეგ უნდა გაიაროთ შემდეგი ეტაპები:
  - ა. ჩამოტვირთეთ სასურველი ბიბლიოთეკა და ZIP ფაილი ატვირთეთ არდუინოს IDE ინტერფეისში (იხ. სურათი 2).

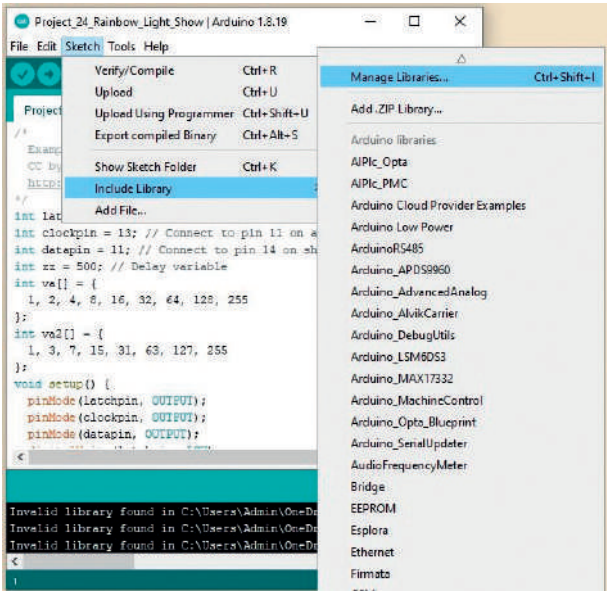


სურათი 2. დამატებითი ბიბლიოთეკის ჩატვირთვა IDE ინტერფეისში

თავიდან გაცხსნათ Arduino IDE და შევამოწმოთ, გაჩნდა თუ არა ახლად დამატებული ბიბლიოთეკა ბიბლიოთეკების სიაში. ეს ხდება მენიუს პუნქტიდან **File->Examples ან Sketch/Include Library** (იხ. სურათი 1).

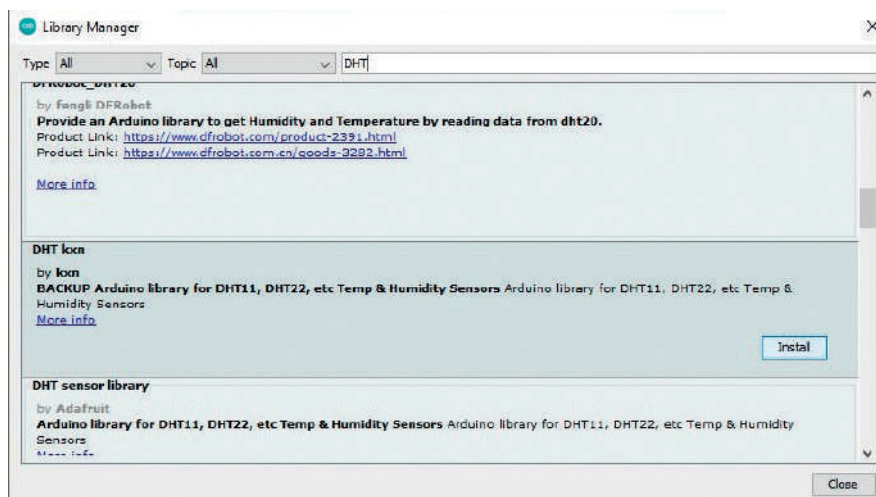
დამატებითი ბიბლიოთეკის ინსტალაცია

მას შემდეგ, რაც თქვენ პირველი ან მეორე მეთოდით (როგორც თქვენთვის არის მოსახერხებელი) ჩატვირთეთ ბიბლიოთეკა არდუინოს IDE ინტერფეისში, უნდა გაიაროთ ბოლო, მესამე ეტაპი. გახსენით არდუინოს IDE და მენიუს ველში მოძებნეთ **Sketch/Include Library/Manage Libraries** (იხ. სურათი 3).



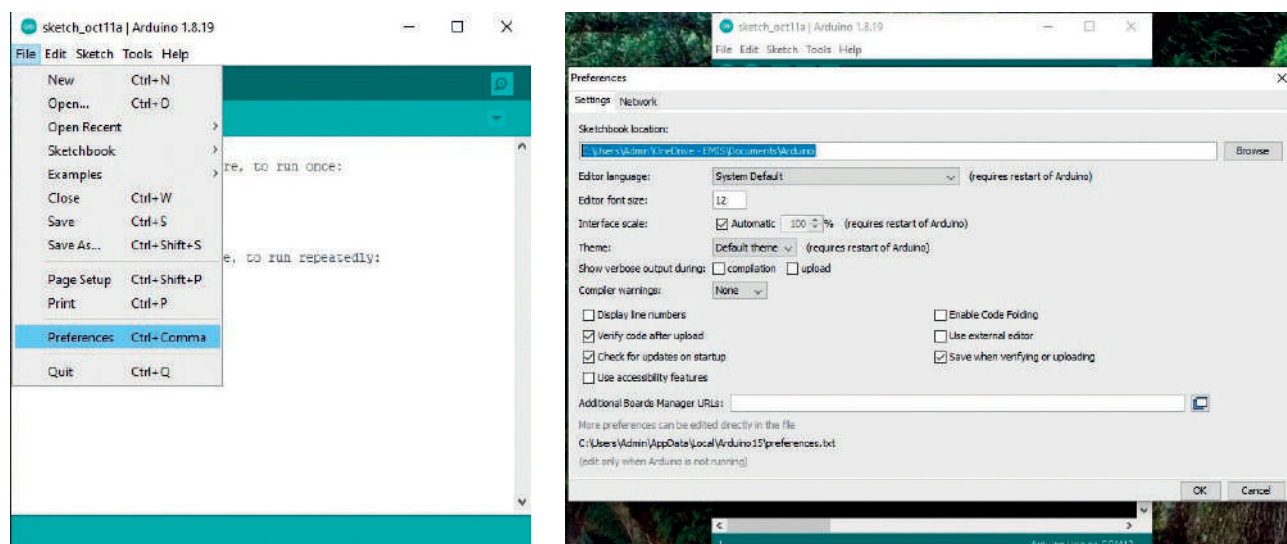
სურათი 3. ბიბლიოთეკის ინსტალაცია

და დაელოდეთ ბიბლიოთეკების დასაინსტალირებელი ფანჯრის გახსნას. როდესაც გაიხსნება ქვემოთ მოცემული ფანჯარა Library Manager, საძიებო ველში ჩაწერეთ თქვენ მიერ ჩატვირთული ბიბლიოთეკის სახელი და დაელოდეთ, ვიდრე არ გამოჩნდება ის მოცემულ ფანჯარაში. ბოლოს დააწკაპუნეთ ღილაკზე Instal (იხ. სურათი 4).



სურათი 4. ბიბლიოთეკების საბოლოო ინსტალაცია

და, ბოლოს, თქვენ შეგიძლიათ იხილოთ თქვენ მიერ დაინსტალირებული ბიბლიოთეკების მისამართები File/Preferences /Sketchbook location.



სურათი 5. დაინსტალირებული ბიბლიოთეკების მისამართი

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| ბიზლიოთეკები და კოდეზი                                          | 3   |
| არლინოს დამაგვითი ბიზლიოთეკის ინსტალაცია                        | 5   |
| I პროექტი: მოციმციმე ინდიკატორი და LED ნათურა                   | 10  |
| II პროექტი: მწკრივში მოციმციმე LED ნათურები                     | 13  |
| III პროექტი: ფერის ამომცნობი                                    | 16  |
| IV პროექტი: 8x8 RGB LED მატრიცა                                 | 19  |
| V პროექტი: RGB LED მატრიცა – მორბენალი სტრიქონი                 | 25  |
| VI პროექტი: მანძილის გაზომვა ულტრაბგერით                        | 28  |
| VII პროექტი: Servo ძრავების მართვა ჯოისტიკით                    | 32  |
| VIII პროექტი: Servo ძრავების მართვა დისტანციურად                | 37  |
| IX პროექტი: Nokia 5110 LCD ეკრანი                               | 42  |
| X პროექტი: ბიჯური ძრავა                                         | 46  |
| XI პროექტი: ტემპერატურის სენსორი LM35                           | 50  |
| XII პროექტი: ციფრული თერმომეტრი                                 | 55  |
| XIII პროექტი: მემბრანული კლავიატურა                             | 59  |
| XIV პროექტი: რეაქციის დროის მთვლეელი                            | 63  |
| XV პროექტი: დეკოდირება კლავიატურით                              | 69  |
| XVI პროექტი: ალკოტესტერი                                        | 74  |
| XVII პროექტი: უსადენო იდენტიფიკაციის სისტემა – RFID წამკითხველი | 79  |
| XVIII პროექტი: შუქდიოდის მართვა ხმოვანი ბრძანებით               | 85  |
| XIX პროექტი: GPS სპიდომეტრი                                     | 92  |
| XX პროექტი: 2DW ტკივანი მანქანა                                 | 97  |
| გამოყენებული ლიზარაგურა                                         | 104 |

# მოსიგშია ინლიკაგორი და LED ნათურა

## არდუინოს გამოყენება

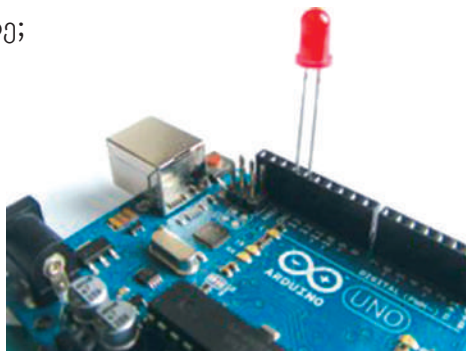
არდუინოსთან მუშაობის დაწყებამდე, გთავაზობთ რამდენიმე საჭირო ინფორმაციას, რომელიც აუცილებლად უნდა იცოდეთ. მოდით, შევხედოთ მოწყობილობას და პროგრამულ უზრუნველყოფას, რომელიც დაგჭირდებათ ამ სახელმძღვანელოსთვის. საჭიროა ჯერ შეამოწმოთ არდუინო შექდიოდის საშუალებით.

ჩვენ უკვე გავცვანით არდუინოს პროგრამირების გარემოს (IDE), მის ინტერფეისს (წიგნი „არდუინოს შემსწავლელი სახელმძღვანელო – 20 STEM პროექტი“ ნაწილი I). ახლა დროა გავაკეთოთ მისი პირველი, კლასიკური პროექტი – შექდიოდის ციმციმი. ეს საშუალებას მოგვცემს არა მარტო შევამოწმოთ, როგორ მუშაობს ჩვენი არდუინო, არამედ, გავცნოთ მარტივ კოდს/sketch-ს. Sketch – ეს არის მხოლოდ ინსტრუქციების სერია. არდუინოს ერთდროულად შეუძლია მხოლოდ ერთი კოდის დამახსოვრება. ამიტომ, მისი დამახსოვრების შემდეგ, რამდენჯერაც აამუშავებთ არდუინოს, შესრულდება ეს კოდი, ვიდრე ახალს არ ჩაწერთ.

ამჯერად ჩვენ მაგალითად ავიღებთ არდუინო IDE-ს მიერ შემოთავაზებულ საბაზო პროექტს Blink – შექდიოდის ციმციმი. ამისთვის დაგვჭირდება ერთი LED ნათურა, რომელიც ერთწამიანი ინტერვალით აინთება და ჩაქრება. **შექდიოდებს ესაჭიროებათ წინაღობა, სხვა შემთხვევაში ისინი გადაიწვება. ასეთი წინაღობა არდუინოში ჩაშენებულია მე-13 პინში.**

მაშ, ასე, გავცვეთ ნაბი-ნაბი:

1. LED ნათურის გრძელი ფეხი (დადებითი პოლუსი, +5V, იგივე ანოდი) მიუერთეთ მე-13 პინს, ხოლო მეორე ფეხი (უარყოფითი პოლუსი, იგივე კათოდი) – GND-ს (ე.წ. დამიწება), როგორც ეს ნაჩვენებია პირველ სურათზე;

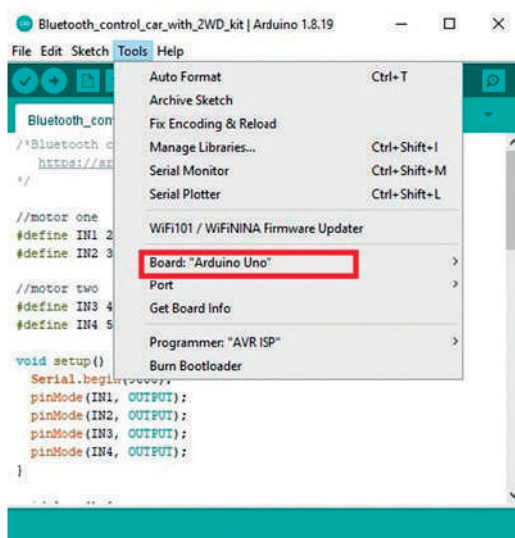


სურათი 1: პროექტი Blink

## გაფრთხილება!

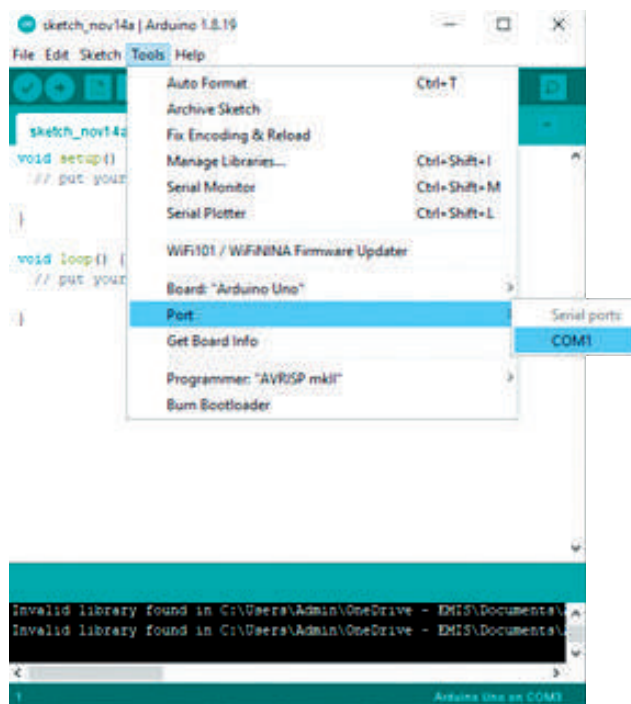
ყოველი პროექტის აწყობისა და დაშლის დროს არდუინო უნდა იყოს გამორთულ მდგომარეობაში.

- დაუკავშირეთ არდუინო USB კაბელით თქვენს კომპიუტერს;
- შეამოწმეთ, სწორად არის თუ არა შერჩეული არდუინოს ტიპი (იხ. სურათი 2);

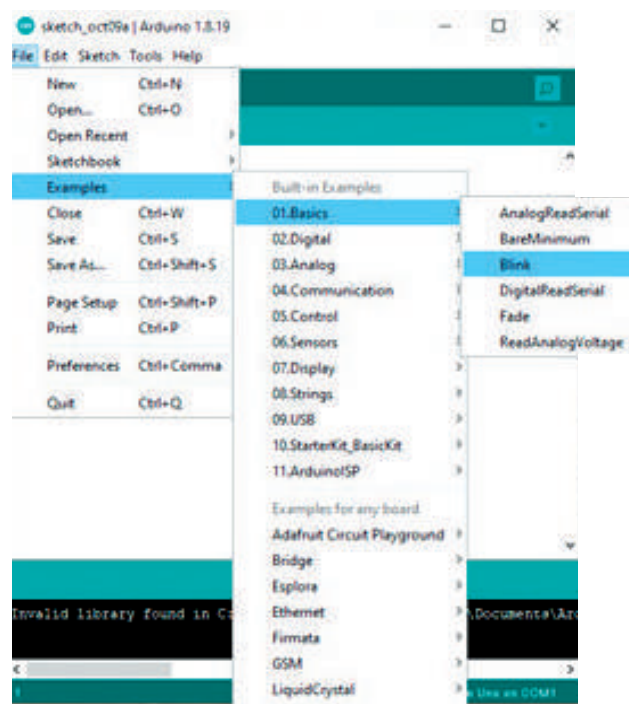


სურათი 2. არდუინოს ტიპის შერჩევა

- შეამოწმეთ, დაკავშირებულია თუ არა არდუინო პორტთან (იხ. სურათი 3);
- ჩასვით პროგრამული კოდი – sketch არდუინო IDE-ს კოდის/sketch-ის ფანჯარაში, როგორც ნაჩვენებია მე-4 სურათზე (File-Examples-Basics-Blink);



სურათი 3: არდუინოს დაკავშირება პორტთან



სურათი 4: როგორ მოძებნოთ კოდი Blink არდუინოში

### Sketch/კოდი Blink:

- // Blinking LED Project
- int led = 13;
- void setup() {
- pinMode(led, OUTPUT);
- }

```

5. void loop() {
6.   digitalWrite(led, HIGH);
7.   delay(1000);
8.   digitalWrite(led, LOW);
9.   delay(1000);
10. }

```

6. დააწკაპუნეთ ღილაკზე **Verify**, რომ დარწმუნდეთ კოდის/sketch-ის მუშაობაში (სწორია თუ არა კოდი);
7. დააწკაპუნეთ ღილაკზე **Upload**, რომ კოდი/sketch ჩაიტვირთოს არდუინოში.

### კოდის/SKETCH-ის განმარტება

პირველი სტრიქონი არის კომენტარი. ნებისმიერი სტრიქონი, რომელიც თქვენს პროგრამაში იწყება ნიშნით `//`, განკუთვნილია მხოლოდ მომხმარებლისთვის და მას არდუინო არ კითხულობს. ის გამოიყენება შენიშვნებისა და კოდის აღწერისთვის (კოდის კომენტირება). თუ კომენტარი ერთ სტრიქონზე მეტია, მაშინ პირველი სტრიქონი იწყება და სრულდება `*/` სიმბოლოთი. მთელი ტექსტი, რაც ამ სიმბოლოთა შორის იქნება, არდუინოს მიერ იგნორირდება.

მეორე სტრიქონი გვეუბნება, რომ ამ კოდში მე-13 გამომყვანმა პინმა მიიღო სახელწოდება LED. რამდენჯერაც Sketch-ში ნახსენები იქნება LED, ის ყოველთვის შეეხება მე-13 პინს.

**Void setup** (მოთავსდება პარამეტრების დასაყენებელი კოდი) ნიშნავს, რომ კოდი/Sketch, მოქცეული ფიგურულ ფრჩხილებში {}, რომელიც მოსდევს შეტყობინებას, პროგრამის გაშვებისას შესრულდება მხოლოდ ერთხელ. { სიმბოლოთი იწყება კოდის/Sketch-ის წერა.

**PinMode** ფუნქციით განვსაზღვრავთ საკონტაქტო გამომყვანს, რომელთანაც ვმუშაობთ და მის მდგომარეობას (INPUT ან OUTPUT). **pinMode** ფუნქციის არგუმენტებია 13 და OUTPUT. თუ ვკითხულობთ ინფორმაციას საკონტაქტო გამომყვანიდან, ეს იქნება INPUT, თუ ვწერთ ინფორმაციას საკონტაქტო გამომყვანში, მაშინ გვექნება OUTPUT. რადგანაც ჩვენ ვწერთ (ვაგზავნით) მე-13 გამომყვანში ინფორმაციას, ამიტომაც **pinMode** ფუნქციის არგუმენტი იქნება OUTPUT. **pinMode (LED, OUTPUT)** ბრძანებით საკონტაქტო გამომყვანის კონფიგურირება ხდება, როგორც გამოსასვლელის. ე.ი. ის ატყობინებს არდუინოს, რომ მე-13 პინი გამომავალია, რაც ნიშნავს, რომ ჩვენ გვინდა კვება მივაწოდოთ LED ნათურას (შუქდიოდს). სიმბოლო } ამთავრებს კოდს.

**void loop** (აქ მოთავსდება ძირითადი კოდი) ქმნის ციკლს. ფიგურულ ფრჩხილებში მოქცეული ინფორმაცია პროგრამის გაშვებისას განმეორდება ციკლურად, დაუსრულებლად, ვიდრე არ გავთიშავთ არდუინოს.

**loop()** ფუნქციის შიგნით ვწერთ **digitalWrite** ფუნქციას. მისი საშუალებით შუქდიოდის ჩართვა ან გამორთვა შეგვიძლია. ჩვენ ვაგზავნით, ვთქვათ 5V სიგნალს მე-13 საკონტაქტო გამომყვანზე, მაშინ არგუმენტად ვწერთ HIGH-ს. ეს ნიშნავს რომ ამ დროს შუქდიოდი აინთება.

შემდეგ პროგრამაში ვრთავთ **delay** ფუნქციას, რომელიც ON (ჩართვა) და OFF (გამორთვა) მდგომარეობას შორის დაყოვნებას უზრუნველყოფს. ამ ფუნქციის პარამეტრი მილიწამებშია მოცემული. ჩვენ შემთხვევაში, დაყოვნების დრო იქნება 1000 მილიწამი (1 წამი).

მერვე სტრიქონში ვცვლით **digitalWrite** ფუნქციას და მე-13 საკონტაქტო გამომყვანზე არგუმენტად ვწერთ LOW-ს. ეს ნიშნავს, რომ ამ დროს შუქდიოდს კვება აღარ მიეწოდება და ის ჩაქრება.

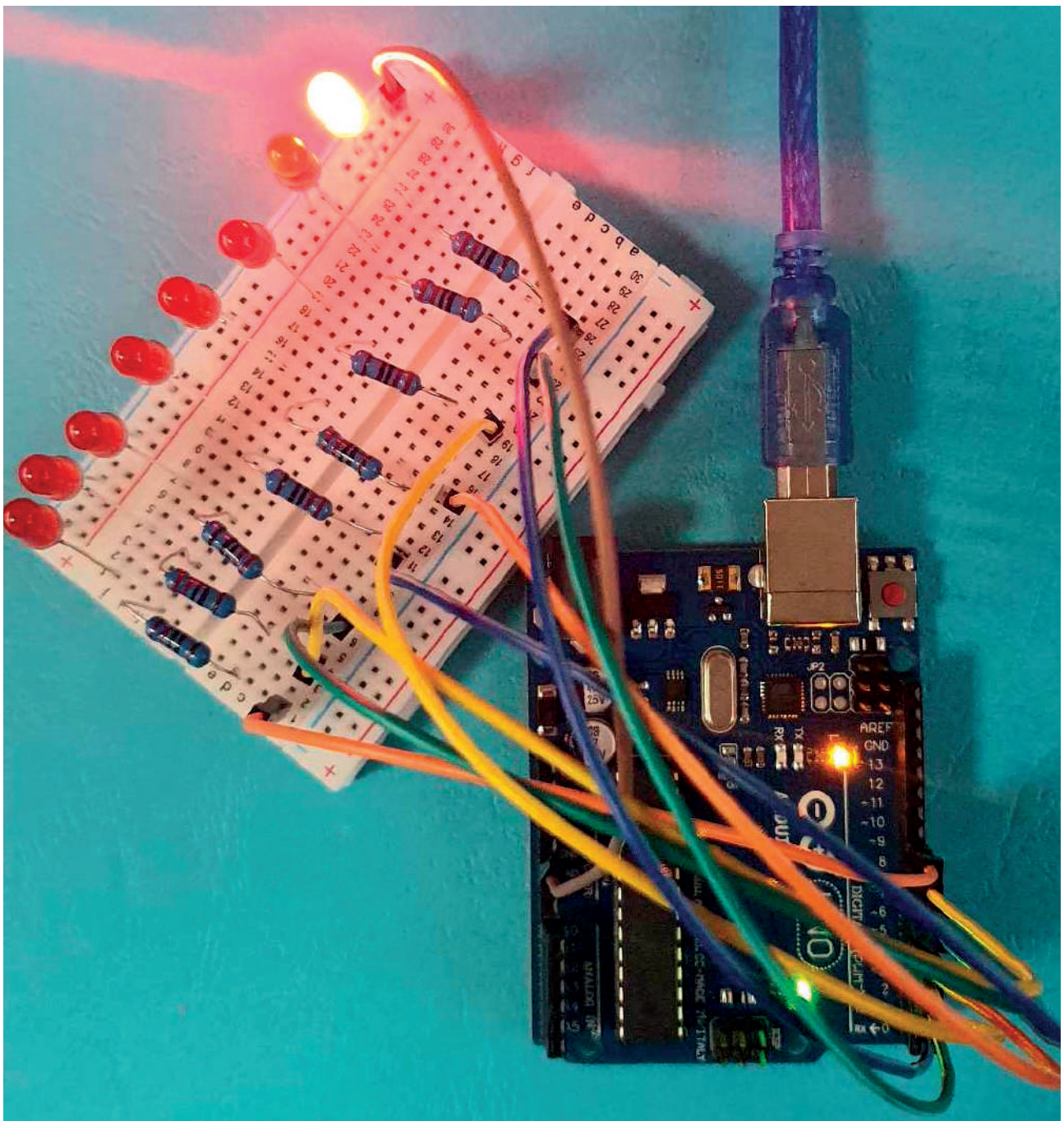
მეცხრე სტრიქონში კვლავ ვუთითებთ **delay** ფუნქციას და დაყოვნების დრო იქნება 1000 მილიწამი (1 წამი).

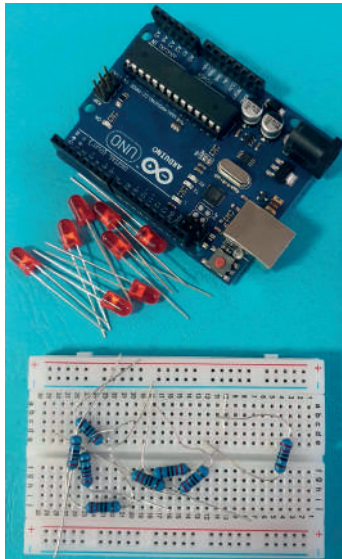
მეთე სტრიქონის ეს სიმბოლო } ნიშნავს ციკლის დასრულებას.

კოდის წერისას განსაკუთრებული მნიშვნელობა ენიჭება ყველა ნიუანსს: წერტილ-მძიმეს, კოდების მოთავსებას სიმბოლოებში {}... ნებისმიერი გადაცდომა არდუინოს მიერ აღიქმება როგორც შეცდომა და პროგრამა არ იმუშავებს.

II კონკრეტი

# მეკრივები მოსივსივე LED ნათურები





### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 8 ცალი LED ნათურა
- 8 ცალი 220 ომი წინაღობა

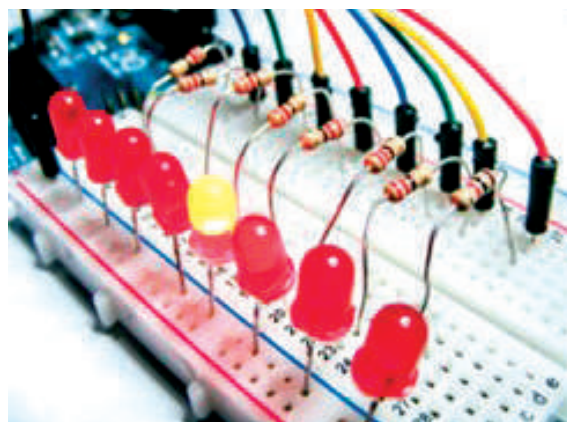
### როგორ მუშაობს LED ნათურა

LED ნათურა მაშინ ანათებს, როცა მასში დაბალი ძაბვა გადის. LED ნათურა პოლარიზებულია, რაც ნიშნავს, რომ მისი ერთი მხარე დადებითია, მეორე კი – უარყოფითი. ეს იმიტომ ხდება, რომ შექდიოდი მხოლოდ მაშინ მუშაობს, როცა მასში ერთი მიმართულებით გადის დენი – დადებითიდან უარყოფითისკენ. LED-ის ერთი ფეხი (გრძელი) დადებითია და ყოველთვის უნდა დაუკავშირდეს კვების დადებით პოლუსს. ამ პროექტში არდუინოს კოდი მართავს LED-ების თანმიმდევრულ ანთებას.

LED ნათურები ნახევრადგამტარებია და მათი მუშაობისთვის საჭიროა დაბალი ძაბვა (2-4 ვოლტი) საჭირო, იმაზე მცირე, ვიდრე არდუინო უზრუნველყოფს. შექდიოდე რომ არ გადაიწვას, უნდა გამოვიყენოთ წინააღობები, რომლებიც არდუინოდან გამომავალ ძაბვას ამცირებენ.

### აპაწყოთ სქემა

1. შექდიოდე დაამაგრეთ სამაკეტო დაფაზე. მათი მოკლე, უარყოფითი ფეხები დაფის უარყოფით პოლუსიან ლიანდაგზე მოათავსეთ (იხ. სურათი 1), ეს უკანასკნელი კი დააკავშირეთ არდუინოს GND კონტაქტთან.



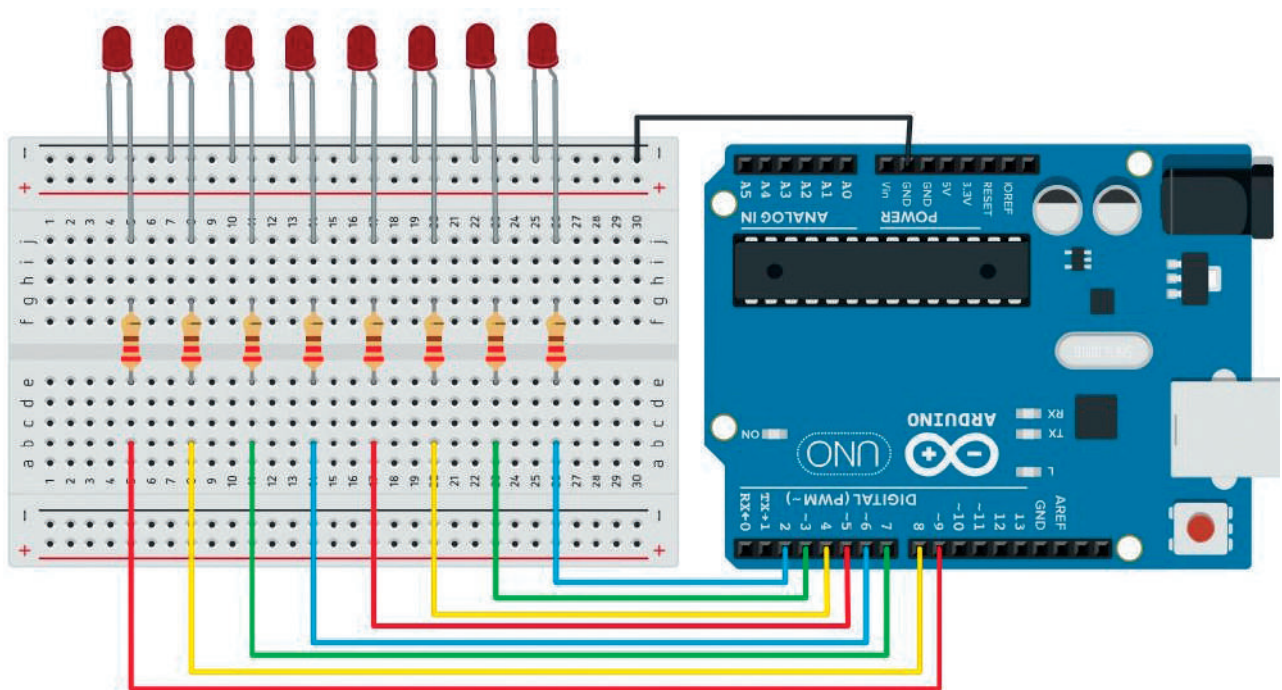
სურათი 1. LED-ების მოკლე ფეხები სამაკეტო დაფის GND-ზეა დამაგრებული. გრძელი (+) ფეხები კი, წინააღობის საშუალებით, დაკავშირებულია არდუინოსთან.

2. დააკავშირეთ შექდიოდები არდუინოს ციფრულ გამომყვანებთან თანმიმდევრობით, 2-დან 9-მდე, ისე, როგორც ეს სქემაზეა ნაჩვენები (იხ. სურათი 2).

220 ომი წინაღობები მოათავსეთ ისე, რომ ისინი მოხვდნენ თითოეული შექდიოდისა და არდუინოს ციფრულ გამომყვანს შორის. მიაქციეთ ყურადღება, რომ წინააღობების ფეხები მოთავსებული იყოს სამაკეტო დაფის ცენტრალური გამყოფი ზოლის ორივე მხარეს.

| LED ნათურები         | ARDUINO                    |
|----------------------|----------------------------|
| დადებითი (+) ფეხები  | Pins 2-9 წინააღობის გავლით |
| უარყოფითი (-) ფეხები | GND                        |

3. კიდევ ერთხელ შეამოწმეთ თქვენი სქემა და ჩატვირთეთ კოდი.



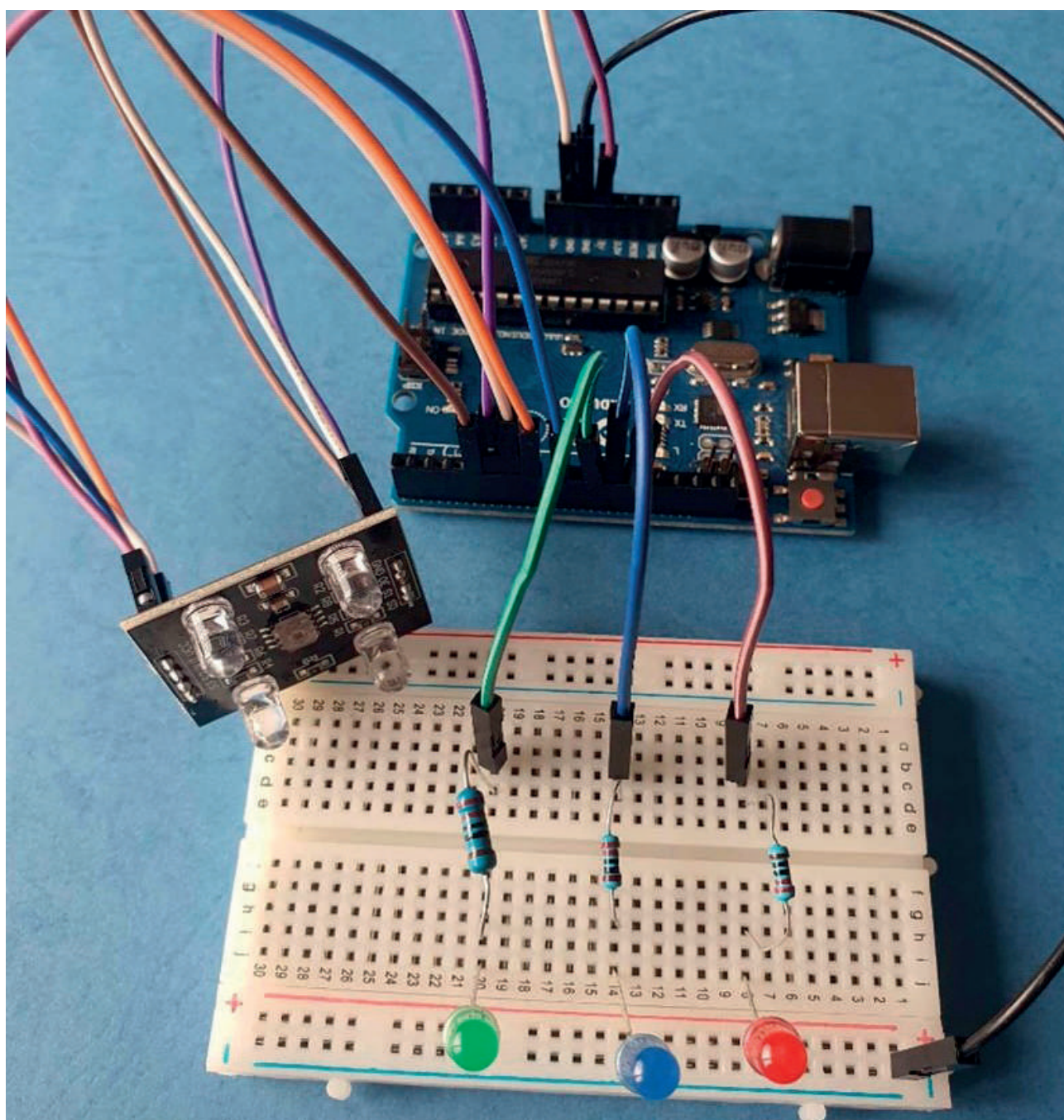
სურათი 2. სქემა

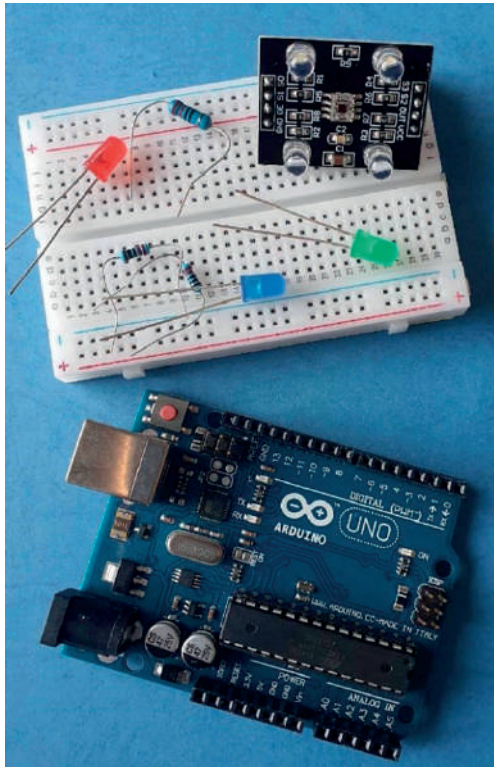
## როგორ გუვაოგ კოდი

კოდი შექდიოდებთან დაკავშირებულ არდუინოს კონტაქტებს განსაზღვრავს, როგორც გამომყვანებს. შემდეგ განსაზღვრავს ყველა შექდიოდის ერთდროული გამორთვის ფუნქციას. ეს ფუნქცია მოწოდებულია ციკლში, რომ შექდიოდები ჯერ გამორთოს, შემდეგ კი თანმიმდევრობით აანთოს 200 მილიწამის დაყოვნებით.

III პრაქტიკა

# ფერის ამოცანები





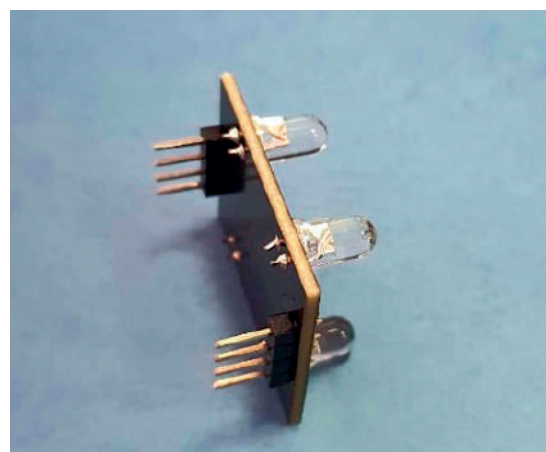
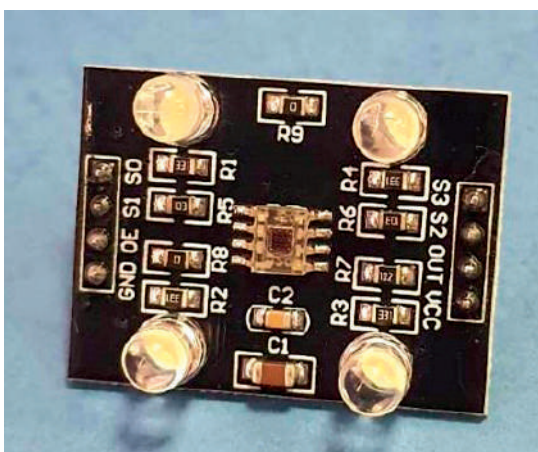
### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 3 ცალი LED ნათურა: წითელი, მწვანე და ლურჯი
- 3 ცალი 220 ომი წინაღობა
- ფერის სენსორი
- მწვანე, ლურჯი და წითელი ფერის ფურცლები

### როგორ გუჰაოგს ფარის სენსორი

ეს სენსორები არის ფოტოელექტრული მოწყობილობები, რომლებსაც შეუძლიათ გამოასხივონ შუქი და აღმოაჩინონ არეკლილი სინათლის ფერი ობიექტიდან. ამ სენსორებს შეუძლიათ ამოიცნონ ობიექტიდან ასახული სინათლის ინტენსივობა და განასხვავონ ძირითადი ფერები: წითელი, ლურჯი და მწვანე. მათ, ასევე, უწოდებენ ფერის დეტექტორებს. ფერის სენსორებს შეუძლიათ გაანათონ ობიექტი ფართო ტალღის სიგრძით, სინათლის თანაფარდობით და განსაზღვრონ ძირითადი ფერების (წითელი, ლურჯი, მწვანე და თეთრი) სინათლის ინტენსივობა. სინათლის ინტენსივობის თანაფარდობა განსაზღვრავს ობიექტის მიერ არეკლილი და შთანთქმული სინათლის რაოდენობას.

ჩვენ მიერ შექმნილ პროექტს შეუძლია ამოიცნოს მწვანე, წითელი და ლურჯი ფერები არდუინოს გამოყენებით და ფერების სენსორით – რომელი ფერის ფურცელსაც დაადებთ სენსორს, შესაბამისი ფერის LED ნათურა აინთება.



სურათი 1. ფერის ამომცნობი სენსორი

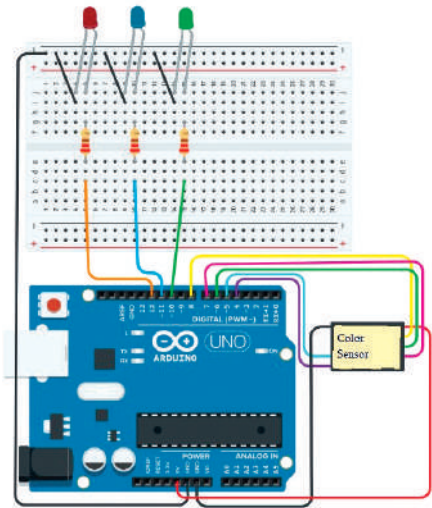
პანელის სქემა

1. LED ნათურების მოკლე, უარყოფითპოლუსიანი ფეხები დააკავშირეთ სამაკეტო დაფის უარყოფითპოლუსიან ლიანდაგს, ხოლო გრძელი, დადებითპოლუსიანი ფეხები 220 ომი წინააღობის გავლით დააკავშირეთ არდუინოს შემდეგ პინებზე: მწვანე LED ნათურა – მე-10 პინზე, ლურჯი – მე-11 პინზე და წითელი – მე-12 პინზე.

| LED ნათურები           | ARDUINO                          |
|------------------------|----------------------------------|
| მწვანე (დადებითი ფეხი) | Pin 10                           |
| ლურჯი (დადებითი ფეხი)  | Pin 11                           |
| წითელი (დადებითი ფეხი) | Pin 12                           |
| უარყოფითი ფეხები       | GND<br>220 ომი წინააღობის გავლით |

2. ფერის სენსორი დააკავშირეთ არდუინოს ციფრულ კონტაქტებს. ჩვენ მიერ გამოყენებულ სენსორს აქვს 8 პინი (იხ. სურათი 1). ჩვენ გამოვიყენებთ 7 პინს. S0 პინი დაუკავშირეთ არდუინოს მე-4 პინს, S1 – მე-5 პინს, S2 – მე-6 პინს, S3 – მე-7 პინს, Out pin – მე-8 პინს, GND – არდუინოს GND-ს და Vcc პინი – არდუინოს +5V-ს.

