



არდუინო და ბლოკური რობოტიკა — წლიური სასწავლო სილაბუსი

აკადემია: ქლევერმაინდ აკადემია (CleverMind Academy)

სამიზნე ასაკი: 9–13 წელი

სწავლის დასაწყისი: 20 სექტემბერი

ინტენსივობა: კვირაში 2 შეხვედრა (თითო 1.5 საათი)

ჯამური ხანგრძლივობა: 32 კვირა / 64 გაკვეთილი / 96 საათი (128 აკადემიური საათი)

ინსტრუმენტები: Arduino Uno, Tinkercad Circuits (Block Editor), ელექტრონული კომპონენტები, სენსორები, ძრავები, რობოტის შასი.



კურსის შემაჯამებელი მონაცემები

მოდული	თემატიკა	ხანგრძლივობა	გაკვეთილები
მოდული 1	ელექტრონიკის საფუძვლები და პირველი ნაბიჯები Tinkercad-ში	8 კვირა	1 – 16
მოდული 2	ანალოგური სენსორები, მონიტორინგი და მექანიკა	8 კვირა	17 – 32
მოდული 3	მობილური რობოტიკა და ძრავები (Real Hardware Integration)	8 კვირა	33 – 48
მოდული 4	კომპლექსური პროექტები და გადასვლა ტექსტურ კოდზე	8 კვირა	49 – 64



მოდული 1: ელექტრონიკის საფუძვლები და პირველი ნაბიჯები Tinkercad-ში

ხანგრძლივობა: კვირა 1–8 (16 შეხვედრა / 24 საათი)

მიზანი: ბაზისური ფიზიკური ცნებების ათვისება, სიმულატორთან მუშაობა, ციფრული გამომყვანების (Digital Outputs) მართვა.

კვირა 1: შესავალი რობოტიკასა და ელექტრონიკაში

- **გაკვეთილი 1:** რა არის რობოტი? არდუინოს (Arduino Uno) აგებულება, პორტები და უსაფრთხოების წესები.
- **გაკვეთილი 2:** Tinkercad Circuits ინტერფეისი. პირველი წრედი: ბატარეა, რეზისტორი და შუქდიოდი (LED). დენის, ძაბვისა და წინააღმდეგობის მარტივი განმარტება.

კვირა 2: პირველი ბლოკური კოდი

- **გაკვეთილი 3:** შუქდიოდის მიერთება არდუინოზე. Built-in LED-ის ციმციმი (Set Built-in LED, Wait).
- **გაკვეთილი 4:** გარე LED-ის მართვა. ციფრული პინების (Digital Pins) HIGH/LOW მდგომარეობები.

კვირა 3: რიტმები და ლოგიკური თანმიმდევრობა

- **გაკვეთილი 5:** რამდენიმე შუქდიოდის მართვა (შუქნიშნის მოდელირება).
- **გაკვეთილი 6:** LED-ების ეფექტები (მაგ. "Knight Rider" / პოლიციის ციმციმა). ციკლების პირველი გაცნობა (Repeat).

კვირა 4: ხმა და ბაზერები (Piezo Buzzer)

- **გაკვეთილი 7:** პიეზო ბაზერის მუშაობის პრინციპი. სიხშირეები და ნოტების დაკვრა (Play tone).
- **გაკვეთილი 8:** მუსიკალური პროექტი: მარტივი მელოდიის აწყობა და LED ეფექტების სინქრონიზაცია.

კვირა 5: ციფრული შემყვანები (Digital Inputs)

- **გაკვეთილი 9:** ლილაკის (Pushbutton) პრინციპი. Pull-up / Pull-down რეზისტორები.
- **გაკვეთილი 10:** If...Else ბლოკის გაცნობა. ლილაკით შუქდიოდის ჩართვა/გამორთვა.

კვირა 6: ცვლადები და მდგომარეობები

- **გაკვეთილი 11:** რა არის ცვლადი (Variables)? ქულების/მდგომარეობის შენახვა.
- **გაკვეთილი 12:** ლილაკის დაჭერის დათვლა (Counter) და რეჟიმების გადართვა.

კვირა 7: ანალოგური სიგნალები (PWM)

- **გაკვეთილი 13:** განსხვავება Digital და Analog სიგნალებს შორის. PWM (Pulse Width Modulation) პინები.
- **გაკვეთილი 14:** RGB LED-ის მართვა — ფერების შექმნა და შერევა.

კვირა 8: მოდულური პროექტი #1

- გაკვეთილი 15: ქვევანი სახლის განათების და სიგნალიზაციის პროექტის დაგეგმვა.
- გაკვეთილი 16: პროექტის აწყობა, პრეზენტაცია და უკუკავშირი.



