Programiranje fizičkih objekata

Katanić, Eni

Master's thesis / Diplomski rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:049246

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-07-03**

Repository / Repozitorij:

Repository of the Faculty of Science - University of Zagreb

SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Eni Katanić

PROGRAMIRANJE FIZIČKIH OBJEKATA

Diplomski rad

Voditelj rada:
doc. dr. sc. Goranka Nogo

Zagreb, 2019.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____ , predsjednik
2. _____ , član
3. _____ , član

Povjerenstvo je rad ocijenilo ocjenom _____ .

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Sadržaj

Programski jezici Logo i Scratch 2
1.1 Logo 2
1.2 Scratch 3
Pločica Pico 5
2.1 Općenito o pločici Pico 5
2.1 Primjeri aktivnosti u školi 6
2.1.1 Aktivnost 1: „Laku noć!“ 6
2.1.2 Aktivnost 2: „Tko je brži?“ 7
Lego WeDo 9
3.1 Općenito o paketu Lego WeDo 9
Pločica Arduino 12
4.1 Općenito o pločici Arduino 12
4.2 Primjeri aktivnosti u školi 15
4.2.1 Aktivnost 1: Prvi program 15
4.2.2 Aktivnost 2: Vlaga u tlu 16
4.2.3 Aktivnost 3: „Svirajmo!“ 20
Pločica Orion 24
5.1 Općenito o pločici Orion 24
5.2 Primjeri aktivnosti u školi 25
5.2.1 Aktivnost 1: Prvi program 25
5.2.2 Aktivnost 2: Temperatura vode 26
5.2.3 Aktivnost 3: Tišina u razredu 29
mBot 34
6.1 Općenito o mBotu 34
6.2 Primjeri aktivnosti u školi 36
6.2.1 Aktivnost 1: Prvi program 36
6.2.2 Aktivnost 2: „Party“ 38
Micro:bit 40

7.1 Općenito o micro:bitu 40
7.2 Primjeri aktivnosti u školi 41
7.2.1 Aktivnost 1: Upoznavanje s micro:bitom 41
7.2.2 Aktivnost 2: Tajno komuniciranje 43
7.2.3 Aktivnost 3: Brojač koraka 45
Prednosti i nedostatci programiranja nabrojenih fizičkih objekata 46
8.1 Pločica Pico 46
8.2 Edukacijski paket Lego WeDo 46
8.3 Pločica Arduino 47
8.4 Pločica Orion 48
8.5 Edukacijski robot mBot 49
8.6 Micro:bit 49
Usporedba s kurikulumom hrvatskih i američkih škola 51
9.1 Kurikulum nastavnog predmeta Informatika za osnovne i srednje škole 51
9.3 Američki nacionalni kurikulum CSTA K-12 55
Zaključak 58
Literatura 59
Sažetak 60
Summary 61
Životopis 62

Uvod

U današnje vrijeme programiranje i algoritamsko razmišljanje samo su neke od kompetencija koje se svakim danom sve više traže. Razumijevanje koncepta programiranja bi, uz znanje korištenja računalom, trebalo biti dio opće pismenosti i za osobe koje se kasnije ne namjeravaju profesionalno baviti time budući da gotovo svi predmeti s kojima se susrećemo u svakodnevnom životu u sebi imaju ugrađen neki elektronički sklop koji izvodi određene operacije. Cilj samog programiranja nije stvoriti buduće programere nego učenike poučavati algoritamskom načinu razmišljanju i pripremati ih za poslove budućnosti. U ovom radu prikazana je mogućnost osuvremenjivanja nastave programiranja korištenjem suvremenih i dostupnih ulaznih i izlaznih jedinica. Time se omogućuje temeljno razumijevanje načina rada uređaja koji imaju neki oblik programiranog djelovanja.

Diplomski rad sastoji se od 9 poglavlja. U prvom poglavlju ukratko su opisana dva programska jezika, Logo i Scratch i dano je nekoliko primjera programiranja u tim programskim jezicima. U drugom poglavlju opisana je pločica Pico i dano je nekoliko primjera aktivnosti koje se s učenicima mogu provesti pri programiranju ove pločice. U trećem poglavlju opisan je edukacijski paket Lego WeDo i način na koji se Ledo WeDo programira. U četvrtom i petom poglavlju dani su kratki opisi pločica Arduino i Orion i na koji način se te pločice programiraju. Dano je i nekoliko primjera aktivnosti za učenike koje se mogu provoditi u školama. U šestom poglavlju dan je opis edukacijskog robota mBot i primjeri aktivnosti koje se mogu s učenicima provesti u školama. U sedmom poglavlju opisan je micro:bit, mikroračunalo s kojim se većina učenika hrvatskih osnovnih škola susrela. Dano je i nekoliko aktivnosti koje se mogu provesti u školama. U osmom poglavlju navedene su prednosti i nedostatci programiranja navedenih fizičkih objekata. Opisani su i potencijalni problemi vezani uz programiranje fizičkih objekata. U devetom poglavlju promatra se kurikulum predmeta Informatika u hrvatskim i američkim školama i koji se ishodi mogu ostvariti programiranjem fizičkih objekata. Na samom kraju rada dan je kratki zaključak o programiranju fizičkih objekata i kakav utjecaj ima na učenike.

Poglavlje 1

Programski jezici Logo i Scratch

Prije negoli prijeđemo na programiranje fizičkih objekata, osvrnut ćemo se na dva programska jezika, Logo i Scratch.

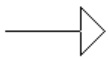
Učenici se s programskim jezikom Logo upoznaju u petom razredu osnovne škole. Za razliku od Loga, u odobrenim udžbenicima iz informatike za osnovnu školu ne obrađuje se programski jezik Scratch. Ipak, učenici se upoznaju s grafičkim editorom koji je sličan grafičkom editoru Scratcha. Riječ je o grafičkom editoru BlokPy kojim se ostvaruje programiranje blokovima u programskom jeziku Python.

1.1 Logo

Među prvim programskim jezicima s kojima se učenici susreću u svom obrazovanju je programski jezik Logo. Osnovna ideja ovog programskog jezika je učenicima pružiti mogućnost crtanja raznih crteža tako što zadaju naredbe kornjači, virtualnom robotu. Od petog razreda osnovne škole pa na dalje, učenici u programskom jeziku Logo uče kako pomoću naredbi pomaknuti kornjaču, odnosno kako se kretati, crtati, itd.

Slijedi jednostavni primjer programa u programskom jeziku Logo.

Primjer 1. Pomakni kornjaču iz početnog položaja 50 koraka u desno.



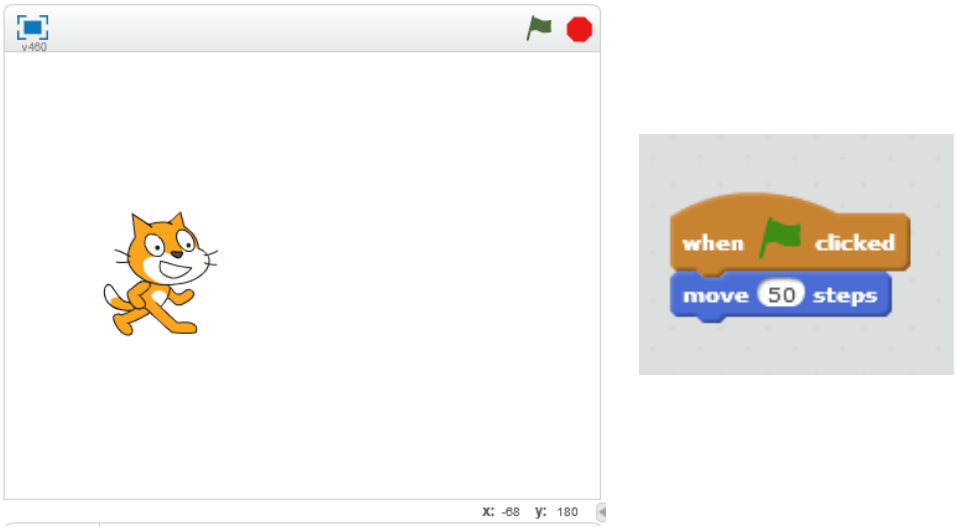
```
rt 90
fd 50
```

1.2 Scratch

Nekoliko desetljeća nakon Loga, pojavljuje se novi programski jezik – Scratch. Ovaj programski jezik nastao je po uzoru na Logo, no od Loga se razlikuje po tome što je prilagođen djeci već i od predškolske dobi. Sintaksu Loga zamjenjuje grafičko sučelje gdje učenici povezuju blokove, poput lego kockica. Različiti tipovi podataka i različite naredbe prikazani su različitim oblicima blokova. Ovakav pristup programiranju onemogućuje sintaktičke greške učenika pri pisanju programa i usmjerava pažnju učenika na problem koji žele riješiti. Programirajući u ovom programskom jeziku učenici mogu lako s ostalim učenicima podijeliti svoje programe, projekte, itd.

Slijedi jednostavni primjer programa u programskom jeziku Scratch.

Primjer 2. Pomakni mačka iz početnog položaja 50 koraka u desno.



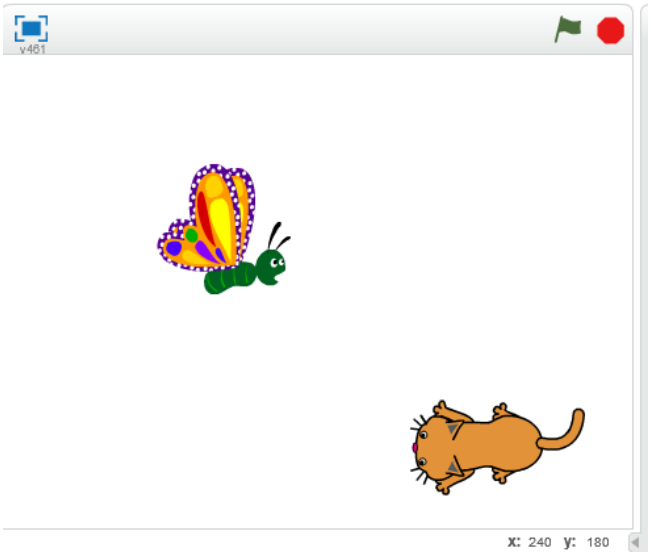
Programiranjem u Scratchu učitelji mogu za učenike napraviti vrlo blag prijelaz iz programiranja virtualnih objekata u programiranje stvarnih, fizičkih objekata.

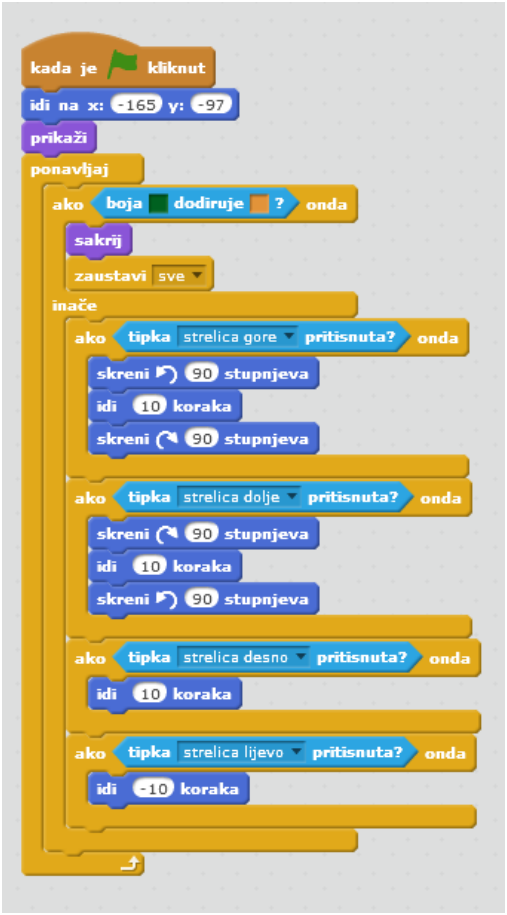
Fizički objekti prepuni su različitih senzora. Najprije učenici trebaju shvatiti što je senzor i na koji način funkcionira. Danas učenici mogu pronaći mnoštvo senzora u svijetu oko sebe. Postoje razni senzori; senzori u mobilnim uređajima, senzori na kliznim vratima koja se otvaraju kada se osoba približi, zatim senzori za parkiranje, igraće konzole kod kojih se upravljanje ostvaruje pomoću različitih ugrađenih senzora, itd.

Senzori su uređaji koji mjere neku fizikalnu veličinu i pretvaraju je u signal koji je pogodan za daljnju obradu, tj. kao izlaz daju digitalno stanje – uključeno ili isključeno. Uz pomoć Scratcha učenici mogu bolje shvatiti što su to senzori i kako se s njima radi. Jedan od blokova u programu Scratch zove se Očitanja (*Sensing*) koji nam omogućuje upravo to, prikupljanje informacija iz okruženja.

Kao uvod u programiranje fizičkih objekata može poslužiti sljedeći primjer programa u Scratchu koji koristi blok Očitanja.

Primjer 3. U novim oblicima odaberi leptira. Neka se leptir može pomicati gore-dolje, lijevo-desno. Negdje na pozornici nalazi se i mačka. Kada leptir dođe do mačke, mačka pojede leptira i leptir nestaje.