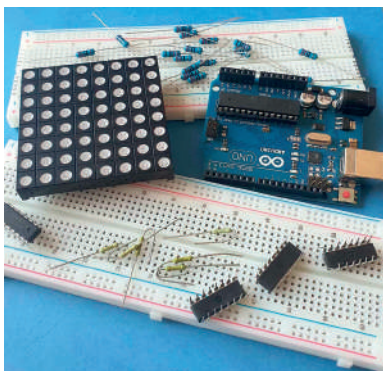
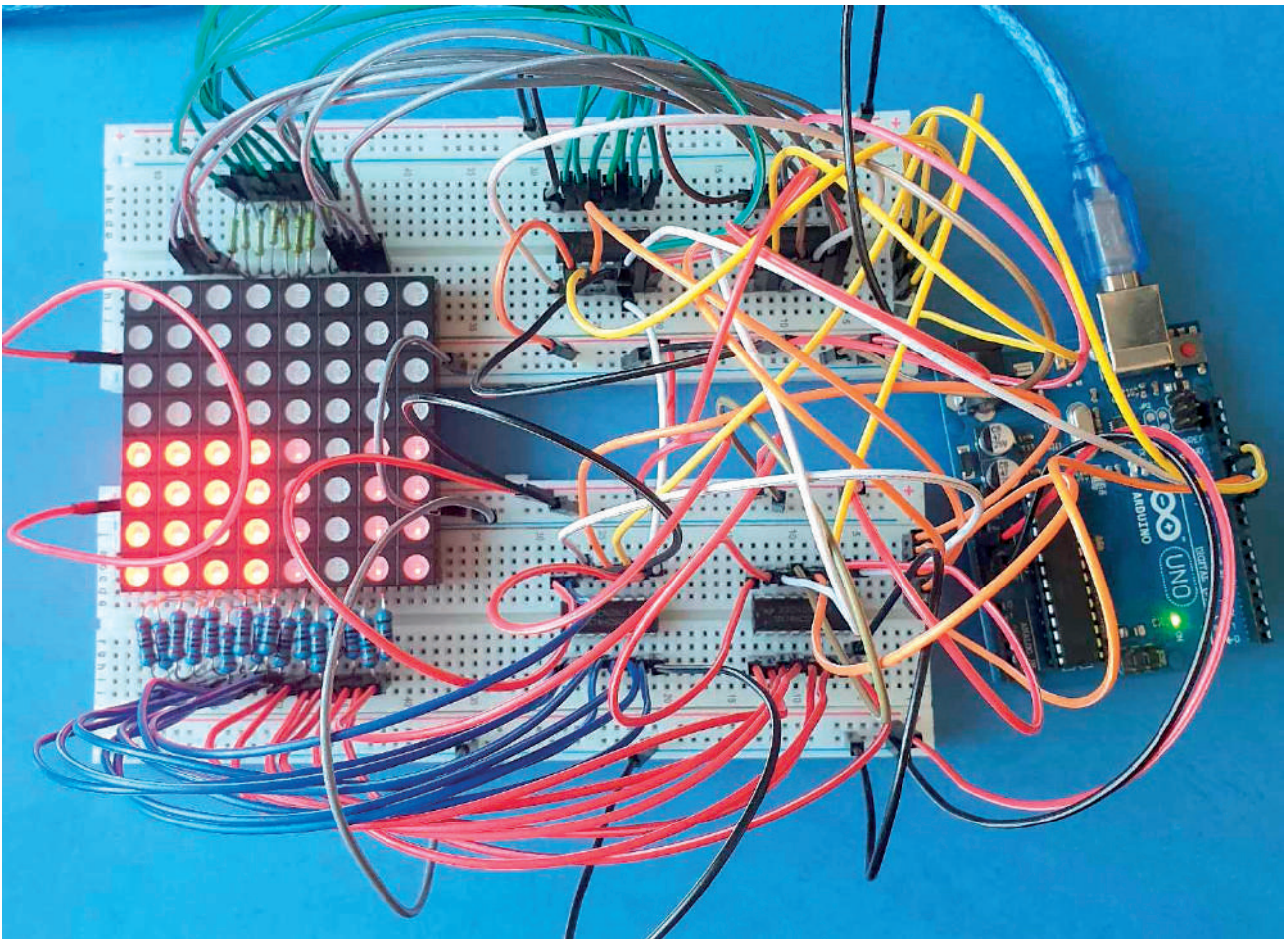
| ფერის სენსორი | ARDUINO |
|---------------|---------|
| Pin S0        | Pin 4   |
| Pin S1        | Pin 5   |
| Pin S2        | Pin 6   |
| Pin S3        | Pin 7   |
| Out Pin       | Pin 8   |
| GND           | GND     |
| Pin Vcc       | +5V     |



სურათი 2. სქემა

## IV პროექტი

# 8X8 RGB LED მატრიცა

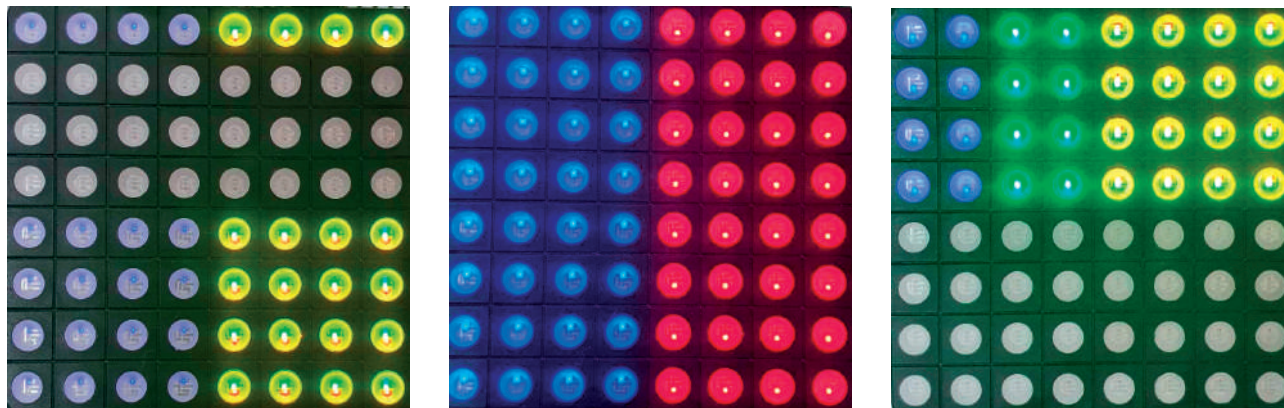


### საჭირო რესურსი

- Arduino UNO
- 2 ცალი დიდი სამაკეტო დაფა
- გამტარები
- 8X8 RGB LED მატრიცა
- 4 ცალი 74HC595 ლოგიკური მიკროსქემა
- 16 ცალი 220 ომი წინაღობა
- 8 ცალი 330 წინაღობა

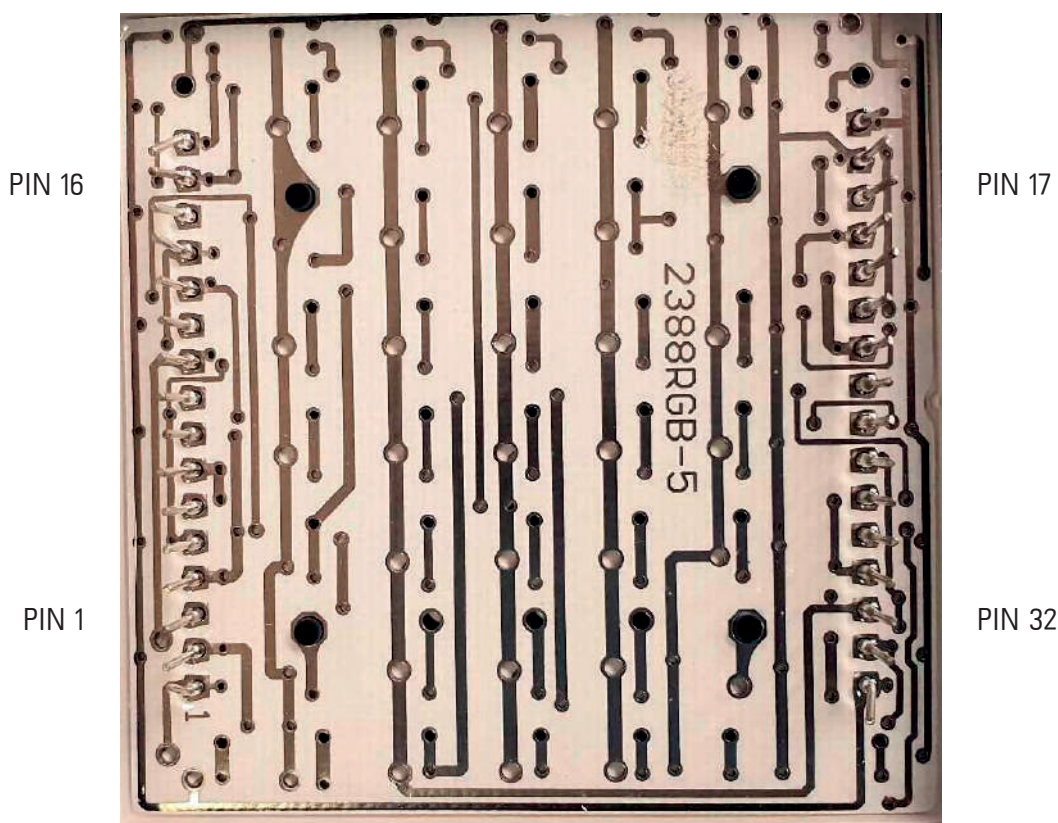
## როგორ მუშაობს RGB მაჭრიცა

RGB მატრიცა წარმოადგენს ბადეს, რომელიც შედგება 64 წითელი, მწვანე და ლურჯი შუქდიოდისგან. თქვენ შეგიძლიათ მართოთ თითოეული შუქდიოდი და შექმნათ ცისარტყელას ყველა ფერი.



სურათი 1. RGB LED მატრიცა

ამ მატრიცას აქვს 32 კონტაქტი. 8 კონტაქტი აკონტროლებს თითოეული შუქდიოდის საერთო ანოდის დადებით ფეხს; 8 კონტაქტი – წითელი, მწვანე და ლურჯი ფერების საშუალებით სხვადასხვა ფერთა გამის მიღებას. ჩვენ მიერ გამოყენებულ მატრიცაში 17-20 და 29-32 კონტაქტები წარმოადგენენ ანოდურს, 9-16 – წითელს, 21-28 – მწვანეს და 1-8 – ლურჯს. არსებობენ ასეთივე მატრიცები, რომელთაც შეიძლება ჰქონდეთ სხვანაირად განლაგებული კონტაქტები. ჩვენს სურათზე კონტაქტების ნომრები საათის ისრის მოძრაობის მიმართულებით არის განლაგებული (იხ. სურათი 2).



სურათი 2. RGB LED მატრიცის კონტაქტები

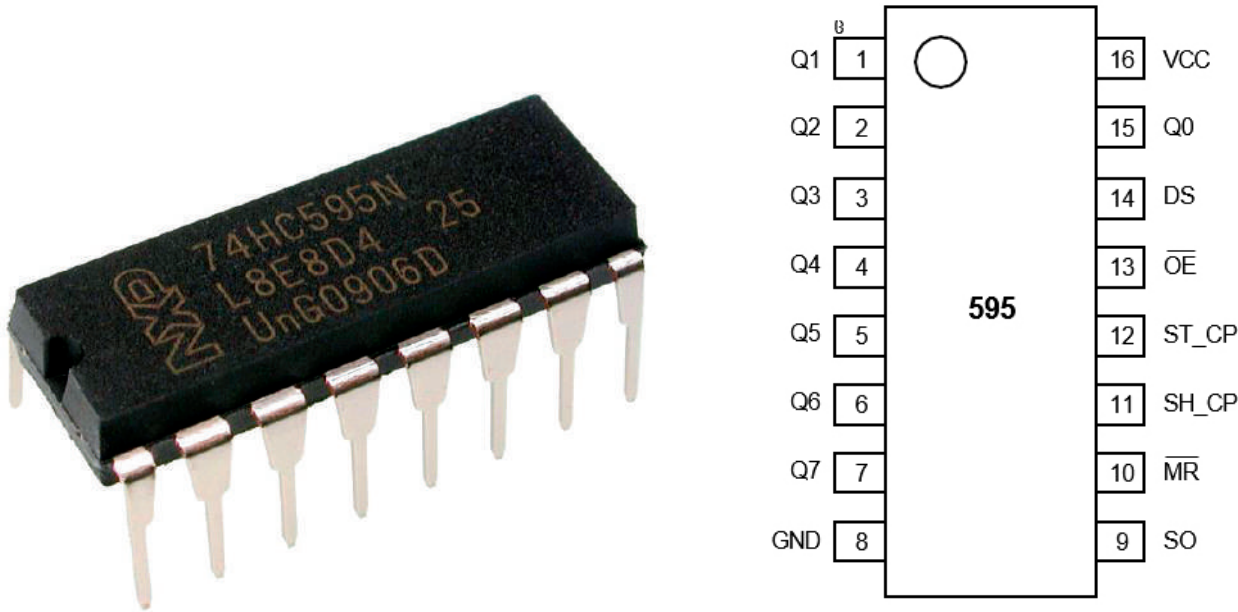
თითოეული ფერის გამომყვანისთვის დაგვჭირდება წინაღობა, რომ თავიდან ავიცილოთ შექ-  
დიოდების გადაწვა. ჩვენ დაგვჭირდება განსხვავებული მნიშვნელობების წინააღობები: ლურჯი და  
მწვანე ფერებისთვის – 220 ომი წინააღობა, ხოლო წითელი ფერისთვის – 330 ომი.

| მატრიცის ფუნქციური პინები | მატრიცის პინების ნუმერაცია     |
|---------------------------|--------------------------------|
| საერთო ანოდი (+)          | 17, 18, 19, 20, 29, 30, 31, 32 |
| წითელი შექდიოდები         | 9, 10, 11, 12, 13, 14, 15, 16  |
| მწვანე შექდიოდები         | 21, 22, 23, 24, 25, 26, 27, 28 |
| ლურჯი შექდიოდები          | 1, 2, 3, 4, 5, 6, 7, 8         |

ცხრილი 1. RGB LED მატრიცის კონტაქტების/პინების კონფიგურაცია

შეიძლება სქემის აწყობა რთულად მოგეჩვენოთ, მაგრამ ეს ასე არ არის. უბრალოდ, ბევრ სხვა-  
დასხვანაირ გამტარს ვიყენებთ.

ვინაიდან დიდი რაოდენობით კავშირების გამო არ გვეყოფა არდუინოს საკონტაქტო გამომყ-  
ვანები, არდუინოს დაფის გასათართოებლად გამოვიყენებთ 74HC595 ლოგიკურ მიკროსქემას (იხ.  
სურათი 3).



სურათი 3. 74HC595 ლოგიკური მიკროსქემა და მისი გამომყვანები

74HC595 ლოგიკური მიკროსქემა წარმოადგენს მეხსიერების ციფრული სქემას, რომელიც გამო-  
იყენება კალკულატორებში, კომპიუტერებსა და მონაცემთა დამუშავების სისტემებში. ამ პროექტში  
74HC595 ლოგიკური მიკროსქემა გამოიყენება 8 გამომყვანის ერთდროული მართვისთვის, რა დრო-  
საც ის არდუინოს მხოლოდ სამ გამომყვანს იყენებს. ჩვენ რამდენიმე რეგისტრს დავაკავშირებთ  
ერთმანეთთან, ერთდროულად რამდენიმე გამომყვანის სამართავად. ამისათვის ერთს ვიყენებთ  
საერთო ანოდის და ერთს – შექდიოდის ყველა ფერის სამართავად. იდენტიფიცირების გასამარტი-  
ვებლად 74HC595 ლოგიკური მიკროსქემის ფუნქციები აღწერილია მეორე ცხრილში.

| ლოგიკური მიკროსქემა | კავშირები | კავშირების ფუნქციები                          |
|---------------------|-----------|-----------------------------------------------|
| Pins 1-7, 15        | Q0-Q7     | გამომყვანი კონტაქტები                         |
| Pin 8               | GND       | დამიწება, VSS                                 |
| Pin 9               | SO        | სერიული გამომყვანი                            |
| Pin 10              | MR        | Master Reclear,<br>აქტიური დაბალი             |
| Pin 11              | SH_CP     | Shift register clock pin<br>(CLOCK pin)       |
| Pin 12              | ST_CP     | Storage register clock pin<br>(LATCH pin)     |
| Pin 13              | OE        | გამომყვანი ჩართულია,<br>აქტიური დაბალი        |
| Pin 14              | DS        | მონაცემთა მიმღევრობითი<br>შემყვანი (DATA pin) |
| Pin 16              | VCC       | დადებითი (+)                                  |

ცხრილი 2. 74HC595 ლოგიკური მიკროსქემის ფუნქციები

პანელის სქემა

- 1. დაამაგრეთ 8X8 RGB LED მატრიცა ორ დიდ სამაკეტო დაფაზე.
- 2. დააკავშირეთ წინააღობები (330 ომი) წითელი შექდიოდის ყველა გამომყვანზე; დააკავშირეთ წინააღობები (220 ომი) მწვანე და ლურჯი შექდიოდების ყველა გამომყვანთან.
- 3. I ლოგიკური მიკროსქემა მოათავსეთ LED მატრიცის საერთოანოდური გამომყვანების ახლოს ისე, როგორც ეს სქემაზეა ნაჩვენები – დაფის ცენტრალური გამყოფი ზოლის ორივე მხარეს (იხ. სურათი 4) შემდეგნაირად:

| საერთოანოდური კავშირები |                | I ლოგიკური სქემის კავშირები |            |
|-------------------------|----------------|-----------------------------|------------|
| LED მატრიცა             | ლოგიკური სქემა | ლოგიკური სქემა              | ARDUINO    |
| 32                      | 15: Q0         | 8: GND                      | GND        |
| 31                      | 1: Q1          | 9: SO                       | Shift 3 DS |
| 30                      | 2: Q2          | 10: MR                      | +5V        |
| 29                      | 3: Q3          | 11: SH-CP                   | 13         |
| 20                      | 4: Q4          | 12: ST-CP                   | 10         |
| 19                      | 5: Q5          | 13: OE                      | GND        |
| 18                      | 6: Q6          | 14: DS                      | Shift 2 SO |
| 17                      | 7: Q7          | 16: VCC                     | +5V        |

ლოგიკურ სქემებს წინააღობები არ სჭირდება.

4. დანარჩენი სამი ლოგიკური სქემა დაამატეთ სამაკეტო დაფებზე. II ლოგიკური სქემა მართავს მწვანე შუქდიოდებს, III – ლურჯს, ხოლო IV – წითელს. გამტარებით დაუკავშირეთ სამივე ლოგიკური სქემა LED მატრიცას შემდეგნაირად:

| მწვანე შუქდიოდის კავშირები |                | II ლოგიკური სქემის კავშირები |            |
|----------------------------|----------------|------------------------------|------------|
| LED მატრიცა                | ლოგიკური სქემა | ლოგიკური სქემა               | ARDUINO    |
| 28                         | 15: Q0         | 8: GND                       | GND        |
| 27                         | 1: Q1          | 9: SO                        | Shift 1 DS |
| 26                         | 2: Q2          | 10: MR                       | +5V        |
| 25                         | 3: Q3          | 11: SH-CP                    | 13         |
| 24                         | 4: Q4          | 12: ST-CP                    | 10         |
| 23                         | 5: Q5          | 13: OE                       | GND        |
| 22                         | 6: Q6          | 14: DS                       | 11         |
| 21                         | 7: Q7          | 16: VCC                      | +5V        |

| ლურჯი შუქდიოდის კავშირები |                | III ლოგიკური სქემის კავშირები |            |
|---------------------------|----------------|-------------------------------|------------|
| LED მატრიცა               | ლოგიკური სქემა | ლოგიკური სქემა                | ARDUINO    |
| 1                         | 15: Q0         | 8: GND                        | GND        |
| 2                         | 1: Q1          | 9: SO                         | Shift 4 DS |
| 3                         | 2: Q2          | 10: MR                        | +5V        |
| 4                         | 3: Q3          | 11: SH-CP                     | 13         |
| 5                         | 4: Q4          | 12: ST-CP                     | 10         |
| 6                         | 5: Q5          | 13: OE                        | GND        |
| 7                         | 6: Q6          | 14: DS                        | Shift 1 SO |
| 8                         | 7: Q7          | 16: VCC                       | +5V        |

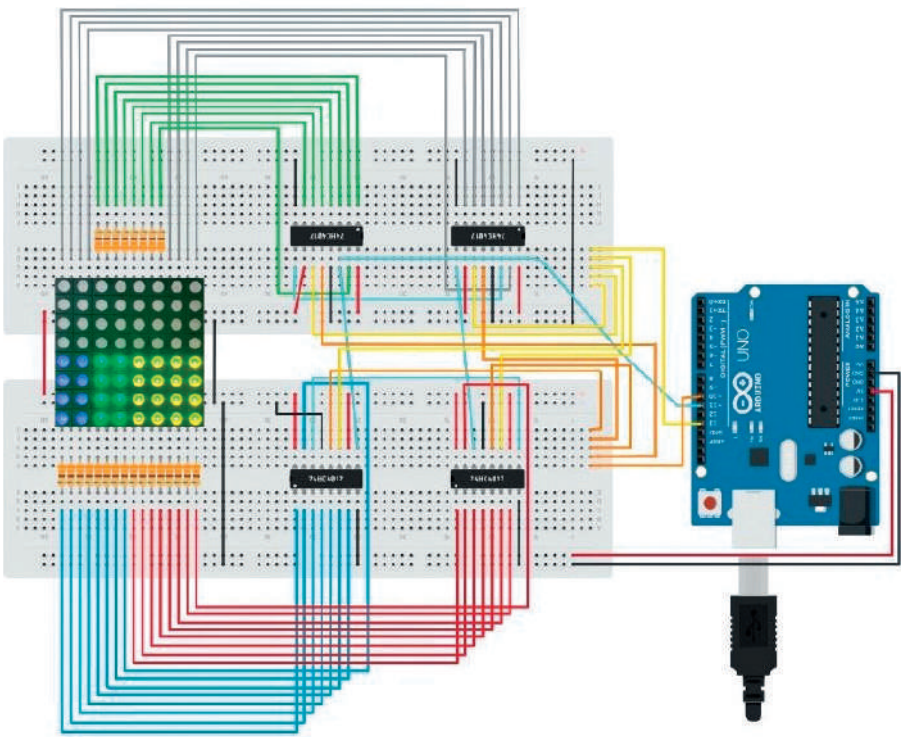
| წითელი შუქდიოდის კავშირები |                | IV ლოგიკური სქემის კავშირები |            |
|----------------------------|----------------|------------------------------|------------|
| LED მატრიცა                | ლოგიკური სქემა | ლოგიკური სქემა               | ARDUINO    |
| 9                          | 15: Q0         | 8: GND                       | GND        |
| 10                         | 1: Q1          | 9: SO                        | Shift 3 DS |
| 11                         | 2: Q2          | 10: MR                       | +5V        |
| 12                         | 3: Q3          | 11: SH-CP                    | 13         |
| 13                         | 4: Q4          | 12: ST-CP                    | 10         |
| 14                         | 5: Q5          | 13: OE                       | GND        |
| 15                         | 6: Q6          | 14: DS                       | Shift 2 SO |
| 16                         | 7: Q7          | 16: VCC                      | +5V        |

LED მატრიცის ფერადი შუქდიოდების ყველა გამომყვანისთვის დაგჭირდებათ წინაღობები.

5. არდუინო შუქდიოდებს აკონტროლებს სამი საკონტაქტო გამომყვანით: one each for clock – თითოეულისთვის განსაზღვრული დრო (ტაქტური სიხშირე), data – მონაცემები და latch – ჩამკეტი. თითოეული გამომყვანი არდუინოს უკავშირდება შემდეგნაირად:

| ლოგიკური მიკროსქემა    | ARDUINO | ფუნქციები          |
|------------------------|---------|--------------------|
| Pin 9 (shift reg 2)    | Pin 11  | data               |
| Pin 12 (all shift reg) | Pin 10  | latch              |
| Pin 11 (all shift reg) | Pin 13  | one each for clock |

6. დარწმუნდით, რომ თქვენ მიერ აწყობილი სქემა ზუსტად იმეორებს ჩვენს სქემას (სურათი 4) და ჩატვირთეთ კოდი.



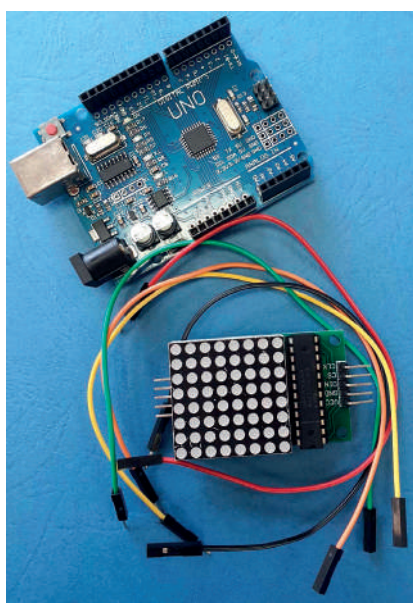
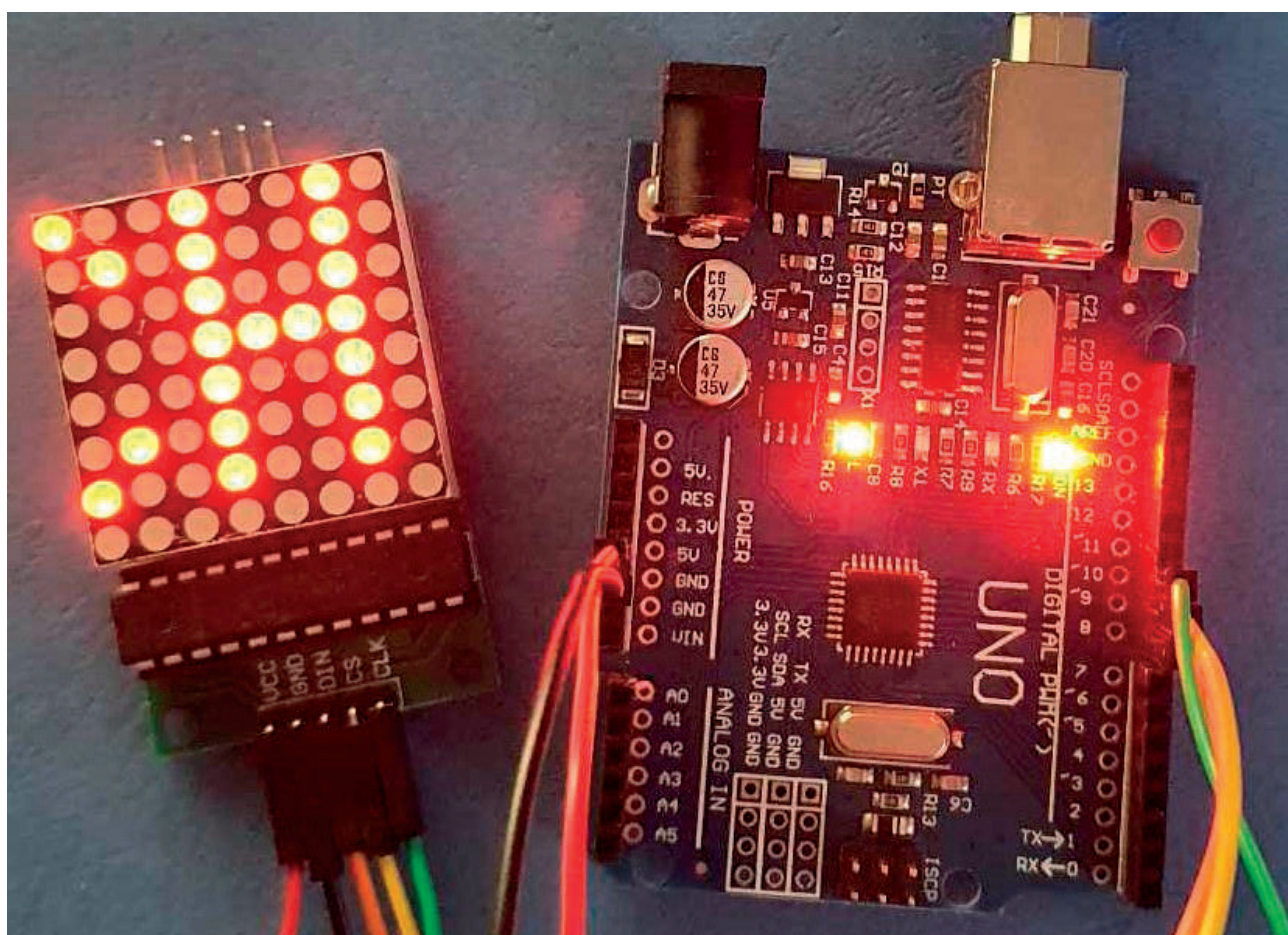
სურათი 4. 8X8 RGB მატრიცის სქემა

### როგორ გუშაოვს კოდი

თავდაპირველად კოდში განისაზღვრება არდუინოს სამი გამომყვანი, რომლებიც მართავენ ლოგიკურ სქემებს. Latch კონტაქტი განისაზღვრება, როგორც არდუინოს მე-10 პინი, the clock კონტაქტი – როგორც მე-13 პინი და data – როგორც მე-11 პინი.

ჩვენ განვსაზღვრავთ ცვლადების რაოდენობას 0-დან 255-მდე, შუქდიოდების ფერების ინტენსივობის გასაკონტროლებლად. შემდეგ, კოდი რიგრიგობით ხართავს თითოეულ შუქდიოდს და აერთიანებს სამ ფერს ცისარტყელას ფერების შესაქმნელად. მაგალითად, მწვანე ჩართული, ლურჯი გამორთული და წითელი ჩართული – გამოჩნდება ყვითელი ფერი. კოდი სრულდება ციკლური, შემთხვევითი ფერების ციმციმით.

# RGB LED მატრიცა – მოკბენალი სტრიქონი



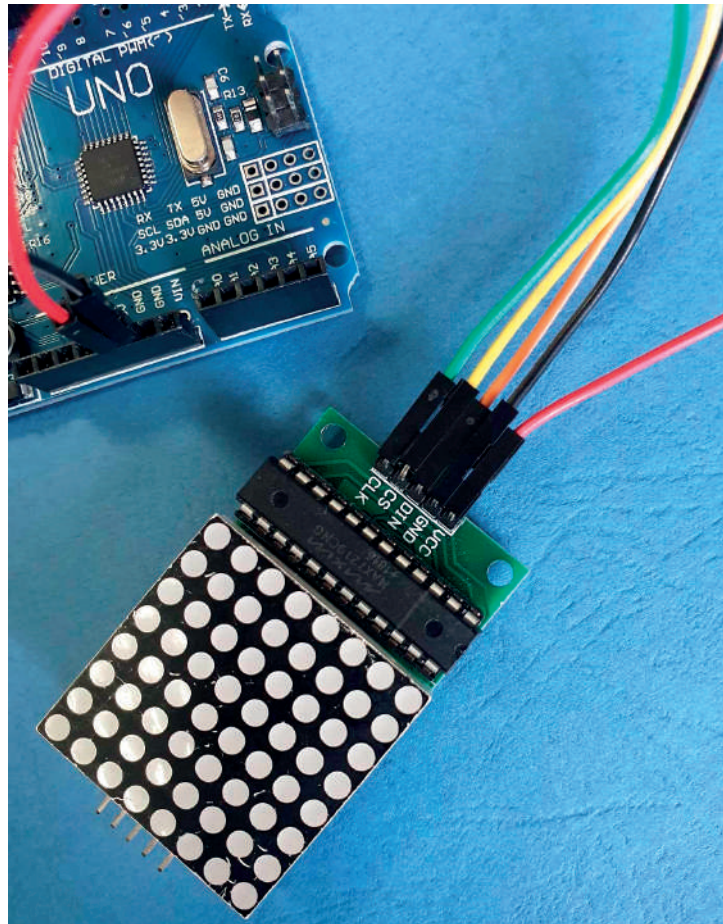
## საჭირო რესურსი

- Arduino UNO
- RGB LED მატრიცა
- დედალ-მამალი გამტარები

## საჭირო ბიბლიოთეკა

- MaxMatrix

ამ პროექტში ჩვენ გამოვიყენებთ დრაივერის ჩაშენებულ მოდულს მორბენალი სტრიქონის შესაქმნელად. 8x8 RGB მატრიცა წარმოადგენს შუქდიოდების ნაკრებს, რომელთა მართვაც შესაძლებელია სათითაოდ. ჩვენ შეგვიძლია შევექმნათ გამოსახულება, ტექსტი, ჩანახატი და ყველაფერი, რის დაპროგრამებასაც მოვისურვებთ. მატრიცას, რომელსაც ჩვენ გამოვიყენებთ, ჩაშენებული აქვს დრაივერის მოდული – დაფა, რომელიც იმართება ჩიპით Maxim 7219, რაც საშუალებას იძლევა, მთელი მატრიცა გმართოთ მხოლოდ 5 გამომყვანით (იხ. სურათი 1).



სურათი 1. ჩიპი Maxim 7219 მართავს RGB LED მატრიცას

მატრიცული მოდულის სამი გამომყვანი DIN, CS და CLK (იხ. სურათი 1) ნიშნავს: DIN – Data IN, CS – Chip Select, და CLK – CLoCK. დანარჩენი ორი კონტაქტი – VCC და GND – უზრუნველყოფენ მატრიცის კვებას. კონტაქტი CLK აღიქვამს იმპულსებს და მართავს არდუინოსა და მატრიცას შორის მონაცემთა სინქრონული გაცვლის სიჩქარეს. მიმდევრობითი პერიფერიული ინტერფეისის პროტოკოლს (SPI) მატრიცა იყენებს არდუინოსთან დასაკავშირებლად, ხოლო კონტაქტი CS განსაზღვრავს, SPI-ს რომელი მოწყობილობა გამოიყენება. კონტაქტი DIN კითხულობს მონაცემებს არდუინოდან – ჩვენ შემთხვევაში, პროექტის კოდს.

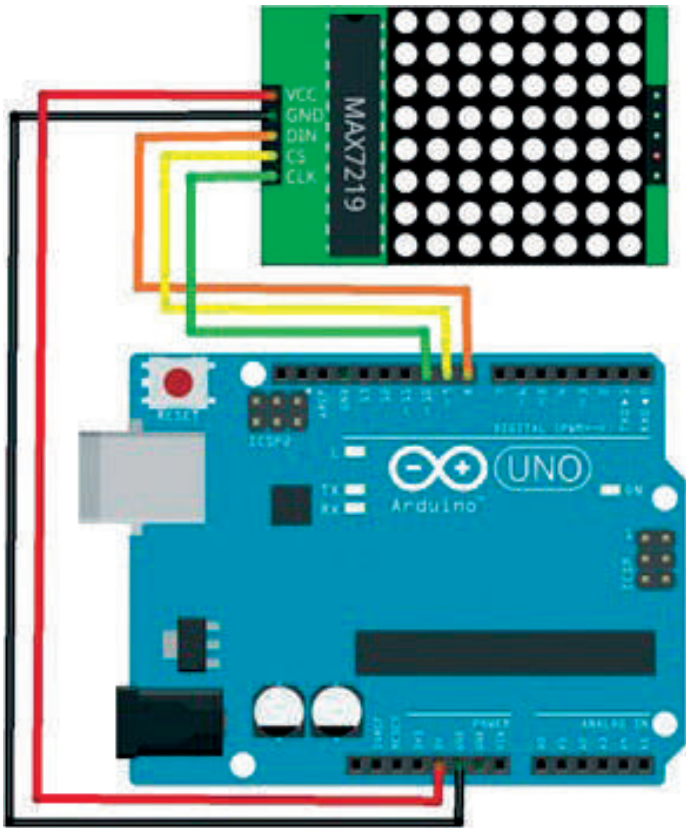
ყველა მოდულს აქვს დამატებითი კონექტორები, რაც ამ მატრიცების ერთმანეთზე ჯაჭვურად გადაბმის შესაძლებლობას იძლევა და იმართება მხოლოდ ერთი კოდით. ეს საშუალებას იძლევა, თქვენ მიერ შექმნილი სიმბოლოთა ერთობლიობა უფრო დიდ სივრცეზე გაშალოთ.

აპაჩოტსქემა

1. მოდული დააკავშირეთ არდუინოსთან დედალ-მამალი გამტარების საშუალებით: მოდულის VCC პინი – არდუინოს +5V-თან, GND – GND-თან, DIN – მე-8 პინთან, CS – მე-9 პინთან, ხოლო CLK – მე-10 პინთან.

| LED MATRIX მოდული | ARDUINO |
|-------------------|---------|
| VCC               | +5V     |
| GND               | GND     |
| DIN               | Pin 8   |
| CS                | Pin 9   |
| CLK               | Pin 10  |

2. დარწმუნდით, რომ სქემა (იხ. სურათი 2) სწორად გაქვთ აწყობილი და ჩატვირთეთ კოდი.



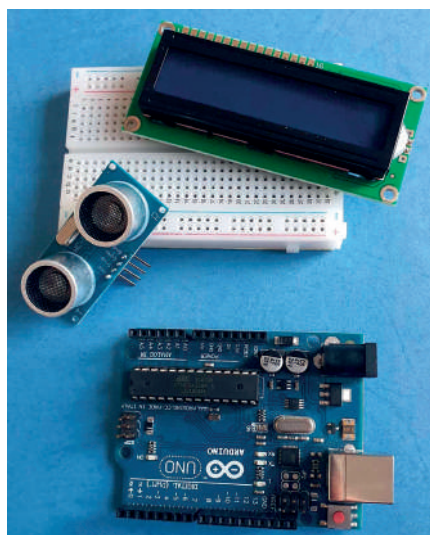
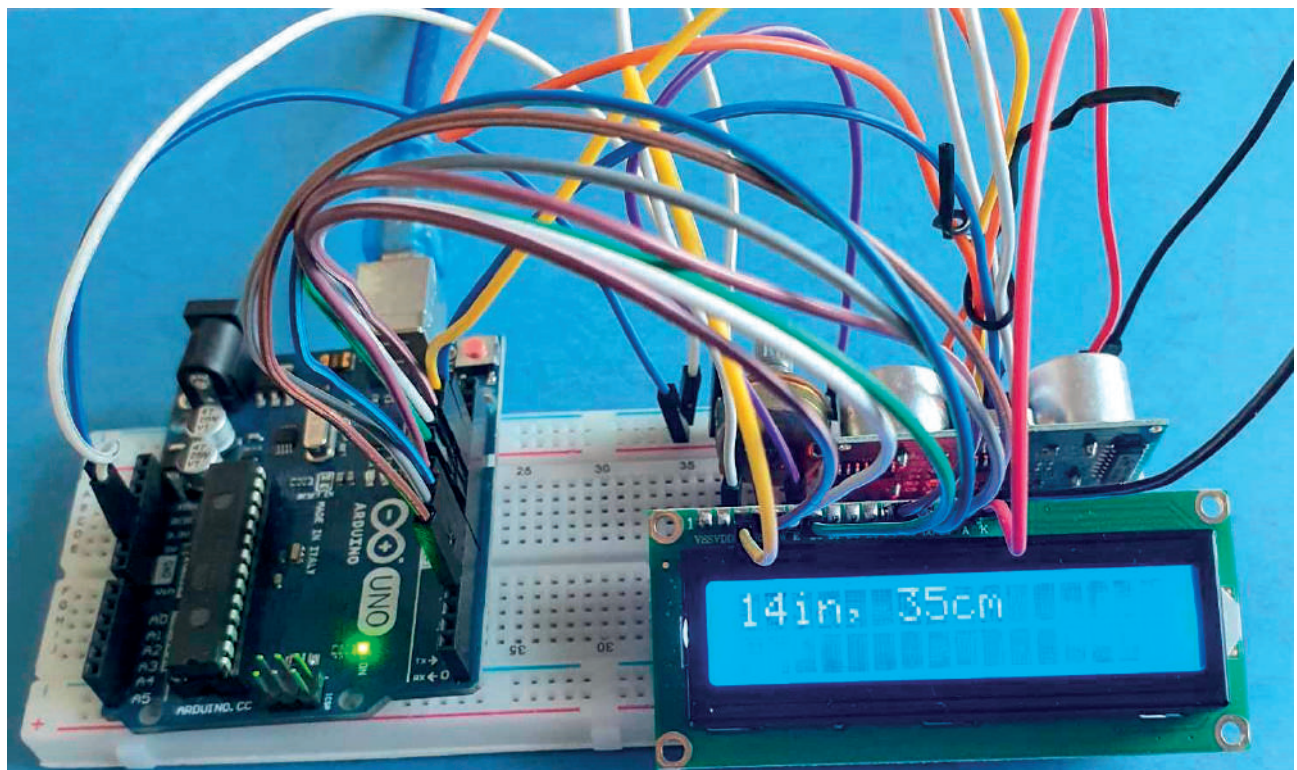
სურათი 2. სქემა

როგორ გუშაოვს კოდი

მატრიცული მოდულის სამართავად კოდი მიმართავს MaxMatrix ბიბლიოთეკას. შემდეგ, განვსაზღვრავთ სიმბოლოებს გამოსახულებისთვის და ვუთითებთ არდუინოს საკონტაქტო პინებს, რომლებიც მართავენ მატრიცას. თქვენი გამოსახულება, იქნება ეს ტექსტი, რიცხვები თუ სხვა რამ, უწყვეტ ციკლად გამოისახება შუქდიოდებზე.

## VI პროექტი

# განძილის გაზომვა ულტრაბგერით



### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 16x2 LCD ეკრანი (HD44780 მხარდაჭერით)
- HC-SR04 ულტრაბგერის სენსორი
- 50 კომი ცვლადი წინაღობა

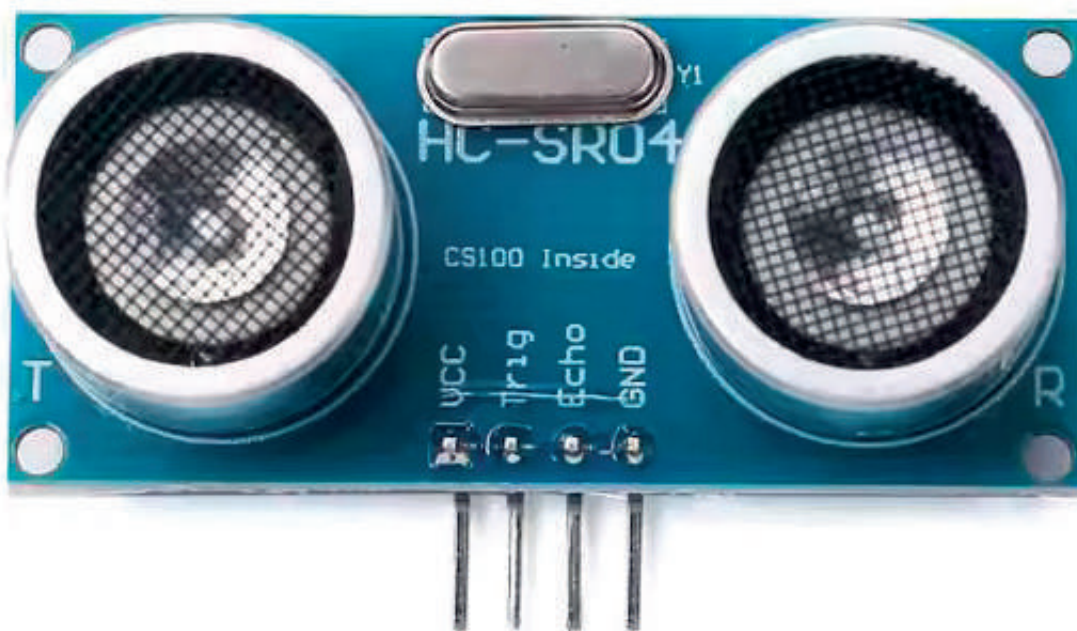
### საჭირო ბიბლიოთეკა

- LiquidCrystal

## როგორ გვემოგვს HC-SR04 ულტრაბგერის სენსორი

ამ პროექტში ჩვენ შევქმენით მარტივ ულტრაბგერით მანძილმზომს ეკრანით, რომელზეც გამო-  
ისახება მანძილი ობიექტამდე (სენსორის მიერ აღქმული იქნება მისგან მაქსიმუმ 5 მეტრით დაშო-  
რებული ობიექტები).

სენსორი აგზავნის ულტრაბგერის ნაკადს და უსმენს ობიექტიდან არეკლილ ექოს. არდუინო აგ-  
ზავნის Trig სენსორის კონტაქტზე მოკლე იმპულსს, რომ გაგზავნოს ულტრაბგერითი ტალღა. შემდეგ  
უსმენს იმპულსს Echo-ს კონტაქტზე პულსენის ფუნქციით. იმპულსის გაგზავნასა და მიღებას შორის  
დრო ტოლია ულტრაბგერის მიერ ობიექტამდე და უკან, სენსორამდე გავლილი დროისა. არდუინო  
ამ დროს გარდაქმნის მანძილში და გამოსახავს მას LCD ეკრანზე.