მოდული 2: ანალოგური სენსორები, მონიტორინგი და მექანიკა

ხანგრძლივობა: კვირა 9–16 (16 შეხვედრა / 24 საათი)

მიზანი: გარემოდან მონაცემების წაკითხვა, სერიული მონიტორი (Serial Monitor), სერვო ძრავები.

კვირა 9: სერიული კომუნიკაცია

- გაკვეთილი 17: Serial Monitor-ის გაცნობა (Print to serial monitor). მონაცემების მონიტორინგი.
- გაკვეთილი 18: პოტენციომეტრი — ანალოგური სიგნალის წაკითხვა (0-1023) და Map ბლოკის გამოყენება.

კვირა 10: სინათლის სენსორი (LDR)

- გაკვეთილი 19: ფოტორეზისტორის მუშაობის პრინციპი.
- გაკვეთილი 20: ქვევანი ქუჩის განათება (ლამის განათება ავტომატური ჩართვით).

კვირა 11: ტემპერატურის სენსორი (TMP36)

- გაკვეთილი 21: ტემპერატურის გაზომვა და მონაცემების გადაყვანა C°-ში.
- გაკვეთილი 22: თერმომეტრი ინდიკატორით (მწვანე/ყვითელი/წითელი LED-ები ტემპერატურის მიხედვით).

კვირა 12: სერვო ძრავი (Servo Motor)

- გაკვეთილი 23: Micro Servo (SG90) მუშაობის პრინციპი (0°–180° კუთხეები).
- გაკვეთილი 24: სერვო ძრავის მართვა პოტენციომეტრით და ლილაკებით (ავტომატური შლაგბაუმი).

კვირა 13: ულტრაბერითი მანძილზომი (HC-SR04)

- გაკვეთილი 25: ექოლოკაციის პრინციპი. ულტრაბერითი სენსორიდან მანძილის წაკითხვა.
- გაკვეთილი 26: მანქანის უკანა პარკინგ-სენსორის სისტემა (ხმოვანი და ვიზუალური სიგნალი).

კვირა 14: LCD ეკრანის ინტეგრაცია (16x2)

- გაკვეთილი 27: ტექსტისა და რიცხვების გამოტანა LCD ეკრანზე.
- გაკვეთილი 28: სენსორების მონაცემების LCD ეკრანზე გამოსახვა (მინისტაციები).

კვირა 15: რთული ლოგიკური ოპერატორები

- გაკვეთილი 29: AND, OR, NOT ოპერატორები ბლოკებში.
- გაკვეთილი 30: მრავალკომპონენტური უსაფრთხოების სისტემა (სინათლე + მანძილი + განგაში).

კვირა 16: მოდულური პროექტი #2

- გაკვეთილი 31: "ჭკვიანი სეიფი" ან "მეტეო-სადგური" (კოდური საკეტი/სერვო/LCD).
- გაკვეთილი 32: პროექტის პრეზენტაცია და შეფასება.

მოდული 3: მობილური რობოტიკა და ძრავები (Real Hardware Integration)

ხანგრძლივობა: კვირა 17–24 (16 შეხვედრა / 24 საათი)

მიზანი: მუდმივი დენის ძრავების (DC Motors) მართვა, რობოტის პლატფორმის აწყობა და ავტონომიური მოძრაობა.

კვირა 17: DC ძრავები და Motor Driver

- გაკვეთილი 33: DC ძრავის კვება და მართვა. H-Bridge (L293D ან L298N) პრინციპი.
- გაკვეთილი 34: ძრავის ბრუნვის მიმართულებისა და სიჩქარის (PWM) მართვა ბლოკებით.

კვირა 18: 2-თვლიანი რობოტის შასი (Robot Chassis Assembly)

- გაკვეთილი 35: რობოტის ფიზიკური აწყობა: შასი, ძრავები, თვლები, ელემენტების ბლოკი.
- გაკვეთილი 36: სადენების მენეჯმენტი და სქემის გადატანა Tinkercad-იდან რეალურ არდუინოზე.

კვირა 19: რობოტის საბაზისო მანევრები

- გაკვეთილი 37: მოძრაობის ფუნქციები: წინ, უკან, მარცხნივ, მარჯვნივ, გაჩერება.
- გაკვეთილი 38: ზუსტი მოძრაობები (მაგ. 90 გრადუსით მობრუნება, კვადრატის ტრაექტორიის გავლით მოძრაობა).

კვირა 20: დაბრკოლებების აცილება (Obstacle Avoidance)

- გაკვეთილი 39: ულტრაბგერითი სენსორის მონტაჟი რობოტის წინა ნაწილზე.
- გაკვეთილი 40: ავტონომიური რობოტი: კედელთან მიახლოებისას გაჩერება და გვერდის ავლით გზის გაგრძელება.