Poglavlje 2

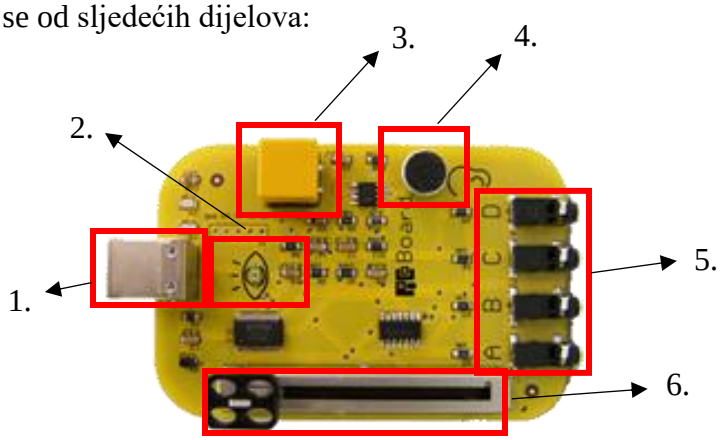
Pločica Pico

Pomoću programskog jezika Scratch učenici mogu programirati pločicu Pico. U ovom poglavlju ukratko je opisana pločica Pico i navedeni su primjeri aktivnosti u kojima se programira pločica Pico.

2.1 Općenito o pločici Pico

Programiranjem pomoću Scratcha moguće je programirati npr. pločicu Pico i Lego WeDo paket.

Pločica Pico sastoji se od sljedećih dijelova:



Slika 1: Pločica Pico (izvor: https://www.picocricket.com)

1. USB konektor pomoću kojeg se pločica spaja s računalom
2. Senzor svjetlosti
3. Dugme
4. Senzor zvuka
5. 4 ulaza (A, B, C i D) kojima se pomoću krokodil štipaljki na pločicu mogu spojiti dodatni senzori
6. Klizač.

2.1 Primjeri aktivnosti u školi

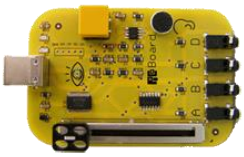
U nastavku je navedeno nekoliko aktivnosti u kojima se programira pločica Pico.

Da bi nam bili ponuđeni blokovi s naredbama za programiranje pločice Pico u Scratchu odaberemo opciju Više blokova (*More Blocks*), Dodaj proširenja (*Add an Extension*) te odaberemo pločicu Pico (*PicoBoard*).

2.1.1 Aktivnost 1: „Laku noć!“

Aktivnost 1 primjerena je za učenike viših razreda osnovne škole.

Potrebni materijali za ovu aktivnost su:

- pločica Pico 
- USB kabel.

Korak 1. Spoji pločicu Pico s računalom pomoću USB kabla.

Zadatak 1. Napravi program u kojem će se ispitivati jačina svjetlosti. Kad padne mrak, odnosno kada je količina svjetlosti mala, mačak postaje pospan, ide spavati i kaže „Laku noć!“. Pri tome se promjeni i pozadina u crno.

Rješenje:

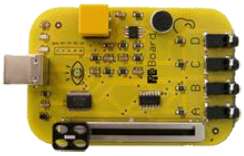
U ovom primjeru učenici mogu sami, prekrivajući senzor svjetlosti, otkrivati kolika je količina svjetlosti kad padne mrak.

Slijedi nešto kompliciraniji primjer programiranja pločice Pico u kojem se, osim senzora svjetlosti koriste i ostali senzori na pločici.

2.1.2 Aktivnost 2: „Tko je brži?“

Aktivnost 2 primjerena je učenicima viših razreda osnovne škole.

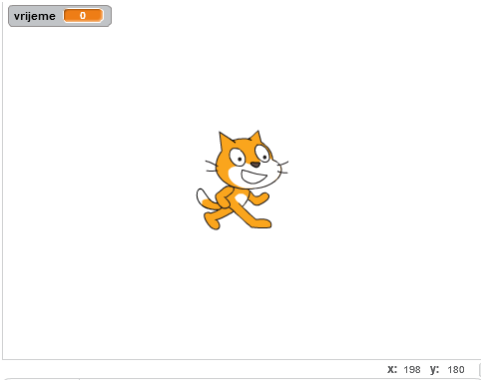
Potrebni materijali za ovu aktivnost su:

- pločica Pico 
- USB kabel.

Zadatak 2. Zaigraj igru sa svojim parom iz klupe. Kada čujete da mačka mijauka, pritisnite dugme na pločici Pico. Napravite program koji mjeri brzinu vaše reakcije i otkrijte tko brže reagira.

a) Tko brže reagira?
b) Pokušajte igru odigrati s drugom rukom. Jeste li reagirali brže?
c) Da bi bilo zanimljivije, pokušajte igru odigrati žmireći.

Rješenje:



Poglavlje 3

Lego WeDo

Ono čime je većina učenika osnovnoškolske, a i srednjoškolske dobi upoznata su lego kocke. Većina učenika je od malih novu slagala zanimljive figurice koristeći lego kocke. Kao spoj lego kockica i programiranja nastao je edukacijski paket Lego WeDo.

U ovom poglavlju ukratko je opisan edukacijski paket Lego WeDo.

3.1 Općenito o paketu Lego WeDo

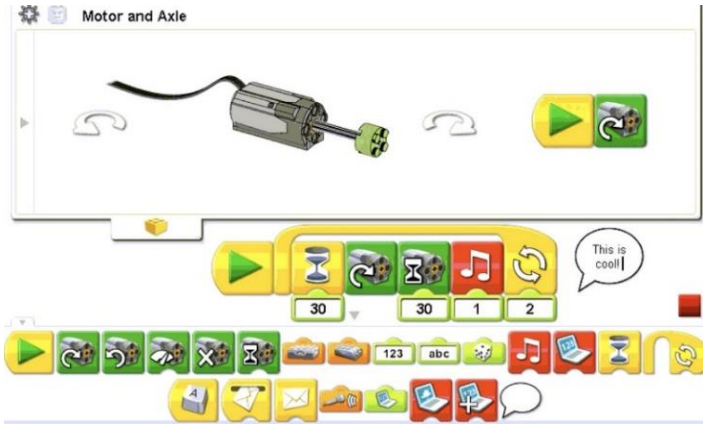
Originalni paket Lego WeDo nastao je 2009. godine, a poboljšana verzija, LegoWeDo 2.0, 2016. godine. Lego WeDo 2.0 je edukacijski paket koji se sastoji od čak 280 dijelova. Osim figurica različitih oblika i boja, paket sadržava i motor, senzor nagiba, senzor pokreta i središnju jedinicu.

Slika 2: Lego WeDo paket (izvor: https://education.lego.com)

Paket je primjeren za učenike osnovnih škola, a uz neke dodatke postaje primjeren i za učenike srednjih škola.

Programi se mogu kreirati pomoću programa Scratch ili softvera WeDo koji se, kao i Scratch, zasniva na „Povuci i pusti“ (Drag and drop) funkcionalnosti. Softver WeDo sadrži i gotove

programe za programiranje paketa Lego WeDo, koji uz to učenicima daje detaljne upute kako će složiti objekt koji se u programu koristi.



Slika 3: Program u softveru WeDo (izvor: https://education.lego.com)

Učenici prije samog programiranja mogu isprobati gotove programe i istraživati mogućnosti paketa Lego WeDo, a onda i sami složiti svoj objekt i kreirati svoj program. Učenici mogu kreirati npr. program u kojem se pomicanjem objekta sprječava poplava, kamion koji odvaja smeće, balerina koja pleše i slično.

Jedan od dodataka kojima ovaj paket postaje primjeren i za učenike srednjih škola je paket Lego Mindstorms.

Slika 4: paket Lego Mindstorms (izvor: https://education.lego.com)

Lego Mindstorms je paket koji se sastoji od različitih senzora, motora i ostalih dijelova pomoću kojih učenici mogu slagati robote. Učenici tako mogu napraviti npr. robota koji uz pomoć senzora može razlikovati osam različitih boja, robota kojim se upravlja daljinskim upravljačem itd.

Neki od robota koji se mogu složiti koristeći ovaj paket su:

Slika 5: Robot TRACK3R (izvor: https://education.lego.com)

Slika 6: Robot GRIPP3R (izvor: https://education.lego.com)

Slika 7: Robot 3RPTAR (izvor: https://education.lego.com)

U poglavlju nisu navedene aktivnosti koje se mogu provoditi koristeći paket Lego WeDo. Zbog same cijene paketa i mnoštva malih, sitnih dijelova, paket nije primjeren za provođenje aktivnosti u školi.

Poglavlje 4

Pločica Arduino

U ovom poglavlju opisana je pločica Arduino i navedeni su primjeri aktivnosti u kojima se programira pločica Arduino i koje se mogu provoditi u školama.

4.1 Općenito o pločici Arduino

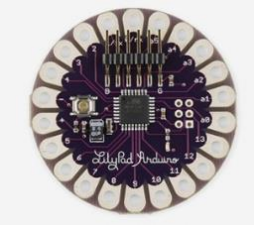
Nešto veći izazov za učenike može biti programiranje pločice Arduino.

Arduino je elektronička platforma namijenjena kreiranju raznih elektroničkih projekata. Pločicu su 2005. godine kreirali talijanski studenti. Arduino se sastoji od dva dijela, fizičkog dijela odnosno same pločice na kojoj se nalazi mikrokontroler i softverskog dijela koji se naziva Arduino IDE (*Integrated Development Environment*) kojim se ostvaruje programiranje pločice.
Cijeli sustav je otvorenog koda (*Open Source*) što znači da je programski kod svima dostupan i da ga mogu slobodno izmjenjivati te stvoriti svoju inačicu pločice Arduino s različitim funkcionalnostima. Tako su nastale različite vrste Arduino pločica pa tako i hrvatska inačica Croduino.

Najpoznatije pločice Arduino su:

- Arduino Uno
- LilyPad Arduino
- RedBoard
- Arduino Mega
- Arduino Leonardo.

Slika 8: Arduino Uno
(izvor: http://www.arduino.cc)



Slika 9: Arduino LilyPad
(izvor: http://www.arduino.cc)

Slika 10: RedBoard
(izvor: http://www.arduino.cc)

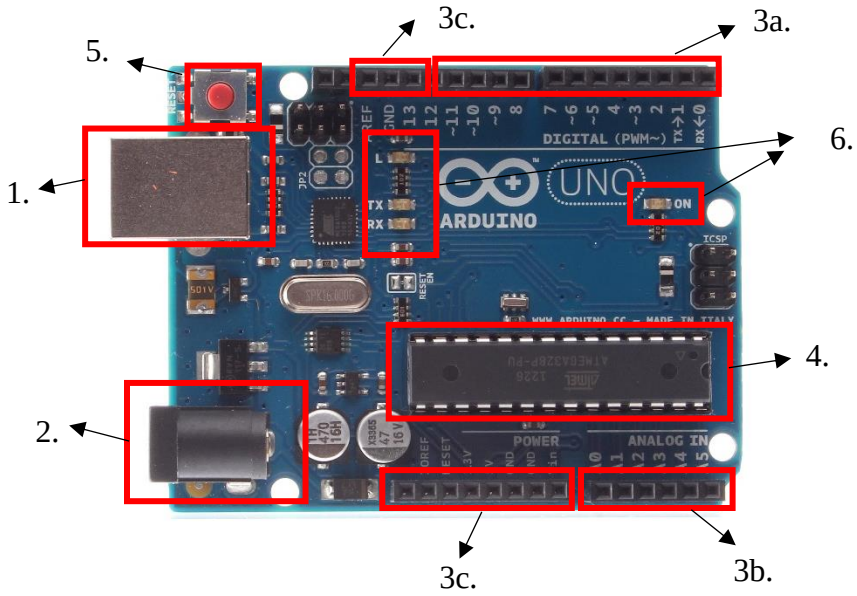
Slika 11: Arduino Mega
(izvor: http://www.arduino.cc/)

Slika 12: Arduino Leonardo
(izvor: http://www.arduino.cc)

Slika 11: Croduino
(izvor: http://e-radionica.com.hr)

Na primjeru pločice Arduino Uno navedeno je od čega se sastoji jedna pločica, a sve nabrojane dijelove možemo lako uočiti i na ostalim pločicama.

Pločica Arduino sastoji se od sljedećih dijelova:



Slika 12: Pločica Arduino Uno (izvor: http://www.arduino.cc)

1. USB konektor pomoću kojeg se pločica spaja s računalom.

2. Utor za napajanje električnom strujom (*Barrel Jack*). Napajanje preko 20 Volti može uništiti Arduino. Preporučuje se napajanje između 6 i 12 Volti.

3. Izvodi (*Pins*) pomoću kojih se senzori i ostale komponente povezuju s pločicom. Svaki izvod ima svoj broj koji se lako može očitati s pločice.

Razlikujemo analogne i digitalne izvode. Digitalni izvodi (1a.) služe za spajanje komponenti kao što su npr. tipke koje signaliziraju je li tipka stisnuta ili nije. Analogni izvodi (1.b) služe za spajanje komponenti kao što su npr. svjetiljka koja može svijetliti punom jačinom, slabijom jačinom itd.

4. Mikrokontroler (*Microcontroller*) koji je „mozak“ cijele pločice upravlja svim procesima koji se na njoj događaju. Mikrokontroler prati što se događa s izvodima i očitava signale koji dolaze na njih. Također, na njega se sprema kod s računala.

5. Dugme za resetiranje (*Reset Button*) koje programski kod pohranjen na pločici izvodi iznova.

6. Pločica Arduino ima 4 lampice (raspored lampica može biti drukčiji na različitim inačicama pločica). Razlikujemo lampice L, TX, RX i ON (PWR). Lampicu L sami možemo kontrolirati i nema posebnu funkciju. Lampica TX kada je uključena signalizira prijenos podataka s Arduina na računalo. Lampica RX pak signalizira prijenos podataka s računala na Arduino. Lampica ON (PWR) je uključena onda kada Arduino ima izvor napajanja.

Programiranje Arduina omogućeno je, između ostalog, programom IDE. Programira se u programskom jeziku C++. Kako je cijeli sustav otvorenog koda, postoji veliki broj biblioteka koje se lako mogu instalirati na računalo i koristiti. Biblioteke se sastoje od velikog broja funkcija koje omogućavaju korisnicima lakše programiranje pločice. Umjesto pisanja koda koji se sastoji od puno linija, mogu se koristiti gotove funkcije.