სურათი 1. HC-SR04 ულტრაბგერის სენსორი

LCD ეკრანი (თხევადკრისტალური ეკრანი) დამზადებულია პოლარიზებული მასალის ორი ფურ-  
ცლისგან, რომელთა შორის მოთავსებულია თხევადკრისტალური ხსნარი. დენის გავლა ხსნარში  
ქმნის გამოსახულებას, ჩვენ შემთხვევაში, სიმბოლოებს. ამ პროექტისთვის დაგჭირდებათ Hitachi  
HD44780 დრაივერთან თავსებადი LCD ეკრანი (16X2, 16 საკონტაქტო პინით). ეკრანზე სიმბოლოების  
გადასაცემად ჩვენ გამოვიყენებთ LiquidCrystal ბიბლიოთეკას, რომელიც უკვე არის ჩაშენებული არ-  
დუინოს IDE ინტერფეისში.

## აპაჩოტსკევა

1. მოამზადეთ LCD ეკრანი ( საკონტაქტო ხვრელებში მიარჩილეთ გამტარები).

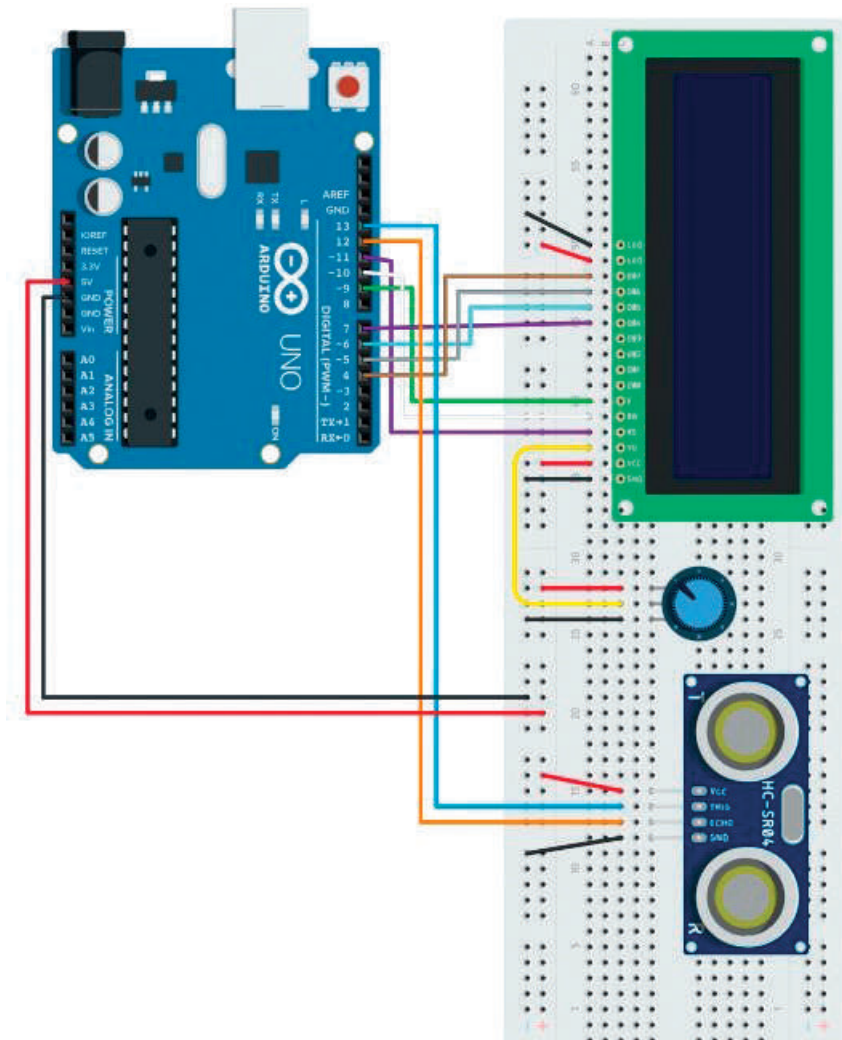
მოათავსეთ ცვლადი წინაღობა სამაკეტო დაფაზე და გამტარების საშუალებით დააკავშირეთ ერ-  
თმანეთთან LCD ეკრანი, ცვლადი წინაღობა და არდუინო ისე, როგორც ქვემოთ მოცემულ ცხრილ-  
შია ნაჩვენები.

| LCD ეკრანი    | ARDUINO                          |
|---------------|----------------------------------|
| 1 VSS         | GND                              |
| 2 VDD         | +5V                              |
| 3 VO contrast | ცვლადი წინაღობის ცენტრალური პინი |
| 4 RS          | Pin 11                           |
| 5 R/W         | Pin 10                           |
| 6 Enable      | Pin 9                            |
| 7 D0          | არ ვიყენებთ                      |
| 8 D1          | არ ვიყენებთ                      |
| 9 D2          | არ ვიყენებთ                      |
| 10 D3         | არ ვიყენებთ                      |
| 11 D4         | Pin 7                            |
| 12 D5         | Pin 6                            |
| 13 D6         | Pin 5                            |
| 14 D7         | Pin 4                            |
| 15 A BcL+     | +5V                              |
| 16 K BcL–     | GND                              |

2. ცვლადი წინაღობის ცენტრალური პინი უკვე უნდა გქონდეთ დაკავშირებული LCD ეკრანთან (V0). ახლა მისი ორი განაპირა პინი დააკავშირეთ GND-სა და +5V-თან. ცვლადი წინაღობის რეგულატორით არეგულირებთ ეკრანის კონტრასტულობას.
3. მოათავსეთ ულტრაბგერითი სენსორი სამაკეტო დაფაზე და დააკავშირეთ მისი VCC პინი +5V-თან, Echo – არდუინოს მე-12 პინთან, GND – GND-თან.

| ულტრაბგერის სენსორი | ARDUINO |
|---------------------|---------|
| VCC                 | +5V     |
| Trig                | Pin 13  |
| Echo                | Pin 12  |
| GND                 | GND     |

4. დააკავშირეთ სამაკეტო დაფის კვების ლიანდაგები არდუინოს +5V-სა და GND-თან.
5. დარწმუნდით, რომ თქვენ მიერ აწყობილი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 2) და ჩატვირთეთ კოდი.

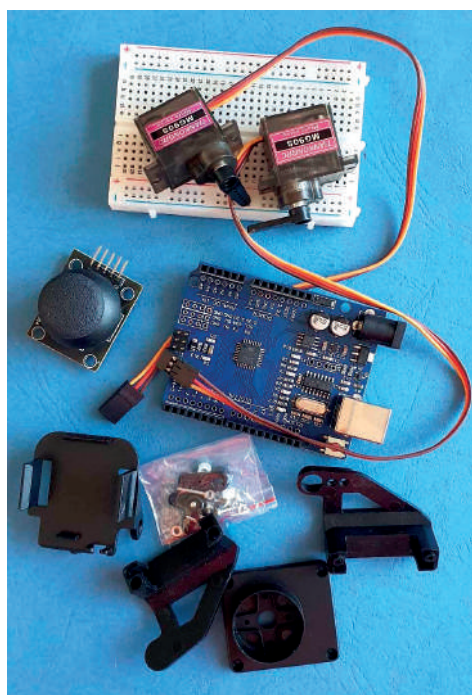
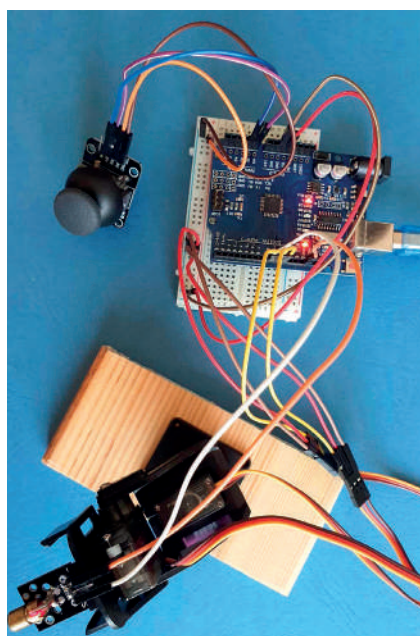
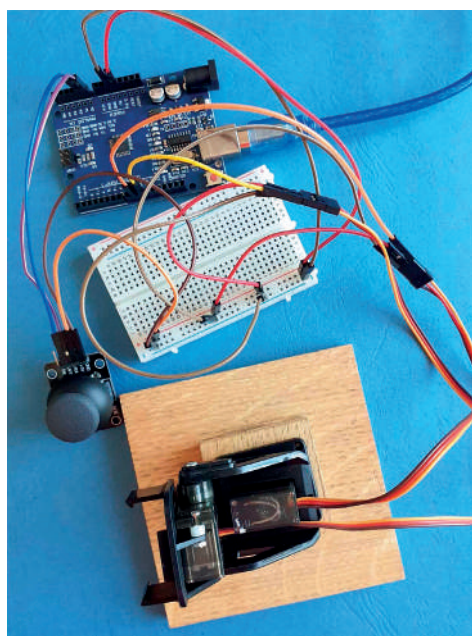


სურათი 2. სქემა

## როგორ გუზავს კოდი

თავდაპირველად კოდი მიმართავს ბიბლიოთეკას LiquidCrystal და განსაზღვრავს არდუინოსთან დაკავშირებულ LCD ეკრანის კონტაქტებს. არდუინოს მე-13 კონტაქტი, რომელიც დაკავშირებულია სენსორის Trig კონტაქტთან, აგზავნის ულტრაბერით სიგნალს, ხოლო არდუინოს მე-12 კონტაქტი, რომელიც დაკავშირებულია სენსორის Echo კონტაქტთან, იღებს უკან დაბრუნებულ სიგნალს. არდუინო გაგზავნილ და დაბრუნებულ სიგნალებს შორის არსებულ დროის ინტერვალს გარდაქმნის მანძილში და შედეგი გამოაქვს LCD ეკრანზე (სანტიმეტრებსა და დიუმებში).

# SERVO ძრავების მართვა ჯოისტიკით



## საჭირო რესურსი

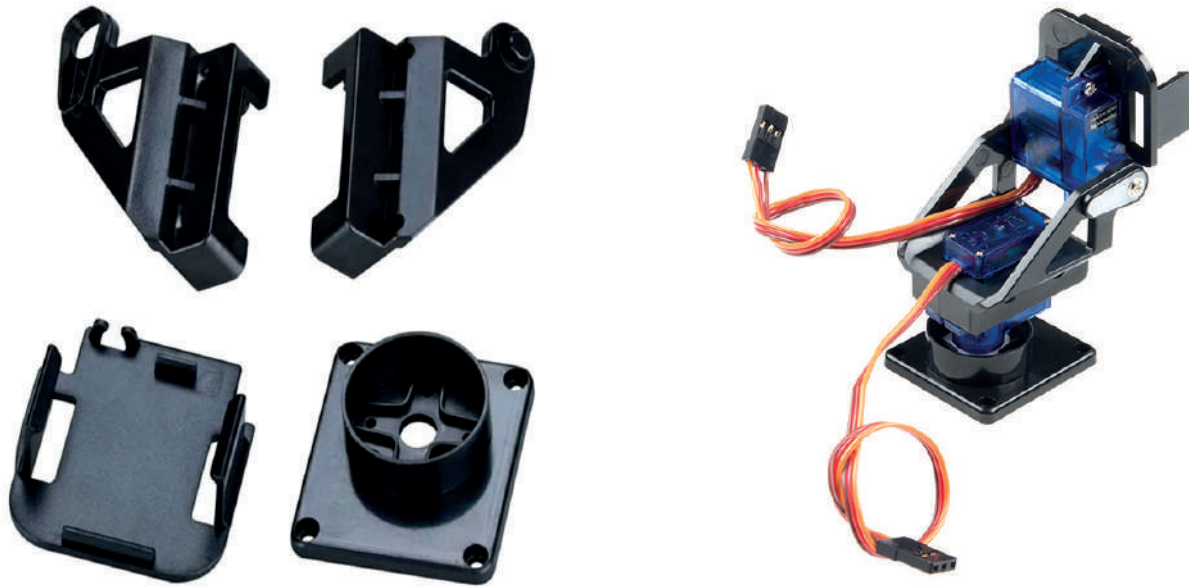
- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 2 ცალი Tower Pro SG90 9g Servo ძრავა
- ორღერძიანი ჯოისტიკის 5-კონტაქტიანი ანალოგური მოდული
- დახრა-მობრუნების კორპუსის მოდული

## საჭირო ბიბლიოთეკა

- Servo

## როგორ მუშაობს პროექტი

Servo – ეს პატარა ძრავაა, რომლის მხარსაც შეუძლია მოტრიალება 0-დან 180 გრადუსამდე. ამ პროექტში Servo ძრავებს მოვათავსებთ დახრა-მობრუნების კორპუსის მოდულში. მისი დამზადება შესაძლებელია როგორც 3D პრინტერზე, ასევე ხელით, მაგ.: მუყაოთი. (იხილეთ სურათი 1).



სურათი 1. დახრა-მობრუნების კორპუსის მოდული

ჩვენ დაგვჭირდება 2 ცალი Servo ძრავა: ერთი – მარჯვნივ და მარცხნივ მოსაბრუნებლად, მეორე – ზემოთ და ქვემოთ დახრისთვის. ძრავას აქვს სამი გამტარი (იხილეთ სურათი 2): დადებითი – წითელი, უარყოფითი (GND) – შავი ან ყავისფერი და სიგნალი – (ყვითელი, ნარინჯისფერი ან თეთრი).



სურათი 2. Servo ძრავა

ვიდრე პროექტის აწყობას დაიწყებთ, ცოტა რამ უნდა იცოდეთ ჯოისტიკის შესახებ (იხ. სურათი 3). ჯოისტიკი, ძირითადად, შედგება ორი პოტენციომეტრისგან და ლილაკისგან, რომლებიც საშუალებას გვაძლევენ ორ განზომილებაში გამოძრაოთ ობიექტი.



სურათი 3. ჯოისტიკის მოდული

პოტენციომეტრი ცვლადი წინაღობაა და მოქმედებს როგორც სენსორი, რომელიც გვაწვდის ძაბვას და ცვლის მას ღერძის გარშემო ჯოისტიკის ტრიალით. როცა ჯოისტიკს ამოძრავებთ თავისი ცენტრის გარშემო, მისი წინაღობა და შესაბამისად, გამომავალი სიგნალიც, იცვლება. პოტენციომეტრის გამომყვანები ანალოგურია, ამიტომ, როცა ხდება მათი წაკითხვა არდუინოს ანალოგური გამომყვანით, მათი მნიშვნელობა არის 0-სა და 1023-ს შორის. ეს რიცხვი აგზავნის იმპულსს არდუინოზე, რომელიც, თავის მხრივ, ატყობინებს Servo ძრავებს, როგორ იმოძრაონ.

ჯოისტიკს აქვს ხუთი საკონტაქტო ფეხი: VRx (x ღერძის სიგნალი), Vry (y ღერძის სიგნალი), SW (რომელსაც ამ პროექტში არ ვიყენებთ), GND (მიწა) და +5V კვება. როცა ჯოისტიკის ღერძი იმოძრავებს მარჯვნივ ან მარცხნივ, შესაბამისი ძრავაც იგივე მიმართულებით იმოძრავებს; როცა ჯოისტიკის ღერძი იმოძრავებს ზევით ან ქვევით, მეორე ძრავაც იმოძრავებს შესაბამისად – ზევით ან ქვევით.

### აპაწყოთ სქემა

1. დააკავშირეთ ორივე Servo ძრავას წითელი გამტარები +5V-თან, ხოლო ყავისფერი გამტარები სამაკეტო დაფის GND-თან.
2. ერთი Servo ძრავას ყვითელი გამტარი დააკავშირეთ არდუინოს მეცხრე საკონტაქტო პინთან, ხოლო მეორე Servo ძრავას ყვითელი გამტარი – არდუინოს მე-10 საკონტაქტო პინთან.

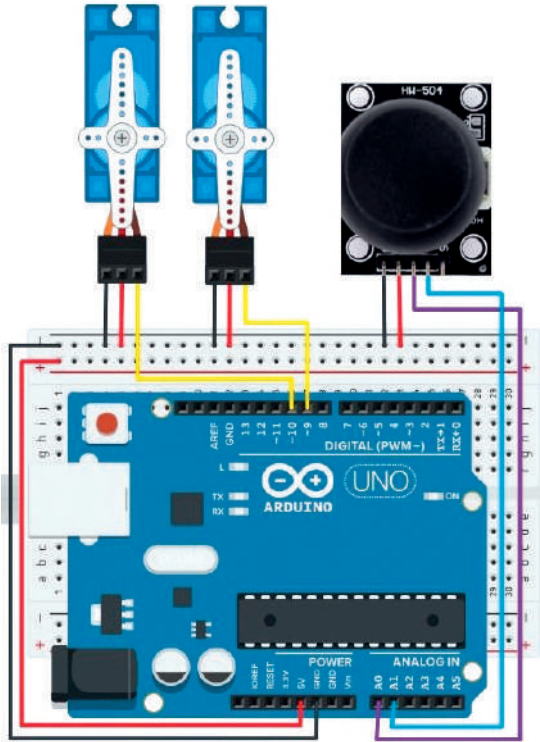
| Servo ძრავები     | ARDUINO |
|-------------------|---------|
| წითელი გამტარი    | +5V     |
| ყავისფერი გამტარი | GND     |
| ყვითელი გამტარი 1 | Pin 9   |
| ყვითელი გამტარი 2 | Pin 10  |

3. ჯოისტიკის მოდულის GND დააკავშირეთ არდუინოს GND-თან, +5V – არდუინოს +5V-თან. VRx პინი დააკავშირეთ უშუალოდ არდუინოს A0 პინთან და Vry პინი – უშუალოდ არდუინოს A1 პინთან. SW გადამრთველს ამ პროექტში არ ვიყენებთ.

| ჯოისტიკი | ARDUINO     |
|----------|-------------|
| +5V      | +5V         |
| GND      | GND         |
| VRx      | A0          |
| VRy      | A1          |
| SW       | არ ვიყენებთ |

4. დააკავშირეთ სამაკეტო დაფის დადებით- და უარყოფითპოლუსიანი ლიანდაგი არდუინოს +5V-თან და GND-თან.

დააკვირდით, ზუსტად შეესაბამება თუ არა თქვენ მიერ აწყობილი სქემა მეოთხე სურათზე გამოსახულ სქემას.



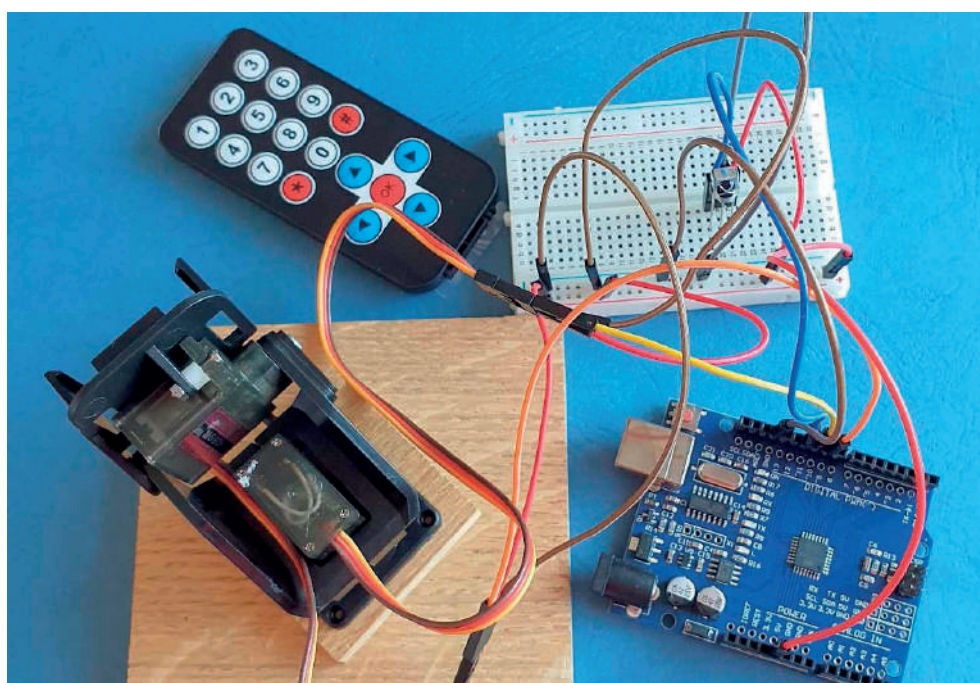
სურათი 4. სქემა

ჯოისტიკის მარცხნივ და მარჯვნივ გადაადგილებით მოძრაობს Servo x-ღერძით, ხოლო ჯოისტიკის ზევით და ქვევით გადაადგილებით მოძრაობს Servo y-ღერძით.

### როგორ მუშაობს კოდი

კოდი ჯერ მიმართავს Servo ბიბლიოთეკას, შემდეგ განსაზღვრავს ორ Servo ძრავას, როგორც დახრასა და მობრუნებას. ჯოისტიკის x ღერძი დაკავშირებულია არდუინოს A0 კონტაქტთან, ხოლო y ღერძი – არდუინოს A1 კონტაქტთან. ეს ორი საკონტაქტო პინი არის ჩვენი INPUT. შემდეგ x და y ღერძები განისაზღვრებიან, როგორც მოძრაობის ცვლადები. დახრის Servo ძრავა დაკავშირებულია არდუინოს მე-9 პინთან, ხოლო მობრუნების – არდუინოს მე-10 პინთან. ეს ორი საკონტაქტო პინი არის ჩვენი OUTPUT. შემდეგ, არდუინო ჯოისტიკიდან კითხულობს INPUT-ს და ამ ძაბვას ცვლის OUTPUT-ზე, ამოძრავებს რა Servo ძრავებს არჩეული მიმართულებიდან გამომდინარე.

# SERVO ძრავების მართვა დისტანციურად

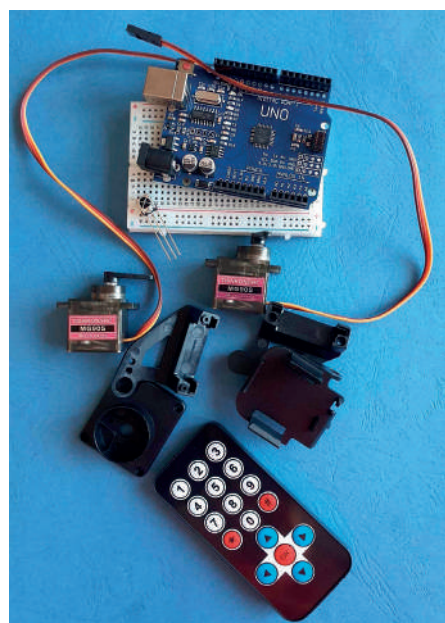


## საჭირო რესურსი

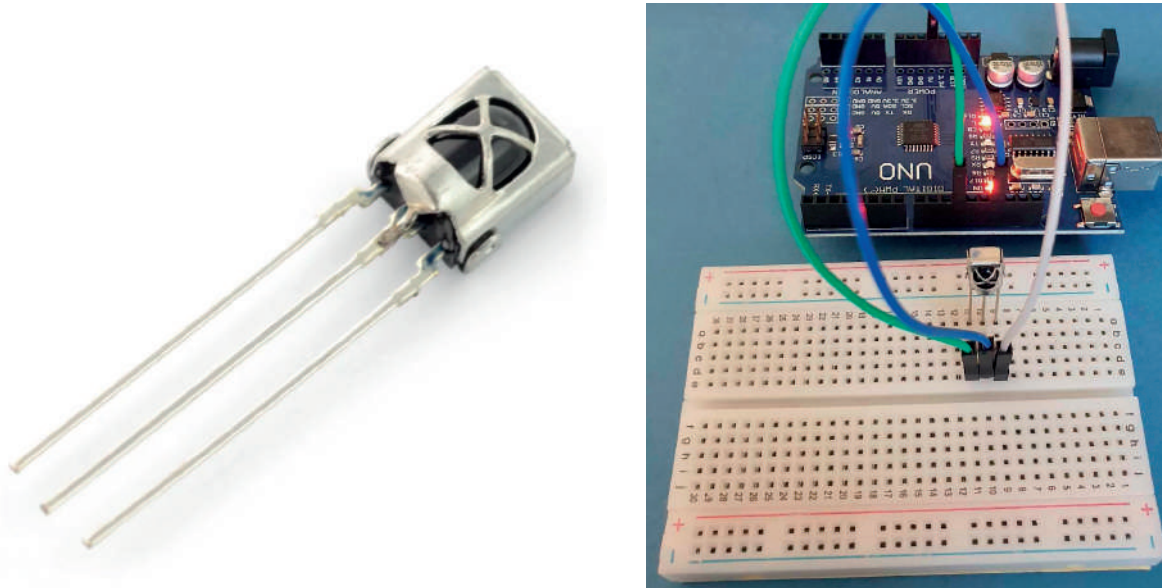
- Arduino UNO
- სააკეტო დაფა
- გამტარები
- 38 kHz ინფრაწითელი მიმღები
- დისტანციური მართვის პულტი
- 2 ცალი Tower Pro SG90 Servo ძრავა
- დახრა-მობრუნების კორპუსის მოდული

## საჭირო ბიბლიოთეკები

- Servo
- I2Cremote



პროექტზე მუშაობას დავიწყებთ დისტანციური მართვის პულტის დეკოდირებით ინფრაწითელი მიმღების საშუალებით. მიმღებს აქვს სამი გამომყვანი: OUT, GND და VCC (იხ. სურათი 1). შეამოწმეთ თქვენი მიმღების სპეციფიკაციები, რომ მისი გამომყვანების განლაგება ჩვენსას შეესაბამებოდეს.



სურათი 1. ინფრაწითელი მიმღების გამომყვანები მარცხნიდან მარჯვნივ: OUT, GND და VCC

ასევე, დაგჭირდებათ დისტანციური მართვის პულტი. დისტანციური მართვის პულტის ღილაკზე დაჭერისას ის აგზავნის ციფრულ მნიშვნელობას, რომელსაც იღებს ინფრაწითელი მიმღები. ეს მნიშვნელობა განსხვავებულია ყველა ღილაკისთვის. ჩვენ გავშიფრავთ თითოეული ღილაკის მნიშვნელობას Arduino-ს საშუალებით და შემდეგ მივანიჭებთ მათ Arduino-ს პინებს კოდში, რათა გავაკონტროლოთ გამოძვარი – ამ შემთხვევაში, Servo ძრავა.

თქვენ მიერ გაშიფრული მნიშვნელობებით, პერსონალიზებული კოდის საშუალებით, შეგიძლიათ დააკავშიროთ გარკვეული ღილაკები გარკვეულ ინსტრუქციებთან და გამოიყენოთ დისტანციური პულტი ძრავების სამართავად. თუ თქვენ უკვე შექმენით დახრა-მობრუნების კორპუსის მოდელი პროექტში „Servo ძრავების მართვა ჯოისტიკით“, შეგიძლიათ ხელახლა გამოიყენოთ იგი აქ. წინააღმდეგ შემთხვევაში, დაუბრუნდით ზემოხსენებულ პროექტს მისი დაყენების ინსტრუქციების სანახავად.

ჩვენ მივანიჭებთ ღილაკს Servo ძრავების მოძრაობის მიმართულებას დახრა-მობრუნების კორპუსში. მთლიანობაში, სულ ოთხი ღილაკი გააკონტროლებს ყველა მოძრაობას: მარცხნივ და მარჯვნივ x-ღერძისთვის და ზევით და ქვევით – y-ღერძისთვის.

## დისკანსიური მართვის პულტის ღილაკების დაყენება

1. ჩამოტვირთეთ IIRemote ბიბლიოთეკა და ჩატვირთეთ არდუინოს IDE ინტერფეისში (ბიბლიოთეკების დაყენების ინსტრუქციები იხილეთ თავში „არდუინოს დამატებითი ბიბლიოთეკების ინსტალაცია“).

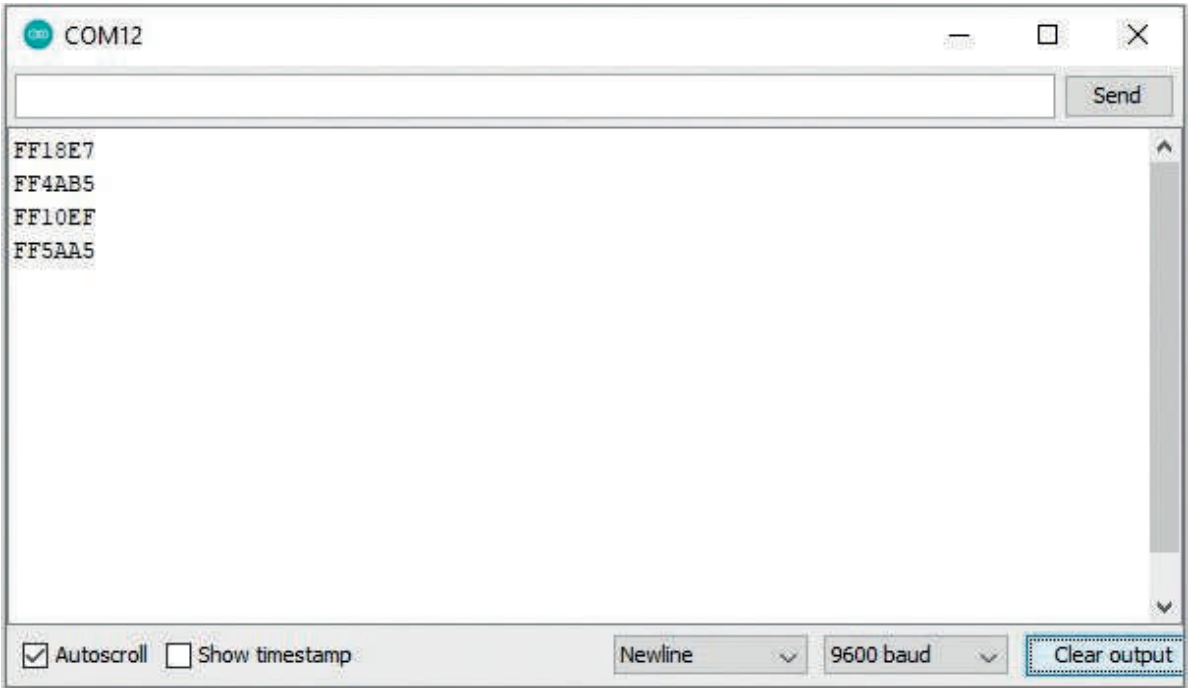
| ინფრაწითელი მიმღები | ARDUINO |
|---------------------|---------|
| OUT                 | Pin 11  |
| GND                 | GND     |
| VCC                 | +5V     |

2. დაამაგრეთ ინფრაწითელი მიმღები სამაკეტო დაფაზე. დააკავშირეთ მიმღების კონტაქტები არდუინოსთან: OUT კონტაქტი არდუინოს მე-11 პინთან, GND – GND-თან, VCC – +5V-თან. შეგახსენებთ, რომ თქვენი მიმღების კონტაქტები შეიძლება განსხვავებული მიმდევრობის იყოს, ამიტომ, შეამოწმეთ მისი სპეციფიკაციები და ისე დააკავშირეთ არდუინოსთან.
3. ჩატვირთეთ Project\_8 ა კოდი (დისტანციური მართვის წამკითხველი).

**როგორ გუზაოგს ლისპანსუარი მართვის წამკითხველი კოდი**

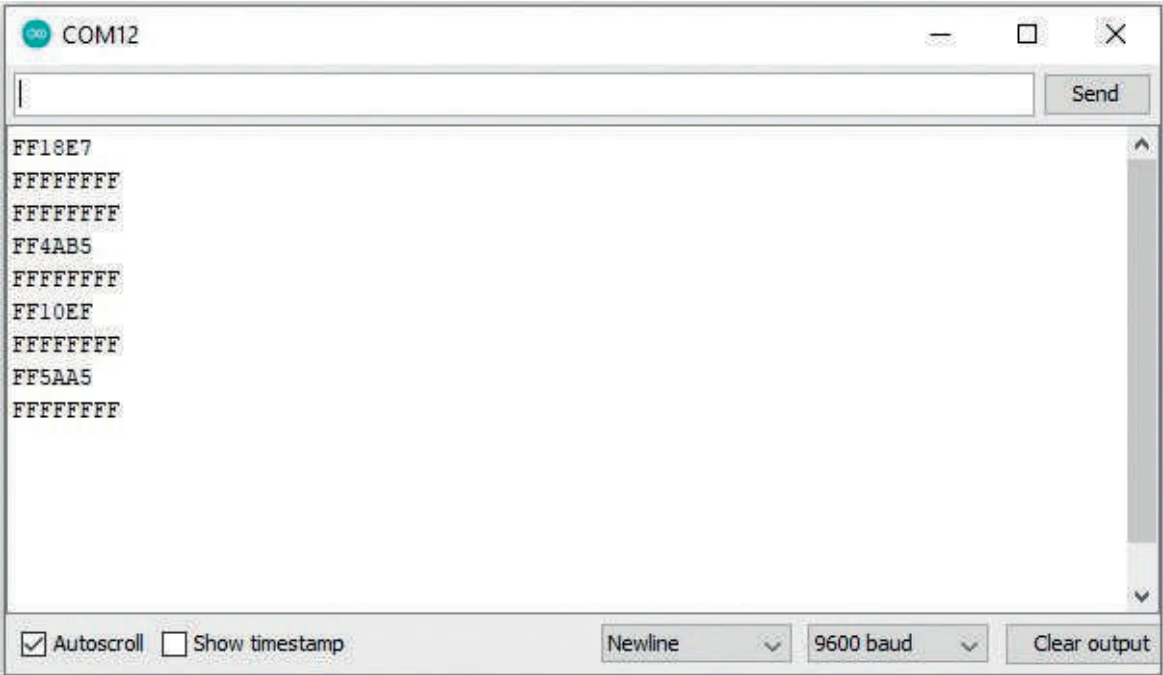
თავდაპირველად კოდი მიმართავს IRremote ბიბლიოთეკას, რომელიც კითხულობს მონაცემებს ინფრაწითელი მიმღებიდან და შესაბამის მონაცემებს აგზავნის არდუინოზე. მიმღებს ვანიჭებთ არდუინოს მე-11 საკონტაქტო პინს და კოდი იწყებს არდუინოს ინტერფეისთან ურთიერთქმედებას, რათა ღილაკზე დაჭერისას შემომავალი მონაცემები რეალურ დროში აისახოს მიმდევრობითი კომუნიკაციის მონიტორზე (Serial Monitor). კოდი მუშაობს ციკლურად და ღილაკზე დაჭერისას გვიჩვენებს მის შესაბამის მნიშვნელობას.

4. გახსენით არდუინოს ინტერფეისში მიმდევრობითი კომუნიკაციის მონიტორი (Serial Monitor).
5. მიმართეთ პულტი მიმღებისკენ და დააჭირეთ სხვადასხვა ღილაკს. მათი დეკოდირებული მნიშვნელობები გამოჩნდება მონიტორზე (იხ. სურათი 2).



სურათი 2. ღილაკების დეკოდირებული მნიშვნელობები

ჩაიწერეთ, რომელ ლილაკს რომელი მნიშვნელობა შეესაბამება. ეს მნიშვნელობები მოგვიანებით დაგჭირდებათ. თუ ლილაკებს ხანგრძლივად დააჭერთ, სურათზე გამოსახული მნიშვნელობების გარდა, სხვა ტექსტსაც დაინახავთ, მაგრამ თქვენ მხოლოდ მეორე სურათზე გამოსახული მსგავსი კონფიგურაციის მნიშვნელობები უნდა ჩაინიშნოთ. ყველა პულტისთვის ეს მნიშვნელობები განსხვავებულია (იხ. სურათი 3).



სურათი 3. როცა ლილაკს ხანგრძლივად ვაჭერთ

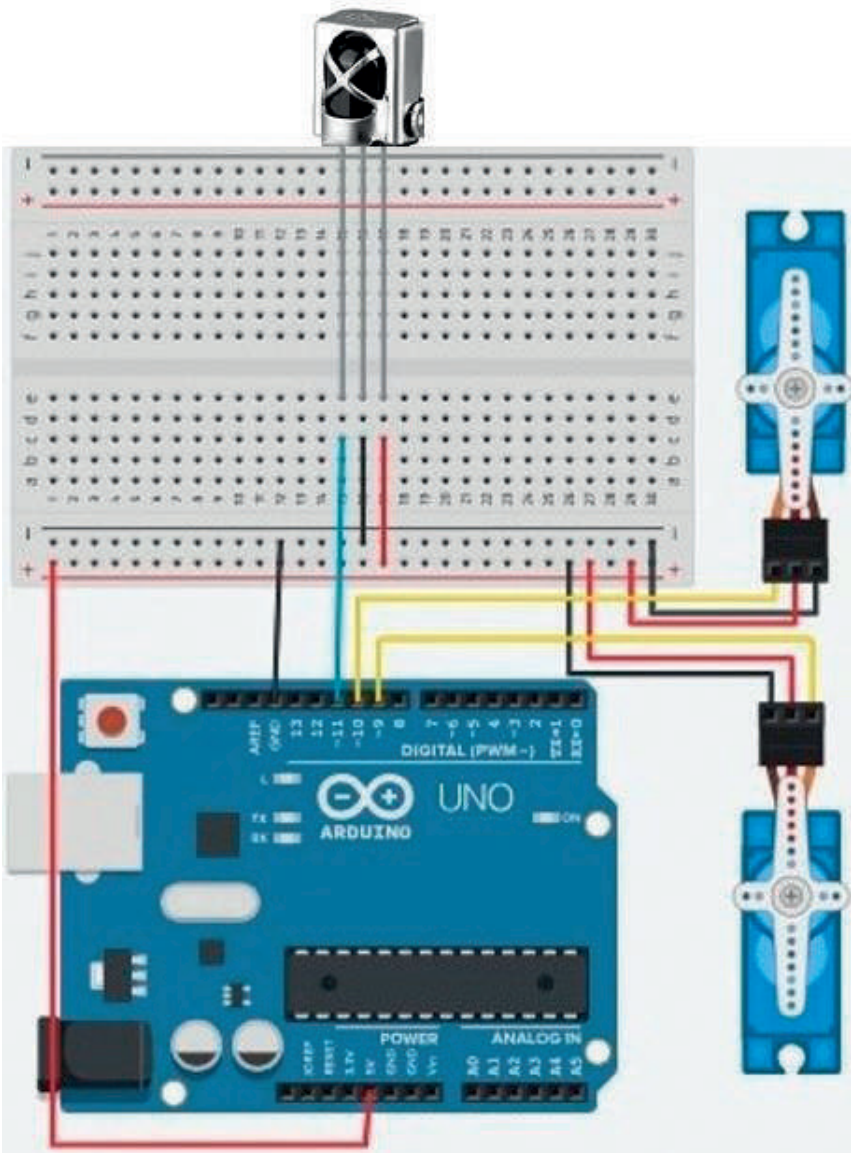
ახლა უკვე, როცა გავშიფრეთ დისტანციური მართვის პულტის ლილაკების მნიშვნელობები, ჩვენ შეგვიძლია მათი გამოყენება ორი Servo ძრავას სამართავად.

აპანუოთსკეპა

- 1. სამაკეტო დაფის გამოყენებით, რომელზეც უკვე გვაქვს მოთავსებული ინფრაწითელი მიმღები, დააკავშირეთ Servo ძრავები არდუინოსთან. ძრავების ყავისფერი გამტარები – არდუინოს GND-თან, წითელი – +5V-თან. ძრავების ყვითელი გამტარები კი დააკავშირეთ ერთი – არდუინოს მე-10 და მეორე – არდუინოს მე-9 საკონტაქტო პინებთან.

| Servo ძრავები             | ARDUINO |
|---------------------------|---------|
| წითელი გამტარი            | +5V     |
| ყავისფერი გამტარი         | GND     |
| ყვითელი გამტარი (servo 1) | Pin 10  |
| ყვითელი გამტარი (servo 2) | Pin 9   |

- 2. არ დაგავიწყდეთ კვების მიწოდება სამაკეტო დაფისთვის.
- 3. შეამოწმეთ, რომ ზუსტად შეესაბამებოდეს თქვენ მიერ აწყობილი სქემა ჩვენს სქემას (იხ. სურათი 4) და ჩატ-ვითრეთ Project\_8 ხ კოდი.



სურათი 4. სქემა

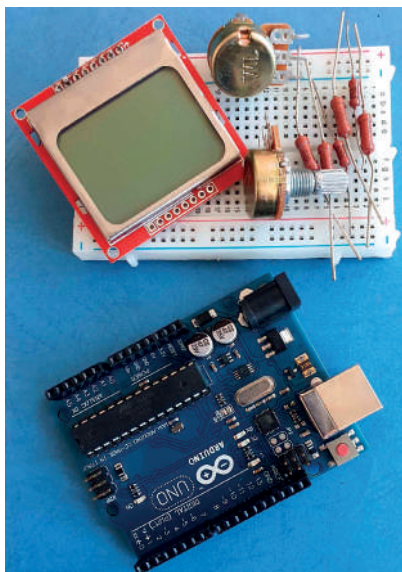
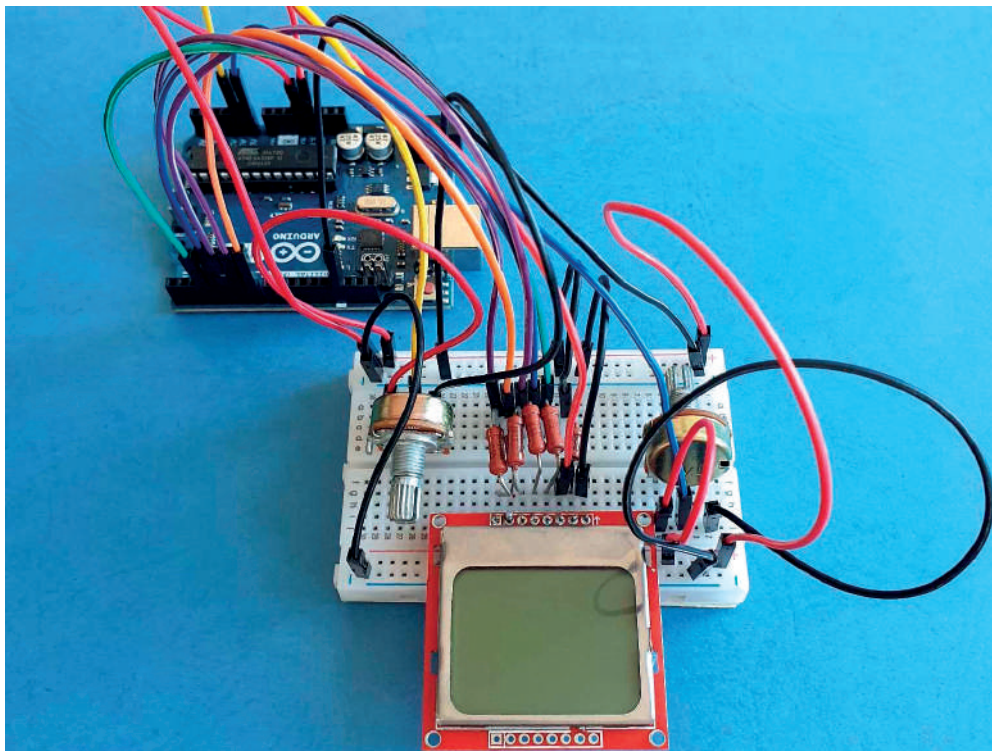
როგორ გუშაოვს კოდი

კოდის ჩატვირთვამდე დარწმუნდით, რომ ჩვენი მნიშვნელობები ჩაანაცვლეთ თქვენ მიერ გაშიფრული დისტანციური მართვის პულტის ღილაკების დეკოდირებული მნიშვნელობებით. მნიშვნელობების შეცვლისას ვინარჩუნებთ 0x-ს და მხოლოდ შემდეგ ვამატებთ ჩვენს მონაცემებს. მაგალითად, ჩვენი პირველი ღილაკის დეკოდირებული მნიშვნელობაა FF18E7, კოდში კი არის:

```
-----
Unsigned long Value1 = 0xFFA05F;
-----
```

ინფრაწითელი მიმღებიდან მონაცემების წასაკითხად კოდი მიმართავს IIRemote ბიბლიოთეკას, ხოლო Servo ძრავების სამართავად – Servo ბიბლიოთეკას.