კვირა 21: ხაზზე მოსიარულე რობოტი (Line Follower) — ნაწილი 1

- გაკვეთილი 41: ინფრაწითელი (IR) ხაზის სენსორების მუშაობის პრინციპი (თეთრი და შავი ზედაპირი).
- გაკვეთილი 42: სენსორების კალიბრაცია და მონაცემების წაკითხვა.

კვირა 22: ხაზზე მოსიარულე რობოტი (Line Follower) — ნაწილი 2

- გაკვეთილი 43: ხაზის მიყოლის ლოგიკის აგება ბლოკებში.
- გაკვეთილი 44: ტრასის გავლის ტესტირება და ალგორითმის ოპტიმიზაცია.

კვირა 23: დისტანციური მართვა (IR / Bluetooth)

- **გაკვეთილი 45:** IR მიმღები და პულტი (ან Bluetooth მოდული HC-05).
- **გაკვეთილი 46:** რობოტის მართვა პულტით/ტელეფონით.

კვირა 24: მოდულური პროექტი #3 (Robo-Race)

- **გაკვეთილი 47:** რობოტების შეჯიბრი (დაბრკოლებების გადალახვა ან ხაზზე რბოლა).
- **გაკვეთილი 48:** შედეგების შეჯამება და ხარვეზების გასწორება.



მოდული 4: კომპლექსური პროექტები და გადასვლა ტექსტურ კოდზე

ხანგრძლივობა: კვირა 25–32 (16 შეხვედრა / 24 საათი)

მიზანი: რამდენიმე სენსორის გაერთიანება, Tinkercad-ის "Blocks + Text" რეჟიმის გაცნობა, ფინალური პროექტები.

კვირა 25: Blocks to C++ Transition (ხიდი ტექსტურ კოდთან)

- **გაკვეთილი 49:** Tinkercad-ში "Blocks + Text" რეჟიმის ჩართვა. ბლოკებისა და გვერდით გენერირებული C++ კოდის შედარება.
- **გაკვეთილი 50:** მარტივი ბლოკების შესაბამისი C++ სინტაქსის გააზრება (void setup(), void loop(), ,, {}).

კვირა 26: ჯოისტიკი და მრავალკომპონენტიანი მართვა

- **გაკვეთილი 51:** Analog Joystick-ის (X, Y ღერძები, ღილაკი) მიერთება.
- **გაკვეთილი 52:** 2 სერვო ძრავის (2-axis Manipulator / Robot Arm) მართვა ჯოისტიკით.

კვირა 27: ჭკვიანი ქალაქი / Smart Home იდეები

- **გაკვეთილი 53:** ავტომატური სანაგვე ყუთი (სერვო + ულტრაბგერა) ან ჭკვიანი ფანჯარა.
- **გაკვეთილი 54:** ჭკვიანი სათბურის მოდელები (ტემპერატურა, განათება, ვენტილაციონი).

კვირა 28: ფინალური პროექტის იდეების გენერირება და დაგეგმვა

- **გაკვეთილი 55:** გუნდების ჩამოყალიბება (2-3 მოსწავლე), იდეების ბრენშტორმინგი, ბიუჯეტირება და სქემის დახატვა.
- **გაკვეთილი 56:** პროექტის არქიტექტურის დამტკიცება და Tinkercad-ში პირველადი მოდელირება.

კვირა 29-31: ფინალურ პროექტებზე მუშაობა (Hackathon Style)

- **გაკვეთილები 57–62:** ფიზიკური კომპონენტების აწყობა და კორპუსების დამზადება, ბლოკური კოდის წერა, ტესტირება, გამართვა (Debugging), პრეზენტაციისათვის მზადება.

კვირა 32: წლიური გამოფენა და სერტიფიცირება

- **გაკვეთილი 63:** პროექტების საბოლოო ტესტირება.
- **გაკვეთილი 64:** Final Demo Day / STEM Exhibition — მშობლებისა და სტუმრების მონახულება, პროექტების

წარდგენა და სერტიფიკატების გადაცემა.

სწავლის შედეგები (Learning Outcomes)

- **ტექნიკური უნარები:** ელექტრონიკის საფუძვლები (ძაბვა, დენი, წინააღმდეგობა), სენსორების მართვა, ბლოკური და C++ სინტაქსის გააზრება, ავტონომიური და დისტანციურად მართვადი რობოტების შექმნა.
- **Soft Skills:** ალგორითმული და ინჟინრული აზროვნება, პრობლემების აღმოფხვრა (Debugging), გუნდური მუშაობა და პროექტების პრეზენტაცია.