Arduino programski kod sastoji se od:

- *setup()* dijela u kojem se inicijaliziraju varijable, uključuju biblioteke, itd.
- *loop()* dijela u kojem se zapisuje kod koji se iznova ponavlja.

Pločica Arduino može se programirati i pomoću programa MBlock koji će detaljnije biti opisan kasnije. Sljedeći programi bit će pisani u programu IDE, a ako učenici nisu upoznati s programskim jezikom C++ programe mogu pokušati napisati u programu MBlock.

Ako učenici za kreiranje programa koriste MBlock, onda su aktivnosti primjerene i za učenike viših razreda osnovne škole.

4.2 Primjeri aktivnosti u školi

4.2.1 Aktivnost 1: Prvi program

Aktivnost 1 primjerena je učenicima srednje škole.

Potrebni materijali za ovu aktivnost su:

- pločica Arduino
- USB kabel.

Korak 1. Pomoću USB kabela spoji pločicu Arduino s računalom.

Zadatak 1. Napiši program tako da se lampica L na Arduino uključuje svakih 5 sekundi.

Rješenje:

```
void setup() {
  pinMode(LED_BUILTIN, OUTPUT); // inicijaliziranje lampice
}

void loop() {
  digitalWrite(LED_BUILTIN, HIGH);// lampica se uključi
  delay(1000);  // lampica svijetli 1 sekundu (1 sekunda = 1000 jedinica)
  digitalWrite(LED_BUILTIN, LOW); // lampice se isključi
  delay(5000);                    // sačekaj 5 sekundi prije nego li se lampica ponovno uključi
}
```

Naredba **pinMode** postavlja izvode kao ulazne (*input*) ili kao ulazne (*output*). U ovom slučaju, ugrađenu lampicu L na Arduinu (LED_BUILTIN) postavili smo kao izlazni izvod.

Naredba **digitalWrite** uključuje ili isključuje digitalni izvod. Naredbom **digitalWrite(pin, LOW)** izvod se uključuje, a naredbom **digitalWrite(pin, HIGH)** izvod se isključuje. Osim **digitalWrite** postoji i naredba **digitalRead** koja omogućava čitanje vrijednosti koje digitalni izvod šalje. Naredbe koje šalju i primaju signale od analognih izvoda su **analogWrite** i

analogRead. Naredbom **delay** ostvaruje se pauza. Primjerice, **delay(1000)** pauzirat će program na 1 sekundu.

Zadatak 2. Istraži kako se šalju signali Morseovom abecedom. Pronađi signal za SOS. Napiši program koji pomoću Arduina šalje signal SOS.

Rješenje:

Morseov kod za SOS: **. . . - - - . . .**

```
void setup() {
  pinMode(LED_BUILTIN, OUTPUT); // inicijaliziranje lampice
}

void loop() {
  for (int i=1; i <=3; i++){
  digitalWrite(LED_BUILTIN, HIGH);// lampica se uključi
  delay(500);  // lampica svijetli 1 sekundu (1 sekunda = 1000 jedinica)
  digitalWrite(LED_BUILTIN, LOW); // lampice se isključi
  delay(500);                      // sačekaj 5 sekundi prije nego li se lampica ponovno uključi
                           }    // dio koda koji se odnosi na slovo S

  for (int i=1; i <=3; i++){
  digitalWrite(LED_BUILTIN, HIGH);// lampica se uključi
  delay(1500);  // lampica svijetli 1 sekundu (1 sekunda = 1000 jedinica)
  digitalWrite(LED_BUILTIN, LOW); // lampice se isključi
  delay(500);                      // sačekaj 5 sekundi prije nego li se lampica ponovno uključi
                           }    // dio koda koji se odnosi na slovo O

  for (int i=1; i <=3; i++){
  digitalWrite(LED_BUILTIN, HIGH);// lampica se uključi
  delay(500);  // lampica svijetli 1 sekundu (1 sekunda = 1000 jedinica)
  digitalWrite(LED_BUILTIN, LOW); // lampice se isključi
  delay(500);                      // sačekaj 5 sekundi prije nego li se lampica ponovno uključi
                           }    // dio koda koji se odnosi na slovo S

  delay(5000);

}
```

4.2.2 Aktivnost 2: Vlaga u tlu

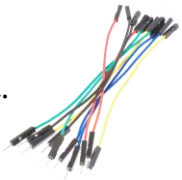
Aktivnost 2 primjerena je učenicima srednje škole.

Potrebni materijali za ovu aktivnost su:

- pločica Arduino

- USB kabel

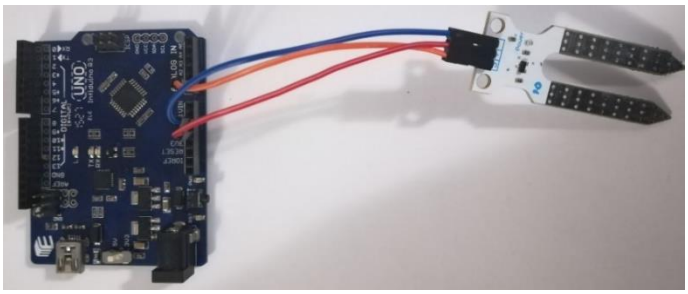
- senzor za vlagu u tlu

- 3 spojne žice s muško-ženskim kontaktima. 

Korak 1. Pomoću spojnih žica spoji senzor vlage u tlu s pločicom Arduino tako da spojiš signal S s izvodom A0 na pločici, signal + s izvodom 5V i signal – s izvodom GND.

Zadatak 1. Napiši program koji će pomoću senzora vlage mjeriti vlažnost tla. Prateći vrijednosti koje senzor daje otkrij kada je tlo suho, kada vlažno i koje vrijednosti senzor daje kada je u vodi.

Rješenje:



```
const int senzor_vlage = A0;  // pin na kojeg je spojen senzor
int vrijednost = 0;         // inicijalizacija varijable


void setup() {
  Serial.begin(9600); // prijenos 9600 bitova u sekundi (možemo promjeniti i u neke od vrijednosti:
                      //300, 600, 1200, 2400, 4800, 14400, 19200, 28800, ...)
}

void loop() {
  vrijednost = analogRead(senzor_vlage);   // senzor vlage tla

  Serial.print("Vlažnost = " );   // ispisi vrijednosti na ekran
  Serial.println(vrijednost);

  delay(1000);      // sačekaj 1 sekundu prije sljedećeg očitanja
}
```

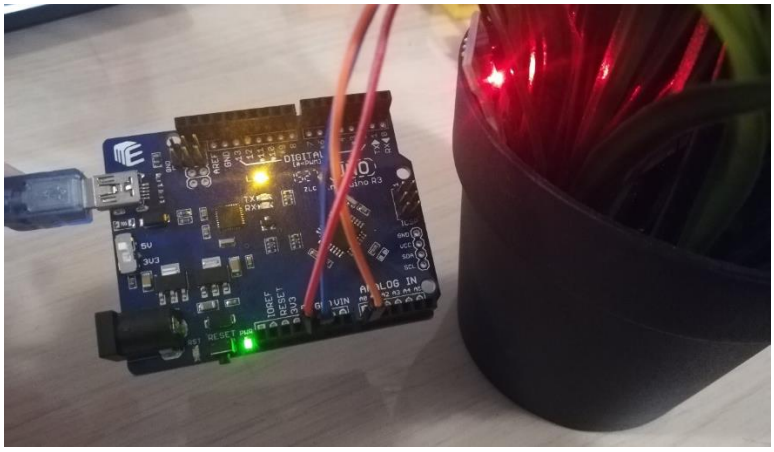
Naredbom **Serial.begin** ostvaruje se serijska komunikacija između računala i pločice Arduino. Broj koji slijedi nakon te naredbe označava koliki je prijenos bitova u sekundi. Naredbe **Serial.print** i **Serial.println** ispisuju podatke na serijskom monitoru (*Serial Monitor*) do kojeg možemo doći iz izbornika Alati (*Tools*). Razlika između tih dviju naredbi je što naredba **Serial.println** dodaje prazni red nakon linije.

Stavljajući senzor u vodu primjećujemo da senzor daje vrijednosti od otprilike 650 do 750. Kada ga stavimo u suho tlo, vrijednosti su od 0 do 200. U vlažnom tlu senzor daje vrijednosti od 200 do 650.

Zadatak 2. Napiši program koji će signalizirati treba li zaliti cvijet. Ako je tlo u kojem je posađen cvijet presuho, onda će se naizmjenično uključivati i isključivati lampica na Arduinu i time signalizirati da se cvijet treba zaliti. Ako je pak tlo dovoljno vlažno, lampica će biti isključena što znači da cvijet ne treba zalijevati.

Rješenje:

```
const int senzor_vlage = A0;  // pin na kojeg je spojen senzor
int vrijednost = 0;          // inicijalizacija varijable


void setup() {
  Serial.begin(9600); // prijenos 9600 bitova u sekundi (možemo promjeniti i u neke od vrijednosti:
                      //300, 600, 1200, 2400, 4800, 14400, 19200, 28800, ...)

  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
  vrijednost = analogRead(senzor_vlage);   // senzor vlage tla

 if ((vrijednost >= 0) && (vrijednost <= 200))
 {
 digitalWrite(LED_BUILTIN, HIGH);
 delay(250);
 digitalWrite(LED_BUILTIN, LOW);
 delay(250);

 }
}
```



4.2.3 Aktivnost 3: „Svirajmo!“

Aktivnost 3 primjerena je učenicima srednje škole.

Potrebni materijali za ovu aktivnost su:

- pločica Arduino
- USB kabel
- zujalica
- 3 spojne žice s muškim kontaktima.

Korak 1. Pomoću USB kabela spoji pločicu Arduino s računalom. Spoji zujalicu s pločicom Arduino tako da signal S na zujalici spojiš na izvod GND na pločici Arduino. Nadalje, signal + spoji s izvodom 5V, a signal – s izvodom 3.

Zadatak 1. Napiši program kojim će zujalica proizvesti zvuk određene frekvencije.

Rješenje:

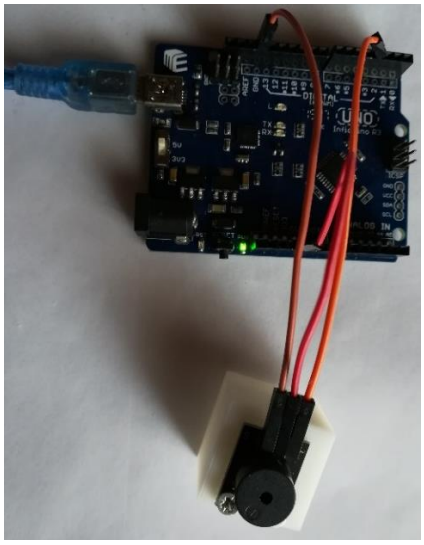
```
const int buzzer = 3; //zvucnik na pin 3

void setup(){

  pinMode(buzzer, OUTPUT); // inicijaliziranje zvucnika

}

void loop(){

  tone(buzzer, 1000); // zvučnik proizvodi zvuk frekvencije 1000 HZ
  delay(200);        // zvuk traje 0.2 sekunde
  noTone(buzzer);     // zaustavi zvuk
  delay(200);        // čekaj 0.2 sekunde

  tone(buzzer, 500);
  delay(1000);
  noTone(buzzer);
  delay(200);

  tone(buzzer, 100);
  delay(500);
  noTone(buzzer);
  delay(200);

  }
```

Da bi zujalica proizvodila zvuk koristimo naredbu **tone(pin, frequency)** koja izvodu šalje signal određene frekvencije. Tako npr. **naredba tone(buzzer, 1000)** omogućava da zujalica (*buzzer*) proizvodi zvuk frekvencije 1000 Hz, odnosno 1 kHz.

Naredba **noTone(pin)** zaustavlja slanje signala izvodu i više ne čujemo tonove koje zujalica proizvodi.

Umjesto korištenja obje naredbe **tone** i **noTone** možemo koristiti naredbu **tone(pin, frequency, duration)**. Tako npr. naredba **tone(buzzer, 500, 1000)** šalje zujalici signal frekvencije 500 i to slanje signala traje 1 sekundu.

Spoj izgleda ovako:



Zadatak 2. Napiši program koji će odsvirati neku pjesmicu po tvom izboru.

Rješenje:

Note za pjesmicu „Sretan rođendan“ glase: D, D, E, D, G, F, D, D, E, D, A, G, D, D, B, G, F, E, E, C, C, B, G, A i G.

Ton A je frekvencije 440 Hz, ton B 493 Hz, ton C 523 Hz, ton D 294 Hz, ton E 329 Hz, ton F 349 Hz i ton G frekvencije 392 Hz.

Dio programskog koda za sviranje pjesmice „Sretan rođendan“ izgleda ovako:

```
const int buzzer = 3; //zvucnik na pin 3

void setup(){

  pinMode(buzzer, OUTPUT); // inicijaliziranje zvucnika

}

void loop(){

  tone(buzzer, 294); // zvučnik proizvodi zvuk frekvencije 294 HZ
  delay(200);        // zvuk traje 0.2 sekunde
  noTone(buzzer);     // zaustavi zvuk
  delay(200);        // čekaj 0.2 sekunde

  tone(buzzer, 294);
  delay(200);
  noTone(buzzer);
  delay(200);

  tone(buzzer, 329);
  delay(500);
  noTone(buzzer);
  delay(200);

  tone(buzzer, 294);
  delay(500);
  noTone(buzzer);
  delay(200);
```

Možemo primijetiti da se potpuni programski kod sastoji od 25 blokova koji se sastoje od naredbi **tone**, **delay**, **noTone** i **delay**.