# NOKIA 5110 LCD ეკრანი



## საჭირო რესურსი

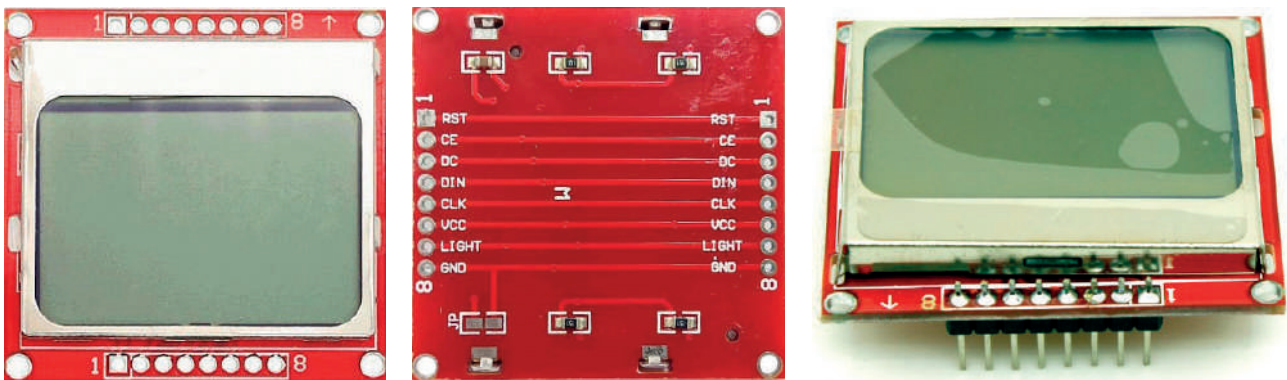
- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- Nokia 5110 LCD ეკრანი
- 2 ცალი 50 ომი ცვლადი წინაღობა (პოტენციომეტრი)
- 4 ცალი 10 კომი წინაღობა
- 2 ცალი 1 კომი წინაღობა

## როგორ მუშაობს NOKIA 5110 LCD ეკრანი

ამ პროექტში გიჩვენებთ, როგორ დავაკავშიროთ Nokia 5110-ის თხევადკრისტალური ეკრანი არდუინოსთან და შევქმნათ მარტივი თამაში. რამდენიმე წლის წინ Nokia-ს ეს ეკრანები გამოიყენებოდა ტელეფონებში. ჩვენ მას დავაკავშირებთ არდუინოსთან და ორი ცვლადი წინააღობის (პოტენციომეტრის) საშუალებით შევქმნით მაგიდის ჩოგბურთის მსგავს თამაშს, სახელწოდებით Pong Game.

Nokia 5110 LED ეკრანი ფუნქციონირებს ისევე, როგორც ამავე სახელმძღვანელოს I ნაწილში განხილული 16x2 LED ეკრანი: თხევადკრისტალური ეკრანი დამზადებულია პოლარიზებული მასალის ორი ფურცლისგან, რომელთა შორის მოთავსებულია თხევადკრისტალური ხსნარი. დენის გავლა ხსნარში ქმნის გამოსახულებას, ჩვენ შემთხვევაში, სიმბოლოებს.

ვიდრე პროექტზე მუშაობას დავიწყებთ, უნდა მოვამზადოთ Nokia 5110 LED ეკრანი. ამისათვის დაგვჭირდება ეკრანის ერთ მხარეს საკონტაქტო ხვრელებში პინების მირჩილვა (იხ. სურათი 1).



სურათი 1. Nokia 5110 LCD და მისი კონტაქტები

## ყურადღება!

Nokia 5110 LCD ეკრანის კვებისთვის გამოიყენეთ არდუინოს +3V საკონტაქტო პინი და არა +5V. წინააღმდეგ შემთხვევაში, ეკრანს დააზიანებთ.

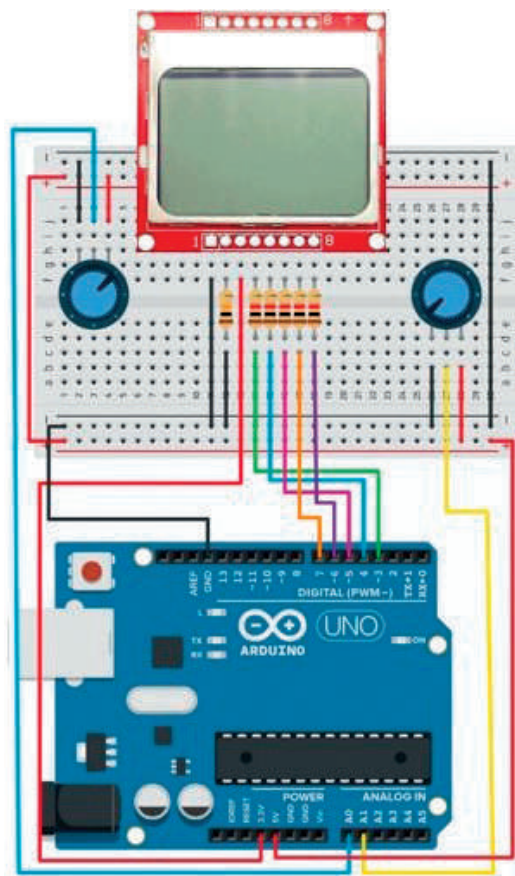
## აუწყობო სქემა

1. Nokia 5110 LCD ეკრანი დაამაგრეთ სამაკეტო დაფაზე.
2. Nokia-ს ეკრანს აქვს რვა გამომყვანი. 10 კომი წინააღობები დააკავშირეთ ეკრანის 1, 3, 4 და 5 კონტაქტებს, ხოლო 1 კომი წინააღობები – ეკრანის 2 და 7 კონტაქტებს. წინააღობები სამაკეტო დაფაზე დაამაგრეთ დაფის ცენტრალური გამყოფი ზოლის ორივე მხარეს, ისე, როგორც ეს სქემაზეა ნაჩვენები (იხ. სურათი 2).
3. გამტარების საშუალებით დააკავშირეთ Nokia 5110 LCD ეკრანი წინააღობების გავლით არდუინოს შესაბამის საკონტაქტო პინებთან.

მიაქციეთ ყურადღება, რომელი ნომინალის წინააღობა რომელ პინს უკავშირდება. ზოგიერთი ეკრანის გამომყვანები შეიძლება განსხვავებული თანმიმდევრობის იყოს, ამიტომ შეუსაბამეთ ერთმანეთს ეკრანისა და არდუინოს გამომყვანები.

| NOKIA 5110 LCD ეკრანი | წინაღობა       | ARDUINO |
|-----------------------|----------------|---------|
| 1 RESET               | 10 კომი        | Pin 6   |
| 2 CE                  | 1 კომი         | Pin 7   |
| 3 DC                  | 10 კომი        | Pin 5   |
| 4 DIN                 | 10 კომი        | Pin 4   |
| 5 CLK                 | 10 კომი        | Pin 3   |
| 6 VCC                 | არ არის საჭირო | +3.3V   |
| 7 Light               | 1 კომი         | GND     |
| 8 GND                 | არ არის საჭირო | GND     |

- დაამაგრეთ ორივე ცვლადი წინაღობა სამაკეტო დაფაზე. ერთის ცენტრალური გამომყვანი დააკავშირეთ არდუინოს A0 პინთან, ხოლო მეორის – A1 პინთან. ორივე წინაღობის თითო გარე პინი დააკავშირეთ სამაკეტო დაფის დადებითპოლუსიან ლიანდაგს (+5V), ხოლო მეორე გარე პინი – სამაკეტო დაფის უარყოფითპოლუსიან ლიანდაგს (GND).
- სამაკეტო დაფის კვების ლიანდაგები – +5V და GND დააკავშირეთ არდუინოს შესაბამის პინებთან (ეს მხოლოდ ცვლადი წინაღობებისთვის. შეგახსენებთ, რომ Nokia 5110 LCD ეკრანს მხოლოდ +3V ესაჭიროება).
- დარწმუნდით, რომ ყველა შეერთება სწორად გაქვთ გაკეთებული და თქვენი სქემა ზუსტად შეესაბამება ჩვენს სქემას. მხოლოდ ამის შემდეგ ჩატვირთეთ კოდი.



სურათი 2. სქემა

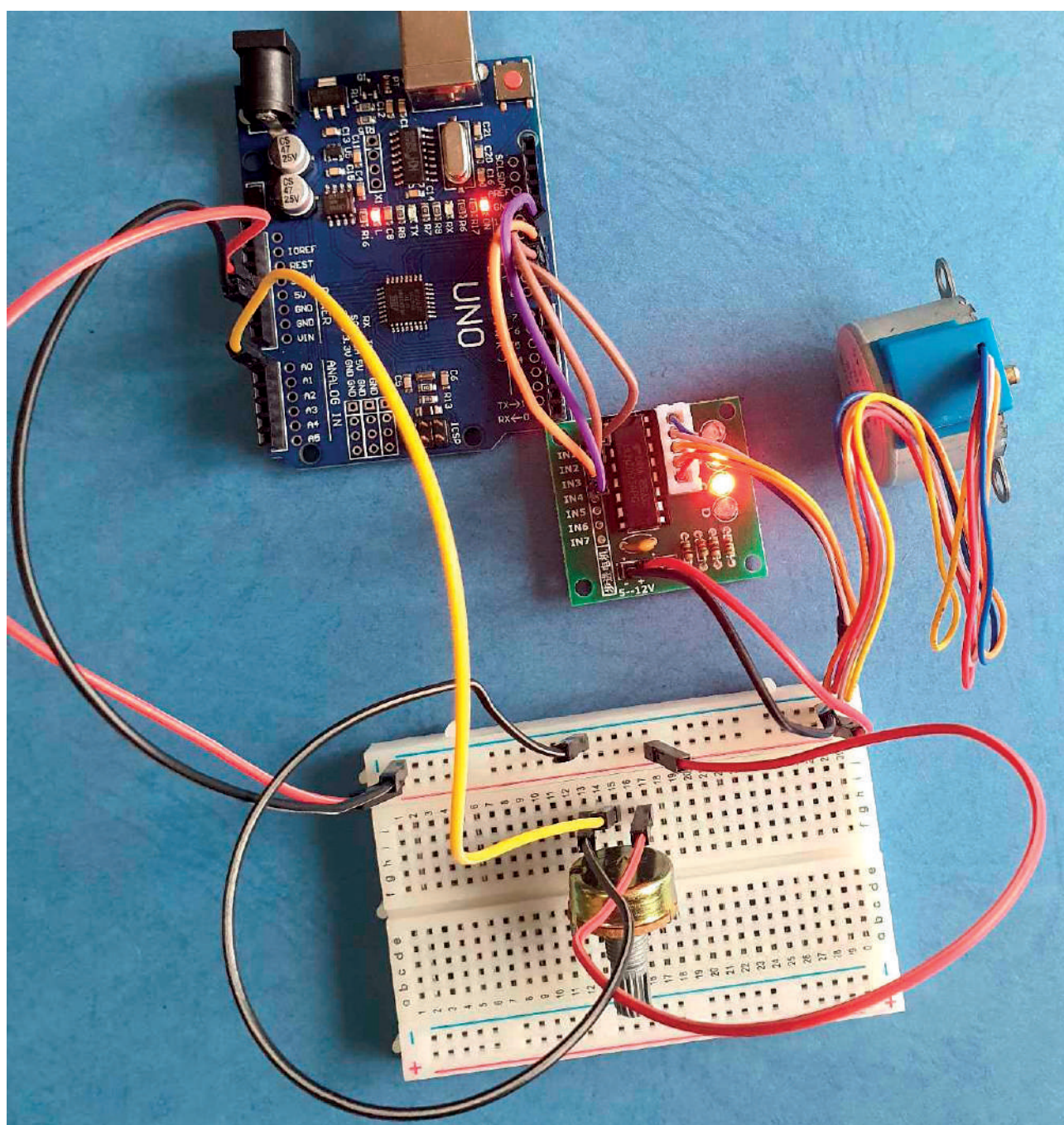
## როგორ გუზავს კოდი

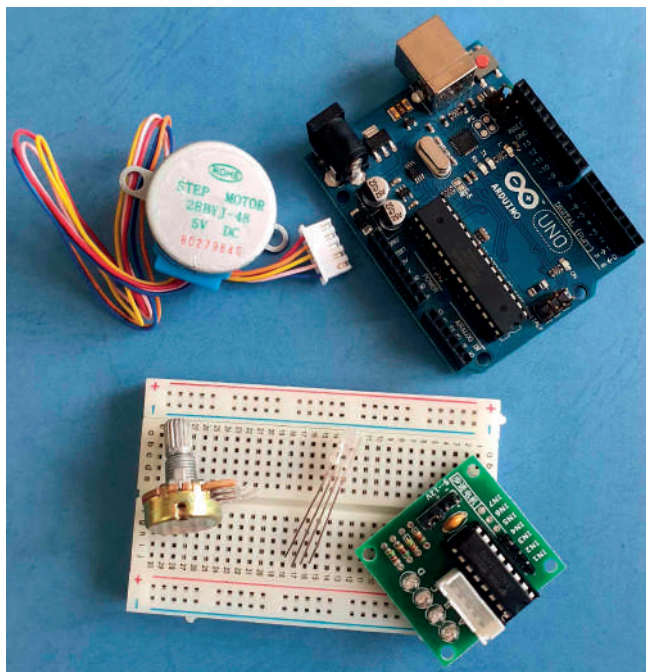
თამაში იწყება ორი მოკლე ზოლით ეკრანის ორ საპირისპირო მხარეს და მათ შორის მოძრაობა ბურთით. თამაშის მიზანია ცვლადი წინააღობების გამოყენება ზოლების გადასადგილებლად, ბურთის წინ დასახვედრად, რათა ის არ გასცდეს თამაშის საზღვრებს (ეკრანის პერიმეტრს მიღმა). ბურთი ეხება ზოლს და თანდათან უფრო და უფრო სწრაფად იწყებს მოძრაობას. თამაში დასრულდება, თუ ბურთს ვერ დაახვედრებთ ზოლს და ის დატოვებს ეკრანს, რის შემდეგაც ეკრანი გადატრიალდება და თამაში თავიდან დაიწყება. გთხოვთ, გაითვალისწინოთ, რომ გრაფიკული შეზღუდვების გამო, ბურთი რაც უფრო სწრაფად იმოძრაავს ეკრანზე, მით უფრო მკრთალი იქნება მისი გამოსახულება.

ცვლადი წინააღობები კითხულობენ ანალოგურ სიგნალს Arduino-ს A0 და A1 გამომყვანებიდან და ეკრანზე გადაადგილებენ შესაბამის ზოლებს, იმისდა მიხედვით, თუ რა მიმართულებით გადაატრიალებთ მათ მარეგულირებელს.

X ჰარმონი

# ბიჯური ძრავა





### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 28BYJ-48 ბიჯური ძრავა ULN2003 დრაივერის მოდულით
- 50 კომი ცვლადი წინაღობა (პოტენციომეტრი)

### საჭირო ბიბლიოთეკა

- Stepper

## როგორ მუშაობს ბიჯური ძრავა

ამ პროექტში გაგაცნობთ ბიჯური ძრავას მუშაობის პრინციპს. ის ელექტროძრავაა მუდმივი დენით, რომელიც მხრის სრულ ბრუნს ანაწილებს რამდენიმე თანაბარ ბიჯად. Servo ძრავასგან განსხვავებით, რომელსაც თქვენ წინა პროექტებში გაეცანით, ბიჯური ძრავა ბრუნავს  $360^{\circ}$ -ით და მისი უპირატესობა ის არის, რომ დიდი სიზუსტით პოზიციონირებს და შეუჩერებელი ბრუნავს.



სურათი 1. ბიჯური ძრავა

ბიჯური ძრავას მახასიათებლებში მითითებულია ბიჯების რაოდენობა, რომელსაც ის ასრულებს ერთი სრული მობრუნებისას. ბიჯი – ეს არის ერთი მოძრაობა ერთი მობრუნების ფარგლებში. ძრავას ერთი სრული ბრუნისთვის ანუ 360°-ით მობრუნებისთვის სჭირდება 200 ბიჯი, ანუ ერთი ბიჯი შეადგენს 1,8 გრადუსს.

თქვენ შეგიძლიათ მართოთ ძრავას პოზიცია და სიჩქარე. რადგან ჩვენ ვიცით, როგორი კუთხე აქვს თითოეულ ბიჯს, შეგვიძლია მივიღოთ მობრუნების ზუსტი კუთხეები და მანძილები. ბიჯური ძრავები ფართოდ გამოიყენება CD და DVD აპარატებში, ასევე 3D პრინტერებში, სადაც მოძრაობები უნდა იყოს ზედმიწევნით ზუსტი.

როცა გადაწყვეტთ ბიჯური ძრავას შეძენას, აუცილებელია გაითვალისწინოთ, აქვს თუ არა მას რედუქტორი. ის აძლიერებს ბრუნვას (გადაადგილების სიძლიერეს), მაგრამ ამცირებს ბრუნვების რაოდენობას წუთში (RPM, ანუ სიჩქარე).

ფაზების გრაგნილების მიხედვით არსებობს ბიჯური ძრავების ორი ტიპი:

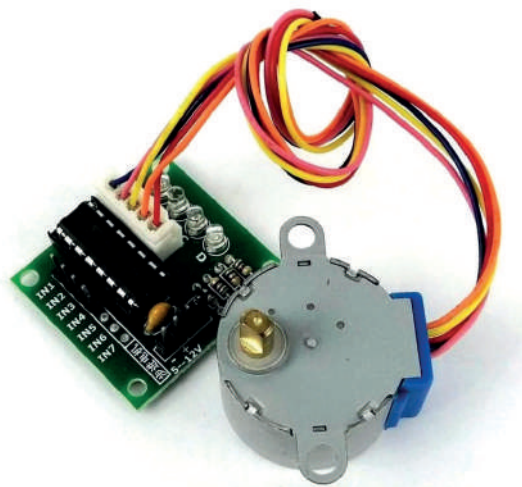
- უნიპოლარულის (ერთპოლარულის);
- ბიპოლარულის (ორპოლარულის).

პოლარულობა – ეს არის დენის მიმართულება. მაგალითად, თუ ჩვენ ადგილებს გავუცვლით +5V-სა და GND-ს, ძრავა იმოძრავებს საპირისპირო მიმართულებით. შეგიძლიათ შეამოწმოთ, უნიპოლარულია თუ ბიპოლარული თქვენი ძრავა – დააკვირდით საკონტაქტო კავშირებს: ბიპოლარულ ძრავას აქვს ოთხი კავშირი, ხოლო უნიპოლარულს – ხუთიდან რვა კავშირამდე. ამ პროექტში ჩვენ გამოვიყენებთ უნიპოლარულ ძრავას, ბიჯურ ძრავას 28BYJ-48 დრაივერის სატესტო მოდულით ULN2003 – დაფით, რომელიც საშუალებას იძლევა მარტივად ვმართოთ ძრავა არდუინოს დახმარებით.

ცვლადი წინაღობის (პოტენციომეტრი) საშუალებით თქვენ ცვლით ბიჯური ძრავას დახრის კუთხეს. ცვლადი წინაღობა – მბრუნავი თავის მქონე (რეგულატორი), სამი ტერმინალისგან შემდგარი წინაღობა, რომლის საშუალებითაც ძაბვის დარეგულირებაა შესაძლებელი. იგი ხშირად გამოიყენება ელექტრომოწყობილობებში, მაგალითად, აუდიოაპარატურებში. ცვლადი წინაღობის რეგულატორის ტრიალით მარჯვნივ და მარცხნივ, შესაძლებელია ძაბვის ნელ-ნელა გაზრდა და შემცირება.

## აპანჟოთსქემა

1. დააკავშირეთ ბიჯური ძრავა დრაივერის დაფასთან (იხ. სურათი 2). ყურადღება მიაქციეთ გამტარების თანმიმდევრობას: ლურჯი, ვარდისფერი, ყვითელი, ნარინჯისფერი, წითელი. ძრავას შეერთება დრაივერის დაფასთან მხოლოდ ასეა შესაძლებელი.



სურათი 2. ძრავას დაკავშირება დრაივერის დაფასთან

2. დააკავშირეთ დრაივერის დაფის საკონტაქტო პინები 1, 2, 3 და 4 უშუალოდ არდუინოს მე-8, 9, 10 და 11 პინებთან.

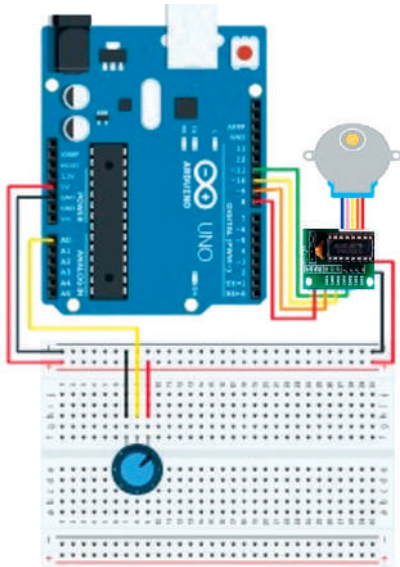
| ბიჯური ძრავას დრაივერის დაფა | ARDUINO |
|------------------------------|---------|
| IN1                          | Pin 8   |
| IN2                          | Pin 9   |
| IN3                          | Pin 10  |
| IN4                          | Pin 11  |
| GND                          | GND     |
| +5V                          | +5V     |

3. მოათავსეთ ცვლადი წინაღობა სამაკეტო დაფაზე. მისი ცენტრალური პინი დააკავშირეთ არდუინოს A0 საკონტაქტო პინთან. დანარჩენი ორი გარე პინი (მხარეებს მნიშვნელობა არ აქვს) დააკავშირეთ არდუინოს +5V-თან და GND-თან.

| ცვლადი წინაღობა | ARDUINO |
|-----------------|---------|
| მარცხენა პინი   | GND     |
| ცენტრალური პინი | A0      |
| მარჯვენა პინი   | +5V     |

4. დააკავშირეთ დრაივერის დაფის GND და +5V სამაკეტო დაფის დადებით- და უარყოფითპოლუსიან ლიანდაგებს, ხოლო სამაკეტო დაფის ეს ლიანდაგები დააკავშირეთ არდუინოსთან შესაბამისად: დაფის GND და +5V არდუინოს GND და +5V-თან.

5. დარწმუნდით, რომ თქვენი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 3) და ჩატვირთეთ კოდი.



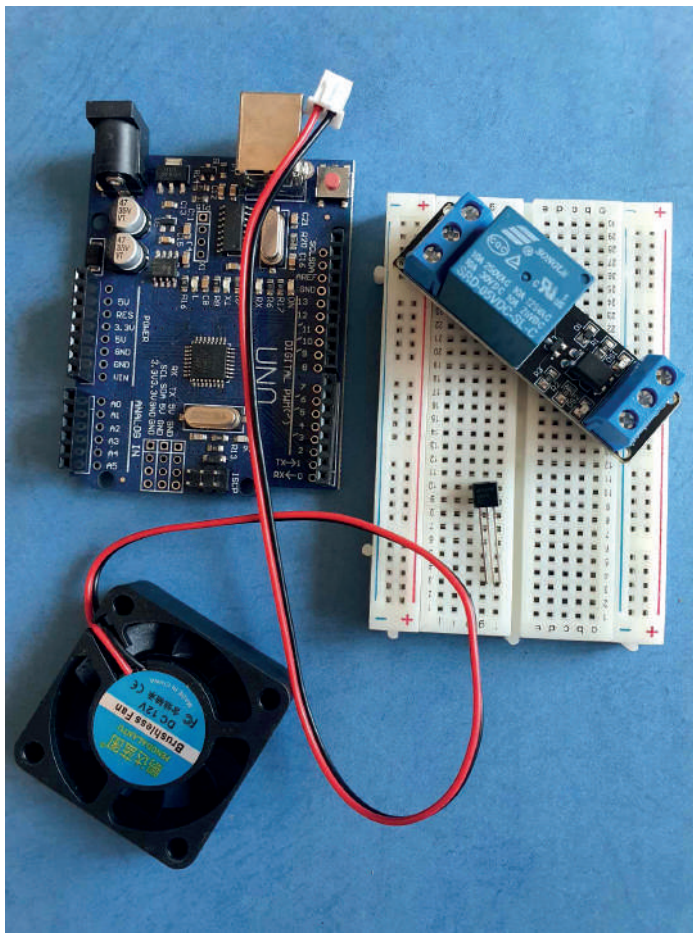
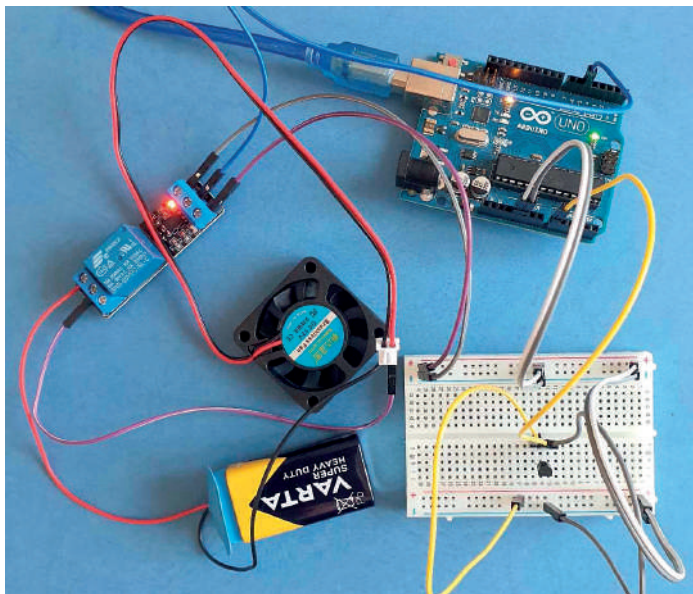
სურათი 3. ბიჯური ძრავას სქემა

ეს კოდი უკვე არის არდუინოს IDE ინტერფეისში: File/Examples/Stepper/MotorKnob. იგი იყენებს ჩაშენებულ ბიბლიოთეკას <Stepper.h>.

ცვლადი წინაღობა დაკავშირებულია არდუინოს ანალოგურ პინთან A0 და რეგულატორის საშუალებით იძლევა ცვლად ძაბვას, რომელიც მართავს ბიჯურ ძრავას.

XI პროექტი

# ტემპერატურის სენსორი LM35



## საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- LM35 ტემპერატურის სენსორი
- 5V-იანი ერთარხიანი რელეს მოდული
- კომპიუტერის 12V-იანი მინი ვენტილატორი
- 9-12V-იანი ელემენტი კვების კაბელით

## როგორ გუგოგს გამპერაჟრის სენსორი LM35

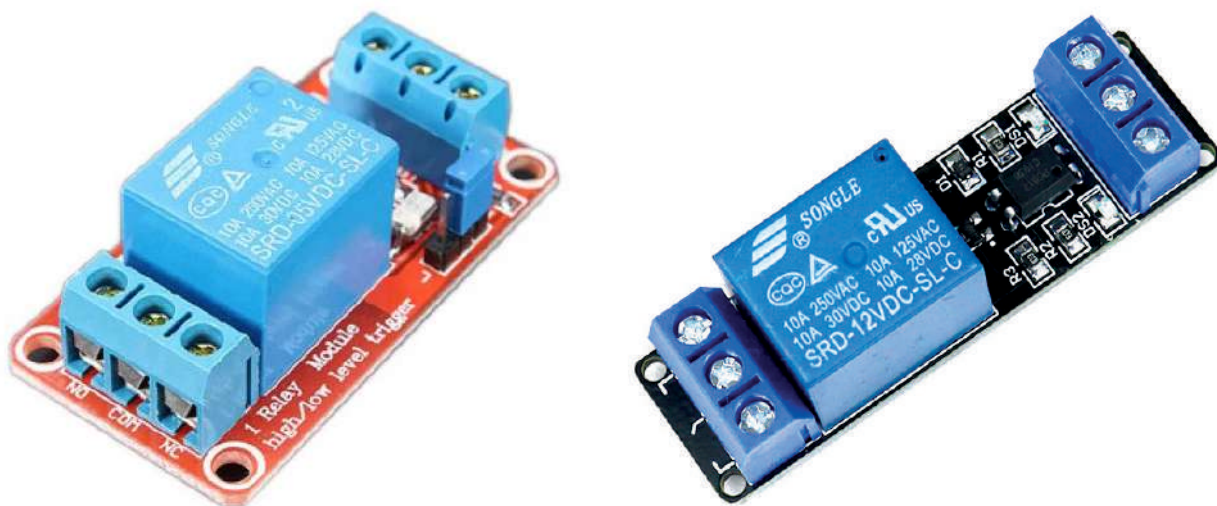
ამ პროექტში ჩვენ გამოვიყენებთ ტემპერატურის სენსორს LM35 (იხ. სურათი 1) პატარა ვენტილატორის ჩასართავად. სენსორი ზომავს ტემპერატურას და ძაბვის სახით აგზავნის არდუინოზე. არდუინო ძაბვის ამ მნიშვნელობას გარდაქმნის ტემპერატურად გრადუსებში ცელსიუსით, ხოლო შემდეგ ეს მნიშვნელობა გადაჰყავს გრადუსებში ფარენგეიტით. როცა ტემპერატურა აჭარბებს 71 გრადუსს ფარენგეიტით, არდუინო ძაბვას აგზავნის რელეზე, რომელიც ჩართავს ვენტილატორს.



სურათი 1. ტემპერატურის სენსორი LM35

(მარცხენა პინი – +5V, ცენტრალური პინი – მონაცემების გამომყვანი, მარჯვენა პინი – GND)

კომპიუტერის ვენტილატორი იყენებს უფრო მეტ სიმძლავრეს, ვიდრე არდუინოს შეუძლია მიაწოდოს მას, ამიტომ ჩვენ დავგჭირდება მისი უზრუნველყოფა საკუთარი კვების წყაროთი: 9 ვოლტიანი ელემენტი. ეს სქემა იმართება ელექტრული რელე-გადამრთველით, რომელიც, ამ შემთხვევაში, წრედის მექანიკური ჩართვა-გამორთვისთვის იყენებს ელექტრომაგნიტს (იხ. სურათი 2). რელე, ჩვეულებრივ, გამოიყენება, როცა ნაკლები სიმძლავრის მოწყობილობამ უნდა ჩართოს ან გამორთოს უფრო ძლიერი მოწყობილობა. ჩვენი რელე იკვებება 5 ვოლტიდან, რომ მართოს მექანიკური გადამრთველი. ამ პროექტში წრედის ძაბვა შეადგენს 9V-ს, თუმცა რელეს შეუძლია მართოს წრედი 240 ვოლტამდე. ჩვენ ამ პროექტში ასეთი მაღალი ვოლტაჟი არ გვჭირდება.



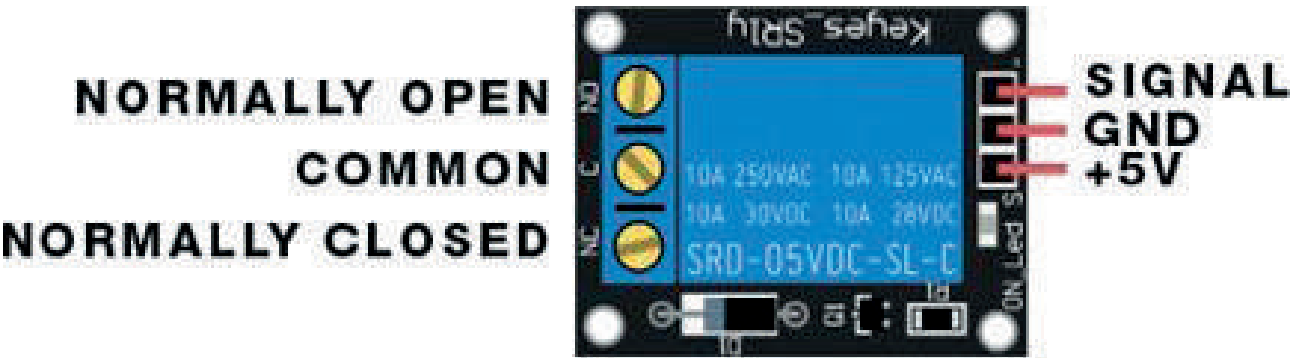
სურათი 2. 5V-იანი ერთარხიანი რელეს მოდული

აპანუოთსქემა

1. დაამაგრეთ LM35 ტემპერატურის სენსორი სამაკეტო დაფაზე (თავის ბრტყელი ზედაპირით თქვენსკენ). მისი მარცხენა გამომყვანი დაუკავშირეთ სამაკეტო დაფის დადებითპოლუსიან ლიანდაგს, ცენტრალური გამომყვანი – არდუინოს A0 გამომყვანს, ხოლო მარჯვენა გამომყვანი – სამაკეტო დაფის GND-ს.

| LM35 სენსორი    | ARDUINO |
|-----------------|---------|
| მარცხენა პინი   | +5V     |
| ცენტრალური პინი | A0      |
| მარჯვენა პინი   | GND     |

2. რელე მუშაობისას შეიძლება დაყენდეს HIGH ან LOW პოზიციაში, რაზეც მიუთითებს პატარა გადამრთველი. ჩვენი პროექტისთვის გადამყვანი დააყენეთ HIGH პოზიციაში, რომ ჩართვისას რელემ კვება მიაწოდოს.



სურათი 3. რელეს კონტაქტები

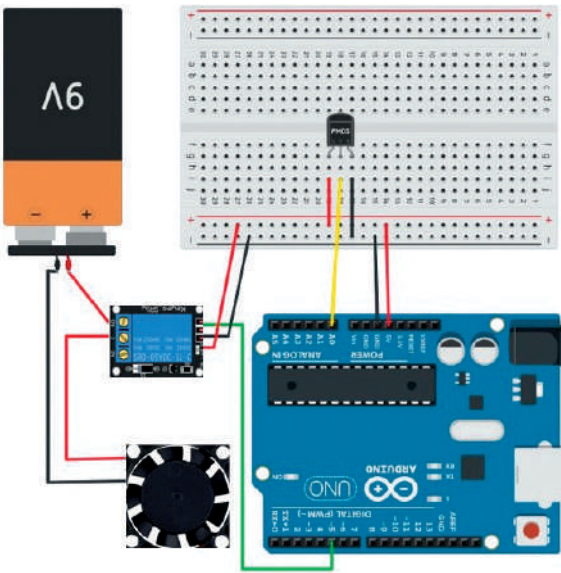
3. დააკავშირეთ რელე არდუინოსთან: რელეს კონტაქტი Signal დააკავშირეთ არდუინოს მე-5 პინ-თან, GND – GND-თან, ხოლო +5V – სამაკეტო დაფის საშუალებით – +5V-თან.

| 5V რელე | ARDUINO |
|---------|---------|
| Signal  | Pin 5   |
| GND     | GND     |
| +5V     | +5V     |

4. რელეს მოდულს ელექტრომაგნიტური გადამრთველის დასაკავშირებლად მარცხენა მხარეს აქვს კავშირების ბუდეები (იხ. სურათი 3). ცენტრალური კონტაქტი არის საერთო კავშირი, მარცხენა კონტაქტი NO – გათიშული წრედი და მისი პოზიცია – გამორთული (გულისხმობის პრინციპით); მარჯვენა კონტაქტი NC – შეკრული წრედი და მისი პოზიცია – ჩართული (გულისხმობის პრინციპით). თუ რელე გადართულია, საერთო გამომყვანი უკავშირდება კონტაქტს NO. რადგან ჩვენ გვსურს, რომ წრედი გათიშული იყოს, ჯერ არ გამოვიყენებთ გადამრთველს, ჩვენ გამოვიყენებთ კონტაქტს NO.
5. ახლა ვენტილატორის GND-ს შავი გამტარი დააკავშირეთ 9V-იანი ელემენტის GND-ს. შემდეგ, როგორც ქვემოთ მოცემულ ცხრილშია ნაჩვენები, ვენტილატორის წითელი, დადებითი გამტარი დააკავშირეთ რელეს Common კონტაქტთან, ხოლო ელემენტის დადებითი გამტარი – რელეს NO კონტაქტთან.

| 5V რელე              | ვენტილატორი/9V ელემენტი       |
|----------------------|-------------------------------|
| NO (normally open)   | 9V ელემენტის დადებითი გამტარი |
| Common               | ვენტილატორის დადებითი გამტარი |
| NC (normally closed) | არ ვიყენებთ                   |

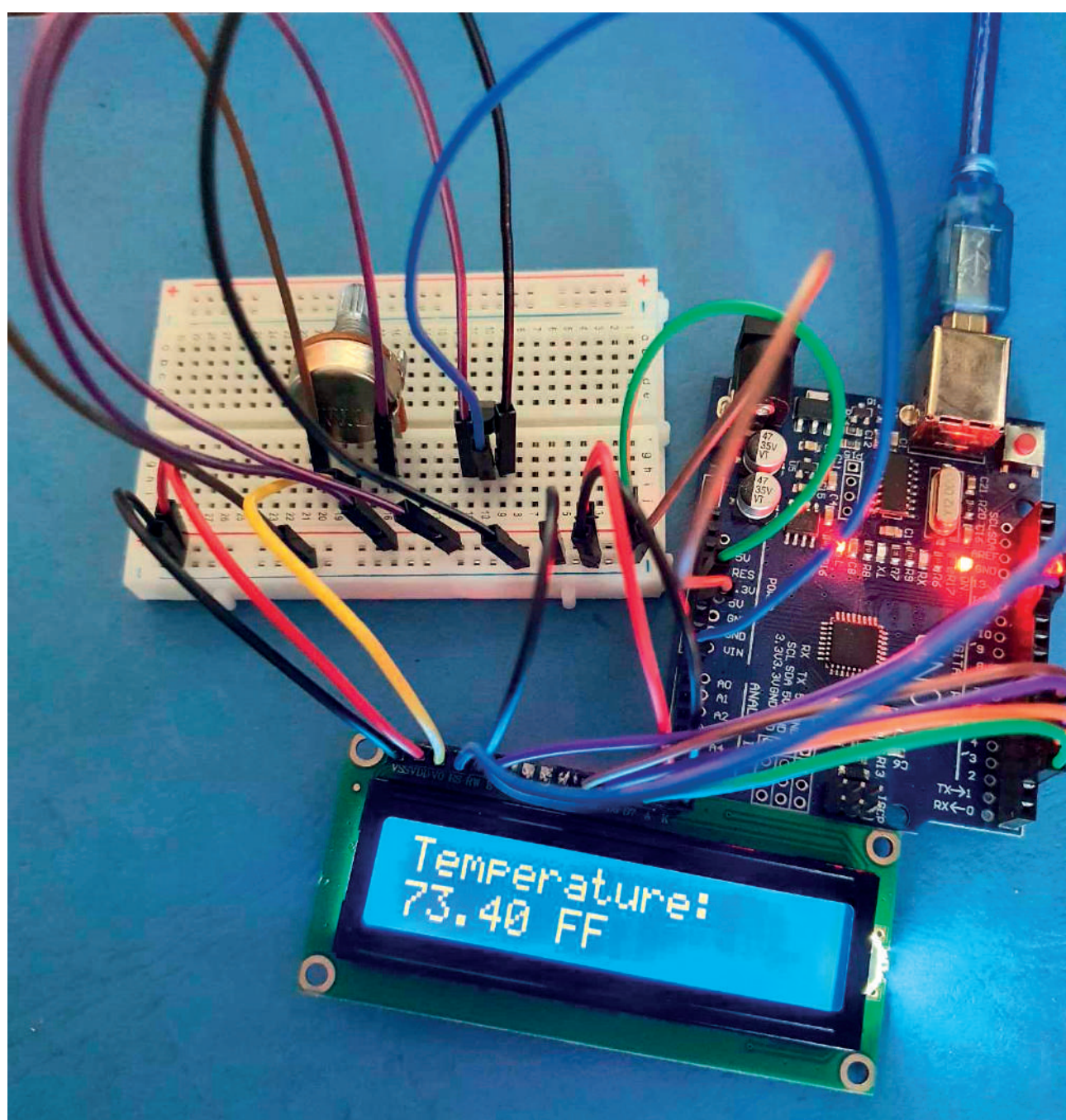
6. დააკავშირეთ სამაკეტო დაფის კვების ლიანდაგები ერთმანეთთან და GND და +5V არდუინოს GND-სა და +5V-თან.
7. დარწმუნდით, რომ თქვენ მიერ აწყობილი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 4) და ჩატვირთეთ კოდი.

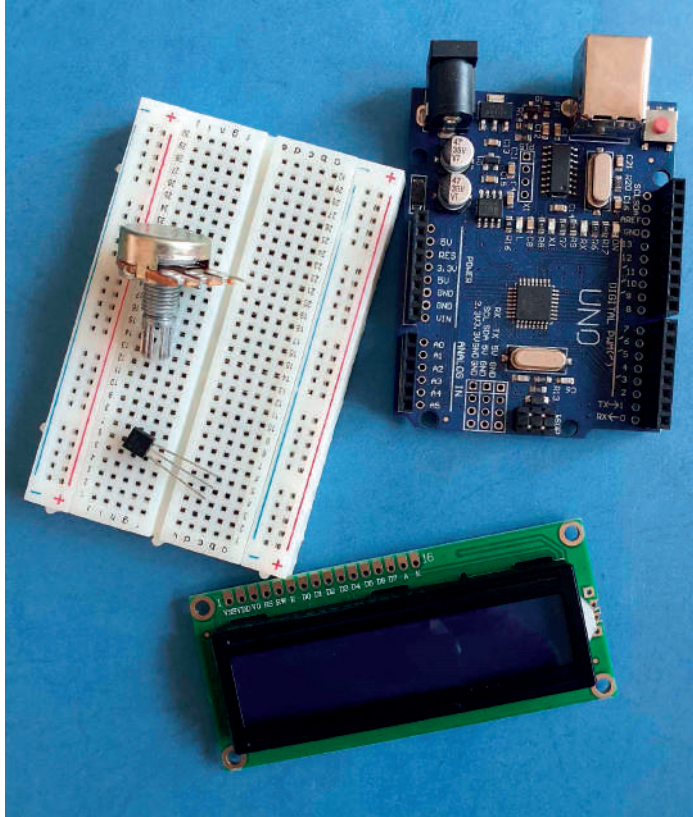


სურათი 4. სქემა

ამ კოდში ჩვენ ჯერ დავაყენებთ LM35 სენსორის კონტაქტს, როგორც A0-ს არდუინოზე, ვენტილატორს განვსაზღვრავთ, როგორც მე-5 კონტაქტს და შევქმნით ცვლადს მნიშვნელობის წასაკითხად LM35-დან. შემდეგ, შევქმნით ცვლადს ტემპერატურის შესანახად და ვენტილატორის გამომყვანს დავაყენებთ როგორც გამომავალს. მცირე გამოთვლა სენსორიდან გამომავალ ძაბვის ჩვენებას გარდაქმნის ტემპერატურის მნიშვნელობად გრადუსებში ფარენგეიტით. ამის შემდეგ არდუინოს IDE ინტერფეისის მიმდევრობითი კომუნიკაციის მონიტორზე (Serial monitor) შეგვიძლია დავინახოთ LM35-ის მაჩვენებლების მნიშვნელობა (ამ დროს არდუინო ჩართული უნდა იყოს). ციკლი სენსორის მაჩვენებლებს ყოველწამიერად ითვლის და თუ ტემპერატურა მიაღწევს 71 გრადუსს ფარენგეიტით, კვება მიეწოდება ვენტილატორის კონტაქტს, რომელიც გაააქტიურებს რელეს და ჩართავს ვენტილატორს. თუ ტემპერატურა 71-ს ქვემოთ დაეცემა, რელე გათიშავს ვენტილატორს.

# XII პროექტი ციფრული თერმომეტრი





### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- HD44780 16x2 LCD ეკრანი
- ტემპერატურის სენსორი LM35
- 50 კომი ცვლადი წინაღობა

### საჭირო ბიბლიოთეკა

- LiquidCrystal

## როგორ გუშაოს პროექტი

ამ პროექტში ჩვენ არდუინოს და LCD ეკრანს დავუმატებთ ტემპერატურის სენსორს LM35-ს და შევქმნით ციფრულ თერმომეტრს. არდუინო კითხულობს ძაბვის ჩვენებას სენსორიდან და ამ ჩვენებას გარდაქმნის ტემპერატურად გრადუსებში ცელსიუსით. შემდეგ ეს მნიშვნელობა კოდს გადაჰყავს გრადუსებში ფარენგეიტით (მნიშვნელობას ამრავლებს 9-ზე, შედეგს ყოფს 5-ზე და უმატებს 32-ს). ტემპერატურის გამოტანა ეკრანზე – ამ რთულ სამუშაოს მთლიანად ასრულებს LiquidCrystal ბიბლიოთეკა lcd.print. ბრძანების დახმარებით. მარტივად შეგიძლიათ ამ პროექტის ადაპტირება და სენსორების დამატებით უნივერსალური მეტეოროლოგიური ცენტრის შექმნა.

## პანწყობის წესები

1. მოამზადეთ LCD ეკრანი. დაამაგრეთ ცვლადი წინაღობა სამაკეტო დაფაზე და ამ უკანასკნელისა და გამტარების დახმარებით დააკავშირეთ ერთმანეთთან LCD ეკრანი, ცვლადი წინაღობა და არდუინო.

| LCD ეკრანი    | ARDUINO                          |
|---------------|----------------------------------|
| 1 VSS         | GND                              |
| 2 VDD         | +5V                              |
| 3 VO contrast | ცვლადი წინაღობის ცენტრალური პინი |
| 4 RS          | Pin 12                           |
| 5 R/W         | GND                              |
| 6 Enable      | Pin 11                           |
| 7 D0          | არ ვიყენებთ                      |
| 8 D1          | არ ვიყენებთ                      |

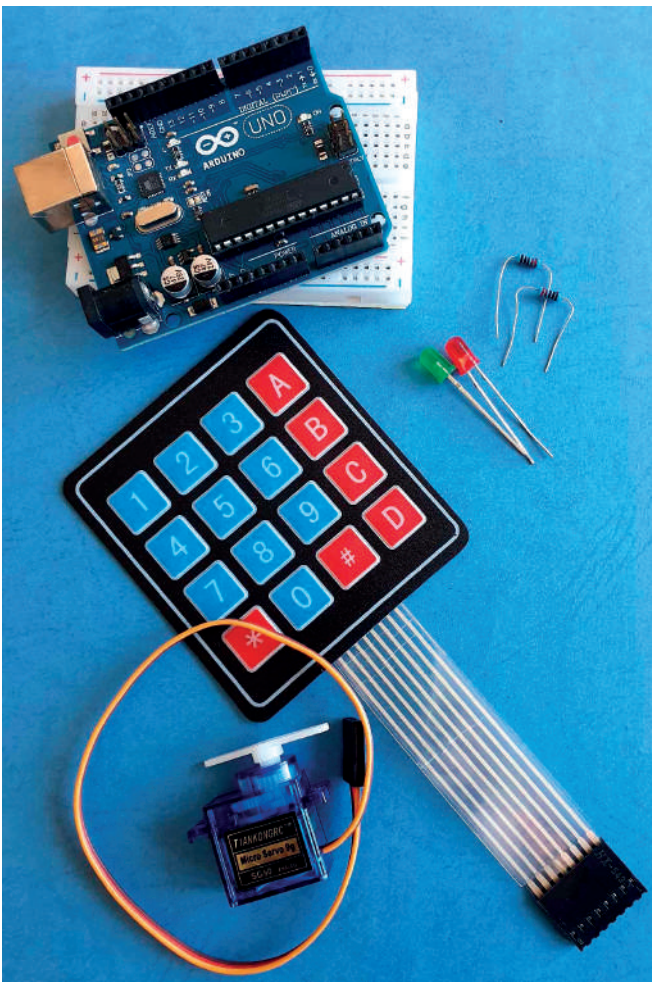
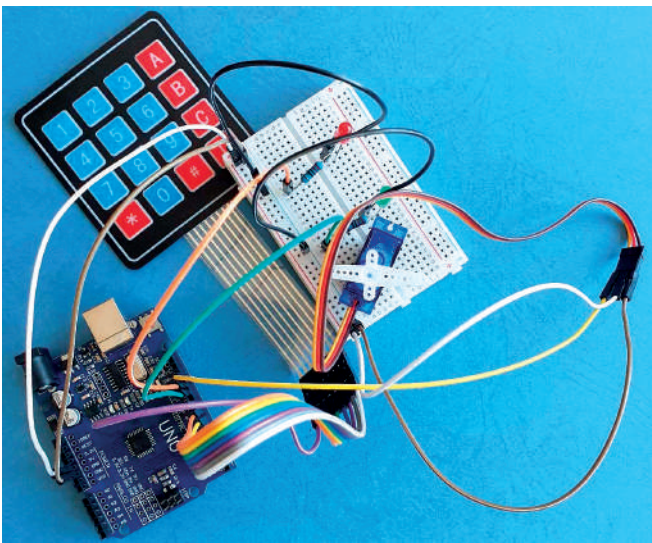


## როგორ მუშაობს კოდი

LCD ეკრანზე იმ მნიშვნელობის გამოსახულების გამოტანისთვის, რომელსაც აღმოაჩენს ტემპერატურის სენსორი LM35, კოდი იყენებს LiquidCrystal ბიბლიოთეკას. სენსორი LM35 ჩვენებებს აგზავნის არდუინოს A0 გამომყვანზე, როგორც ძაბვას. კოდი ძაბვის მაჩვენებლებს გარდაქმნის ტემპერატურად გრადუსებში ცელსიუსით, ხოლო შემდეგ (რამდენიმე დამატებითი გამოთვლით), გამოჰყავს საბოლოო შედეგი გრადუსებში ფარენგეიტით. კოდი ჩვენებებს ყოველწამიერად აახლებს და გამოაქვს ეკრანზე.

# XIII პროექტი

## მემბრანული კლავიატურა



### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 2 ცალი LED ნათურა
- 2 ცალი 220 ომი წინაღობა
- Servo ძრავა
- მემბრანული კლავიატურა

### საჭირო ბიბლიოთეკები

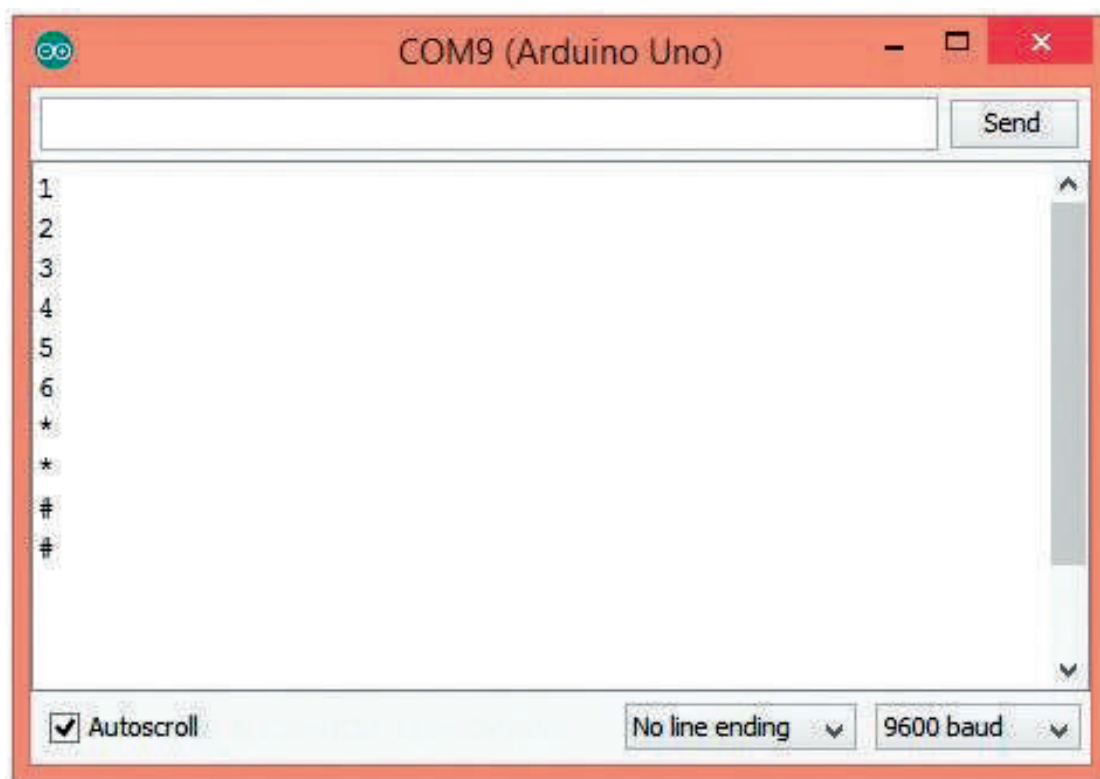
- Keypad
- Servo
- Password

ამ პროექტში ჩვენ ვიყენებთ 4X4 მემბრანულ კლავიატურას რვა გამტარით. გამტარები დანომრილია მარცხნიდან მარჯვნივ 1-დან 8-მდე. პირველი 4 გამტარი შეესაბამება ჰორიზონტალურ, ხოლო დანარჩენი 4 – ვერტიკალურ სვეტებს.

პროექტში კლავიატურას ვაკავშირებთ Servo ძრავასთან და რამდენიმე შექდიოდთან, რომ შევქმნათ ჩამკეტი სისტემა. ჩამკეტი რომ გამოიყენოთ, ჩაწერეთ კოდი და დადასტურებისთვის კლავიატურაზე დააჭირეთ ფიფქის (\*) კლავიშს. თუ კოდი სწორია, აინთება მწვანე შექდიოდი და Servo ძრავას მხარი მოტრიალდება 90°-ით. თუ სურვილი გაქვთ, კოდში მოცემული რიცხვები (პაროლი) შეგიძლიათ შეცვალოთ. ამისათვის ვიყენებთ კლავიშს # და ახლიდან ჩავტვირთავთ შეცვლილ კოდს.

ჯერ უნდა შევამოწმოთ, როგორ მუშაობს ჩვენი კლავიატურა. ამ პროექტს ორი კოდი აქვს: a – კლავიატურის შესამოწმებელი და b – კოდი. ამიტომ, ჯერ დაგვჭირდება a კოდის ჩატვირთვა.

კოდის ჩატვირთვის შემდეგ გახსენით არდუინოს IDE ინტერფეისში მიმდევრობითი კომუნიკაციის მონიტორი – Serial Monitor (იხილეთ სურათი 1).



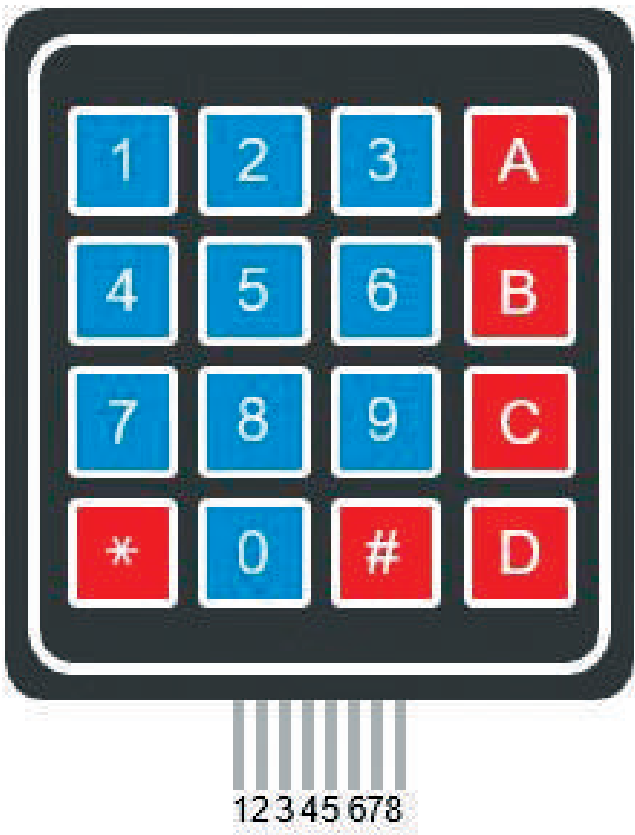
დააკავშირეთ კლავიატურის გამტარები მარცხნიდან მარჯვნივ, თანმიმდევრობით, არდუინოს 9-2 კონტაქტებთან (სურათი 2 და ცხრილი). კოდის ჩატვირთვის შემდეგ დააჭირეთ კლავიატურის რამდენიმე კლავიშს თითო და დარწმუნდით, რომ ეკრანზე შესაბამისი სიმბოლოები აისახა. რიცხვები იქნება ეს თუ ასოები, ეკრანზე ისინი ვერტიკალურ სვეტად უნდა გამოისახონ, ისე, როგორც ეს სურათზეა ნაჩვენები.

სურათი 1. კლავიატურის შემოწმება

აპაწყოს სქემა

1. კლავიატურის გამტარები დააკავშირეთ არდუინოს შემდეგ პინებთან:

| კლავიატურა | ARDUINO |
|------------|---------|
| Pin 1      | Pin 9   |
| Pin 2      | Pin 8   |
| Pin 3      | Pin 7   |
| Pin 4      | Pin 6   |
| Pin 5      | Pin 5   |
| Pin 6      | Pin 4   |
| Pin 7      | Pin 3   |
| Pin 8      | Pin 2   |



სურათი 2. კლავიატურის პინები და მათი დაკავშირება არდუინოსთან

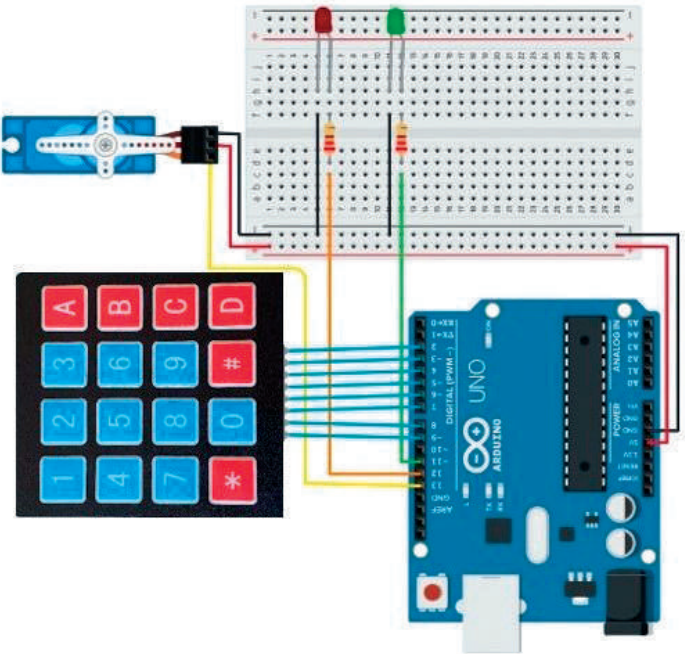
2. მწვანე და წითელი შუქდიოდები მოათავსეთ სამაკეტო დაფაზე. მათი მოკლე, უარყოფითპოლუსიანი ფეხები დააკავშირეთ არდუინოს GND-თან. გრძელი, დადებითპოლუსიანი ფეხები, 220 ომი წინაღობების გავლით, დააკავშირეთ: მწვანე შუქდიოდი – არდუინოს მე-11 გამომყვანთან, წითელი შუქდიოდი – არდუინოს მე-12 გამომყვანთან.

| LED ნათურები     | ARDUINO                            |
|------------------|------------------------------------|
| დადებითი ფეხები  | Pin 11 და Pin 12                   |
| უარყოფითი ფეხები | 220 ომი წინააღობების გავლით<br>GND |

3. დააკავშირეთ Servo ძრავა არდუინოსთან: ძრავას ყავისფერი გამტარი – არდუინოს GND-თან, წითელი გამტარი – არდუინოს +5 V-თან, ყვითელი/თეთრი გამტარი – არდუინოს მე-13 პინთან.

| Servo             | ARDUINO |
|-------------------|---------|
| ყავისფერი გამტარი | GND     |
| წითელი გამტარი    | +5V     |
| ყვითელი გამტარი   | Pin 13  |

4. დარწმუნდით, რომ თქვენ მიერ აწყობილი სქემა სრულად შეესაბამება ჩვენს სქემას (იხ. სურათი 3) და მხოლოდ შემდეგ ჩატვირთეთ b – კოდი.

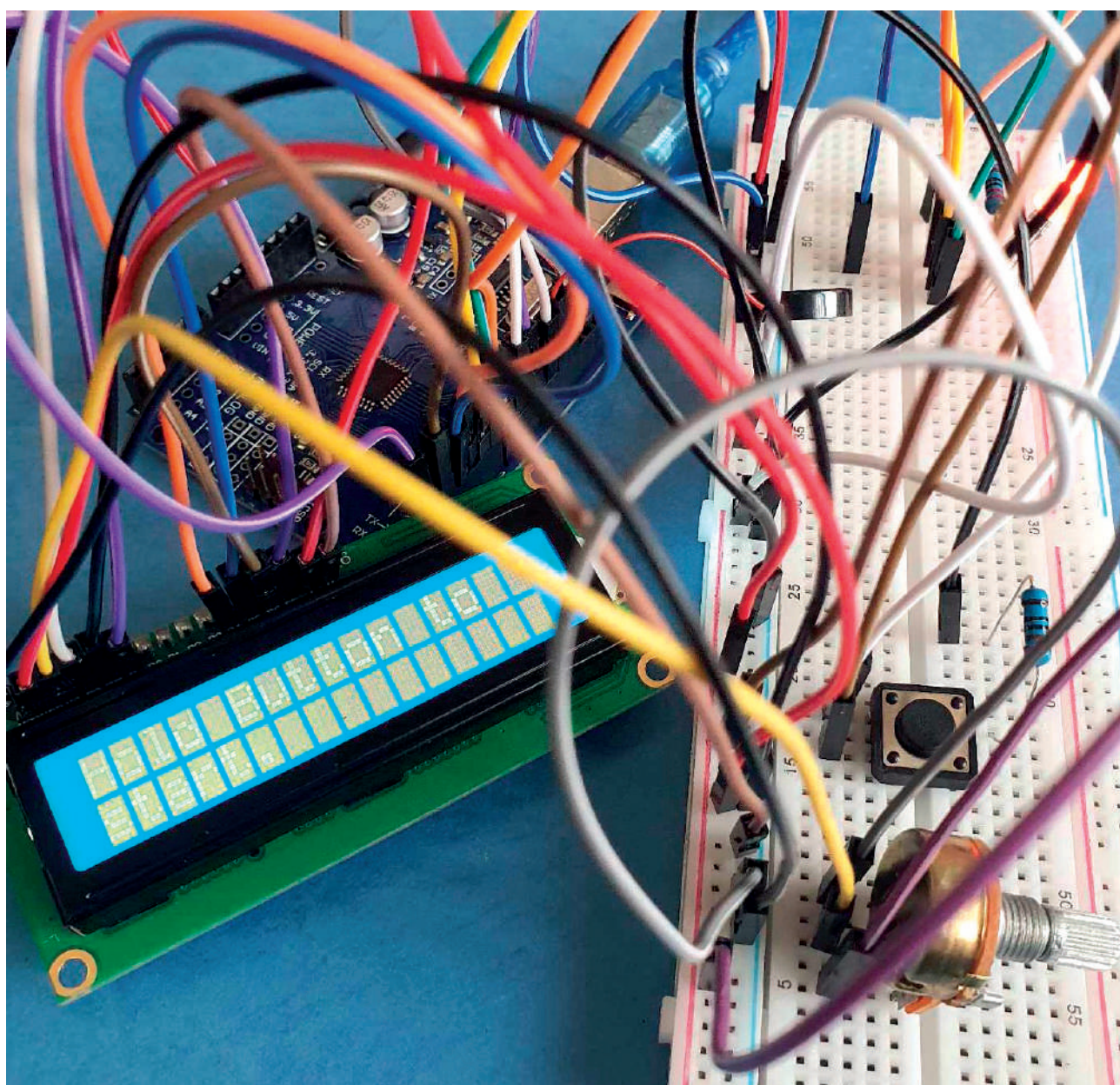


სურათი 3. სქემა

**როგორ გუშაობს კოდი**

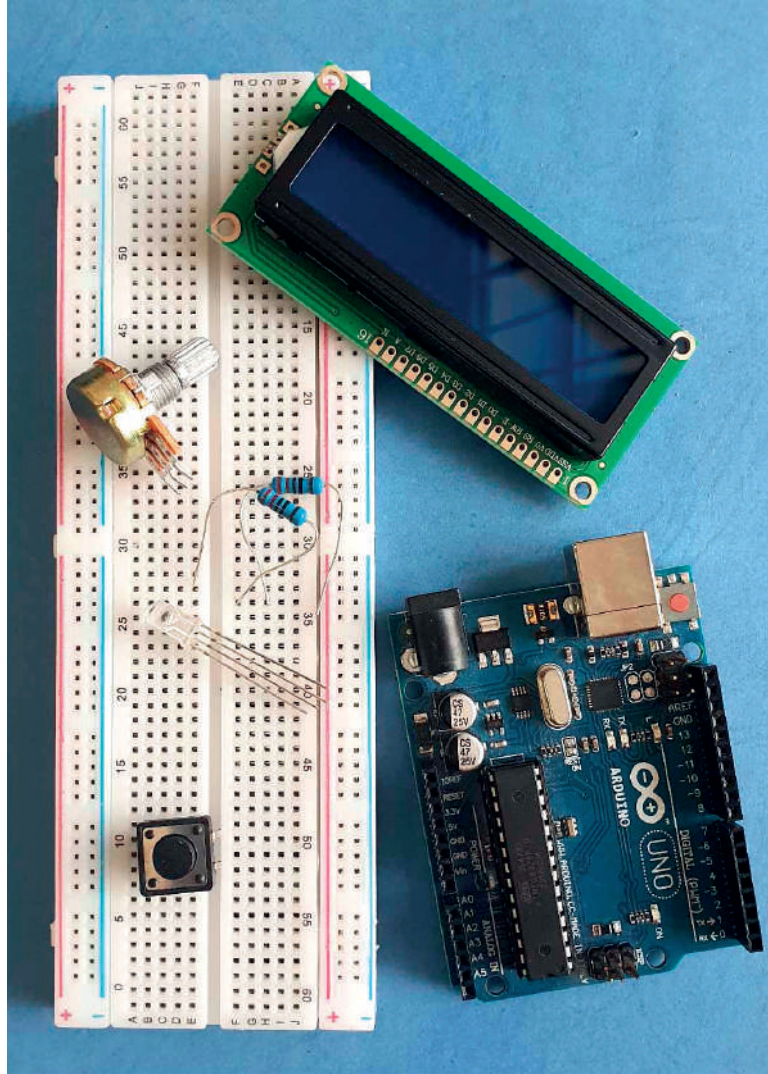
არდუინო ელოდება კლავიატურიდან კოდის შეყვანას სიმბოლოთი \*. სიმბოლოზე \* დაჭერისას კოდი ამოწმებს პაროლთან შესაბამისობას. თუ პაროლი არასწორია, აინთება წითელი შუქდიოდი, ხოლო თუ პაროლი სწორია – აინთება მწვანე შუქდიოდი და Servo ძრავას მხარი მოტრიალდება 90°-ით. იმ შემთხვევაში, თუ პაროლის შეცვლა გსურთ, დააჭირეთ სიმბოლოს # და შეცვალეთ ახალი პაროლით. პაროლის შესაცვლელად შეცვალეთ რიცხვები (მხოლოდ რიცხვები) შემდეგ სტრიქონში Password password = Password(“2468”);

# ჩეაქსიის ღრმის მთვლელები



საჭირო ბიბლიოთეკა

■ LiquidCrystal



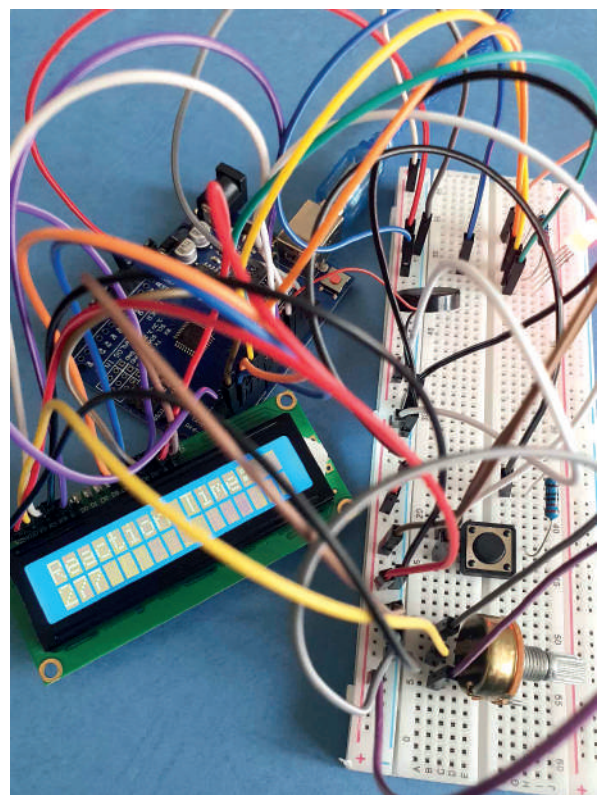
#### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- 16X2 LCD ეკრანი  
(თავსებადი Hitachi HD44680)
- RGB LED ნათურა  
(საერთოანოდური)
- პასიური Piezo დინამიკი
- 50 კომი ცვლადი რეზისტორი  
(პოტენციომეტრი)
- არაფიქსირებადი ლილაკი
- 220 ომი წინაღობა

#### როგორ გუშაოვს პროექტი

ამ პროექტში ჩვენ გავაერთიანებთ LCD ეკრანს, RGB შუქდიოდს და პიეზო დინამიკს, რომ გამოვ-  
თვალოთ ჩვენი რეაქციის დრო.

თამაში იწყება არაფიქსირებად ლილაკზე თითის დაჭერით. ამ დროს RGB შუქდიოდი იწყებს შემ-  
თხვევითი ფერების ანთებას (წითელი, მწვანე, ლურჯი). თქვენი მიზანია, რაც შეიძლება სწრაფად  
აუშვათ ლილაკს თითი წითელი ფერის ანთებისას. ამ დროს LCD ეკრანზე გამოისახება თქვენი რეაქ-  
ციის დრო მილიწამებში – წითელი ფერის ანთებიდან თქვენ მიერ ლილაკიდან თითის აღებამდე (იხ.  
სურათი 1).

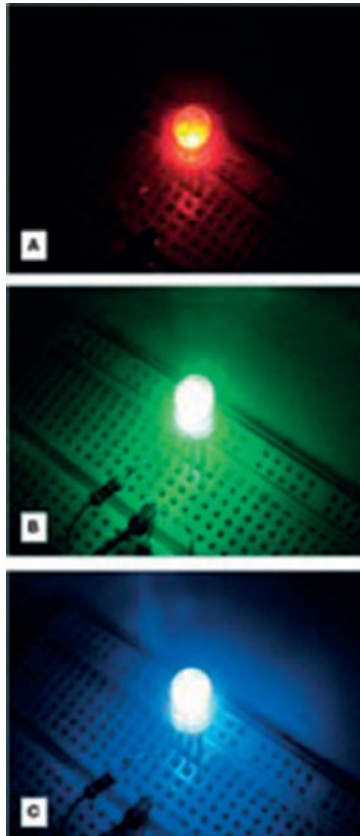


სურათი 1. თქვენი რეაქციის დრო LCD ეკრანზე

Piezo დინამიკი ხმაურით ცდილობს თქვენი ყურადღების გაფანტვას. თუ თქვენ დროზე ადრე აუშვებთ თითს ლილაკიდან, LCD ეკრანი ამასაც გატყობინებთ და მოგიწევთ თავიდან სცადოთ ბედი.

### როგორ მუშაობს RGB ნათურა

LED შუქდიოდი არსებობს სხვადასხვა ფორმისა და ფერის, მაგრამ ერთი-ერთი ყველაზე საინტერესო – RGB შუქდიოდი. როგორც სახელწოდებიდან ჩანს, RGB შუქდიოდი არის სამი შუქდიოდი ერთში: წითელი (Red), მწვანე (Green) და ლურჯი (Blue). იხილეთ სურათი 2.



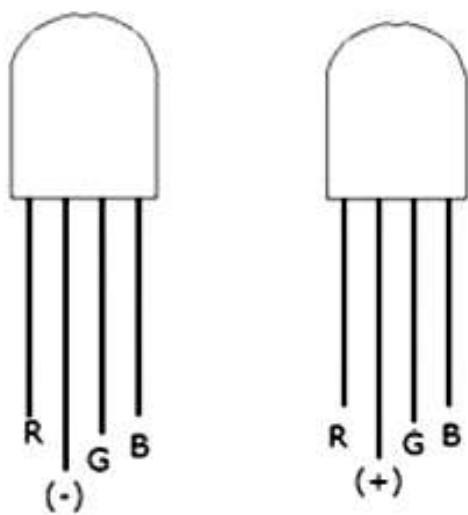
სურათი 2. RGB შუქდიოდის ძირითადი ფერები.

RGB შუქდიოდი არის ტექნოლოგია, რომელიც გამოიყენება სხვადასხვა ფერის კომბინაციის შესაქმნელად ერთი ნათურის საშუალებით, რაც ნიშნავს ორი ან მეტი ფერის კომბინირებით სხვა ფერის მიღებას. წითელი, მწვანე და ლურჯი – ძირითადი ფერებია და გამოიყენება სხვა ფერების მისაღებად (იხილეთ სურათი 3).



სურათი 3. RGB – ფერების კომბინაციის მოდელი

განვიხილოთ RGB შუქდიოდი უფრო დაწვრილებით.



სურათი 4. RGB შუქდიოდი საერთო კათოდით და საერთო ანოდით

როგორც ხედავთ, აქამდე ჩვენ მიერ გამოყენებული შუქდიოდებისგან განსხვავებით, RGB შუქდიოდს ორის ნაცვლად ოთხი ფეხი აქვს (იხ. სურათი 4): თითო-თითო ფეხი წითელი, მწვანე და ლურჯი ფერებისთვის. მეოთხე ფეხი ან კათოდია (უარყოფითი), ან ანოდი (დადებითი). *თუმცა, ამ პროექტში ჩვენ გამოვიყენებთ RGB შუქდიოდს საერთო ანოდით, ანუ მისი ოთხი ფეხიდან ერთი, ყველაზე გრძელი, ამ შემთხვევაში არის დადებითპოლუსიანი და უკავშირდება +5V-ს.*

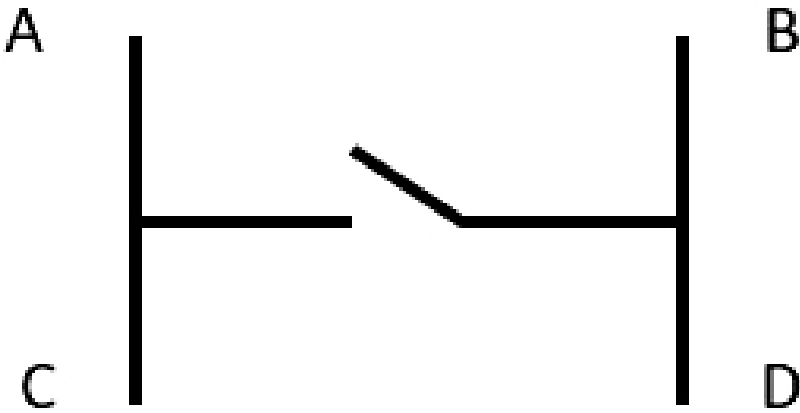
პანელის დაყვანა

- 1. მოამზადეთ LCD ეკრანის საკონტაქტო პინები კავშირებისთვის.
- 2. მოათავსეთ ცვლადი წინაღობა სამაკეტო დაფაზე. გამოიყენეთ სამაკეტო დაფა და გამტარები LCD ეკრანის, არდუინოსა და ცვლადი წინაღობის ერთმანეთთან დასაკავშირებლად.

| LCD ეკრანი    | ARDUINO                          |
|---------------|----------------------------------|
| 1 VSS         | GND                              |
| 2 VDD         | +5V                              |
| 3 VO contrast | ცვლადი წინაღობის ცენტრალური პინი |
| 4 RS          | Pin 11                           |
| 5 R/W         | GND                              |
| 6 Enable      | Pin 12                           |
| 7 D0          | არ ვიყენებთ                      |
| 8 D1          | არ ვიყენებთ                      |
| 9 D2          | არ ვიყენებთ                      |
| 10 D3         | არ ვიყენებთ                      |

|            |       |
|------------|-------|
| 11 D4      | Pin 5 |
| 12 D5      | Pin 4 |
| 13 D6      | Pin 3 |
| 14 D7      | Pin 2 |
| 15 A BcL + | +5V   |
| 16 K BcL – | GND   |

3. ზემოთ მოცემული ცხრილის მიხედვით თქვენ უკვე უნდა დაგეგავშირებინათ ცვლადი წინაღობის ცენტრალური პინი LCD ეკრანის V0 გამომყვანისთვის. ახლა მისი ორი განაპირა პინები დააკავშირეთ GND-სა და +5V-თან. ის არეგულირებს LCD ეკრანის კონტრასტულობას.
4. დაამაგრეთ არაფიქსირებადი ლილაკი სამაკეტო დაფაზე ისე, რომ მისი ორ-ორი პინი მოთავსებული იყოს სამაკეტო დაფის ცენტრალური გამყოფი ზოლის ორივე მხარეს.



სურათი 5. არაფიქსირებადი ლილაკის საკონტაქტო პინები

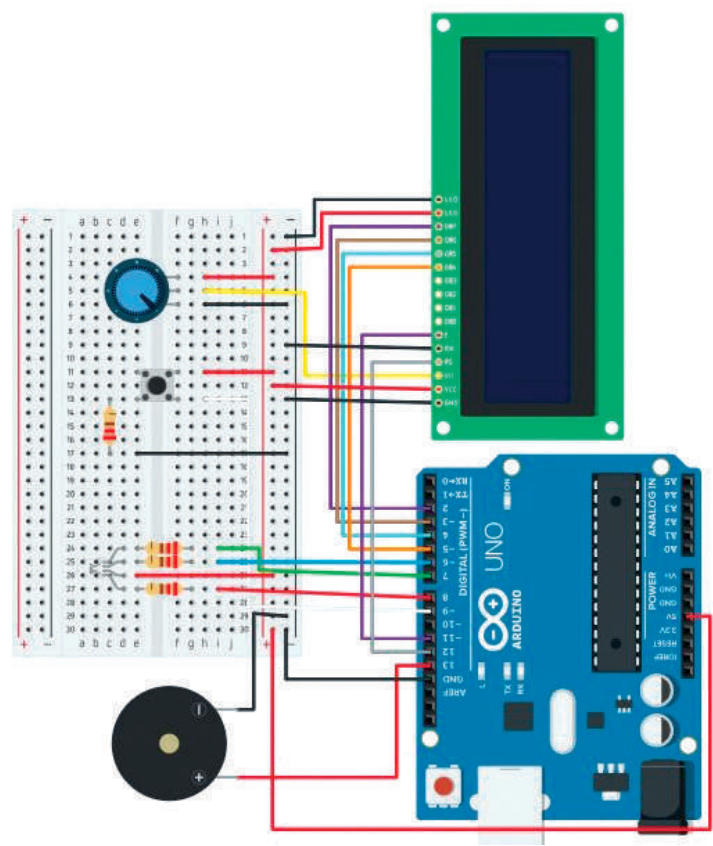
დააკავშირეთ ლილაკის A პინი 220 ომი წინაღობის გავლით GND-თან, C პინი – არდუინოს მე-9 გამომყვანთან, ხოლო D პინი – +5V-თან. არაფიქსირებადი ლილაკის მუშაობის პრინციპს ჩვენ გავეცანით წიგნში „არდუინოს შემსწავლელი სახელმძღვანელო – 20 STEAM პროექტი“, I ნაწილი (II პროექტი, შუქდიოდის მართვა ლილაკის საშუალებით).

| ლილაკი | გენტილატორი/9V ელემენტი          |
|--------|----------------------------------|
| Pin A  | GND<br>220 ომი რეზისტორის გავლით |
| Pin C  | Pin 9                            |
| Pin D  | +5V                              |

5. მოათავსეთ RGB შუქდიოდი სამაკეტო დაფაზე: წითელი ფეხი დააკავშირეთ არდუინოს მე-8 პინთან, მწვანე – მე-6 პინთან, ლურჯი – მე-7 პინთან და გრძელი ფეხი, ანოდი – +5V-თან.

| RGB LED          | ვენტილატორი/9V ელემენტი |
|------------------|-------------------------|
| წითელი ფეხი      | Pin 8                   |
| მწვანე ფეხი      | Pin 6                   |
| ლურჯი ფეხი       | Pin 7                   |
| საერთო ანოდი (+) | +5V                     |

- Piezo დინამიკის წითელი გამტარი დააკავშირეთ უშუალოდ არდუინოს მე-13 პინთან, ხოლო შავი – GND-თან.
- შეამოწმეთ, გაქვთ თუ არა სქემა (იხ. სურათი 6) სწორად აგებული და ჩატვირთეთ კოდი.



სურათი 6. სქემა

როგორ გუშაოვს კოდი

როცა ღილაკზე თითი გაქვთ დაჭერილი, RGB შუქდიოდი ანათებს სხვადასხვა ფერად, მათ შორის წითლად. ფერების თანმიმდევრობა და განათებებს შორის დროის ინტერვალები შემთხვევითობის პრინციპით მუშაობს კოდში, რაც ნიშნავს, რომ თქვენ ვერც ფერების თანმიმდევრობას დაიმახსოვრებთ და ვერც იმას, როდის აინთება წითელი. თქვენ შეგიძლიათ გაართულოთ თამაში, თუ გაზრდით ინტერვალების ხანგრძლივობას კოდის ამ მონაკვეთში:

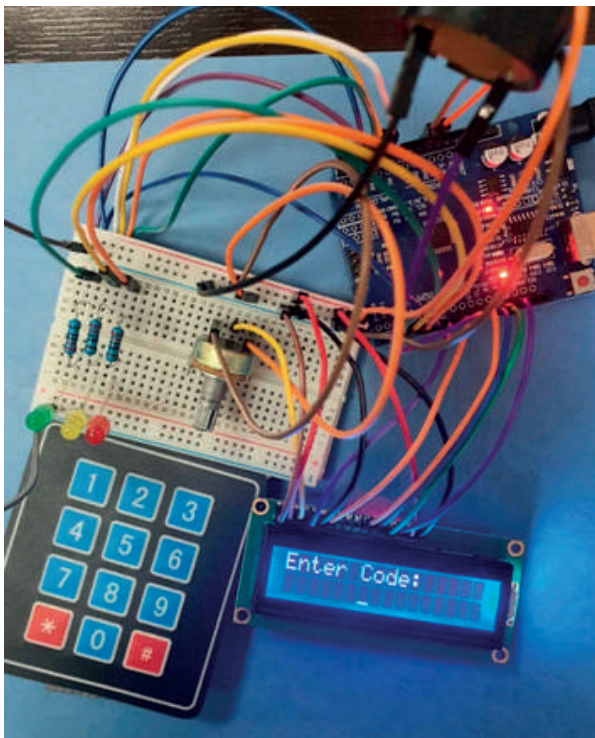
```

-----
PSE = random(500, 1200);
-----

```

# XV პროექტი

## დეკოდირება კლავიატურით

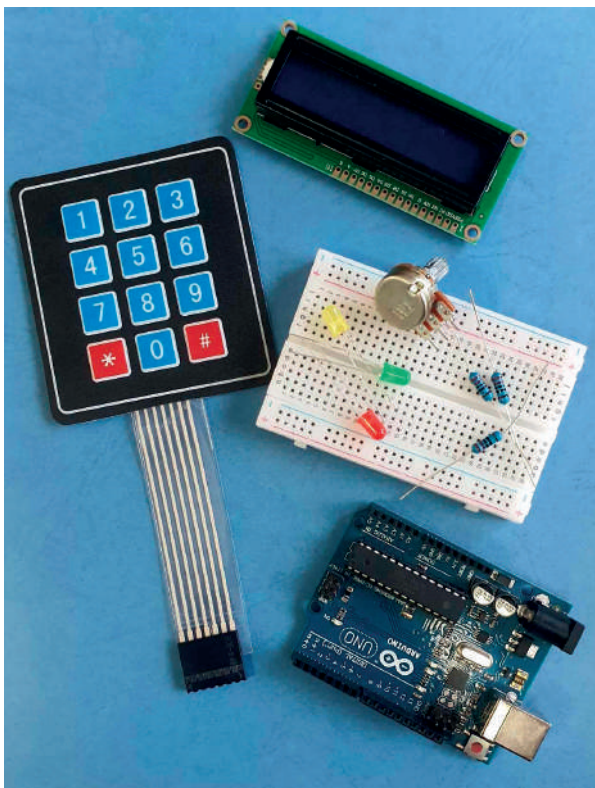


### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- HD44780 16x2 LCD ეკრანი
- 10-დან 50 კომამდე ცვლადი წინაღობა
- პასიური Piezo დინამიკი
- 3x4 მემბრანული კლავიატურა
- 3 ცალი 220 ომი წინაღობა
- LED ნათურები: წითელი, ყვითელი, მწვანე

### საჭირო ბიბლიოთეკები

- LiquidCrystal
- Keypad
- Tone



როგორ გუშაოს პროექტი

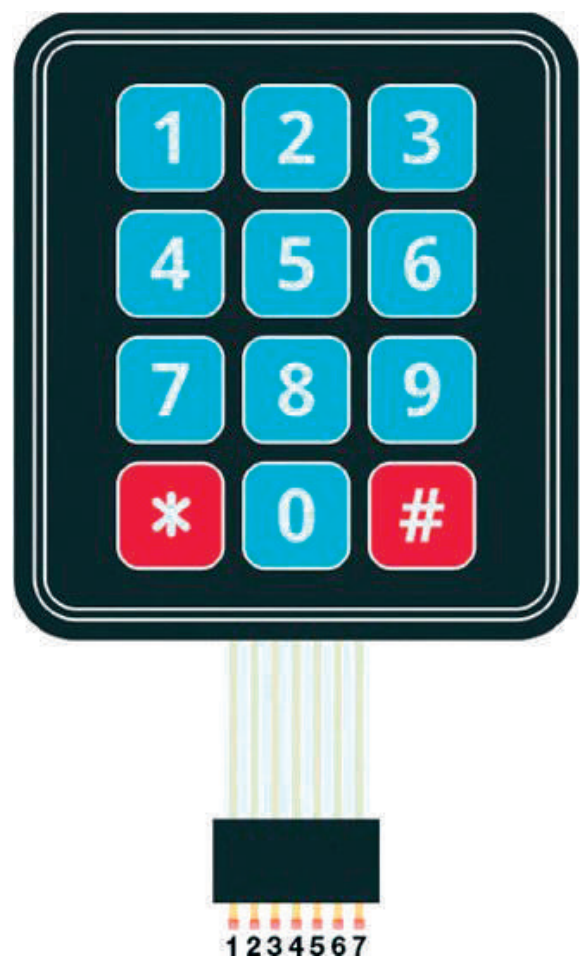
არდუინოს რომ ჩართავთ, ერთ-ერთი მოთამაშე შეიყვანს ოთხნიშნა კოდს, რომ ბომბის დროის მთვლელი ჩაირთოს. მეორე მოთამაშე დააწვება\* ლილაკს, რომ დაიწყოს ბომბის გაუვნებელყოფა – ამ მოთამაშემ უნდა გამოიცნოს კოდი, რომ დროულად გააუვნებელყოს ბომბი. თუ მოთამაშე შეცდება და არასწორ ლილაკს დააჭერს თითს, თავიდან დაწყება შეუძლია # ლილაკზე თითის დაჭერით. ხოლო თუ არასწორ კოდს შეიყვანს ან დროში ვერ ჩაეტევა, ბომბი აფეთქდება. თამაშის დროს ციმციმებს ყვითელი შუქდიოდი, ხოლო Piezo დინამიკი ხმოვან სიგნალს გამოსცემს დროის უკუთვლის ტაქტში. LCD ეკრანი მიმდევრობით გვიჩვენებს ყველა მოქმედებას. თუ მოთამაშე („გამნაღმველი“) არასწორ კოდს შეიყვანს, ამ შემთხვევაში ყველა შუქდიოდი აციმციმდება და დინამიკი აფეთქების ხმას გამოსცემს. ვინაიდან კლავიატურა ოთხნიშნა კოდის უამრავ ვარიანტს შეიცავს, კარგი იქნებოდა „გამნაღმველისთვის“ დაგვესვა ოთხი შეკითხვა, რომელთაგან თითოეული მისცემდა კოდის ერთ ციფრს. მოთამაშეს უნდა მიეცეს განსაზღვრული დრო კითხვებზე პასუხის მოსაფიქრებლად და კოდის შესაყვანად.