Programski kod može se skratiti korištenjem naredbe **tone(pin, frequency, duration)**, korištenjem varijabli ili korištenjem petlji.

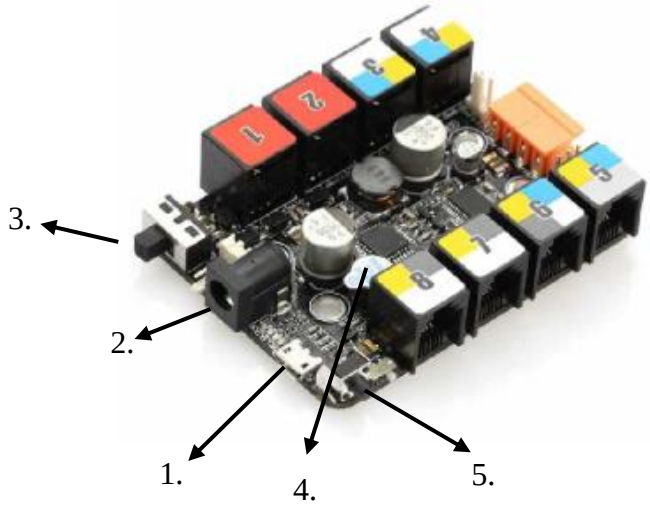
Poglavlje 5

Pločica Orion

Izgledom i nekim karakteristikama pločica slična pločici Arduino je pločica Orion. U ovom poglavlju opisana je pločica Orion i dane su aktivnosti koje se mogu provoditi u školama.

5.1 Općenito o pločici Orion

Pločica Orion sastoji se od sljedećih dijelova:



Slika 13: Pločica Orion (izvor: https://www.makeblock.com/)

1. USB konektor pomoću kojeg se pločica spaja s računalom.

2. Utor za napajanje električnom strujom (*Barrel Jack*). Kao i kod Arduina, preporučuje se napajanje između 6 i 12 Volti.

3. Prekidač za uključivanje i isključivanje.

4. Zujalica (*Buzzer*).

5. Dugme za resetiranje (*Reset Button*) koje programski kod pohranjen na pločici izvodi iznova.

Na pločici se nalazi i 8 ulaza, označenih brojevima od 1 do 8, pomoću kojih se različiti senzori, motori i slično spajaju na pločicu.

Pločica Orion može se programirati koristeći softver Arduino IDE ili programsku podršku MBlock u kojem se programira metodom „Povuci i pusti“. MBlock je modificirana verzija Scratch-a verzije 2.0 koji omogućava lako upravljanje senzorima, motorima, itd. Ovaj način programiranja prihvatljivi je za učenike koji se još nisu susreli s programskim jezikom C++.

Prije samog programiranja, u izborniku Pločice (*Boards*) potrebno je odabrati pločicu Startet/Ultimate (Orion).

5.2 Primjeri aktivnosti u školi

5.2.1 Aktivnost 1: Prvi program

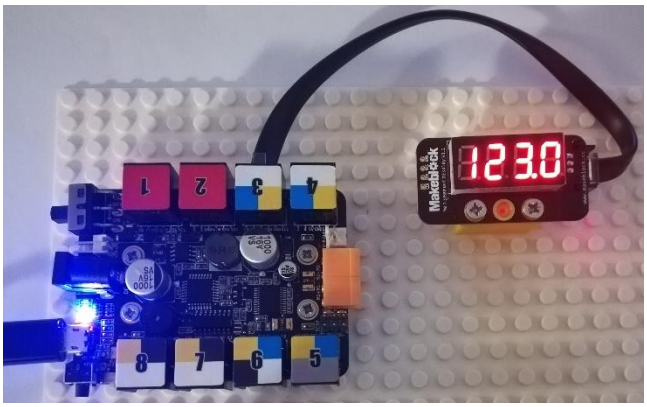
Aktivnost 1 primjerena je učenicima viših razreda osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

- pločica Orion
- mikro USB kabel
- 7-segmentni displej
- 1x kabel.

Zadatak 1. Napiši program koji na 7-segmentni displej ispisuje broj 123.

Rješenje:

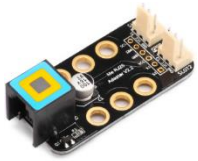


5.2.2 Aktivnost 2: Temperatura vode

Aktivnost 2 primjerena je učenicima viših razreda osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

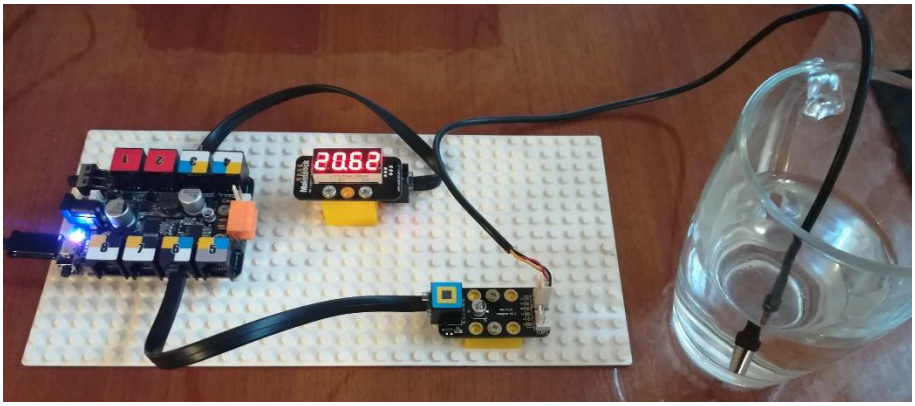
- pločica Orion
- mikro USB kabel
- 7-segmentni display
- 2x kabel

- termometar - adapter RJ25. 

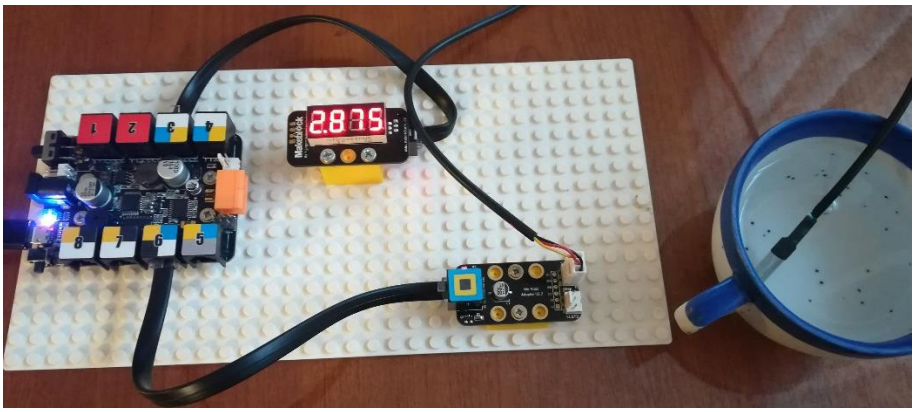
Korak 1. Spoji termometar s Orion pločicom. Da bi to mogao učiniti, potreban je adapter RJ25. Prvo spoji termometar s adapterom RJ25, a zatim njega, pomoću kabela, spoji s pločicom Orion. Da bi znao na koji ulaz spajaš senzor ili adapter, pronađi odgovarajući blok u MBlocku i u padajućem izborniku odaberi jedan od ulaza. Na taj ulaz spoji i senzor ili adapter.

Zadatak 1. Napiši program koji na 7-segmentnom displeju ispisuje kolika je temperatura tekućine.

Rješenje:



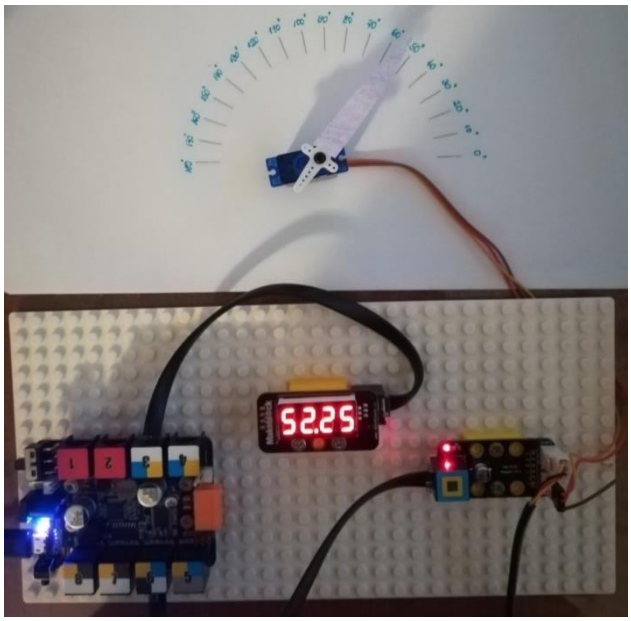
Temperatura vode je 20.62°C. Kada termometar stavimo u hladnu vodu, dobivamo:

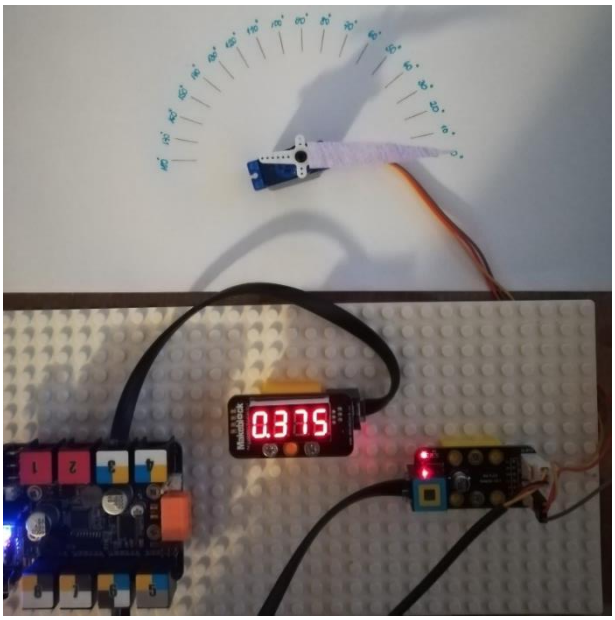


Korak 2. Spoji Mikro servo s pločicom Orion. Mikro servo, kao i senzor za temperaturu spaja se na RJ25 Adapter, no ovaj put na druge izvode.

Zadatak 2. Napiši program koji mjeri temperaturu tekućine pomoću termometra i ovisno o temperaturi pomiče mikro servo. Na listu papira nacrtaj skalu i označi brojeve na njoj i postavi mikro servo na papir tako da, kako se mikro servo pomiče, na papiru pokazuje na broj koji odgovara temperaturi vode. Na 7-segmentnom displeju prikazuje se temperatura tekućine.

Rješenje:





5.2.3 Aktivnost 3: Tišina u razredu

Aktivnost 3 primjerena je učenicima viših razreda osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

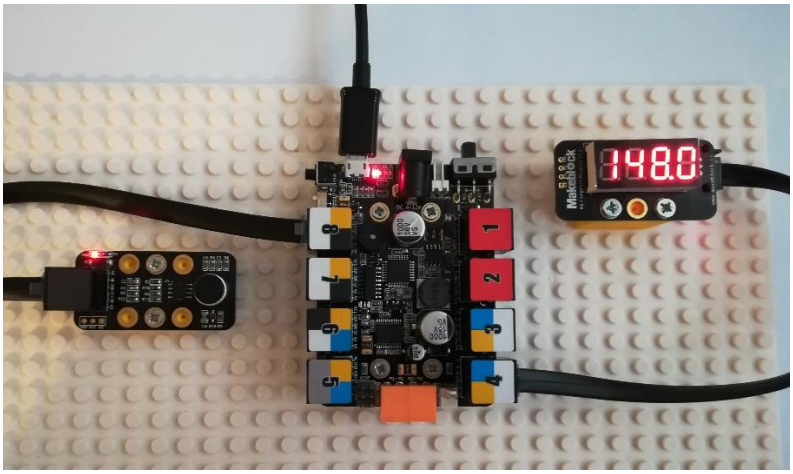
- pločica Orion
- mikro USB kabel
- 7-segmentni displej
- 2x kabel
- LED lampice
- senzor za zvuk.

Korak 1. Spoji 7-segmentni displej i senzor za zvuk s pločicom Orion.

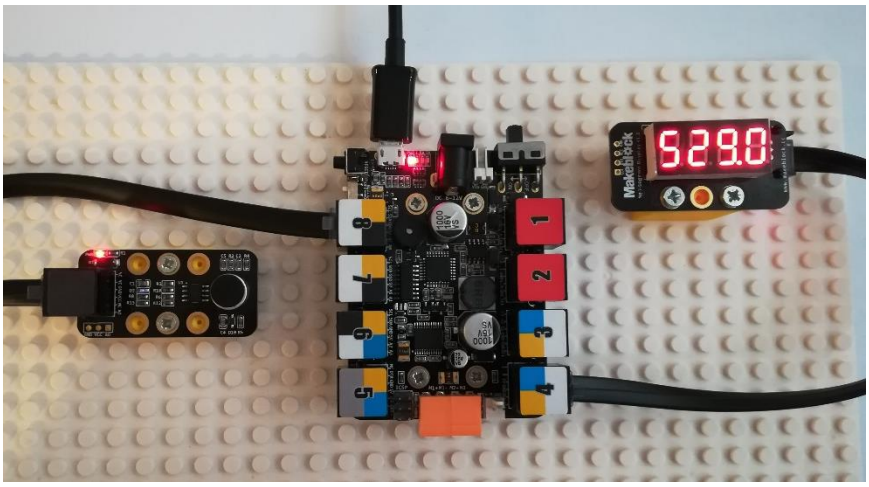
Zadatak 1. Napiši program koji na 7-segmentnom displeju ispisuje jačinu zvuka. Prati i zapiši koje vrijednosti daje senzor kada je u prostoriji tišina, a koje vrijednosti kada je u prostoriji bučno.

Rješenje:

Kada je u prostoriji tišina, senzor daje vrijednosti oko 150.



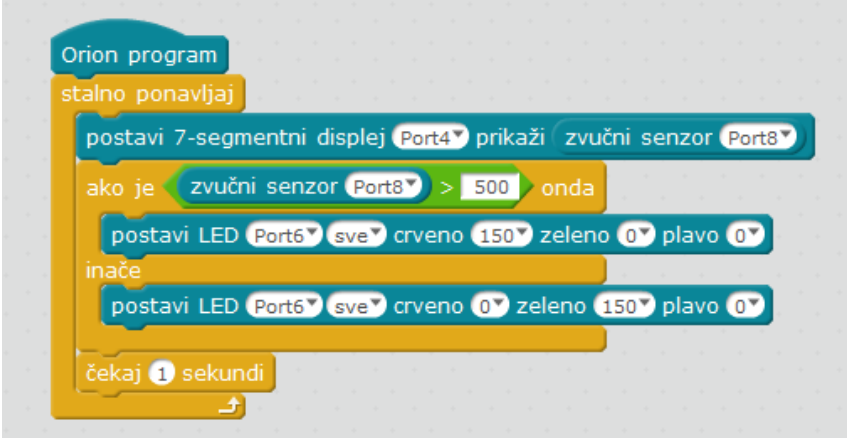
Kada je u prostoriji bučno, senzor daje vrijednosti oko 500.



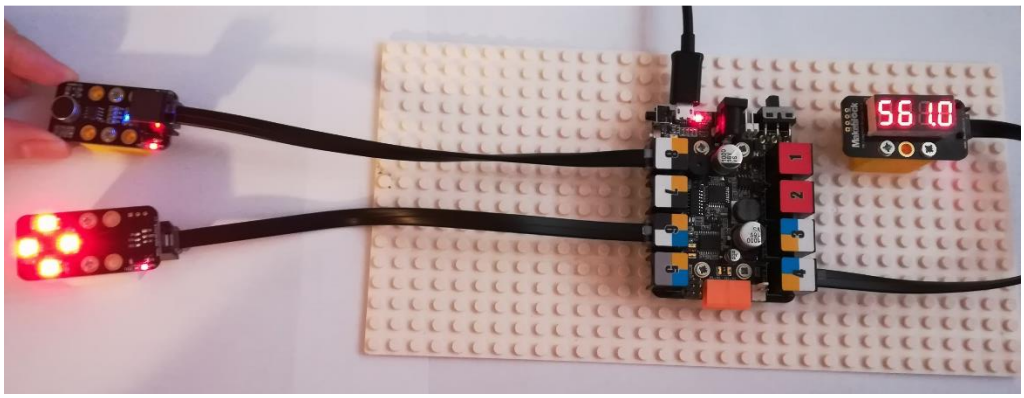
Korak 2. Spoji LED lampice sa pločicom Orion.

Zadatak 2. Napiši program koji na 7-segmentnom displeju ispisuje jačinu zvuka. Kada u prostoriji postane bučno, neka se upali crveno svjetlo na LED lampicama. Ako u prostoriji nije bučno, tada svijetli zeleno svjetlo.

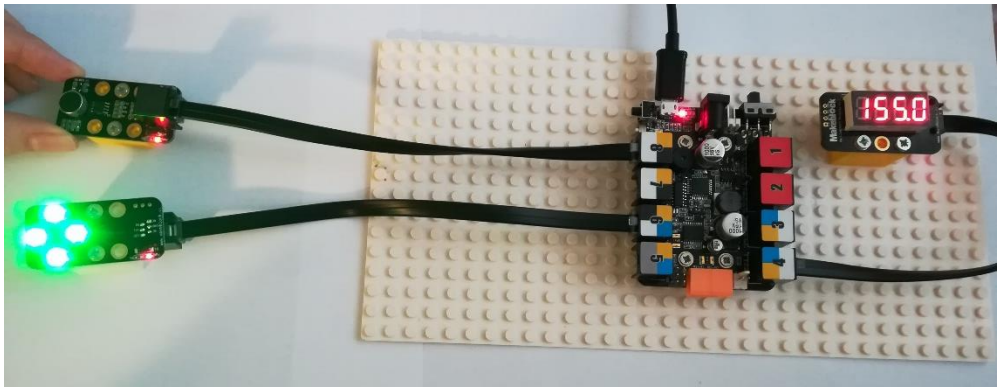
Rješenje:



Kada je u prostoriji bučno:



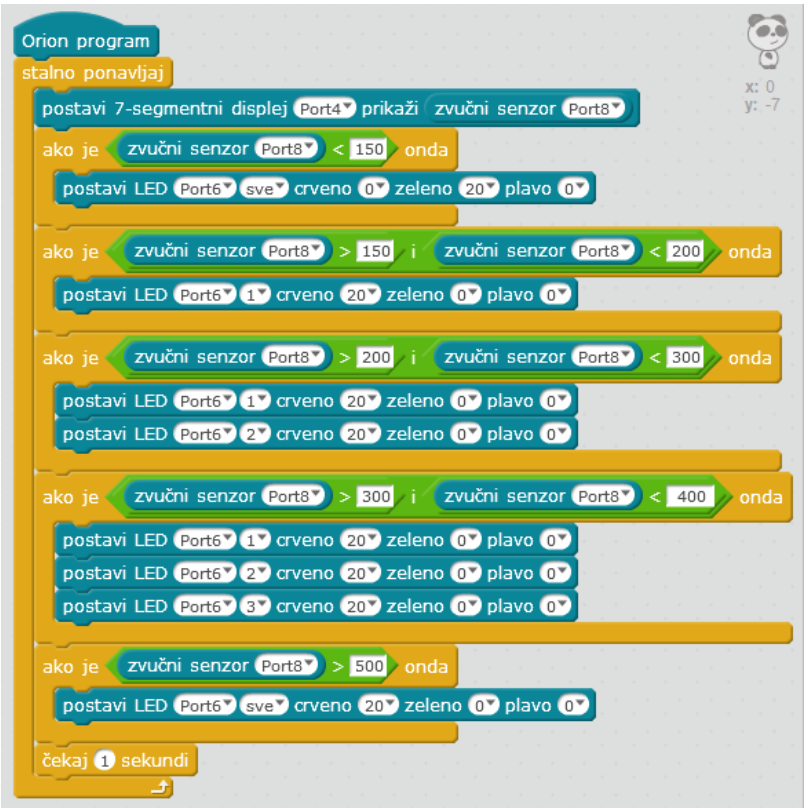
Kada u prostoriji nije bučno:



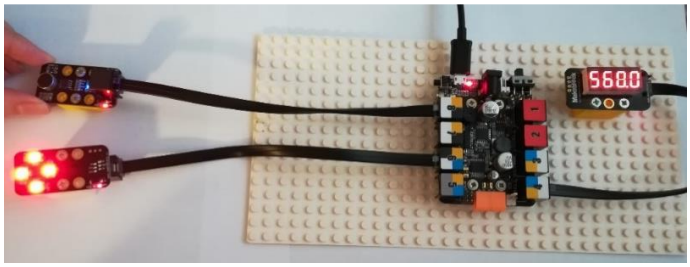
Zadatak 3. Napiši program koji na 7-segmentnom displeju ispisuje jačinu zvuka. Ovisno o tome koliko je bučno u prostoriji, LED lampice će svijetliti drukčije. Ako je u prostoriji tišina, tj. ako senzor za zvuk daje vrijednosti manje od 150, onda će svijetliti zeleno svjetlo. Kako se jačina zvuka povećava, tako se pale crvene lampice i to na sljedeći način. Ako senzor daje vrijednosti:

- između 150 i 200, pali se jedna crvena lampica
- između 200 i 300, pale se dvije crvene lampice
- između 300 i 400, pale se tri crvene lampice
- više od 500, u prostoriji jako bučno i pale se četiri crvene lampice.

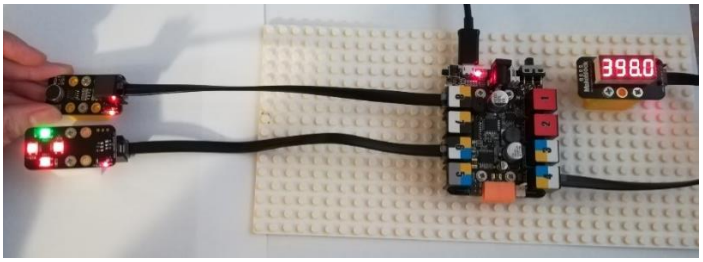
Rješenje:



Primjer 1:



Primjer 2:



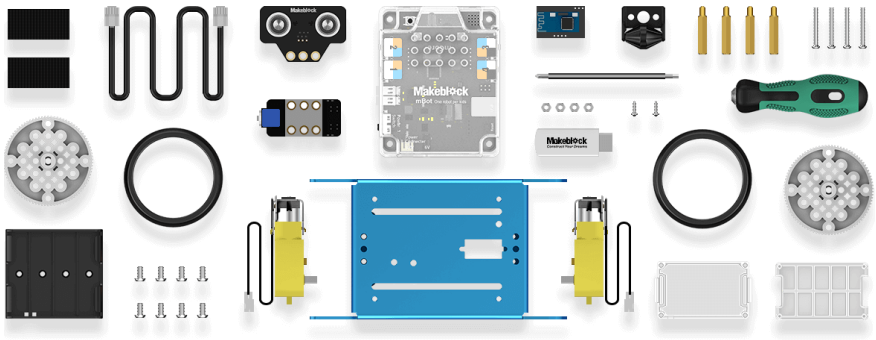
Poglavlje 6

mBot

U ovom poglavlju opisan je edukacijski robot mBot koji je proglašen jednim od najboljih edukacijskih robota na svijetu. Edukacijski robot učenicima na zoran i praktičan način daje uvid u robotiku i programiranje. Opisano je i nekoliko aktivnosti koje se mogu provoditi u školama i pomoću kojih se programira mBot.

6.1 Općenito o mBotu

Edukacijski robot mBot sastoji se od oko 40 dijelova. MBot s računalom može komunicirati Bluetoothom ili pomoću wireless modula. MBot se kreće pomoću dva kotača koje pokreću elektromotori. Osim toga, robot ima i nekoliko senzora: senzor svjetlosti, ultrazvučni senzor, senzor za praćenje linija itd.



Slika 14: Dijelovi edukacijskog paketa mBot (izvor: https://www.makeblock.com/)

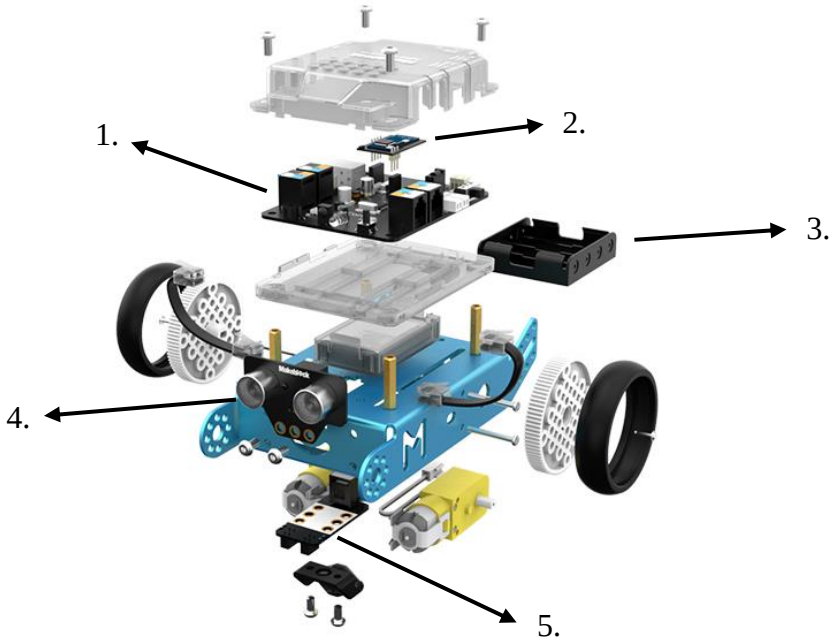
Ovako mBot izgleda kada je sastavljen:

Slika 15: Sastavljeni mBot (izvor: https://www.makeblock.com/)

Uz neke dodatke mBot možemo preoblikovati i stvoriti robote poput ovih na slikama:

Slika 16: Roboti varijacije mBota (izvor: https://www.makeblock.com/)

MBot se sastoji od sljedećih dijelova:



Slika 17: Dijelovi mBota (izvor: https://www.makeblock.com/)

1. Pločica (*mCore*) koja je „mozak“ mBota. Pločica je vrlo slična pločici Arduino Uno, a dizajnirana je posebno za mBot. Na pločici se nalaze zujalica, senzor svjetlosti, dugme, RBG lampice, izvodi za motore mBota itd.

2. Bluetooth modul koji se spaja direktno na pločicu. Modul omogućava upravljanje mBotom računalom ili mobilnim uređajem putem Bluetootha.

3. Kućište za baterije. Da bi mBot mogao raditi potreban mu je izvor napajanja. Za napajanje potrebne su četiri AA baterije koje se stavljaju u kućište za baterije.

4. Ultrazvučni senzor – „oči“ robota. Pomoću ultrazvučnog senzora robot detektira prepreke ispred sebe.

5. Senzor za praćenje linija pomoću kojeg robot može pratiti liniju.

Robot se još sastoji i od nekih fizičkih dijelova koji povezuju sve navedene komponente. MBot se još sastoji i od dva motora koja omogućavaju kretanje robota.

Upravljanje mBota moguće je, osim pomoću Bluetootha, daljinskim upravljačem, aplikacijom Makeblock na mobilnom uređaju ili jednostavno pohranjivanjem programa na pločicu. Programe za mBot možemo kreirati koristeći program mBlock.