აპანოთსება

- 1. მოათავსეთ ცვლადი წინაღობა სამაკეტო დაფაზე.
- 2. დააკავშირეთ ერთმანეთთან LCD ეკრანი, ცვლადი წინაღობა და არდუინო.

| LCD ეკრანი    | ARDUINO                          |
|---------------|----------------------------------|
| 1 VSS         | GND                              |
| 2 VDD         | +5V                              |
| 3 VO contrast | ცვლადი წინაღობის ცენტრალური პინი |
| 4 RS          | Pin 7                            |
| 5 R/W         | GND                              |
| 6 Enable      | Pin 8                            |
| 7 D0          | არ ვიყენებთ                      |
| 8 D1          | არ ვიყენებთ                      |
| 9 D2          | არ ვიყენებთ                      |
| 10 D3         | არ ვიყენებთ                      |
| 11 D4         | Pin 10                           |
| 12 D5         | Pin 11                           |
| 13 D6         | Pin 12                           |
| 14 D7         | Pin 13                           |
| 15 A BcL+     | +5V                              |
| 16 K BcL–     | GND                              |

3. ცვლადი წინაღობის ცენტრალური პინი უკვე დაკავშირებული გაქვთ LCD ეკრანთან (V0). ახლა კი მისი ორი განაპირა პინიდან ერთი დააკავშირეთ GND-თან, ხოლო მეორე – +5V-თან (პროექტში რამდენიმე კავშირი იქნება GND-თან, ამიტომ ამისთვის გამოიყენეთ სამაკეტო დაფა).
4. კლავიატურის კონტაქტები დანომრილია 1-დან – 7-მდე, მარცხნიდან მარჯვნივ (იხ. სურათი 1). დააკავშირეთ მისი კონტაქტები არდუინოსთან.



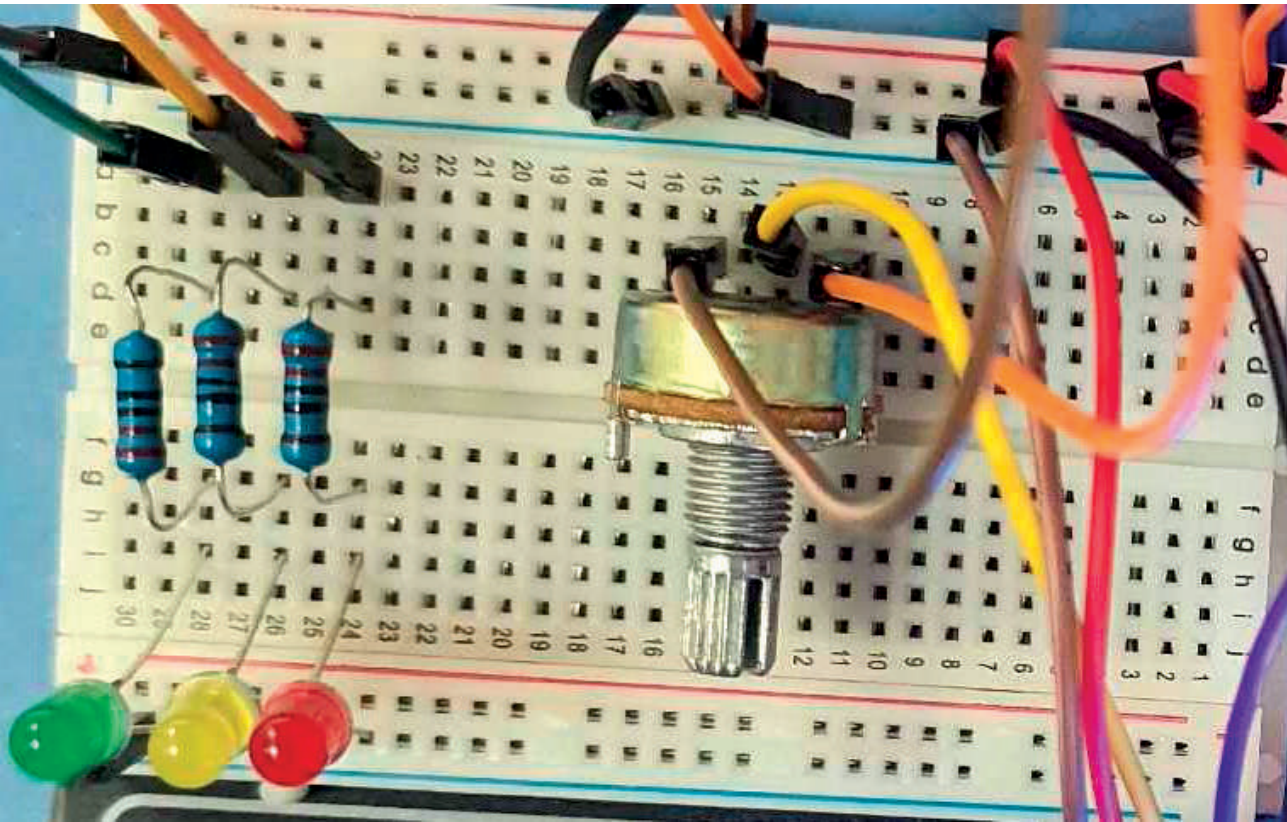
სურათი 1. 3x4 კლავიატურის პინების ნუმერაცია

| კლავიატურა | ARDUINO |
|------------|---------|
| Pin 1      | Pin 5   |
| Pin 2      | Pin A5  |
| Pin 3      | Pin A4  |
| Pin 4      | Pin A2  |
| Pin 5      | Pin A1  |
| Pin 6      | Pin A0  |
| Pin 7      | Pin A3  |

5. Piezo დინამიკის წითელი გამტარი დააკავშირეთ არდუინოს მე-9 საკონტაქტო პინთან, ხოლო შავი გამტარი – არდუინოს GND-თან.

| PIEZO დინამიკი | ARDUINO |
|----------------|---------|
| წითელი გამტარი | Pin 9   |
| შავი გამტარი   | GND     |

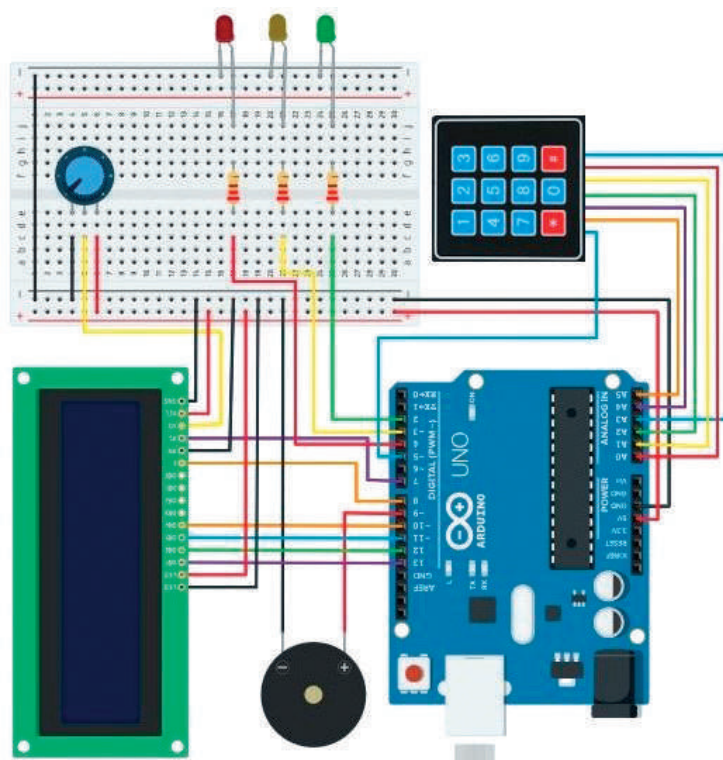
6. მოათავსეთ შუქდიოდები სამაკეტო დაფაზე (იხ. სურათი 2). მათი მოკლე ფეხები დააკავშირეთ დაფის უარყოფითპოლუსიან ლიანდაგზე, ხოლო გრძელი, დადებითპოლუსიანი ფეხები 220 ომი წინააღობების საშუალებით დააკავშირეთ არდუინოსთან: მწვანე – პინი 2, ყვითელი – პინი 3 და წითელი – პინი 4.



სურათი 2. შუქდიოდების განლაგება სამაკეტო დაფაზე

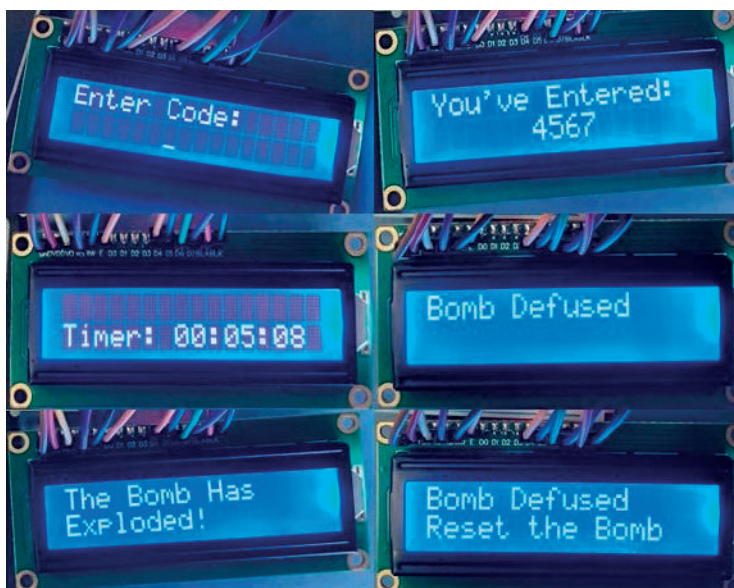
| LED ნათურები          | ARDUINO                           |
|-----------------------|-----------------------------------|
| უარყოფითი ფეხები      | GND                               |
| მწვანე დადებითი ფეხი  | Pin 2 – 220 ომი წინააღობის გავლით |
| ყვითელი დადებითი ფეხი | Pin 3 – 220 ომი წინააღობის გავლით |
| წითელი დადებითი ფეხი  | Pin 4 – 220 ომი წინააღობის გავლით |

- სამაკეტო დაფის დადებით- და უარყოფითპოლუსიანი ლიანდაგები დააკავშირეთ არდუინოს შე-საბამის პინებთან: GND – GND, +5V – +5V.
- დარწმუნდით, რომ თქვენი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 3) და ჩატვირთეთ კოდი.



სურათი 3. სქემა

ახლა გიჩვენებთ LCD ეკრანის გამოსახულებებს თამაშის სხვადასხვა ეტაპზე (იხ. სურათი 4).

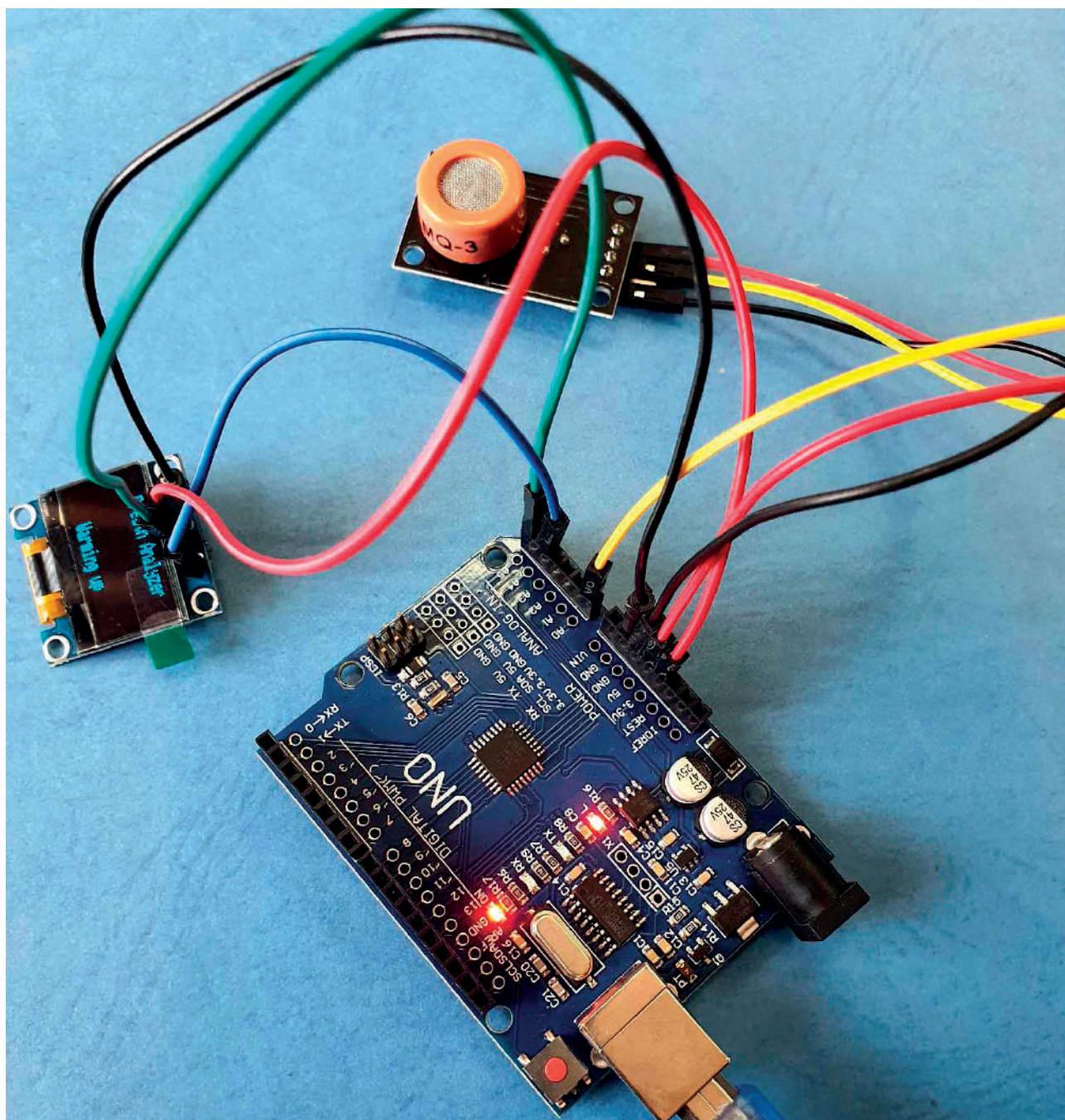


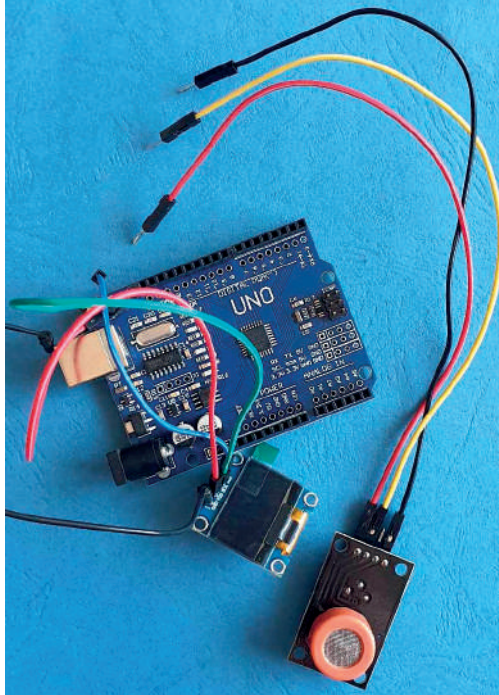
სურათი 4. LCD ეკრანზე ასახული თამაშის ეტაპები

1. შეიყვანეთ კოდი.
2. ბომბი ადასტურებს შეყვანილ კოდს.
3. დროის მთვლეელი იწყებს უკუთვლას.
4. ყვითელი შუქდიოდი ანათებს უკუთვლის ტაქტში.
5. მეორე მოთამაშე აჭერს კლავიატურის\* ლილაკს და ბომბის გასაუვნებელყოფად შეჰყავს კოდი.
6. ეკრანზე არ აისახება ბომბის გასაუვნებელყოფად შეყვანილი კოდი.
7. სწორი კოდის შეყვანის შემთხვევაში ბომბი გაუვნებელყოფილია.
8. თუ არა და ... ბუმ!

XVI პროექტი

# ელკომუნიკაციები





#### საჭირო ბიბლიოთეკები

- SPI
- Wire
- Adafruit GFX
- Adafruit\_SSD1306

#### საჭირო რესურსი

- Arduino UNO
- დედალ-მამალი გამტარები
- Keyes MQ3 ალკოტესტერი
- OLED ეკრანი (128x64)

### როგორ გუზაოვს KEYES MQ3 ალკოჰესვაი

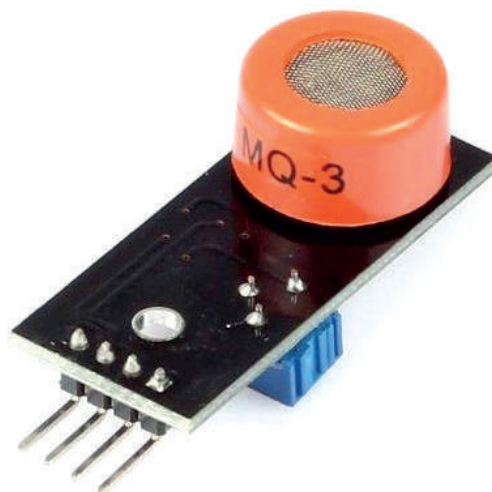
გაზის სენსორების ოჯახი, რომელიც ასევე მოიცავს MQ2-ს, რომელიც მგრძნობიარეა მეთანის, ბუტანისა და კვამლის მიმართ; MQ4 – მგრძნობიარეა შეკუმშული ბუნებრივი აირის მიმართ;

MQ6 – მგრძნობიარეა ბუტანისა და თხევადი აირის მიმართ; MQ7 – მგრძნობიარეა ნახშირბადის მონოქსიდის მიმართ და MQ3, რომელიც მგრძნობიარეა ალკოჰოლისა და ეთანოლის მიმართ, ამიტომ სწორედ ამას გამოვიყენებთ ჩვენს ალკოანალიზატორში.

**ეს სახალისო პროექტია და არ უნდა იყოს გამოყენებული ალკოჰოლის შემცველობის ზუსტი დონის განსაზღვრისთვის!**

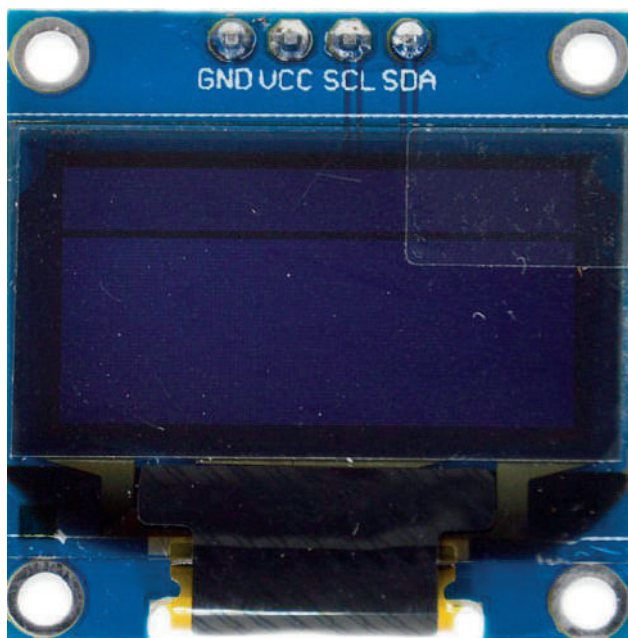
Keyes MQ3 მოდული აღჭურვილია ამ პროექტისთვის საჭირო რესურსით, მათ შორის, ჩაშენებული პოტენციომეტრითა (ცვლადი წინაღობით) და წინაღობით. მოდულს აქვს სამი პინი: OUT, VCC და GND.

როგორც MQ სენსორების უმეტესობა, MQ3 მოდულს შიგნით აქვს პატარა გამათბობელი ელექტროქიმიური სენსორით, რომელიც გამოიყენება გაზის დონის გასაზომად. მაჩვენებლების მნიშვნელობა იგზავნება OUT პინზე, რომელიც შემდეგ ჩვენს არდუინოზე ანალოგური გამომყვანით იკითხება (იხ. სურათი 1).



სურათი 1. Keyes MQ3 ალკოჰოლის სენსორის მოდული

სენსორის მაჩვენებლების წასაკითხად ჩვენ გამოვიყენებთ OLED ეკრანს (სურათი 2). OLED ეკრანი შედგება თხელი მრავალშრიანი ორგანული ფირისგან, რომელიც მოთავსებულია ანოდსა და კათოდს შორის. ძაბვის მიწოდებისას გამოსახულება იქმნება ელექტროლუმინესცენციით, ამიტომ, ეკრანი არ საჭიროებს განათებას. ჩვენი OLED ეკრანი არის I2C 128x64-ის მონოქრომული ვერსია, რაც იმას ნიშნავს, რომ ჩვენ შეგვიძლია მისი მართვა არდუინოს მხოლოდ ორი პინით და მისი ზომაა 128x64 პიქსელი. ეს ეკრანი იყენებს იმავე საკომუნიკაციო პროტოკოლს, როგორც ჩვენი თხევადკრისტალური LCD ეკრანი.



სურათი 2. მონოქრომული OLED ეკრანი 128x64

როდესაც MQ3 კითხულობს მნიშვნელობას, არდუინო აგზავნის შეტყობინებას OLED ეკრანზე, სადაც მითითებულია, იყო თუ არა აღმოჩენილი ალკოჰოლი.

## ყურადღება!

**MQ3 სენსორთან მუშაობის დროს იყავით ფრთხილად!**

როგორც ზემოთაღ აღვნიშნეთ, MQ3 სენსორში გამოიყენება შიდა გამათბობელი, რომელიც სენსორის სამუშაო პროცესის ნაწილია. სენსორის მუშაობისას გამათბობელი შეიძლება გაცხელდეს 120-140<sup>0</sup>-მდე.

## პანაუთსკება

1. ვიდრე დავიწყებთ სენსორის გამოყენებას, ის უნდა „გამოვწვათ“. ეს პროცესი – რამდენიმე საათის განმავლობაში სენსორი უნდა გავაჩეროთ ჩართულ მდგომარეობაში, რომ მექანიზმი შიგნიდან გახურდეს – აუმჯობესებს სენსორის სიზუსტეს. ამისათვის საჭიროა MQ3-ის VCC და GND კონტაქტები გამტარებით დააკავშიროთ არდუინოს +5V და GND კონტაქტებთან. როცა არდუინოს მიაწვდით კვებას, ის, თავის მხრივ, მიაწვდის ძაბვას MQ3 სენსორს. დატოვეთ სენსორი ჩართული 2-3 საათის განმავლობაში – ამ დროს თქვენ შეიძლება იგრძნოთ დამწვრის სუნი და სენსორი გახურდეს, მაგრამ ეს ნორმალურია.

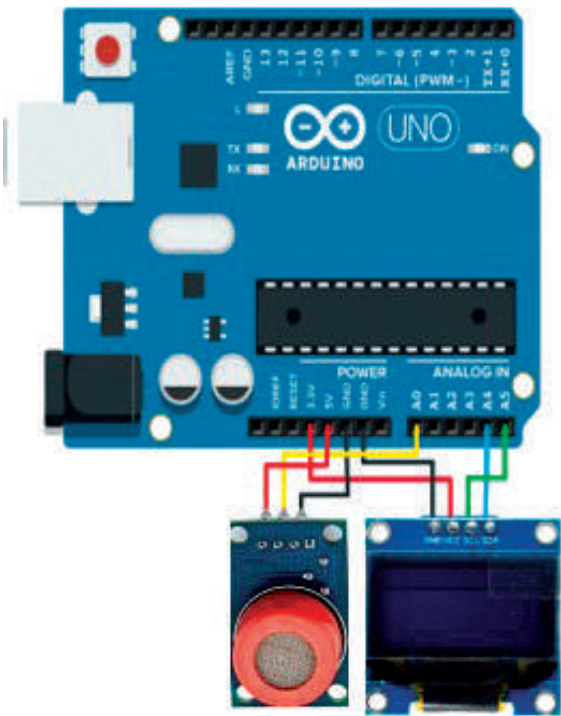
2. მას შემდეგ, რაც სენსორი „გამოიწვება“, გაუთიშეთ კვება არდუინოს და MQ3 სენსორის OUT პინი გამტარით დააკავშირეთ არდუინოს A0 გამომყვანს, ხოლო პინები VCC და GND დატოვებულ იქნას პოზიციებზე, როგორც გქონდათ მითითებული.

| MQ3 სენსორი | ARDUINO |
|-------------|---------|
| OUT         | Pin A0  |
| VCC         | +5V     |
| GND         | GND     |

3. დააკავშირეთ OLED ეკრანი არდუინოსთან: SCL – A5-თან, SDA – A4-თან, VCC – +3.3V-თან და GND – GND-თან.

| OLED ეკრანი | ARDUINO |
|-------------|---------|
| SCL         | Pin A5  |
| SDA         | Pin A4  |
| VCC         | +3.3V   |
| GND         | GND     |

4. ამ პროექტის სწორად ამუშავებისთვის თქვენ დაგჭირდებათ რამდენიმე ბიბლიოთეკის ჩატვირთვა ადრუინოში. SPI და Wire ბიბლიოთეკები უკვე ჩატვირთულია არდუინოს IDE-ში, მაგრამ OLED ეკრანის სამართავად თქვენ ასევე დაგჭირდებათ ბიბლიოთეკები **Adafruit\_GFX** და **Adafruit\_SSD1306**. ჩვენ წინამდებარე სახელმძღვანელოში დაწვრილებით განვიხილეთ ბიბლიოთეკების დამატება არდუინოს IDE ინტერფეისში (იხ. თავი „არდუინოს დამატებითი ბიბლიოთეკების ინსტალაცია“).



სურათი 3. სქემა

5. დარწმუნდით, რომ თქვენი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 3) და მხოლოდ ამის შემდეგ ჩატვირთეთ კოდი.
6. სენსორის შიგნით მოთავსებული გამათბობელის გათბობას დასტირდება დაახლოებით 4 წუთი იმისათვის, რომ ზუსტად იმუშაოს. კოდში მითითებულია მთვლელები, ამიტომ, ეკრანზე მნიშვნელობები ჩართვისთანავე არ აისახება, საჭიროა გავიდეს გარკვეული დრო. ტექსტი „Warming up“ გამოსახული იქნება ეკრანზე მანამ, სანამ სენსორი არ იქნება მზად მუშაობისთვის.

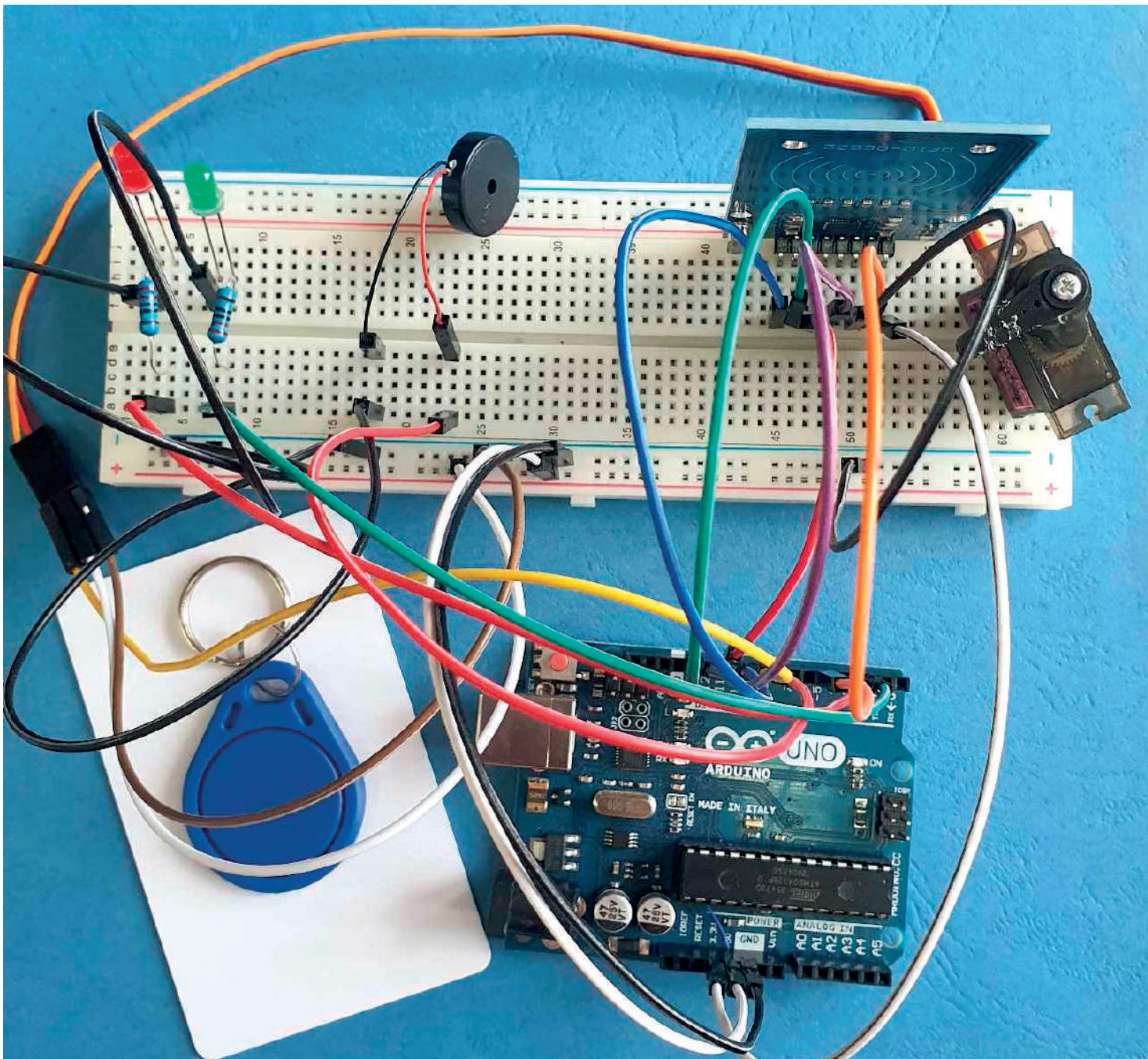
## როგორ მუშაოს კოდი

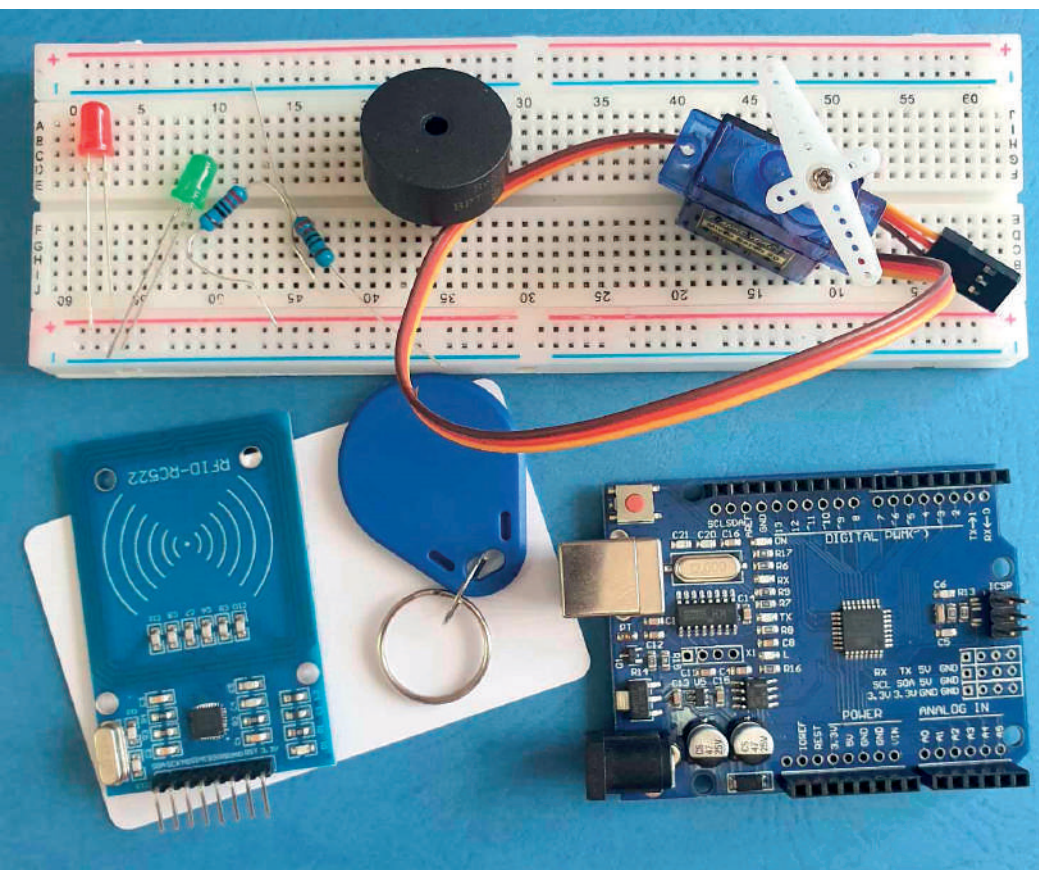
კოდი კომუნიკაციისა და OLED ეკრანის სამართავად თავდაპირველად მიმართავს ბიბლიოთეკებს SPI, Wire, Adafruit\_GFX და Adafruit\_SSD1306. გახურების დროდ განვსაზღვრავთ 4 წუთს. არდუინო ეკრანზე აგზავნის ანალოგური პინიდან წაკითხულ სხვადასხვა შეტყობინებებს. მაგალითად, თუ სენსორის მაჩვენებელი აღემატება 200-ს, არდუინო შეგეკითხებათ, დალიეთ თუ არა ლუდი. თუ მნიშვნელობა დაბალია, არდუინო გეტყვით, რომ თქვენ ფხიზელი ხართ. ალკოჰოლის მინიმალური დონე, რომელსაც MQ3 კითხულობს, შეადგენს 180-ს. გადაჭარბებისას კი (450) ალკოტესტერი შეგატყობინებთ, რომ თქვენ მთვრალი ხართ.

ალკოტესტერის გამოყენებისთვის საჭიროა დაელოდოთ, ვიდრე სენსორი გათბება, შემდეგ ფრთხილად ამოისუნთქეთ. ეცადეთ, სენსორი არ დაასველოთ. ამ შემთხვევაში მაჩვენებლები მცდარი იქნება.

XVII პროექტი

# უსაღნო იღანვოჟოკასიოს სისგამე – RFID ნამკითხველი





### საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- პასიური Piezo დინამიკი
- Servo ძრავა Pro SG90 9g
- 2 ცალი LED ნათურა (წითელი და მწვანე)
- 2 ცალი 220 ომი წინაღობა
- RFID მოდული MFRC522 (13,56 მჰც)

### საჭირო ბიბლიოთეკები

- RFID
- SPI
- Wire
- Servo
- Pitches

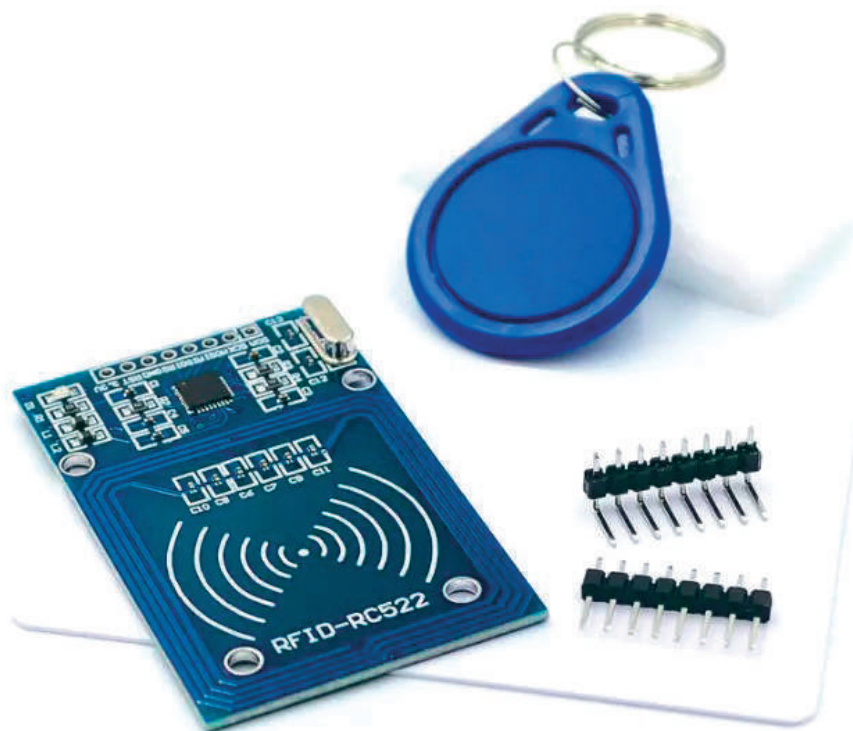
## როგორ გუშაობს RFID წამკითხველი

ამ პროექტში ჩვენ გამოვიყენებთ რადიოსიხშირულ საიდენტიფიკაციო წამკითხველს (RFID) ბარათების უსადენო საიდენტიფიკაციო სისტემის შესაქმნელად.

RFID წამკითხველი იყენებს უსადენო ტექნოლოგიას ბარათის, ბრელოკის... უკონტაქტო იდენტიფიცირებისთვის. წამკითხველი რეაგირებს, როცა მას მიუახლოებენ ბარათს. თავდაპირველად ჩვენ გვჭირდება, რომ წამკითხველმა წაიკითხოს ჩვენი RFID ბარათის უნიკალური ნომერი. შემდეგ დავამატებთ Servo ძრავას, რომელიც იმოძრავებს იმის მიხედვით, ამოიცნობს თუ არა RFID წამკითხველი ბარათს. იდენტიფიცირების ამ სისტემის გამოყენება შესაძლებელია კარის ან ყუთის საკეტზე. წამკითხველები და RFID ბარათები ხშირად გამოიყენება, როგორც იდენტიფიკატორები შეზღუდული დაშვების მქონე ზონებში, მაგალითად, ზესაიდუმლო ლაბორატორიებსა და დახურულ დასახლებებში.

არსებობს ორი სახის წამკითხველი: აქტიური და პასიური. ორივე სისტემა წამკითხველსა და ბარათს შორის სიგნალის გასაცვლელად იყენებს რადიოსიხშირეს. ეს სიგნალი შეიცავს ბარათის უნიკალურ კოდს და იმ შემთხვევაში, თუ RFID წამკითხველი ამოიცნობს კოდს, რეაგირებს შესაბამისად — მაგალითად, გააღებს კარს...

აქტიური წამკითხველები ძვირია და გამოიყენება გაცილებით რთულ მოწყობილობებში, ამიტომ ჩვენი პროექტისთვის დაგვჭირდება RFID-ის პასიური სისტემა: Mifare RFID-RC522 წამკითხველი, რომელსაც თან ახლავს ბარათი და ბრელოკი (იხ. სურათი 1).

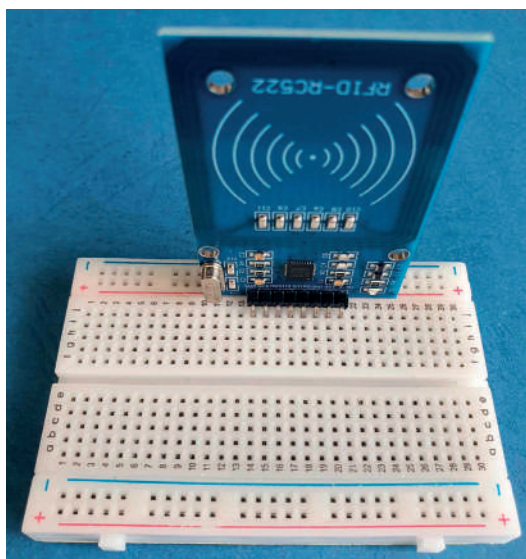


სურათი 1. RFID წამკითხველი, ბარათი და ბრელოკი

წამკითხველი მუშაობს 13,56 მეგჰც-ზე და მისი მოქმედების ზონა ძალიან მცირეა – ამის გათვალისწინება აუცილებელია.

### აპაზოტსება

1. თავდაპირველად დაგჭირდებათ RFID წამკითხველის საკონტაქტო პინების მირჩილვა, რომ შეძლოთ მისი სამაკეტო დაფაზე დამაგრება.
2. დაამაგრეთ წამკითხველი სამაკეტო დაფაზე (იხ. სურათი 2).

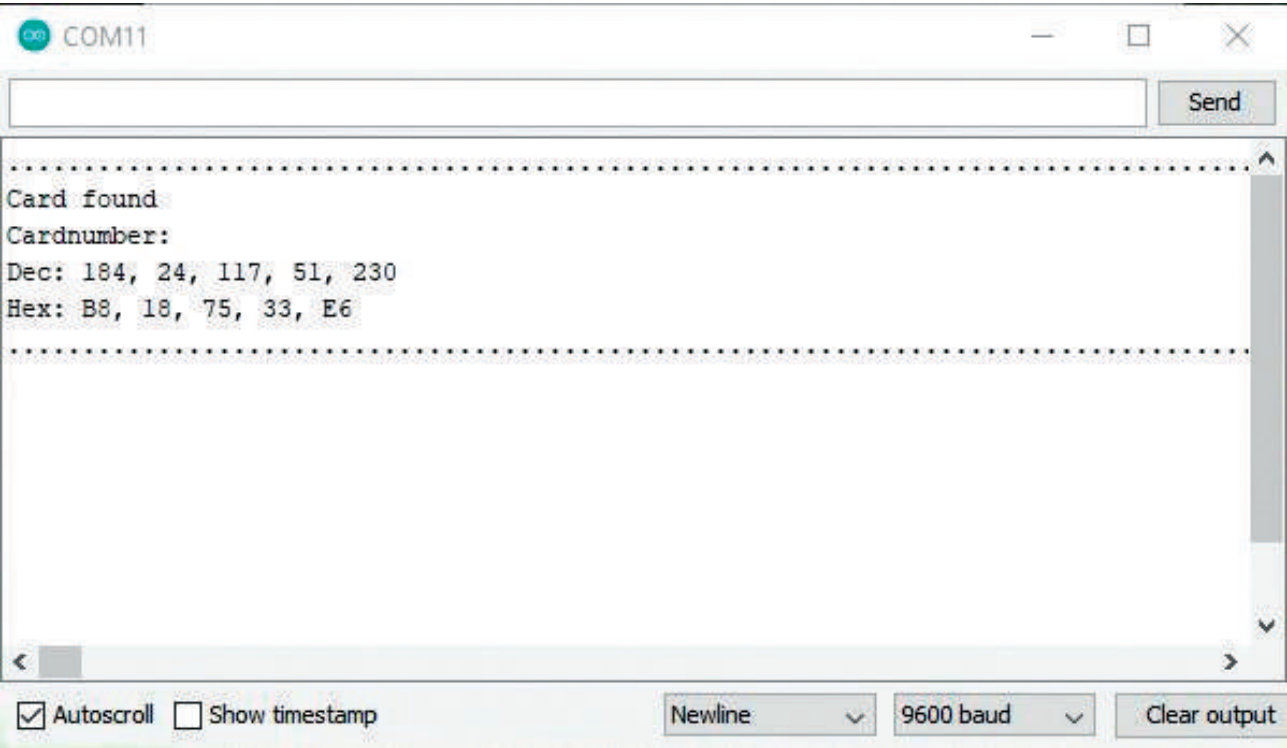


სურათი 2. RFID მოდული სამაკეტო დაფაზე

და გამტარებით დააკავშირეთ არდუინოსთან. გაითვალისწინეთ, რომ წამკითხველი უნდა დააკავშიროთ არდუინოს +3.3 V-თან (და არა +5V-თან), თორემ მოდულს დააზიანებთ.

| RFID | ARDUINO     |
|------|-------------|
| 3.3V | 3.3V        |
| RST  | Pin 5       |
| GND  | GND         |
| IRQ  | არ ვიყენებთ |
| MISO | Pin 12      |
| MOSI | Pin 11      |
| SCK  | Pin 13      |
| SDA  | Pin 10      |

- 3. ახლა უნდა შევამოწმოთ, მუშაობს თუ არა ჩვენი RFID მოდული (ამისათვის თქვენ უკვე უნდა გქონდეთ ჩატვირთული RFID ბიბლიოთეკა). ჩატვირთეთ წამკითხველის სატესტო კოდი **Project\_17a** არდუინოში.
- 4. გახსენით მიმდევრობითი კომუნიკაციის მონიტორი (serial monitor) არდუინოს IDE ინტერფეისში.
- 5. ბარათი ან ბრელოკი მიუახლოვეთ RFID მოდულს. მონიტორზე უნდა გამოჩნდეს უნიკალური ნომერი (იხ. სურათი 3). ჩაიწერეთ ეს ნომერი, რადგან მოგვიანებით დაგჭირდებათ მისი გამოყენება. ჩვენ შემთხვევაში ბარათის ნომერია B8, 18, 75, 33, E6 .



სურათი 3. ბარათის ნომერი

6. სამაკეტო დაფაზე დაამაგრეთ LED ნათურები. მათი მოკლე, უარყოფითპოლუსიანი ფეხები დაუკავშირეთ სამაკეტო დაფის უარყოფით ლიანდაგს, ხოლო გრძელი, დადებითპოლუსიანი ფეხები 220 ომი წინააღობის გავლით: წითელი LED ნათურა – არდუინოს მე-3 საკონტაქტო პინს, მწვანე LED ნათურა – არდუინოს მე-2 საკონტაქტო პინს.

| LED ნათურები           | ARDUINO                         |
|------------------------|---------------------------------|
| უარყოფითი ფეხები       | GND                             |
| დადებითი ფეხი (წითელი) | Pin 3 220 ომი წინააღობის გავლით |
| დადებითი ფეხი (მწვანე) | Pin 2 220 ომი წინააღობის გავლით |

7. დააკავშირეთ Servo ძრავა არდუინოსთან: წითელი გამტარი – არდუინოს +5V-თან, ყავისფერი/შავი გამტარი – არდუინოს GND-თან, ხოლო ყვითელი გამტარი – არდუინოს მე-9 საკონტაქტო პინთან.

| Servo ძრავა            | ARDUINO |
|------------------------|---------|
| წითელი გამტარი         | +5V     |
| შავი/ყავისფერი გამტარი | GND     |
| ყვითელი გამტარი        | Pin 9   |

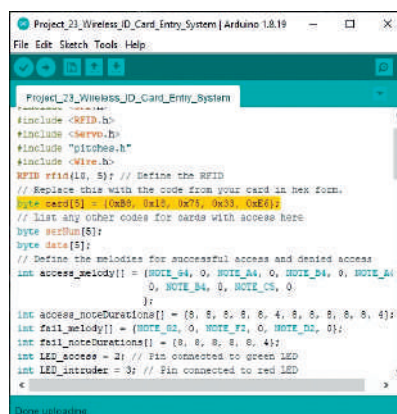
8. დააკავშირეთ Piezo დინამიკი არდუინოსთან: წითელი გამტარი – არდუინოს მე-8 საკონტაქტო პინთან, ხოლო შავი – GND-თან.

| Piezo დინამიკი     | ARDUINO |
|--------------------|---------|
| წითელი გამტარი (+) | Pin 8   |
| შავი გამტარი (-)   | GND     |

9. ახლა გახსენით პროექტის კოდი Project\_17b და შეცვალეთ შემდეგი სტრიქონი თქვენი ბარათის მონაცემებით.

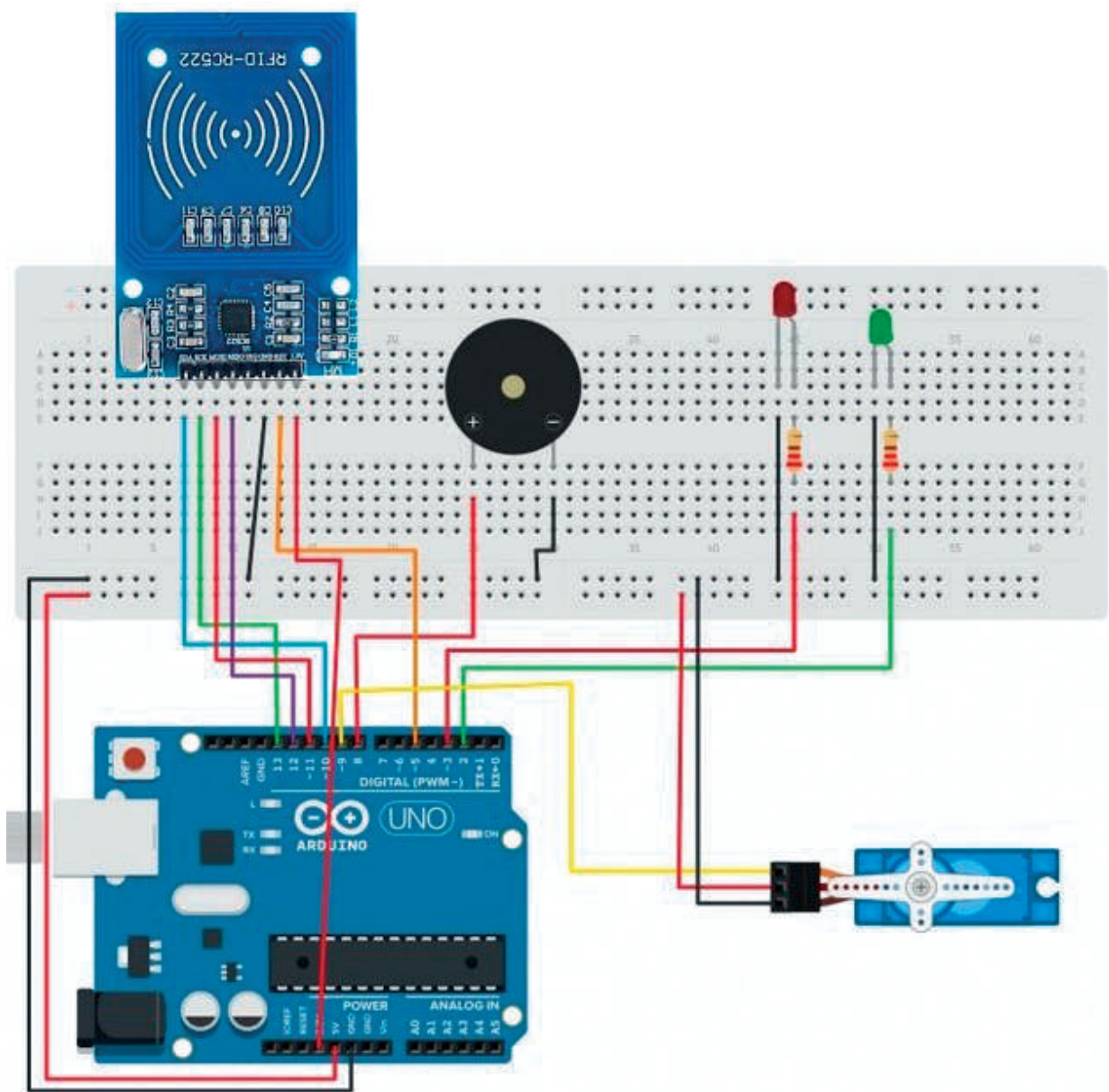
Byte card {5} = {0xB8, 0x18, 0x75, 0x33, 0xE6};

გაითვალისწინეთ, 0x ტოვებთ უცვლელად და დანარჩენს ცვლით თქვენი ბარათის ნომრით (იხ. სურათი 4).



სურათი 4. ბარათის ნომრის სტრიქონი

10. თუ თქვენ ნაბიჯ-ნაბიჯ და ზუსტად შეასრულეთ ჩვენი რჩევები, და ამასთანავე, თქვენი სქემა ზუსტად შეესაბამება ჩვენს სქემას (იხ. სურათი 5), შეგიძლიათ ჩატვირთოთ კოდი Project\_17b.



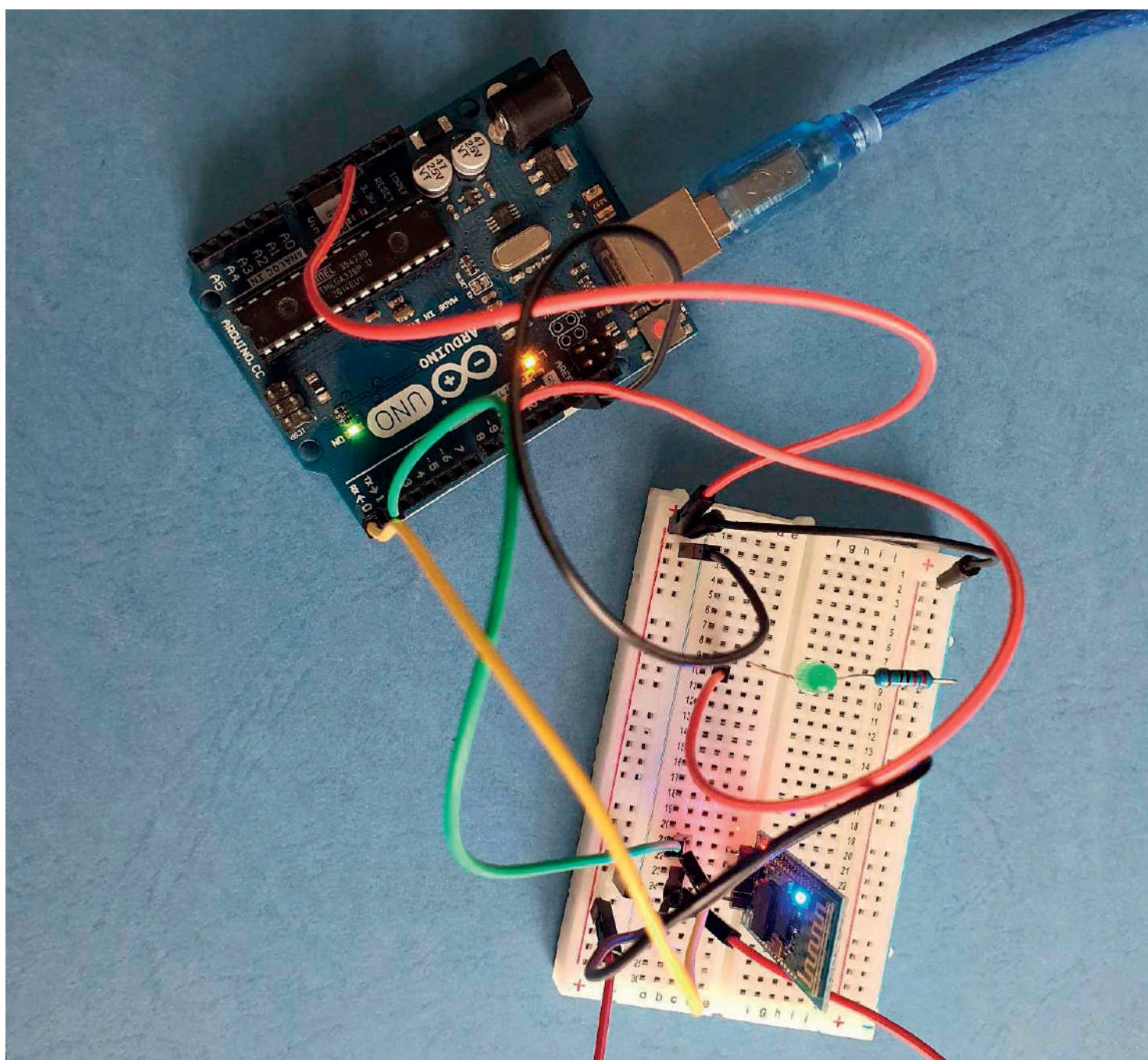
სურათი 5. სქემა

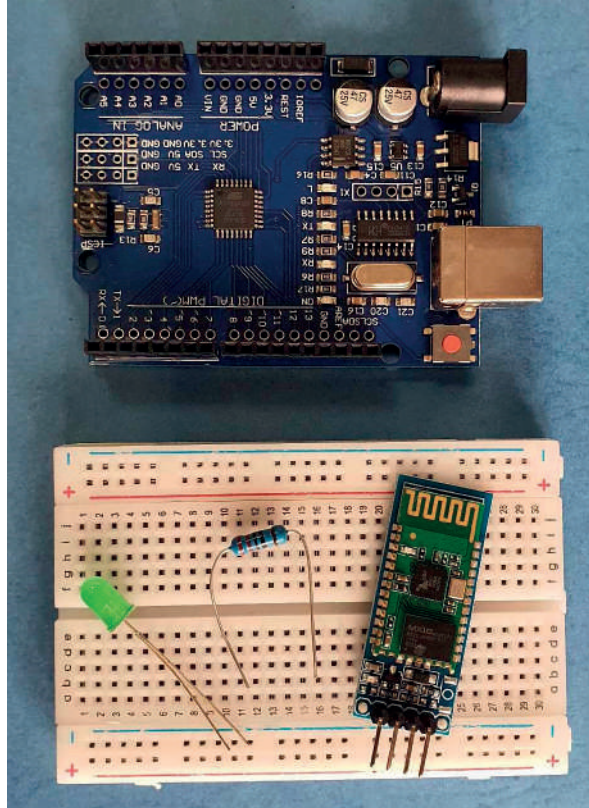
## როგორ გუშაოვს კოდი

კოდი იწყება SPI, RFID, Servo, Pitches და Wire ბიბლიოთეკებისადმი მიმართვით არდუინოს, RFID მოდულსა და Servo ძრავას შორის კავშირის სამართავად. გვაქვს ორი სახის მელოდია: ერთი – დადებითი, ბარათის ამოცნობის შემთხვევაში, მეორე – უარყოფითი, უცნობი ბარათის შემთხვევაში.

გაატარეთ ბარათი RFID მოდულის წინ (ახლო მანძილზე). თუ ბარათის უნიკალური ნომერი ემთხვევა თქვენ მიერ კოდში შეტანილ ნომერს (თუ სწორად შეიტანეთ), აინთება მწვანე შუქდიოდი, გაისმება მელოდია და Servo ძრავას მხარი მოტრიალდება. თუ შეცდომაა, ან სხვა ბარათს გამოიყენებთ, აინთება წითელი შუქდიოდი და გაისმება განსხვავებული მელოდია, Servo ძრავას მხარი კი უმოქმედოდ იქნება.

# შექმოდის მართვა ხმოვანი ბრძანებით





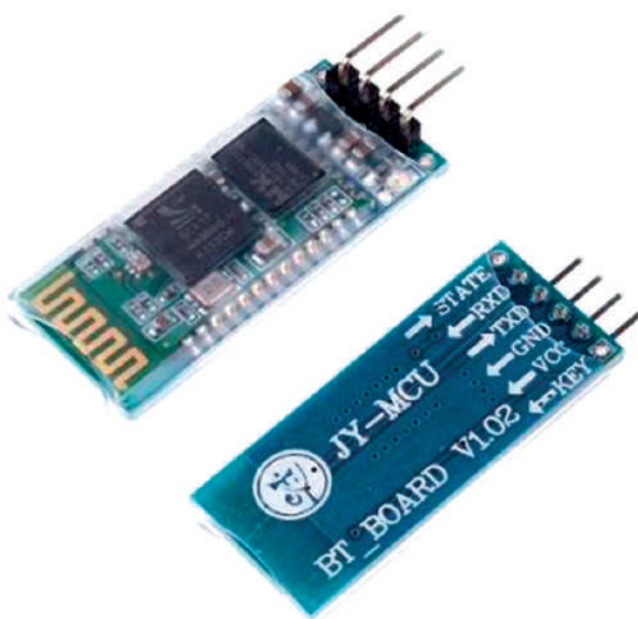
## საჭირო რესურსი

- Arduino UNO
- სამაკეტო დაფა
- გამტარები
- HC-06 Bluetooth მოდული
- LED ნათურა
- 220 ომი წინაღობა
- ანდროიდი სმარტფონი

## როგორ გუგაოს HC-06 BLUETOOTH მოდული

ამ პროექტში ჩვენ გამოვიყენებთ Bluetooth მოდულს, სმარტფონსა და ხმის ამომცნობ აპლიკაციას, რომ ხმოვანი ბრძანებებით ვმართოთ შუქდიოდი.

Bluetooth უსადენო ტექნოლოგია იყენებს რადიოტალღებს მონაცემების გადაცემისა და გაცვლისთვის მოკლე მანძილზე. სმარტფონები, ნოუტბუქები და მულტიმედიური მოწყობილობები, მაგალითად, აკუსტიკური სისტემები, იყენებენ Bluetooth-ს, როგორც საერთო სტანდარტს. ჩვენ გამოვიყენებთ იაფ HC-06 Bluetooth მოდულს (იხ. სურათი 1) არდუინოს სმარტფონთან დასაკავშირებლად, რაც საშუალებას მოგვცემს, დისტანციურად ვმართოთ შუქდიოდი ხმის ამომცნობი აპლიკაციის საშუალებით. ამ მოდულს აქვს ექვსი პინი, ჩვენ მხოლოდ შუა 4-ს გამოვიყენებთ. პინების აღმნიშვნელები მოდულს წინა მხარეს უნდა ჰქონდეს..



სურათი 1. HC-06 Bluetooth მოდული

აპლიკაცია, რომელსაც ჩვენ გამოვიყენებთ, არის BroxCode-ის Arduino Bluetooth Control და ის ხელმისაწვდომია Google Play-ში ანდროიდ მოწყობილობებისთვის, ასევე Play Store-ში. არსებობს ბევრი სხვა მსგავსი უფასო აპლიკაცია და მათი გამოყენების პრინციპი ერთი და იგივეა, მაგრამ BroxCode აპლიკაციას აქვს დამატებითი ფუნქციები, რომლებსაც ჩვენ ვიყენებთ ამ პროექტში, მაგალითად, ხმის ამოცნობა Google Assistant-ის საშუალებით.

აპაწყოთსკემა

Bluetooth კონტროლერის აწყობამდე აუცილებელია არდუინოში ჩატვირთოთ კოდი. ამას იმიტომ ვაკეთებთ, რომ მიმდევრობითი კავშირი კომპიუტერიდან არდუინოზე იყენებს იმავე კონტაქტებს, რომლებიც შემდეგში დაგვჭირდება Bluetooth მოდულთან დასაკავშირებლად.

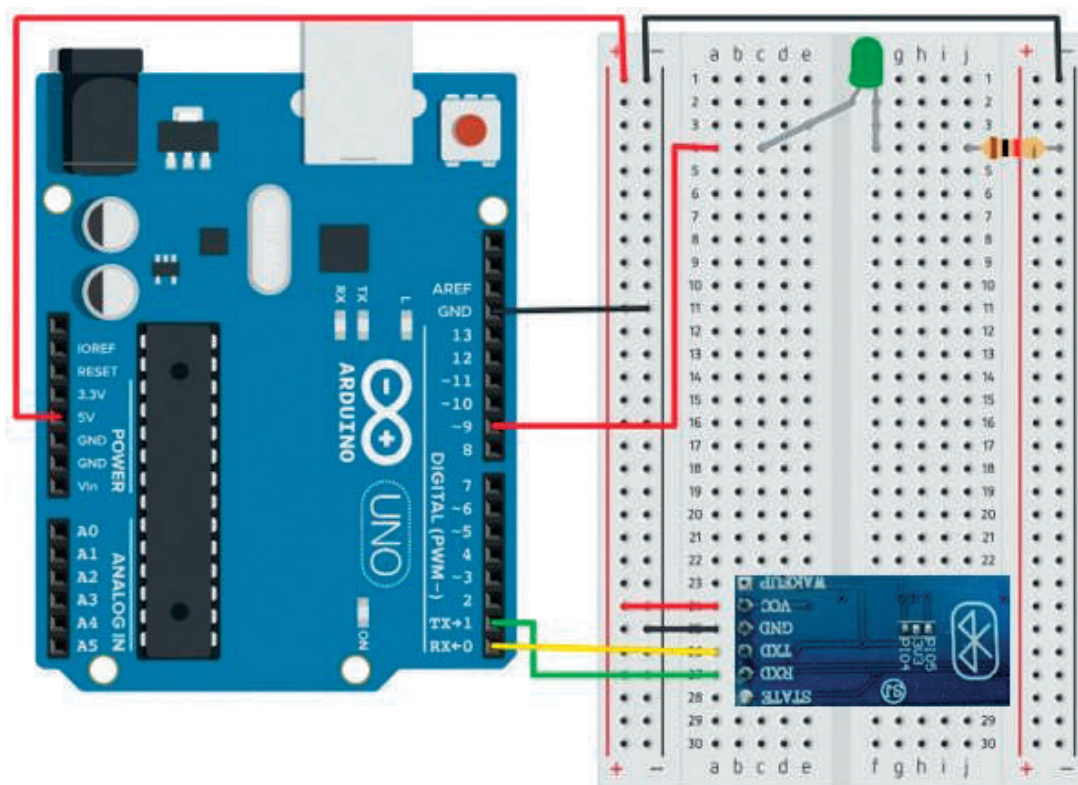
1. ჩატვირთეთ Bluetooth-ის კოდი არდუინოში. შემდეგ მოათავსეთ Bluetooth მოდული სამაკეტო დაფაზე და მისი VCC და GND საკონტაქტო პინები დააკავშირეთ სამაკეტო დაფის შესაბამის პოლუსიან ლიანდაგებთან: VCC – დადებითპოლუსიან ლიანდაგს, ხოლო GND – უარყოფითპოლუსიან ლიანდაგს. მოდულის TXD პინი დააკავშირეთ არდუინოს 0 (RX) გამომყვანთან, ხოლო RXD პინი – არდუინოს 1 (TX) გამომყვანთან.

| HC-06 BLUETOOTH მოდული | ARDUINO    |
|------------------------|------------|
| VCC                    | +5V        |
| GND                    | GND        |
| TXD                    | Pin 0 (RX) |
| RXD                    | Pin 1 (TX) |

2. შექდიოდი მოათავსეთ სამაკეტო დაფაზე ისე, რომ მისი ფეხები დაფის ცენტრალური გამყოფი ზოლის ორივე მხარეს მოხვდეს. შექდიოდის მოკლე, უარყოფითი ფეხი 220 ომი წინააღობის გავლით დააკავშირეთ სამაკეტო დაფის უარყოფითპოლუსიან ლიანდაგთან, ხოლო გრძელი, დადებითი ფეხი გამტარის საშუალებით დააკავშირეთ არდუინოს მე-9 საკონტაქტო პინთან.

| LED ნათურა     | ARDUINO |
|----------------|---------|
| დადებითი ფეხი  | Pin 9   |
| უარყოფითი ფეხი | GND     |

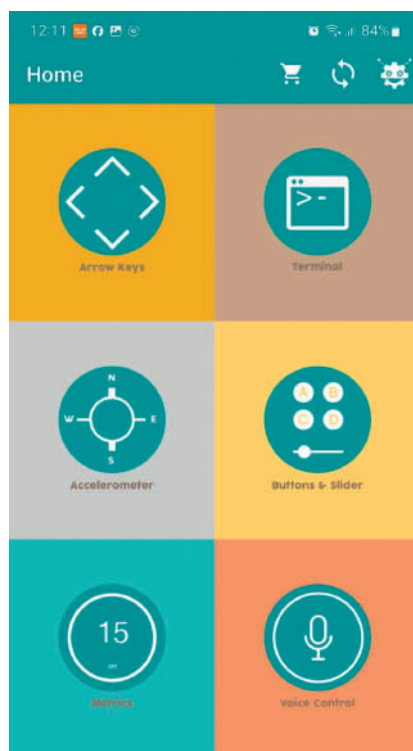
3. სამაკეტო დაფის დადებით- და უარყოფითპოლუსიანი ლიანდაგები დააკავშირეთ არდუინოს შესაბამის პინებთან: +5V და GND-თან.
4. დარწმუნდით, რომ თქვენი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 2).



სურათი 2. სქემა

## ARDUINO BLUETOOTH CONTROL

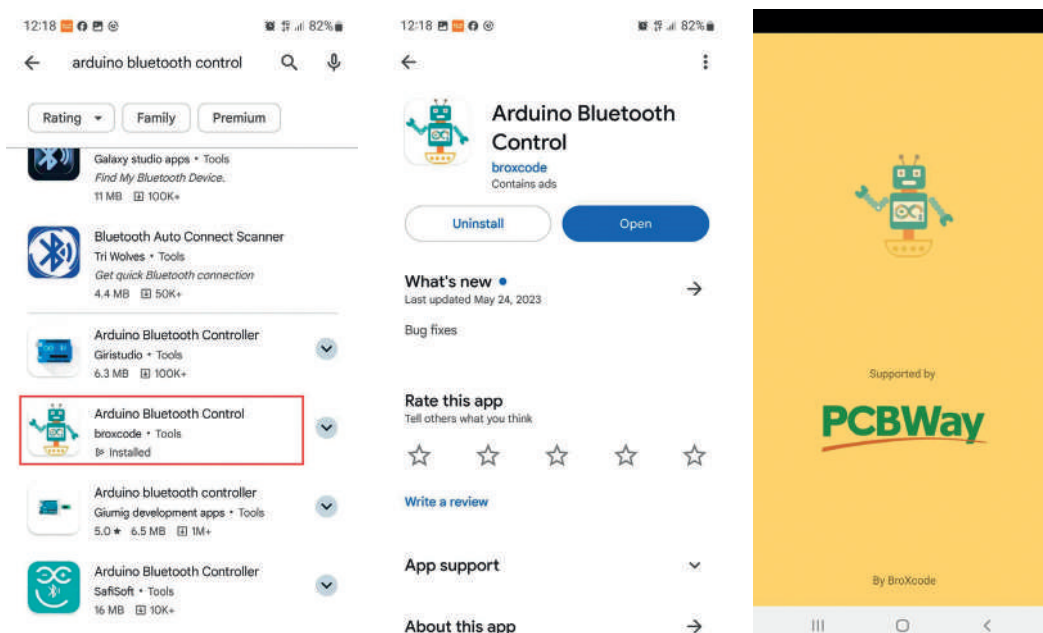
აპლიკაცია Arduino Bluetooth Control გვთავაზობს მართვის ექვს ვარიანტს და თითოეული მათგანი არდუინოზე მონაცემებს აგზავნის სხვადასხვა მეთოდით. თქვენ შეგიძლიათ გამოიყენოთ ის, რომელსაც მიაჩნებთ უპირატესობას.



სურათი 3. Arduino Bluetooth Control აპლიკაციის მენიუ

ჩვენ ამ პროექტში გამოვიყენებთ Voice Control-ს – ეს ინსტრუმენტი იყენებს Google-ის ხმოვან ბრძანებებს თქვენს ანდროიდ მოწყობილობებზე და საშუალებას გაძლევთ, მართოთ არდუინო თქვენი ხმოვანი ბრძანებების საშუალებით.

ახლა ჩამოტვირთეთ და დააყენეთ თქვენს ტელეფონში Arduino Bluetooth Control აპლიკაცია.

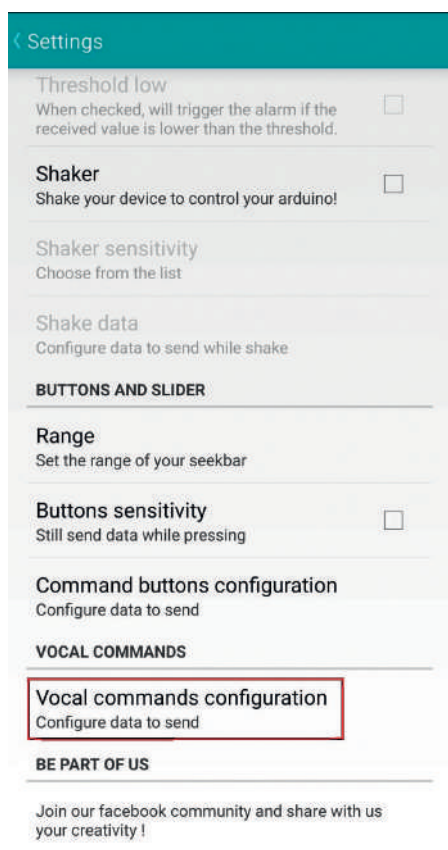


სურათი 4. Arduino Bluetooth Control აპლიკაცია

2. აპლიკაციის ჩამოტვირთვის შემდეგ ბლუთუზის გასააქტიურებლად ჩართეთ არდუინო. შედით სმარტფონის ბლუთუზის პარამეტრებში, ჩართეთ ბლუთუზი და აირჩიეთ MORE SETTINGS დასაკავშირებელი მოწყობილობების სანახავად. ხელმისაწვდომ მოწყობილობებში უნდა გამოგიჩვენოთ მოდული HC-06. აირჩიეთ ის. მოწყობილობასთან დასაკავშირებლად მოგთხოვთ პაროლის შეყვანას. მოდულისთვის გამოიყენება პაროლი 1234, ზოგჯერ 0000. იმ შემთხვევაში, თუ პირველი პაროლი არ იმუშავებს, მოსინჯეთ მეორე.
3. როდესაც მოწყობილობა გააქტიურდება, გახსენით Arduino Bluetooth Control აპლიკაცია. ყველა ხელმისაწვდომი მოწყობილობების გამოსახულების ფანჯრიდან აირჩიეთ HC-06 მოდული (იხ. სურათი 5). თქვენ არ დაგჭირდებათ მოწყობილობის არჩევა ყოველი ჩართვის დროს, აპლიკაცია დაიმახსოვრებს მას.



სურათი 5. დაკავშირება მოწყობილობასთან

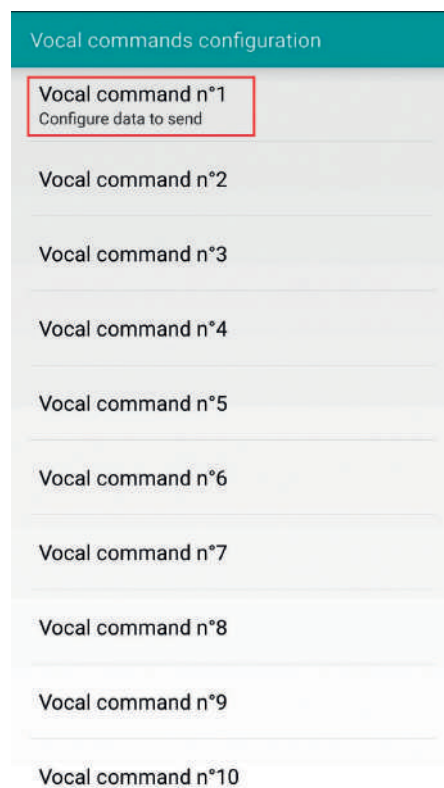


სურათი 6. ხმოვანი ბრძანებების კონფიგურაცია

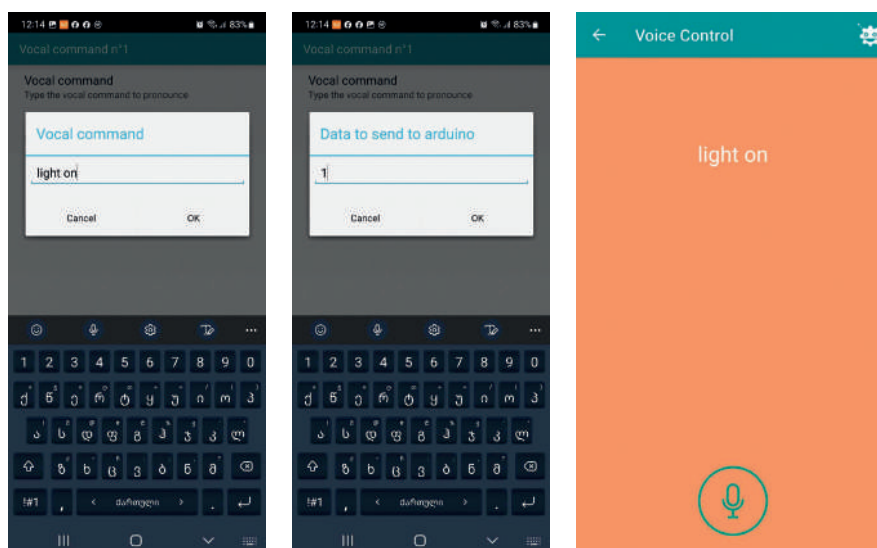
6. აქ თქვენ წერთ შემავალ მონაცემებს, რომელიც გამოიძახებს პირველ ფუნქციას. ჩაწერეთ ტექსტი light on (იხ. სურათი 8). ამის შემდეგ აპლიკაცია მოგთხოვთ გამოშვებულ მონაცემებს, რომლებიც უნდა გაიგზავნოს არდუინოზე, როცა მისცემთ ბრძანებას. ამ ეკრანზე შეიყვანეთ ციფრი 1. ეს ციფრი გაიგზავნება არდუინოზე, როგორც შემავალი სიგნალი, შუქდიოდს მიეწოდება კვება და აინთება.

4. ახლა ჩვენ უნდა გამოვიყენოთ ხმოვანი მართვის ფუნქცია, რომ გარკვეული ბრძანების წარმოთქმისას შუქდიოდი აინთოს და ჩაქრეს. აირჩიეთ ხმოვანი მართვის ფუნქცია. თქვენ მოხვდებით პარამეტრების მენიუში (იხ. სურათი 6). აირჩიეთ ხმოვანი ბრძანებების კონფიგურაცია (Vocal commands configuracion). მისი დახმარებით ჩვენ განვსაზღვრავთ შემავალ და გამოშვალ ფუნქციებს.

5. აირჩიეთ Vocal command n°1 (იხ. სურათი 7).

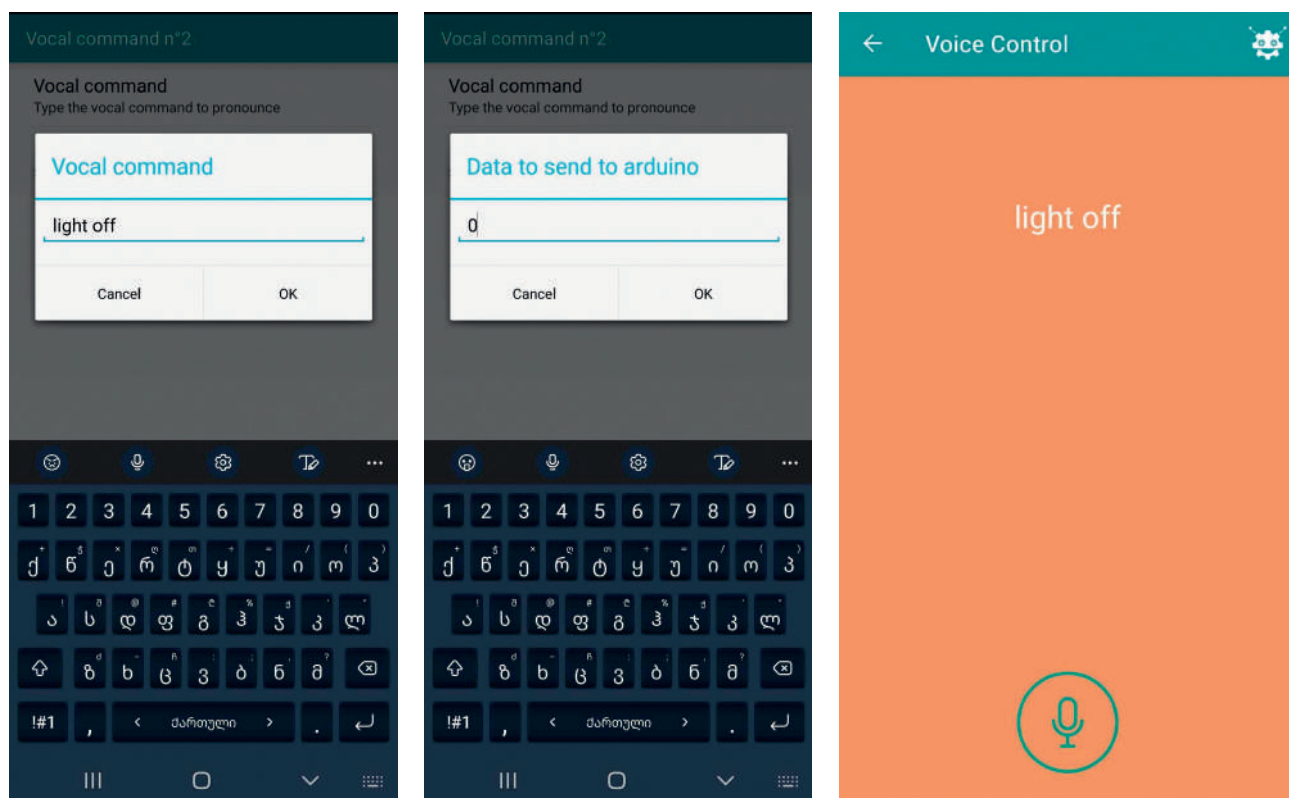


სურათი 7. თქვენი პირველი ხმოვანი ბრძანების პარამეტრი



სურათი 8. შუქდიოდის ანთება ხმოვანი ბრძანებით „light on”

7. შეასრულეთ იგივე მოქმედებები, რომ განსაზღვროთ n°2 ბრძანება – ამ შემთხვევაში ჩაწერეთ ტექსტი light off და 0. ეს ბრძანება შეუქმნის ჩაქრობს (იხ. სურათი 9).



სურათი 9. შეუქმნის ჩაქრობა ხმოვანი ბრძანებით „light off”

მაშ ასე, ჯერ ღილაკზე voice command და შემდეგ მიკროფონზე დაჭერისას, როცა წარმოთქვამთ ტექსტს light on – შეუქმნის აინთება, light off – შეუქმნის ჩაქრობა.

აპლიკაციას ასევე აქვს ფუნქცია, რომელიც საშუალებას იძლევა, არდუინო ვმართოთ sms-ის მეშვეობით.

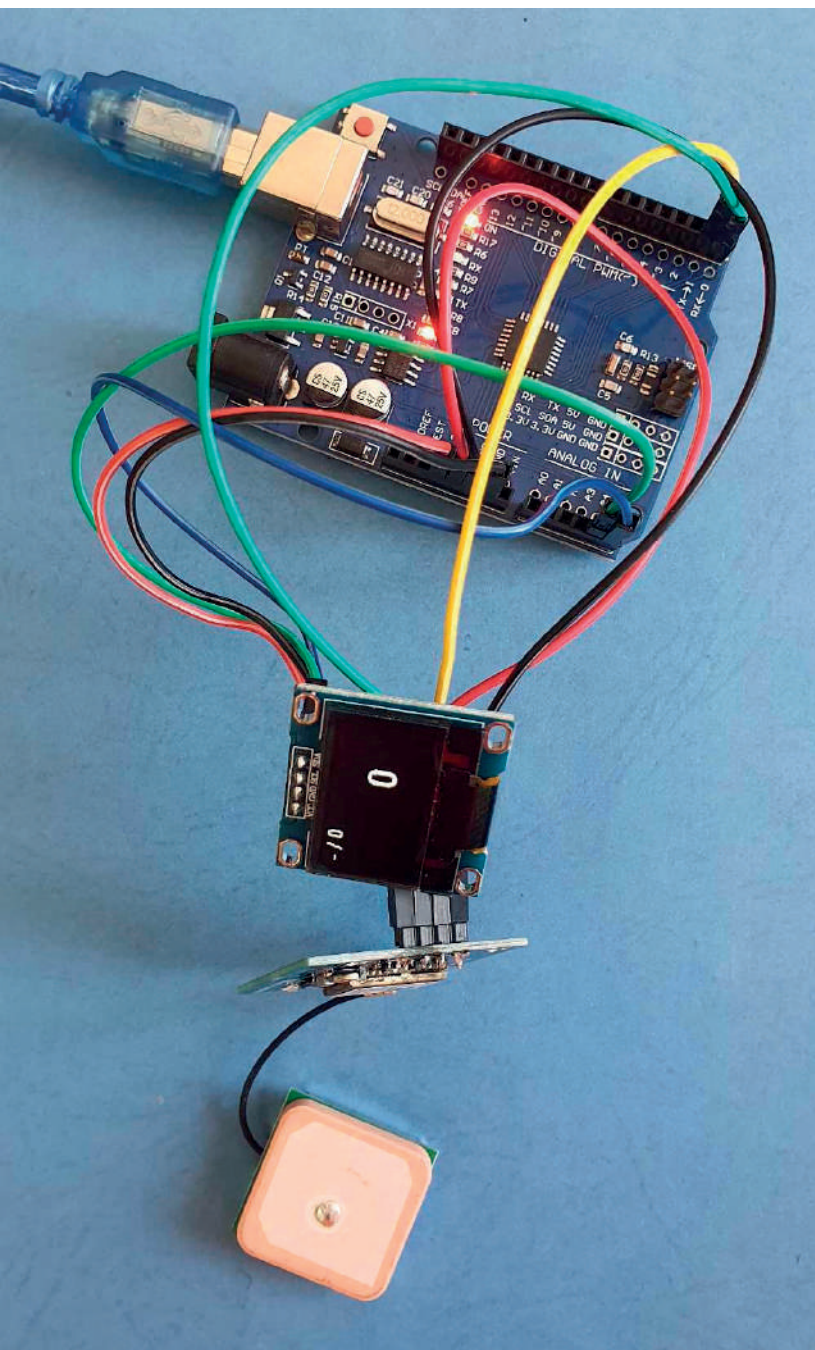
### როგორ გუგაოვს კოდი

პროექტის კოდი საკმაოდ მარტივია. ის იწყება ცვლადის შექმნით, რომელიც ინახავს მონაცემებს Bluetooth მოდულიდან. მონაცემთა გაცვლის სიჩქარედ ის განსაზღვრავს 9600 და შეუქმნის გამოყვანად განსაზღვრავს არდუინოს მე-9 საკონტაქტო პინს. ციკლში მოწმდება, ეგზავნება თუ არა არდუინოს მონაცემები Bluetooth მოდულიდან. ციკლი კითხულობს მონაცემებს, ასევე აგზავნის მათ მიმდევრობითი კომუნიკაციის მონიტორზე (Serial Monitor), რაც საშუალებას გვაძლევს, შევამოწმოთ, სწორად მუშაობს თუ არა ყველაფერი. თუ არდუინო მიიღებს აპლიკაციიდან 1-ს, მე-9 კონტაქტი განისაზღვრება როგორც HIGH და შეუქმნის აინთებას. თუ არდუინო მიიღებს 0-ს, მე-9 კონტაქტი განისაზღვრება როგორც LOW და შეუქმნის ჩაქრობა.

ამ პრინციპების გამოყენებით, შეგიძლიათ შეუქმნის ნაცვლად გამოიყენოთ სხვა რესურსი/სენსორი და ნებისმიერი ადგილიდან მართოთ თქვენი სახლი, გახადოთ ის ავტომატიზებული.