Prije samog programiranja mBota, mogu se isprobati neki od pohranjenih programa na mBotu. Prvi program omogućava upravljanje mBotom pomoću daljinskog upravljača. Drugi program omogućava kretanje robota tako da izbjegava prepreke i treći program kojim robot prati liniju nacrtanu na podu. Programi se mijenjaju pritiskom na dugme na pločici. Kada je pokrenut prvi program RBG lampice na pločici svijetle svijetloplavom bojom. Kada je pokrenut drugi program svijetle zelenom bojom i kada je pokrenut treći program svijetle plavom bojom.

6.2 Primjeri aktivnosti u školi

6.2.1 Aktivnost 1: Prvi program

Aktivnost 1 primjerena je učenicima viših razreda osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

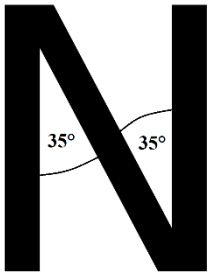
- 2 mBota
- USB kabel.

Korak 1. Spoji mBot s računalom.

Zadatak 1. Napiši program kojim će se mBot kretati u obliku slova N.

Rješenje:

Da bismo znali zadati naredbe robotu, najprije na papir skiciramo slovo N.



Robot najprije ide ravno, zatim se zarotira za $180° - 35° = 145°$ u desno, ponovno ide ravno, zatim se ponovno zarotira za $180° - 35° = 145°$ ovaj put u lijevo i nastavi opet ravno.

Programski kod izgleda ovako:

Uočavamo da umjesto zadavanja kuta zaokreta postavljamo brzine dvaju motora da bismo ostvarili rotaciju. Tako smo blokovima **motoru M1 zadaj brzinu 200**, **motoru M2 zadaj brzinu 0**, **čekaj 1 sekundu** ostvarili okretaj za otprilike 145°. Kasnije smo za okretaj od 145° koristili nešto drukčije vrijednosti. Razlog tome je što su jačine motora različite. Također, zbog jačine napajanja program se ne mora uvijek izvoditi isto.

Dakle, učenici bi trebali isprobavati i mijenjati vrijednosti da bi postigli željeni učinak.

Zadatak 2. Programski kod koji si napisao za svoj mBot prebaci na mBot svog prijatelja iz klupe. Pokreni program na oba mBota. Ponašaju li se mBoti drukčije? Ako da, što misliš zašto?

Rješenje:

Vrlo vjerojatno da će se mBoti ponašati drukčije. Jedan od razloga je to što su jačine motora dvaju mBota različite pa se tako i mBoti ponašaju drukčije. Drugi razlog je što napajanje tih mBota može biti različito, odnosno baterije jednog mBota mogu biti praznije od baterija drugog mBota.

6.2.2 Aktivnost 2: „Party“

Aktivnost 2 primjerena je učenicima viših razreda osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

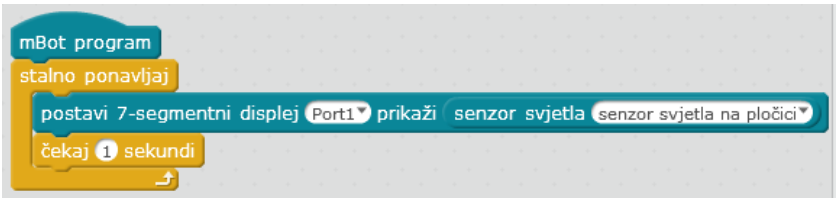
- mBot
- USB kabel
- 7-segmentni displej
- 2x kabel.

Zadatak 1. Napiši program kojim će se mBot vrtjeti u krug kada je ugašeno svjetlo. Kada je svjetlo upaljeno mBot miruje.

Rješenje:

Najprije učenici trebaju otkriti koje vrijednosti daje senzor svjetlosti kada je svjetlo ugašeno, a koje vrijednosti kada je svjetlo upaljeno. To mogu otkriti tako da 7-segmentni displej spoje na mBot.

Programski kod za taj dio izgleda ovako:



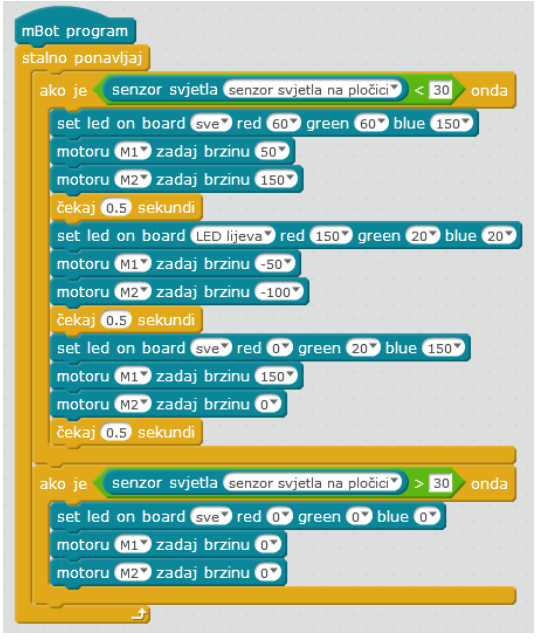
Kada je svjetlo ugašeno senzor svjetlosti daje vrijednosti od 0 do 50. Kada je svjetlo upaljeno senzor svjetlosti daje vrijednosti veće od 50.

Programski kod za konačno rješenje onda izgleda ovako:

Uočavamo da smo to da se mBot vrti u krug ostvarili blokovima **motoru M1 zadaj brzinu 100**, **motoru M2 zadaj brzinu 0**.

Zadatak 2. Napiši program tako da mBot pleše, tj. motori se pokreću kada je ugašeno svjetlo. Kada je svjetlo upaljeno mBot miruje. Dok mBot pleše RGB lampice svijetle različitim bojama.

Rješenje:



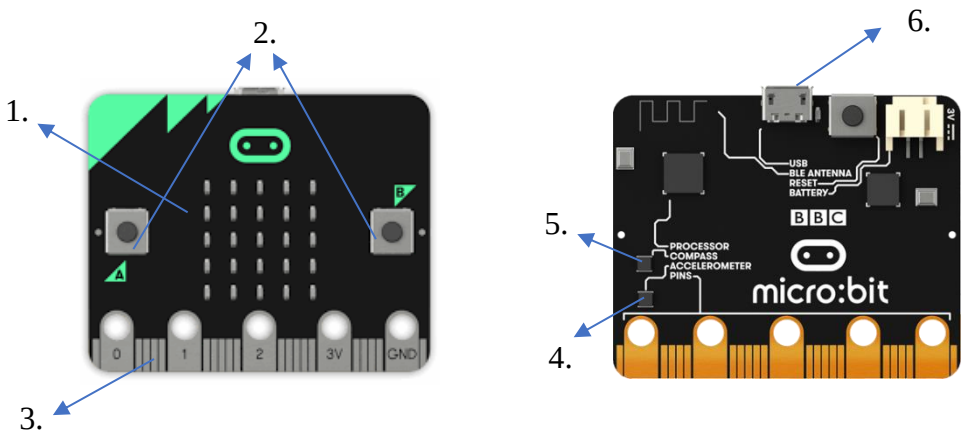
Poglavlje 7

Micro:bit

U ovom poglavlju ukratko je opisan micro:bit. Micro:bit je programabilno mikroračunalo pomoću kojega učenici mogu stvoriti različite zanimljive projekte kao što su npr. brava s brojčanom kombinacijom, kalkulator za zbrajanje do 20, program za brojanje koraka i slično. Opisano je i nekoliko aktivnosti koje se mogu provoditi u školama.

7.1 Općenito o micro:bitu

Micro:bit je maleno programabilno mikroračunalo dimenzije 4 cm x 5 cm koje se sastoji od sljedećih dijelova:



Slika 18: Micro:bit sprijeda i straga (izvor: https://microbit.org/)

1. 25 LED lampica koje se mogu programirati pojedinačno i omogućuju prikaz brojeva, slika i teksta.

2. Dvije tipke, tipke A i B.

3. Izvodi 0, 1, 2, 3V i GND koji služe za povezivanje različitih senzora na pločicu.

4. Mjerač ubrzanja koji mjeri brzinu i može detektirati drhtanje, slobodni pad i nagib.

5. Pločica na sebi ima i kompas kako bi otkrili u kojem smjeru je okrenut micro:bit. Da bi kompas davao precizne rezultate potrebno je kompas kalibrirati.

6. USB konektor pomoću kojeg se micro:bit spaja s računalom.

Na pločici nalaze se još i senzor svjetlosti, senzor temperature, dugme za resetiranje, utor za bateriju i procesor. Micro:bit ima i radio koji omogućuje komunikaciju među micro:bitovima. Da bi micro:bit mogao bežično komunicirati s računalom, mobilnim uređajem i tabletom zaslužan je Bluetooth.

Micro:bit je moguće programirati koristeći uređivač MakeCode koji se zasniva na „Povuci i pusti“ metodi (*Drag and drop*) i kojeg nije potrebno instalirati na računalo nego ga možemo pokrenuti preko web preglednika. Za one koji žele programirati micro:bit u Pythonu tu je Python uređivač koji je također moguće pokrenuti preko web preglednika. Nadalje, micro:bit možemo programirati pomoću programa MBlock i Scratch koje smo već prethodno spomenuli. Da bismo mogli programirati micro:bit programom MBlock potrebna nam je verzija MBlock 5. Za programiranje pomoću Scratcha potrebna nam je verzije Scratch 3.0. Micro:bit je moguće programirati i pomoću aplikacije micro:bit na mobilnom uređaju ili tabletu.

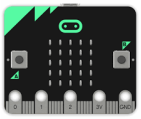
7.2 Primjeri aktivnosti u školi

U ovom poglavlju dani su primjeri aktivnosti koje se mogu provoditi u školama.

7.2.1 Aktivnost 1: Upoznavanje s micro:bitom

Aktivnost 1 primjerena je učenicima osnovne škole i nižih razreda srednje škole.

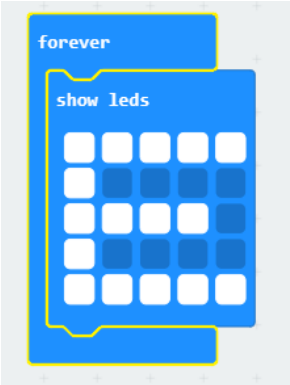
Potrebni materijali za ovu aktivnost su:

- micro:bit 

- micro USB kabel.

Zadatak 1. Napiši program tako da se na micro:bitu prikazuje slovo E.

Rješenje:



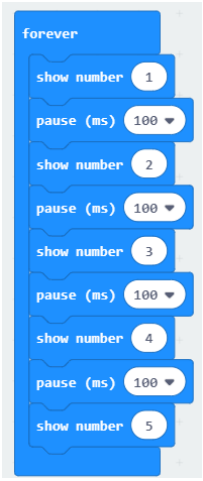
Za kreiranje ovog programa korišten je uređivač MakeCode. Nakon što se kreira program za micro:bit, micro:bit se pomoću micro USB kabela spoji s računalom, program se preuzme i prebaci na micro:bit. Da bi se program mogao izvoditi, možemo ili micro:bit spojiti s računalom kao izvorom napajanja ili spojiti bateriju na micro:bit.

Zadatak 2. Napiši program tako da se na micro:bitu prikazuju brojevi od 1 do 5.

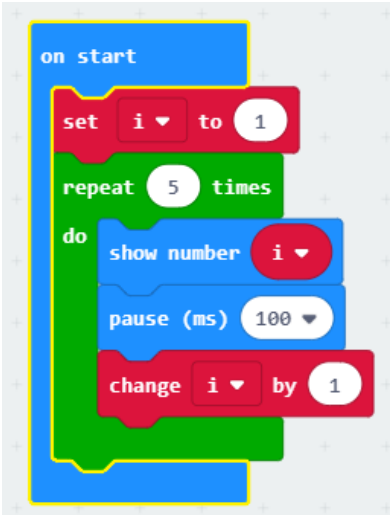
Rješenje:

Prikazana su dva rješenja. U drugom rješenju koriste se petlja i varijabla. Ako učenici nisu upoznati s varijablama i petljama prvi način je prikladniji za njih.

Prvo rješenje:



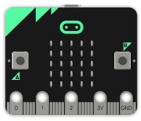
Drugo rješenje:



7.2.2 Aktivnost 2: Tajno komuniciranje

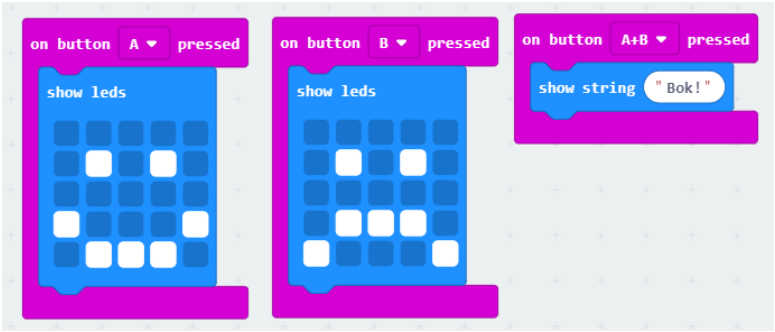
Aktivnost 2 primjerena je učenicima osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

- 2 micro:bita 
- micro USB kabel
- 2 kućišta za baterije
- 2 seta od dvije AAA baterije.

Zadatak 1. Napiši program tako da se na micro:bitu pritiskom na tipku A prikazuje emotikon sretnog lica, a pritiskom na tipku B emotikon tužnog lica. Ako se pak pritisnu tipke A i B istovremeno prikaže se tekst „Bok!“.