XIX პროექტი

# GPS სპილოვარი

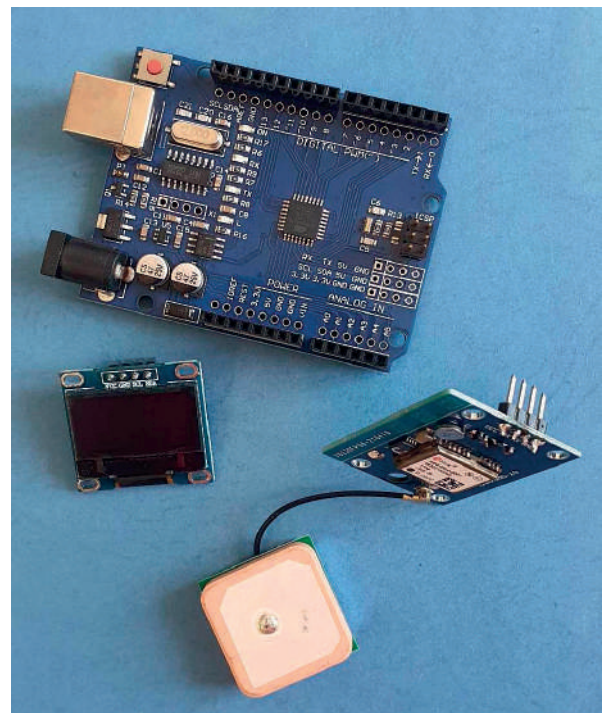


## საჭირო რესურსი

- Arduino UNO
- დედალ-მამალი გამტარები
- OLED ეკრანი (მონოქრომული, 128x64)
- GPS მოდული Ublox NEO-6M და ანტენა

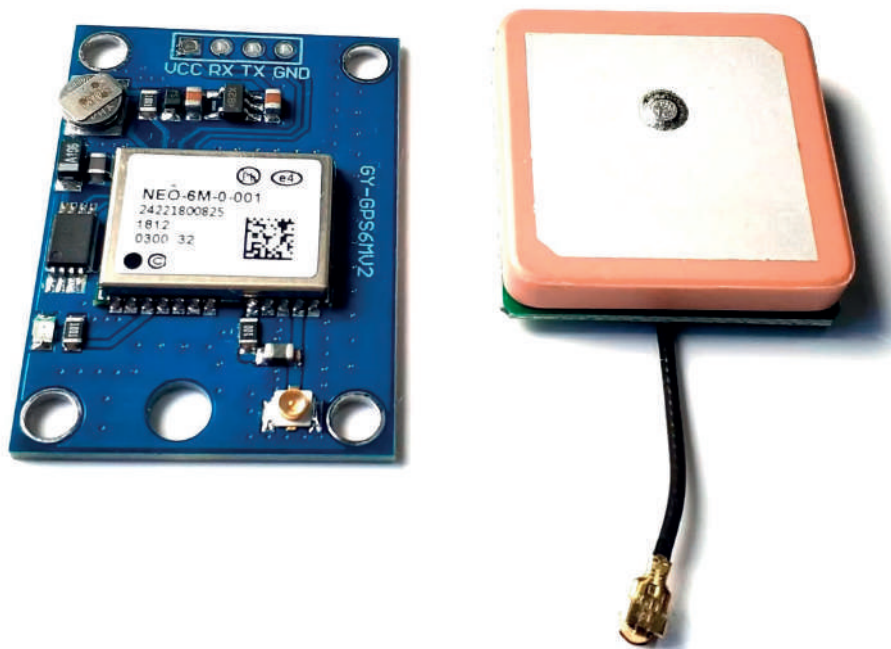
## საჭირო ბიბლიოთეკა

- U8glib



## როგორ გვემოგვს GPS მოდული UBLOX NEO-6M

ამ პროექტში ჩვენ დაგაკავშირებთ GPS მოდულსა და OLED ეკრანს არდუინოსთან და შევექმნით მარტივ GPS სპიდომეტრს, რომელიც გაზომავს თქვენს სიჩქარეს თანამგზავრიდან. GPS მოდული Ublox NEO-6M ჩვეულებრივ გამოიყენება თვითმფრინავებისა და უბილოტო აპარატების ადგილმდებარეობის დასადგენად. მოდულის შეძენისას გაითვალისწინეთ, რომ მას თან უნდა მოჰყვებოდეს ანტენა (იხ. სურათი 1).



სურათი 1. GPS მოდული Ublox NEO-6M და ანტენა

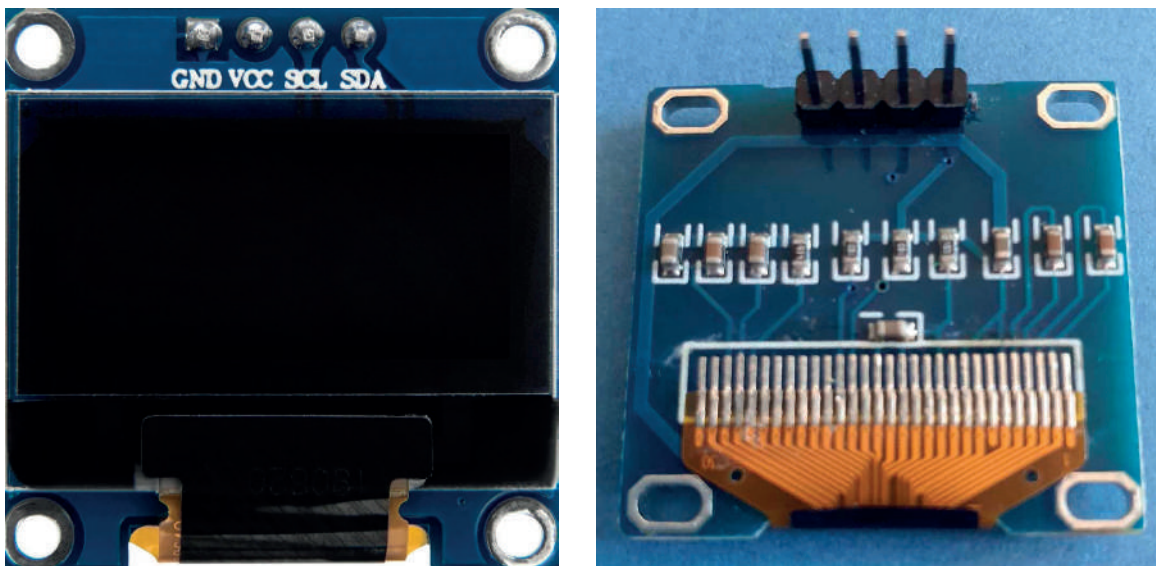
მოდული იყენებს GPS ტექნოლოგიას (Global Positioning System), რომ განსაზღვროს არდუინოს ზუსტი ადგილმდებარეობა და OLED ეკრანზე ასახოს მისი სიჩქარე კმ/სთ-ში. GPS შედგება დედამიწის გარშემო მოძრავი 32 თანამგზავრისგან და მთელ მსოფლიოში გამოიყენება ყოველდღიურ ტექნოლოგიებში, მაგ.: ავტომობილების თანამგზავრული სანავიგაციო სისტემებში, სმარტფონებში, ტრეკერებში...

გლობალური პოზიციონირების სისტემა (GPS) Navstar ამერიკის შეერთებული შტატების მთავრობის მიერ შეიქმნა 1970-იან წლებში და თავდაპირველად სამხედრო მიზნებისთვის გამოიყენებოდა, მაგრამ ახლა ის თავისუფლად ხელმისაწვდომია GPS მიმღების მქონე ნებისმიერი ადამიანისთვის. ზუსტად რომ განისაზღვროს მიმღების ადგილმდებარეობა, სისტემა იყენებს თანამგზავრებს, საკონტროლო სადგურებს დედამიწაზე და თქვენს მოწყობილობებს, რომ გამოთვალოს მანძილი, სიჩქარე და დრო გაგზავნილი და მისაღები სიგნალებისთვის – მათი დახმარებით ხდება შესაძლებელი თქვენი ადგილმდებარეობის დადგენა.

ჩვენს პროექტში GPS მოდული Ublox NEO-6M განუწყვეტლივ იღებს სატელიტურ სიგნალებს და აგზავნის მათ არდუინოზე თქვენი ადგილმდებარეობის დასადგენად. როგორც კი თქვენ მოძრაობას დაიწყებთ, თქვენი სიჩქარე გადაიცემა OLED ეკრანზე კმ/სთ-ში. ასე მივიღებთ მარტივ GPS სპიდომეტრს.

მიუხედავად იმისა, რომ ჩვენი პროექტის ფუნქციონალი საკმაოდ რთულია, მარტივია მისი აწყობა.

1. მოამზადეთ OLED ეკრანი (იხ. სურათი 2) არდუინოსთან დასაკავშირებლად. ამისათვის თქვენ დაგჭირდებათ დედალ-მამალი გამტარები. გახსოვდეთ, ეკრანი კვებისთვის იყენებს +3.3 V ძაბვას და ის არდუინოს +3.3V საკონტაქტო პინთან უნდა დააკავშიროთ და არა +5V-თან. წინააღმდეგ შემთხვევაში, თქვენი ეკრანი დაზიანდება.



სურათი 2. მონოქრომული OLED ეკრანი

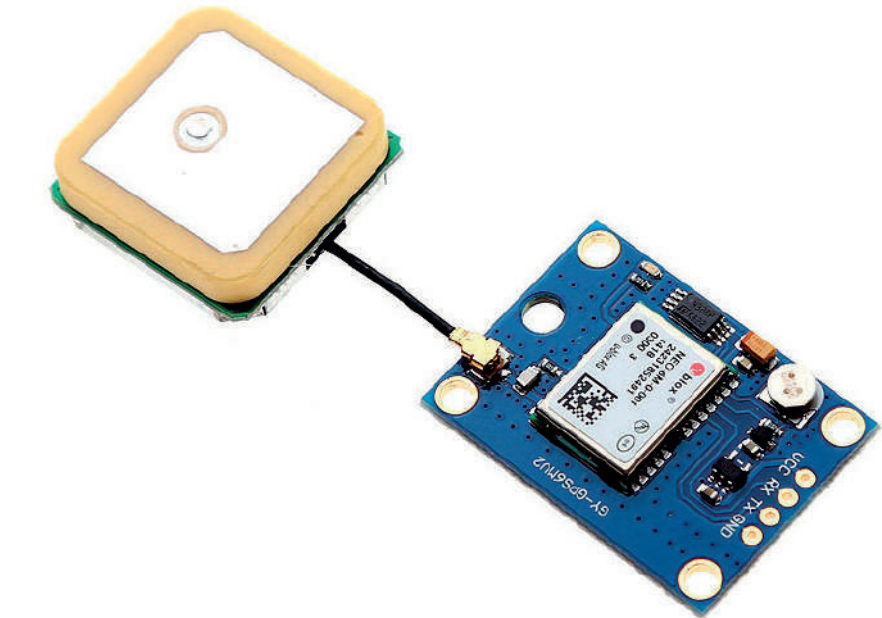
დააკავშირეთ ეკრანის VCC საკონტაქტო პინი არდუინოს +3.3V-თან, GND – GND-თან, SCL – მე-5 და SDA – მე-6 საკონტაქტო პინებთან.

| OLED ეკრანი | ARDUINO |
|-------------|---------|
| VCC         | + 3.3V  |
| GND         | GND     |
| SCL         | Pin A5  |
| SDA         | A4      |

2. GPS მოდული იყენებს არდუინოს RT და TX გამომყვანებს კავშირისთვის. ამავე დროს, ეს კონტაქტები საჭიროა კოდის ჩასატვირთად. ამიტომ, ჯერ ჩატვირთეთ კოდი Softwarserial არდუინოში ისე, რომ ეს კონტაქტები იყოს თავისუფალი. არ დაგავიწყდეთ, ამ პროექტისთვის დაგჭირდებათ U8glib ბიბლიოთეკა (იხ. თავი „არდუინოს დამატებითი ბიბლიოთეკების ინსტალაცია“).
3. ჩატვირთეთ კოდი, გამორთეთ არდუინო კომპიუტერიდან და მხოლოდ ამის შემდეგ დააკავშირეთ GPS მოდულის VCC კონტაქტი არდუინოს +5V-თან, GND – GND-თან, GPS მოდულის TX კონტაქტი არდუინოს 0 პინთან (RX) და GPS მოდულის RX კონტაქტი – არდუინოს 1 პინთან.

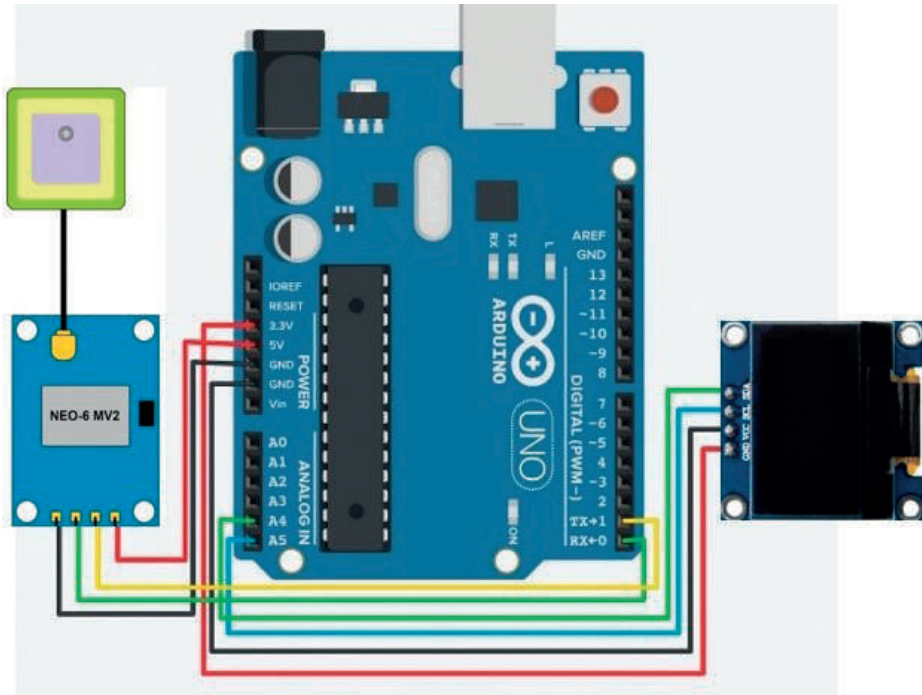
| GPS მოდული | ARDUINO    |
|------------|------------|
| VCC        | +5V        |
| GND        | GND        |
| TX         | Pin 0 (RX) |
| RX         | Pin 1 (TX) |

4. დაამაგრეთ ანტენა GPS მოდულზე ისე, როგორც ეს ნაჩვენებია მესამე სურათზე.



სურათი 3. GPS მოდულზე დამაგრებული ანტენა

5. დარწმუნდით, რომ თქვენი სქემა შეესაბამება ჩვენს სქემას (იხ. სურათი 4).



სურათი 4. სქემა

6. მიაწოდეთ კვება არდუინოს და GPS სპიდომეტრი მზად არის. ანტენის ზედაპირი (იხ. სურათი 3) მიმართული უნდა იყოს ზემოთ. ის უკეთესად მუშაობს გარეთ (შიძლება მოსინჯოთ შენობაშიც, მაგრამ ფანჯარასთან ახლოს), რადგან GPS მოდულს ესაჭიროება პირდაპირი ხედვა ორბიტალურ თანამგზავრებთან.
7. GPS მოდულს თანამგზავრთან დასაკავშირებლად დასჭირდება დაახლოებით 30 წამი. თუ კავშირი წარმატებულია, მოდულის ინდიკატორი აციმციმდება, ხოლო OLED ეკრანის ზედა, მარცხენა კუთხეში სიმბოლო დაიწყებს ტრიალს.

## კოდი

კოდი ჯერ მიმართავს U8glib ბიბლიოთეკას, შემდეგ განსაზღვრავს OLED-ს, რომ შევძლოთ ჩვენი ეკრანის მართვა. ჩვენ განვსაზღვრავთ GPS მოდულს, როგორც მიმდევრობით შეერთებას და მივუთითებთ მას, როგორი ინფორმაციის მიღება გვსურს თანამგზავრიდან.

### შენიშვნა

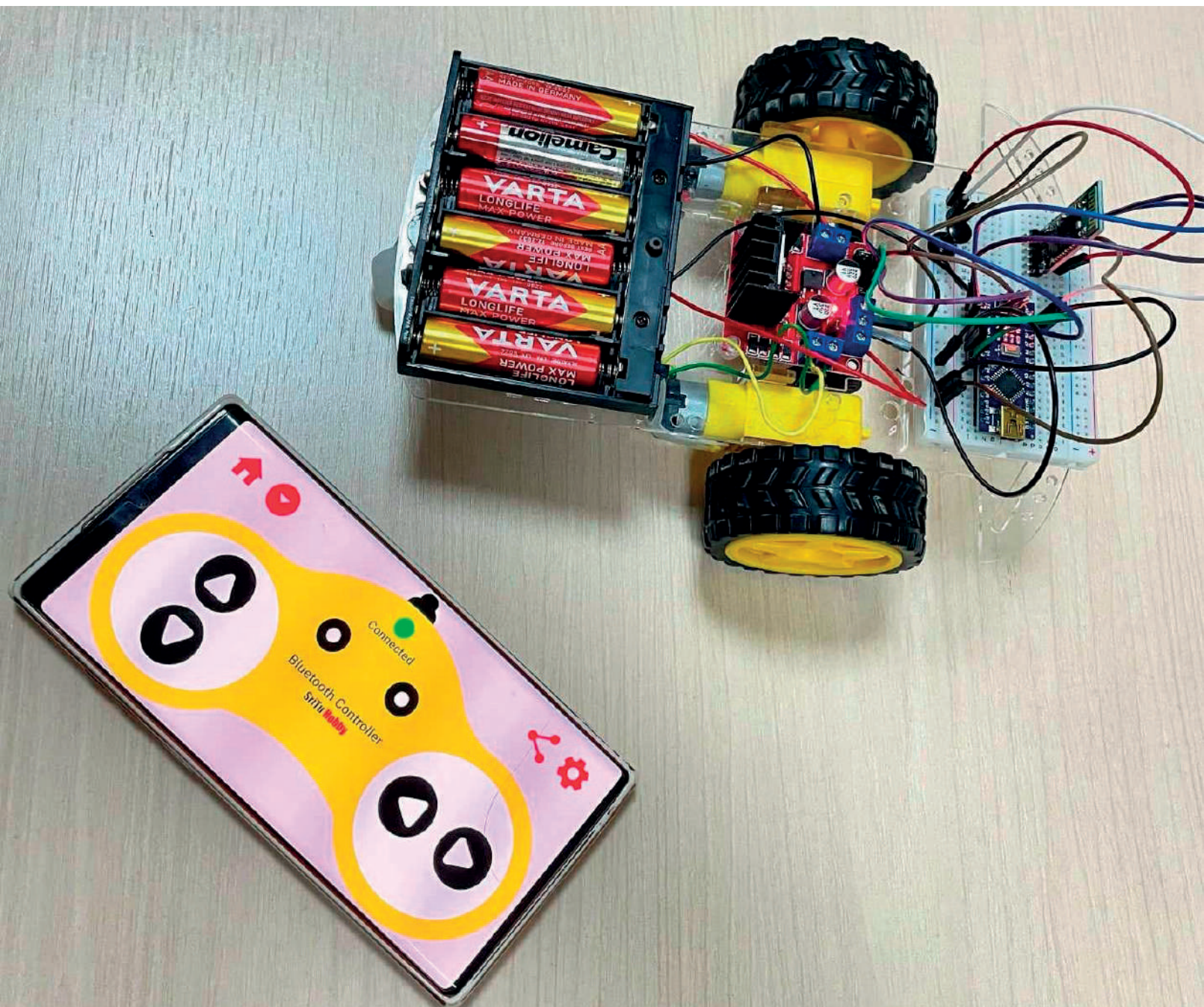
არ დაგავიწყდეთ კოდის ჩატვირთვამდე არდუინოს 0 (RX) საკონტაქტო პინის გათავისუფლება და შემდეგ კვლავ დაკავშირება.

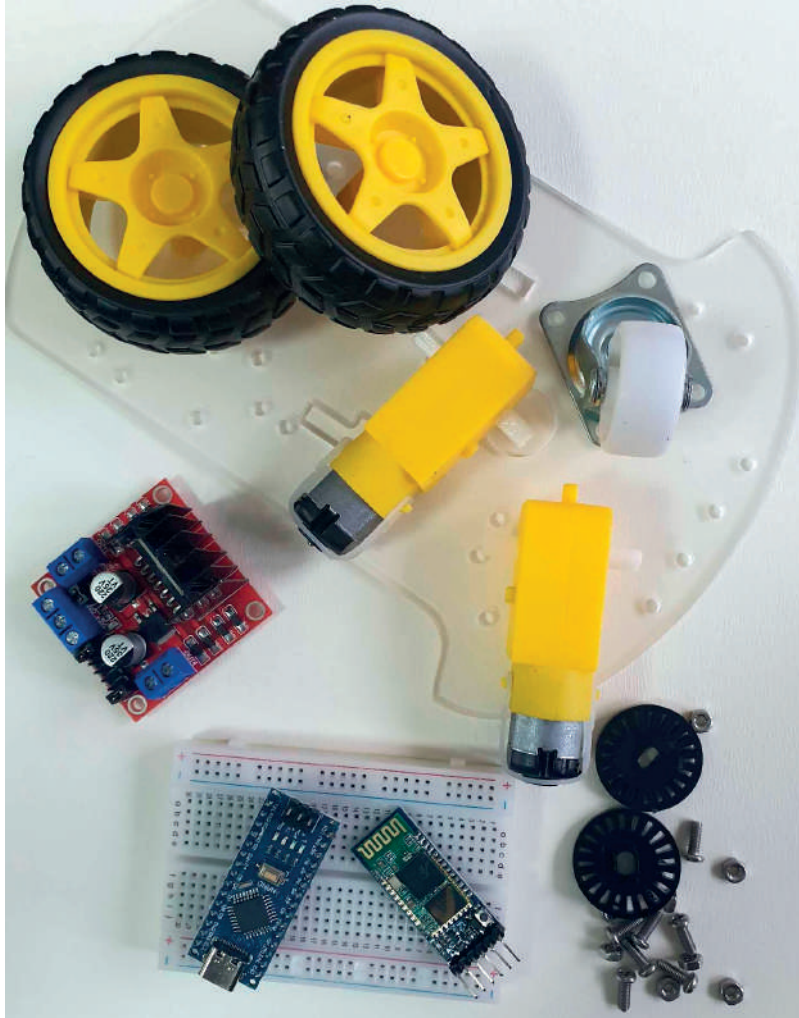
ციკლი კოდის ბოლოს ამოწმებს, ხდება თუ არა მონაცემების მიღება და თუ ეს ასეა, შესაბამისი სიმბოლო აისახება OLED ეკრანის ზედა, მარცხენა კუთხეში. პირველი სიმბოლო გვიჩვენებს, რომ ეკრანის განახლება ხდება სწორად, ხოლო მეორე – GPS-ის მონაცემების პაკეტი მიიღება თანამგზავრებიდან. ეკრანის ზედა, მარცხენა კუთხეში ასევე აისახება თანამგზავრების რაოდენობა, რომლებთანაც შედგა კავშირი.

თუ ყველა მონაცემი სწორად მოგვეწოდება, ეკრანზე გამოისახება მოძრაობის სიჩქარე კმ/სთ-ში.

XX პერიოდი

## 2WD ჭკვიანი მანქანა



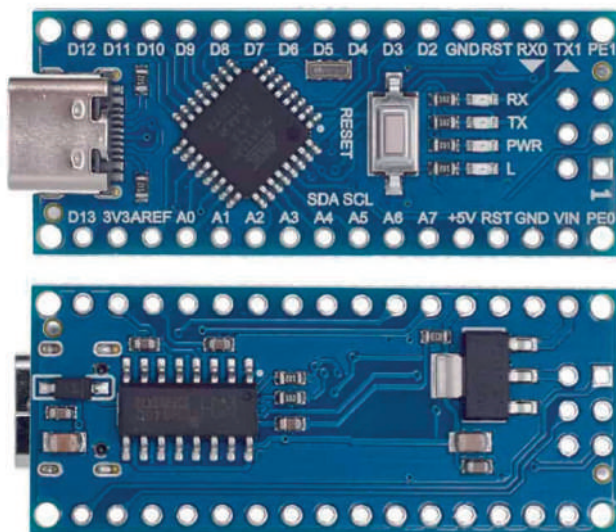


### საჭირო რესურსი

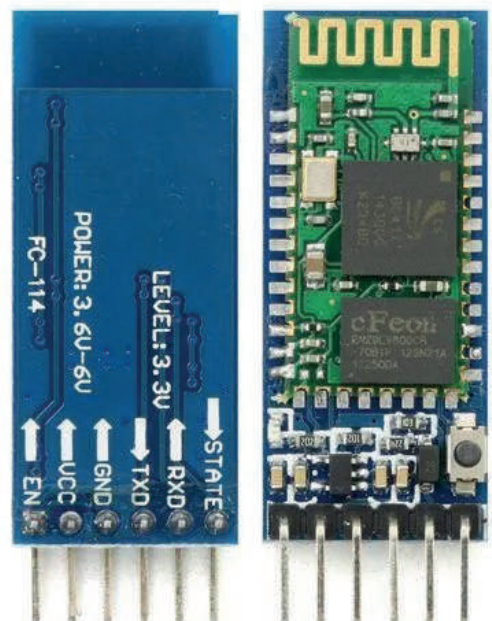
- Arduino Nano
- 2WD ტკვიანი მანქანა (კომპლექტი)
- სამაკეტო დაფა
- გამტარები
- Bluetooth HC-05 მოდული
- ძრავას დრაივერი L298N
- ელემენტები და ბუდე AA x 6
- Bluetooth კონტროლერი SriTu Hobby

### როგორ გუშაოვს ტკვიანი მანქანა 2WD

ამ პროექტში ჩვენ შევექმნით მანქანას, რომელიც იმართება ბლუთუზით და პირველად გამოვიყენებთ Arduino Nano-ს. Arduino Nano გამოყენებული კომპონენტების შედარებით მცირე ზომებით განსხვავდება Arduino Uno-სგან. შესაბამისად, დაფაც უფრო მცირე ზომისაა.



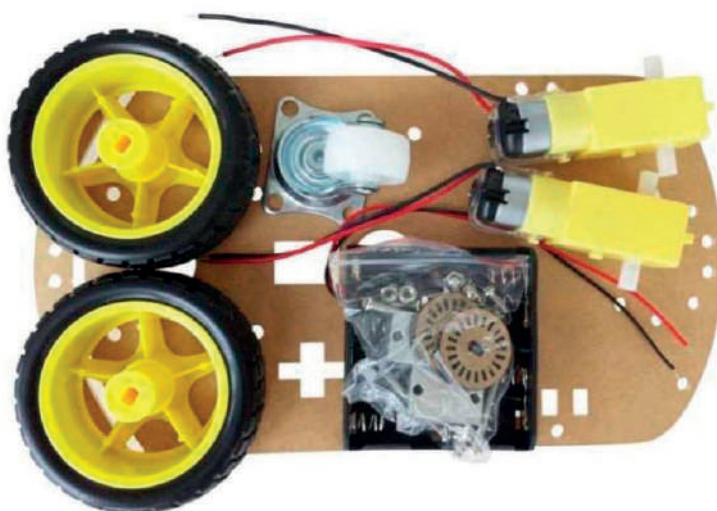
სურათი 1. Arduino Nano



სურათი 2. Bluetooth HC-05 მოდული

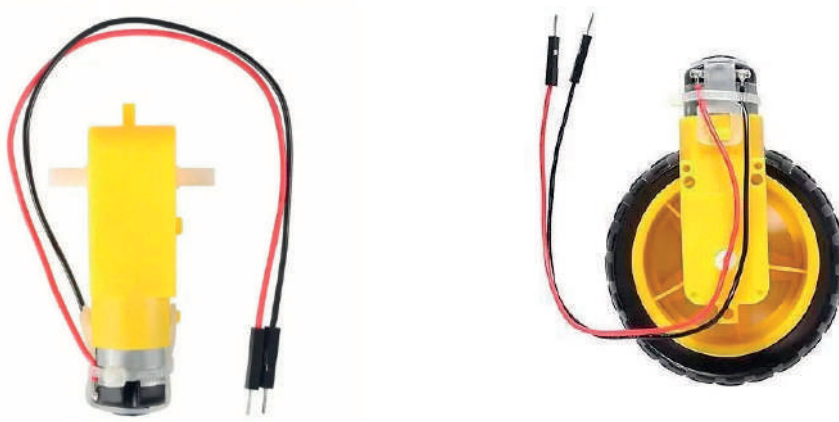
მიჰყევით ინსტრუქციას ნაბიჯ-ნაბიჯ:

1. თავდაპირველად, გახსენით მანქანის კომპლექტი და ყველა დეტალს გაეცანით ყურადღებით. აკრილის დეტალებიდან მოხსენით დამცავი ფირები.



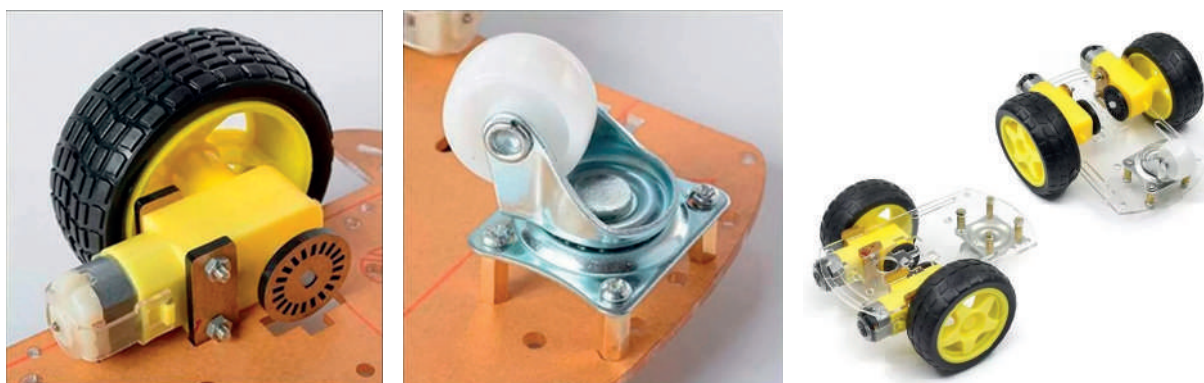
სურათი 3. 2WD მანქანის კომპლექტი

2. ძრავებს მირჩილვის წესით დაამაგრეთ გამტარები (იხ. სურათი 4).



სურათი 4.

3. დაამაგრეთ ძრავები და წინა ბორბალი ძირითად დაფაზე (იხ. სურათი 5)

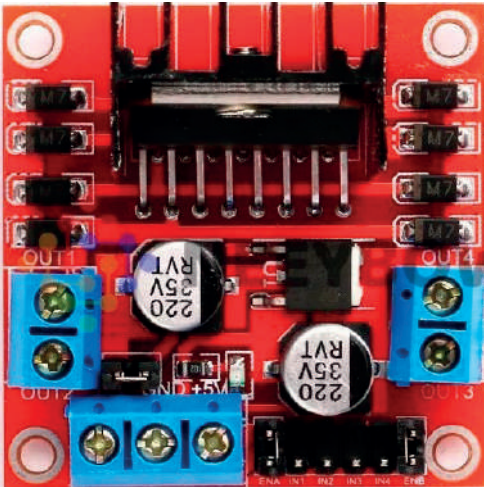


სურათი 5.

4. დაამაგრეთ ელემენტების ბუდე AA x 6 დაფაზე.
5. მოათავსეთ არდუინო ნანო და Bluetooth HC-05 მოდული სამაკეტო დაფაზე. დააკავშირეთ ერთ-მანეთთან: არდუინოს TX პინი – ბლუთუზის RX პინს და არდუინოს RX – ბლუთუზის TX პინს (ანუ, პირიქით).

| Arduino Nano | Bluetooth HC-05 |
|--------------|-----------------|
| TX           | RX              |
| RX           | TX              |

6. არდუინო და ძრავას დრაივერი დააკავშირეთ ერთმანეთთან: არდუინოს პინები D2-დან D5-მდე – დრაივერის პინებს IN1-დან IN4-მდე.

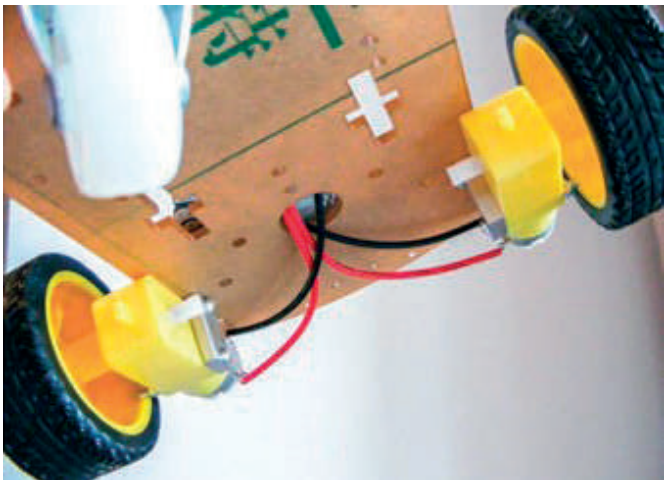


სურათი 6. ძრავას დრაივერი (მართავს რელუქტორულ ძრავებს, რომელთა საშუალებითაც ავტომობილი გადაადგილდება წინ და უკან, მოტრიალდება მარჯვნივ და მარცხნივ).

| Arduino Nano | ძრავას დრაივერი L298N |
|--------------|-----------------------|
| D2           | IN1                   |
| D3           | IN2                   |
| D4           | IN3                   |
| D5           | IN4                   |

7. დაამაგრეთ ძრავას დრაივერი დაფაზე. დააკავშირეთ ძრავები ძრავას დრაივერთან (იხ. სურათი 9 – სქემა და ქვემოთ მოცემული ცხრილი).

| ძრავები    | ძრავას დრაივერი L298N |
|------------|-----------------------|
| მარცხენა + | OUT1                  |
| მარცხენა - | OUT2                  |
| მარჯვენა + | OUT3                  |
| მარჯვენა - | OUT4                  |



სურათი 7. ძრავების გამტარების პოზიცია დრაივერთან დასაკავშირებლად

8. დააკავშირეთ ელემენტების ბუდე ძრავას დრაივერთან.

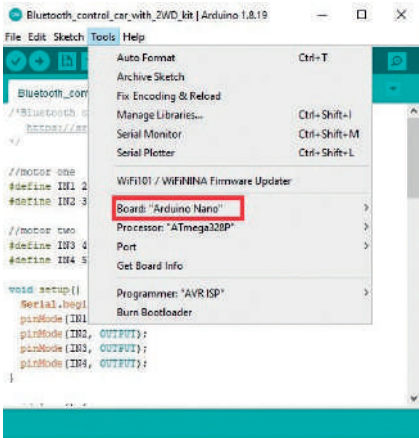
| ელემენტების ბუდე AAx6 | ძრავას დრაივერი L298N |
|-----------------------|-----------------------|
| წითელი გამტარი (+)    | +12                   |
| შავი გამტარი (-)      | სამაკეტო დაფის GND    |

9. დააკავშირეთ არდუინო, ძრავას დრაივერი და ბლუთუზი სამაკეტო დაფის კვების ლიანდაგებზე.

| Arduino Nano | სამაკეტო დაფა | ძრავას დრაივერი L298N | სამაკეტო დაფა | Bluetooth HC-05 | სამაკეტო დაფა |
|--------------|---------------|-----------------------|---------------|-----------------|---------------|
| VIN          | +             | +5V                   | +             | VCC             | +             |
| GND          | GND           | GND                   | GND           | GND             | GND           |

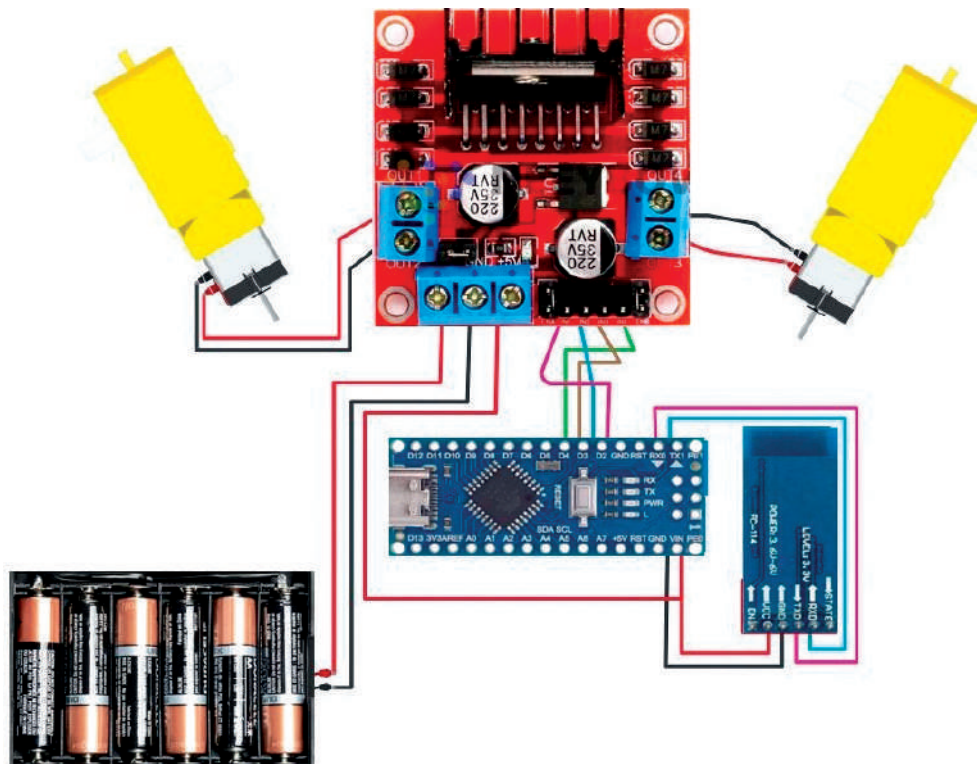
10. დააკავშირეთ არდუინო კომპიუტერთან. კოდის ჩატვირთვამდე Bluetooth მოდულის RX და TX საკონტაქტოები გათიშეთ (ამოიღეთ გამტარები დაფის საკონტაქტო ხვრელებიდან).

არ დაგავიწყდეთ, კოდის ჩატვირთვამდე არდუინოს IDE ინტერფეისის მენიუს ზოლში გახსენით Tools და მონიშნეთ პორტი და board-ზე – Arduino Nano (იხ. სურათი 8).



სურათი 8. არდუინოს IDE ინტერფეისში მონიშნეთ პორტი და ბორდზე Arduino Nano

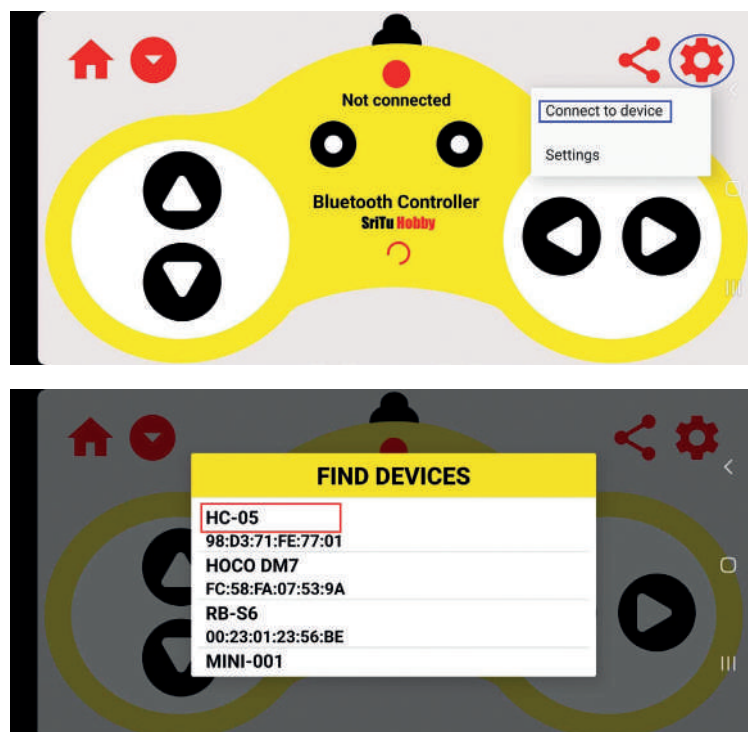
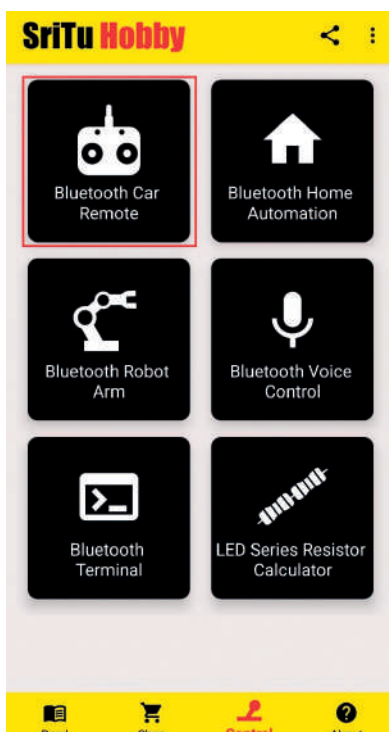
11. ჩატვირთეთ კოდი და კვლავ დააბრუნეთ გამტარები RX და TX კონტაქტებთან.

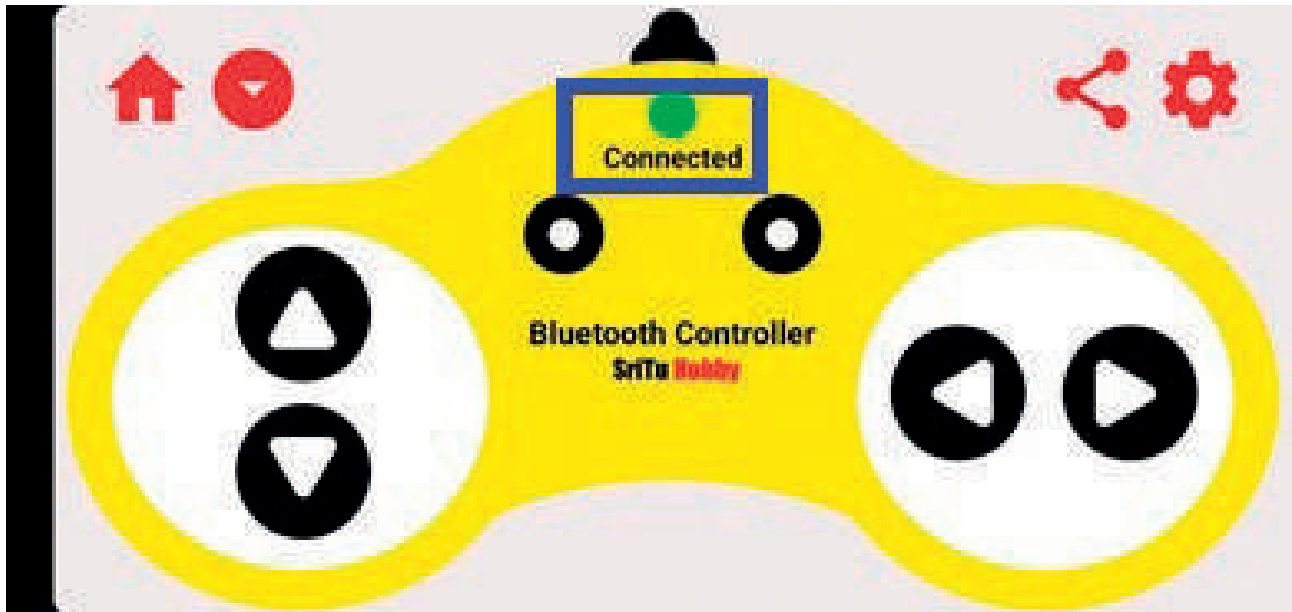


სურათი 9. სქემა

12. გამორთეთ არდუინო კომპიუტერიდან. ჩაალაგეთ ელემენტები ბუდეში.

13. Play Store-დან ჩამოტვირთეთ ტელეფონში აპლიკაცია SriTu Hobby. გახსენით აპლიკაცია, გადა-  
დით კონტროლერზე და აირჩიეთ ბლუთუზის დისტანციური მართვის პულტი. დააჭირეთ ღილაკს  
Connect to device და აირჩიეთ თქვენი ბლუთუზის სახელი HC-05. დისტანციური მართვის პულტზე  
დაინახავთ მწვანე ინდიკატორს (იხ. სურათი 10).





სურათი 10. SriTu Hobby აპლიკაცია და მისი დისტანციური მართვის პულტი

ახლა კი შეგიძლიათ მართოთ თქვენი ავტომობილი.

### გამოყენებული ლიტერატურა:

1. Mark Geddes – Arduino Project Handbook. 2016
2. ზ. ტაბატაძე, თ. თოდუა – არდუინო. პრაქტიკული სახელმძღვანელო დამწყებთათვის. თბილისი, 2019
3. ჯ. გრიგალაშვილი – Arduino-ს ვიზუალური დაპროგრამება FLProg გარემოში. საქართველოს ტექნიკური უნივერსიტეტი. თბილისი, 2015
4. <https://srituhobby.com/>