Rješenje:

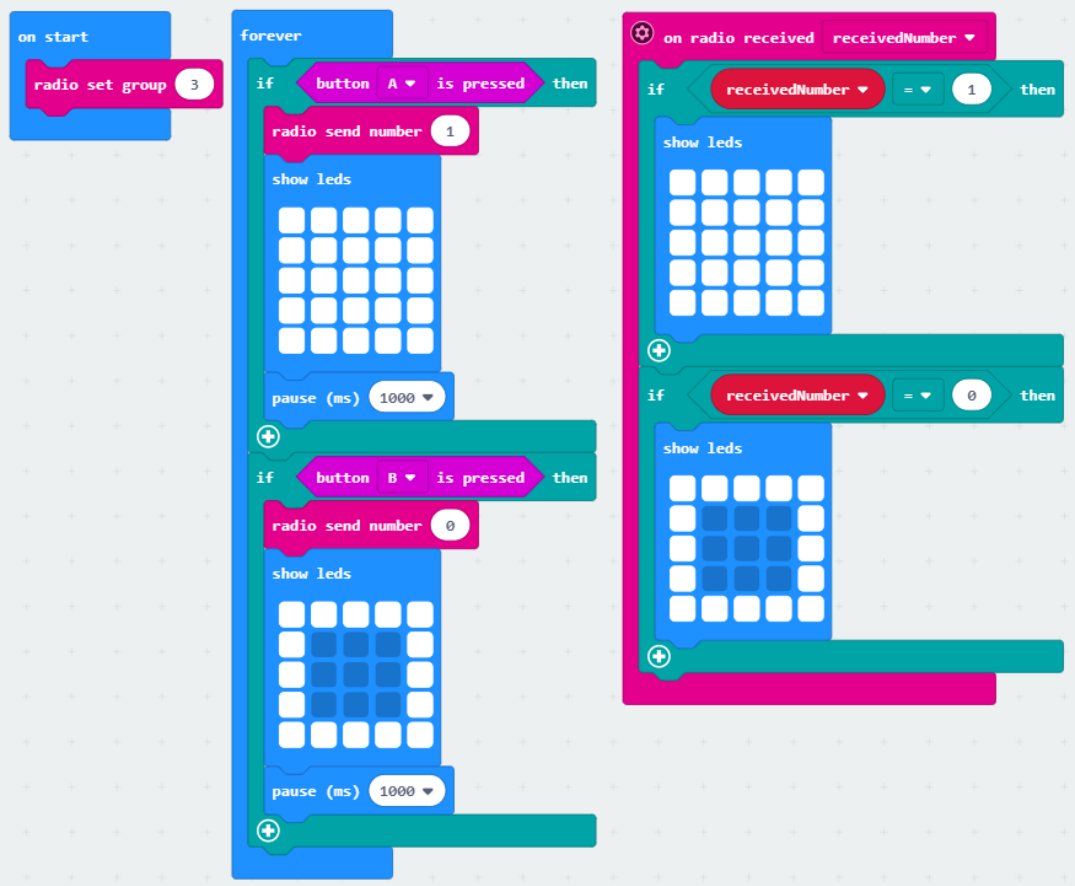


Zadatak 2. Skupa s svojim prijateljem iz klupe napravi program za tajno komuniciranje preko vaših micro:bitova. Micro:bitovi komuniciraju preko radija. Program radi tako da:

- Kada pritisneš tipku A, tvoj micro:bit šalje poruku „DA”. Na tvom i prijateljevom micro:bitu pojavi se ispunjeni kvadratić dimenzije 5x5.
- Kada pritisneš tipku B, tvoj micro:bit šalje poruku „NE”. Na tvom i prijateljevom micro:bitu pojavi se prazni kvadratić dimenzije 5x5.
- Vrijedi i suprotno, tj. kada prijatelj pritisne tipku A na svom micro:bitu tada se na njegovom i tvom micro:bitu prikaže ispunjeni kvadratić itd.
- Da bi tajnost bila veća, prijatelji dogovaraju koji kanal će koristiti.

Tvoj prijatelj i ti sada možete tajno komunicirati, tj. možeš prijatelju naglas postaviti pitanje, a on će ti odgovoriti tako da samo ti znaš odgovor.

Rješenje:



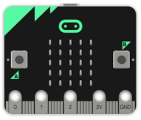
Program je potrebno pohraniti na oba micro:bita. Program radi tako da ako je pritisnuta neka od tipki na bilo kojem micro:bitu, tada se šalje signal preko radija. Drugi micro:bit taj signal prima i na temelju toga izvršava neku određenu naredbu. Na oba micro:bita možemo spojiti baterije kao izvor napajanja ili ih spojiti s računalom.

Na samom početku blokom **radio set group 3** postavljamo kanal koji će se pri komunikaciji koristiti. Ako se ova aktivnost provodi u većoj grupi učenika važno je napomenuti učenicima da svaki par učenika mora odabrati različiti kanal za komunikaciju kako ne bi došlo do poteškoća pri komunikaciji.

7.2.3 Aktivnost 3: Brojač koraka

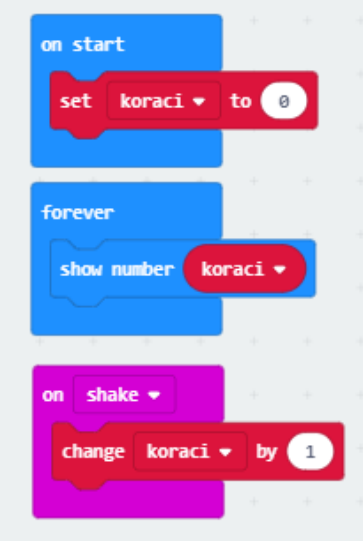
Aktivnost 3 primjerena je učenicima osnovne škole i nižih razreda srednje škole.

Potrebni materijali za ovu aktivnost su:

- micro:bit 
- micro USB kabel
- kućište za baterije
- set od dvije AAA baterije.

Zadatak 1. Napiši program kojim će micro:bit brojati tvoje korake.

Rješenje:



Micro:bit će brojati korake tako što će registrirati trzaje do kojih dođe kada osoba hoda. Trzaje micro:bit može registrirati pomoću bloka on shake.

Poglavlje 8

Prednosti i nedostatci programiranja nabrojenih fizičkih objekata

U ovom poglavlju bit će nabrojene prednosti i nedostatci programiranja pločice Pico, edukacijskog paketa Lego WeDo, pločice Arduino, pločice Orion, edukacijskog robota mBot i micro:bita.

8.1 Pločica Pico

Ako su učenici upoznati s programom Scratch, onda neće imati poteškoća s programiranjem pločice Pico. Kako se pločica programira u Scratchu, učenicima to može biti blagi prijelaz s programiranja virtualnih objekata na programiranje stvarnih, fizičkih objekata. Sama pločica ne sadrži mnogo senzora, ali one koje ima dovoljni su za stvoriti neke jednostavne programe. Mogućnosti pločice Pico mogu se proširiti dodatkom (PicoBoard Lab Pack) koji sadrži potenciometar, termometar, motor, dva dodatna gumba itd.

Za razliku od pločica Arduino i Orion koje je moguće programirati koristeći više programa, pločicu Pico moguće je programirati samo koristeći program Scratch. Cijena same pločice je oko 45$, a cijena pločice i dodatka oko 70$. Ono što još može predstavljati nedostatak ove pločice su izvodi. Naime, pločica ima ukupno 4 izvoda na koja je moguće spojiti ukupno 4 dodatka istovremeno. Pločicu je moguće povezati s računalom samo pomoću USB kabela, nema mogućnost spajanja pomoću Bluetootha i sl.

8.2 Edukacijski paket Lego WeDo

Kao što je već navedeno, većina učenika je već upoznata s lego kockama pa im ovaj spoj programiranja i lego kockica može biti jako zanimljiv. Sada ne samo da mogu stvarati različite likove pomoću lego kockica nego sada ti likovi mogu i nešto raditi. Pomoću ovog edukacijskog paketa učenici mogu stvoriti različite zanimljive projekte ili isprobati gotove projekte kojih na internetu ima pregršt. Osim osnovnog paketa postoje i neki dodaci kao što su LEGO MINDSTORMS koji pružaju mnoštvo dodatnih mogućnosti. Lego WeDo

programira se koristeći softver WeDo koji je, osim na računalu, dostupan i na mobilnim uređajima i tabletima. Pomoću softvera učenici dobivaju vrlo detaljne upute prvo kako sastaviti lik, a onda i kako ga programirati. LegoWeDo može se spojiti s računalom pomoću Bluetootha što učenicima olakšava korištenje i programiranje paketa jer ne postoji fizička povezanost s računalom. Osim Bluetoothom, LegoWeDo s računalom se može spojiti i pomoću USB kabela.

Edukacijski paket sastoji se od 280 dijelova među kojima ima i jako sitnih dijelova i ti dijelovi mogu se onda lako zagubiti. Također, zbog velike količine dijelova potrebno je puno prostora u učionici da bi učenici mogli bez poteškoća koristiti paket. Cijena edukacijskog paketa LegoWeDo je oko 200$, dok je cijena paketa LEGO MINDSTORMS oko 400$.

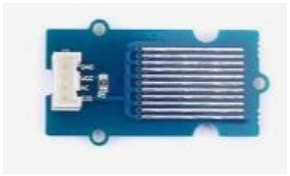
8.3 Pločica Arduino

Pločicu Arduino moguće je programirati koristeći Arduino IDE ili MBlock. Za učenike koji nisu upoznati s programskim jezikom C++ prikladnije je programiranje metodom „Povuci i pusti“ u programu MBlock. Ipak, da bismo neke naredbe u MBlocku mogli koristiti trebamo na računalo instalirati i neke dodatke. Učenicima srednje škole veći izazov je programirati pomoću softvera Arduino IDE. Učenici nakon što napišu program pomoću programa Arduino IDE mogu program napisati i koristeći grafički editor u MBlocku. Arduino IDE moguće je pokrenuti i pomoću programa MBlock. Postoji veliki broj različitih pločica Arduino pa učenici tako mogu istraživati funkcionalnosti istih. Prosječna cijena Arduino pločica je oko 20$. Na internetu je moguće pronaći i jeftinije pločice koje su jako slične najpoznatijim pločicama Arduino. Cijena takvih pločica je oko 4$. Bitno je napomenuti da su i te verzije legalne sve dok na sebi ne sadrže logo Arduino pošto je cijeli koncept otvorenog koda i ne krše se nikakva autorska prava. Kopije različitih pločica Arduino su lošije kvalitete i mogu se ponašati drukčije od svojih originala u različitim uvjetima kao što su temperatura, vlažnost itd. Na internetu se lako mogu pronaći različiti materijali za programiranje pločice Arduino. Na pločicu Arduino mogu se spojiti različiti senzori do kojih je lako doći i cjenovno su prihvatljivi. Tako se npr. senzor za otisak prsta može pronaći po cijeni od oko 10$, senzor za vlagu za 1$, senzor za otkucaje srca za 2$, senzor za mjerenje tlaka zraka za 1$, itd.

Glavni nedostatak pločice Arduino je spajanje senzora na pločicu. Senzori se s pločicom spajaju pomoću spojnih žica s muškim i muško-ženskim konektorima.

Slika 19: Spojne žice
(izvor: http://www.kitronik.co.uk)

Slika 20: pločica Arduino
(izvor: http://www.arduino.cc)

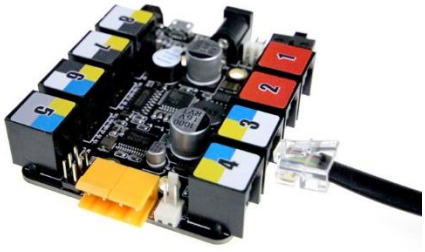


Slika 21: Senzor za Arduino
(izvor: http://www.arduino.cc)

Kod spajanja nekih senzora s pločicom Arduino sam spoj je dosta nestabilan i žice se mogu lako izvući. Učenici bi mogli imati problema s rukovanjem ako su spojena npr. 3 senzora. Tada je minimalno 9 žica potrebno da bismo sve povezali.

8.4 Pločica Orion

Problem s velikim brojem žica nemamo kod programiranja pločice Orion. Dodatne komponente na pločicu spajamo pomoću kabela kako je prikazano na slici 22. Za spajanje jedne komponente koristimo jednu žicu. Takav spoj je i čvršći u odnosu na spajanje komponenti na pločicu Arduino.



Slika 22: Spoj na pločici Orion (izvor: https://www.makeblock.com)

Nadalje, postoji veliki broj komponenti koje možemo spojiti na pločicu Orion kao što su senzor plamena, senzor za temperaturu i vlažnost, bluetooth modul, Wi Fi modul, LED RGB traka, itd. Cijena same pločice je oko 20$, a cijena dodataka kao što su senzor plamena, senzor za temperaturu i vlažnosti i LED RGB traka je oko 7$. Cijena dodataka poput bluetooth modula i Wi-Fi modula koji služe za komunikaciju s pločicom Orion je oko 20$. Dobro je

napomenuti da se nabrojene komponente, iako su prilagođene za pločicu Orion, mogu lako spojiti i s pločicom Arduino.

Pločicu Orion moguće je programirati pomoću softvera MBlock koji je za Orion pločicu jako dobro razvijen. Ipak, postoje neki senzori za koje ne postoje blokovi u grafičkom editoru. Jedan od takvih senzora je i kompas. Da bismo mogli koristiti i kompas možemo program pisati pomoću softvera Arduino IDE. To je još jedan način na koji možemo kreirati programe za pločicu Orion. U odnosu na broj materijala koje možemo pronaći za programiranje pločice Arduino, materijala za programiranje pločice Orion je znatno manje.

8.5 Edukacijski robot mBot

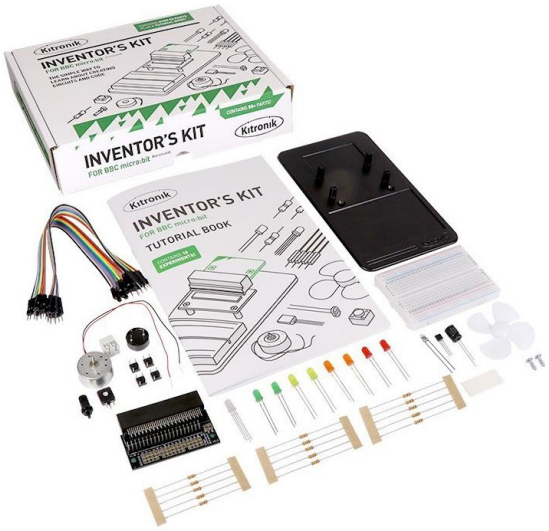
Ako su učenici istraživali mogućnosti pločice Orion onda neće imati problema s programiranjem ovog edukacijskog paketa. Pločica mCore od koje se mBot sastoji vrlo je slična pločici Orion pa tako učenici mogu na nju spojiti i dodatke koje su spajali na pločicu Orion. Tako npr. mogu spojiti 7-segmentni displej na mBot. Nadalje, na internetu je moguće pronaći i mnogo materijala i aktivnosti za programiranje mBota. Učenici mogu stvoriti različite zanimljive projekte. Ono što je prednost programiranja mBota je što je njega moguće programirati, osim pomoću računala, i mobilnim uređajem ili tabletom. To može biti korisno ako nema dovoljno računala u prostoriji da bi svaki učenik mogao programirati mBot.

Nedostatak mBota je što je učenicima potrebno mnogo prostora da bi istraživali njegove mogućnosti pa je nezgodno provoditi aktivnosti u učionici. Za pokretanje mBota potrebne su 4 AA baterije, a za daljinski upravljač 2 AAA baterije. Baterije ne dolaze s edukacijskim robotom pa ih je potrebno posebno kupiti. Baterije se brzo troše pa bi uz sve to bilo dobro imati i punjač za baterije. Nadalje, daljinski upravljač za mBot potrebno je dosta približiti mBotu i često ne funkcionira.

8.6 Micro:bit

Za programiranje micro:bita postoji pregršt materijala na internetu, kako na engleskom jeziku tako i na hrvatskom jeziku. Micro:bit odličan je za učenike nižih razreda osnovne škole koji se tek upoznaju s robotikom. Učenici mogu stvarati različite zanimljive programe. Pločica se može spojiti i s nekim drugim pločicama kao što su Raspberry Pi i Arduino. Micro:bitov displej koji se sastoji od 25 LED lampica može biti dosta ograničavajući čimbenik u kreiranju

različitih programa. Nadalje, pločica ima samo 2 tipke i svega par senzora na sebi. Ipak, mogućnosti micro:bita mogu se uvelike proširiti korištenjem Kitronik seta za micro:bit.



Slika 23: Kitronik paket (izvor: https://www.kitronik.co.uk)

Kitronik set sastoji se od nekoliko lampica, potenciometra, motora, ventilatora itd. Cijena samog micro:bita je oko 20$, a cijena kitronik seta je oko 35$. Micro:bit se može programirati i pomoću aplikacije na mobilnom uređaju ili tabletu što također može biti velika prednost ako su učenici u prostoriji gdje nema računala.

Micro:bit je dimenzija 4 cm x 5 cm što je čini najmanjom od do sada nabrojenih pločica. Uz micro:bit dolazi i USB kabel koji je dužine svega 15 cm što bi učenike moglo dosta ograničavati.

Poglavlje 9

Usporedba s kurikulumom hrvatskih i američkih škola

U ovom poglavlju dani su ishodi iz hrvatskih i američkih škola koji se mogu ostvariti programiranjem fizičkih objekata.

9.1 Kurikulum nastavnog predmeta Informatika za osnovne i srednje škole

U dokumentu *Kurikulum nastavnoga predmeta Informatika za osnovne i srednje škole* nailazimo na sljedeće odgojno-obrazovne ishode koji se mogu ostvariti programiranjem fizičkih objekata:

Osnovna škola – 70 sati godišnje u svim razredima.

5. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon pete godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik: B. 5. 1 koristi se programskim alatom za stvaranje programa u kojemu se koristi ulaznim i izlaznim vrijednostima te ponavljanjem B. 5. 2 stvara algoritam za rješavanje jednostavnoga zadatka, provjerava ispravnost algoritma, otkriva i popravlja pogreške.

6. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon šeste godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik:

	B. 6. 1 stvara, prati i preuređuje programe koji sadrže strukture grananja i uvjetnoga ponavljanja te predviđa ponašanje jednostavnih algoritama koji mogu biti prikazani dijagramom, riječima govornoga jezika ili programskim jezikom B. 6. 2 razmatra i rješava složeniji problem rastavljajući ga na niz potproblema.

7. razred	
Domena	Ishod
Informacije i digitalna tehnologija	Nakon sedme godine učenja predmeta Informatika u domeni Informacije i digitalna tehnologija učenik: A. 7. 2 primjenjuje strategije za prepoznavanje i rješavanje rutinskih hardverskih/softverskih problema do kojih može doći tijekom uporabe računalne tehnologije. A. 7. 3 prikuplja i unosi podatke kojima se analizira neki problem s pomoću odgovarajućega programa, otkriva odnos među podatcima koristeći se različitim alatima programa te mogućnostima prikazivanja podataka.
Računalno razmišljanje i programiranje	Nakon sedme godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik: B. 7. 1 razvija algoritme za rješavanje različitih problema koristeći se nekim programskim jezikom pri čemu se koristi prikladnim strukturama i tipovima podataka B. 7. 4 koristi se simulacijom pri rješavanju nekoga, ne nužno računalnoga, problema.
Digitalna pismenost i komunikacija	Nakon sedme godine učenja predmeta Informatika u domeni Digitalna pismenost i komunikacija učenik: C. 7. 1 koristi i upoznaje se s različitim platformama i programima, koje prema potrebi pronalazi i instalira.

8. razred	
Domena	Ishod
Informacije i digitalna tehnologija	Nakon osme godine učenja predmeta Informatika u domeni Informacije i digitalna tehnologija učenik: A. 8. 4 prepoznaje i proučava interdisciplinarnu primjenu računalnoga razmišljanja analiziranjem i rješavanjem odabranih problema iz različitih područja učenja.
Računalno razmišljanje i programiranje	Nakon osme godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik: B. 8. 1 identificira neki problem iz stvarnoga svijeta, stvara program za njegovo rješavanje, dokumentira rad programa i predstavlja djelovanje programa drugima.
Digitalna pismenost i komunikacija	Nakon osme godine učenja predmeta Informatika u domeni Digitalna pismenost i komunikacija učenik: C. 8. 2 samostalno pronalazi informacije i programe, odabire prikladne izvore informacija te uređuje, stvara i objavljuje/dijeli digitalne sadržaje C. 8. 3 dizajnira, razvija, objavljuje i predstavlja radove s pomoću sredstava informacijske i komunikacijske tehnologije primjenjujući suradničke aktivnosti.

Opće, jezične, klasične i prirodoslovne gimnazije – 70 sati godišnje

1. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon prve godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik: B. 1. 1 analizira problem, definira ulazne i izlazne vrijednosti te uočava korake za rješavanje problema B. 1. 2 primjenjuje jednostavne tipove podataka te argumentira njihov odabir, primjenjuje različite vrste izraza, operacija, relacija i standardnih funkcija za modeliranje jednostavnoga problema u odabranome programskom jeziku

	B. 1. 3 razvija algoritam i stvara program u odabranome programskom jeziku rješavajući problem uporabom strukture grananja i ponavljanja.

2. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon druge godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik: B. 2. 2 u zadanome problemu uočava manje cjeline, rješava ih te ih potom integrira u jedinstveno rješenje problema B. 2. 4 u suradnji s drugima osmišljava algoritam, implementira ga u odabranom programskom jeziku, testira program, dokumentira i predstavlja drugima mogućnosti i ograničenja programa.
e-Društvo	Nakon druge godine učenja predmeta Informatika u srednjoj školi u domeni e-Društvo učenik: D. 2. 2 analizira i procjenjuje utjecaj informacijske i komunikacijske tehnologije na učinkovitost i produktivnost u raznim područjima i poslovima.

3. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon treće godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik: B. 3. 3 koristeći neki grafički modul vizualizira i grafički prikazuje neki problem iz svoje okoline B. 3. 4 rješava problem primjenjujući složene tipove podataka definirane zadanim programskim jezikom B. 3. 5 definira problem iz stvarnoga života i stvara programsko rješenje prolazeći sve faze programiranja, predstavlja programsko rješenje ostalima i vrednuje ga.

4. razred	
Domena	Ishod
Računalno razmišljanje i programiranje	Nakon četvrte godine učenja predmeta Informatika u srednjoj školi u domeni Računalno razmišljanje i programiranje učenik: B. 4. 4 definira problem iz stvarnoga života i stvara programsko rješenje prolazeći sve faze programiranja, predstavlja programsko rješenje i vrednuje ga.

9.3 Američki nacionalni kurikulum CSTA K-12

Američki nacionalni kurikulum CSTA K-12 obuhvaća tri razine. U prvoj razini opisani su standardi učenja za učenike od prvog do šestog razreda, u drugoj razini od šestog do devetog razreda i treća razina od devetog do dvanaestog razreda. U svakoj razini razlikujemo ishode vezano za sljedeća područja: Computational Thinking, Collaboration, Computing Practise & Programing, Computers and Communications Devices and Community Global, and Ethical Impacts. Standardi su navedeni u originalu.

Level 1	
Computational Thinking	The student will be able to: 1. Understand and use the basic steps in algorithmic problem-solving (e.g., problem statement and exploration, examination of sample instances, design, implementation, and testing). 2. Develop a simple understanding of an algorithm (e.g., search, sequence of events, or sorting) using computer-free exercises.
Computing Practice and Programming	The student will be able to: 1. Construct a set of statements to be acted out to accomplish a simple task (e.g., turtle instructions). 2. Construct a program as a set of step-by-step instructions to be acted out (e.g., make a peanut butter and jelly sandwich activity). 3. Implement problem solutions using a block based visual programming language.

Level 2	
Computational Thinking	The student will be able to: 1. Use the basic steps in algorithmic problem solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing, evaluation). 2. Describe the process of parallelization as it relates to problem solving. 3. Define an algorithm as a sequence of instructions that can be processed by a computer. 4. Evaluate ways that different algorithms may be used to solve the same problem.
Computers & Communications Devices	The student will be able to: 1. Describe what distinguishes humans from machines focusing on human intelligence versus machine intelligence and ways we can communicate. 2. Describe ways in which computers use models of intelligent behavior (e.g., robot motion, speech and language understanding, and computer vision).
Collaboration	The student will be able to: 1. Work in a team to design and develop a software artifact. 2. Use collaborative tools to communicate with project team members (e.g., discussion threads, wikis, blogs, version control, etc.). 3. Describe how computing enhances traditional forms and enables new forms of experience, expression, communication, and collaboration. 4. Identify how collaboration influences the design and development of software products.

Level 3	
Computational Thinking	The student will be able to: 1. Evaluate algorithms by their efficiency, correctness, and clarity.
Computing Practice and Programming	The student will be able to: 1. Describe a variety of programming languages available to solve problems and develop systems.

	2. Use project collaboration tools, version control systems, and Integrated Development Environments (IDEs) while working on a collaborative software project. 3. Evaluate programs written by others for readability and usability.
Computers & Communications Devices	The student will be able to: 1. Describe the unique features of computers embedded in mobile devices and vehicles (e.g., cell phones, automobiles, airplanes). 2. Develop criteria for purchasing or upgrading.

Zaključak

U radu su opisani neki fizički objekti koje učenici mogu programirati. Osim nabrojenih objekata postoje još i npr. pločice Raspberry Pi i Neuron, roboti Fable, Tenergy Odev Tomo i Codey Rocky.

Uz brojne prednosti, ali i nedostatke programiranja nabrojenih fizičkih objekata, smatramo da bi se u hrvatskim školama trebalo provoditi više aktivnosti vezano za programiranje fizičkih objekata. Neki učenici u hrvatskim školama upoznati su s edukacijskim robotom Mbot pošto su škole koje sudjeluju u robotičkoj ligi Croatian Makers dobile nekoliko ovih edukacijskih robota. Učenici su upoznati i s micro:bitom pošto je u sklopu STEM revolucije preko 35 000 učenika dobilo svoj primjerak micro:bita. S druge strane, potrebno je i više ulagati u školstvo pošto oprema za programiranje ovakvih objekata zahtijeva ulaganje velike količine novca.

Kod programiranja fizičkih objekata učitelj može učenicima dati gotove materijale za programiranje, učenici sami mogu pronaći materijale na internetu ili sami osmisliti neke zanimljive projekte, mogućnosti su neiscrpne. Programiranjem fizičkih objekata učenici osim što razvijaju algoritamski način razmišljanja, uče raditi u timu i uče vrednovati svoj, ali i tuđi rad.

Literatura

1. M. Babić, N. Bubica, S. Leko, Z. Dimovski, M. Stančić, N. Mihočka, I. Ružić, B. Vejnović, Moj portal 8, Školska knjiga, Zagreb, 2018.
2. I. Kniewald, Terrapin Logo, Profil, 2005.
3. G. Nogo, Nastavni materijali s predavanja, akademska godina 2017./2018.
4. G. Nikolić, Robotska edukacija i robotska pismenost, Andragoški glasnik, 2016., Broj 1-2, str. 25-57, https://hrcak.srce.hr/file/256293, pristupljeno 21.09.2018.
5. J. C. Olabe, Programming and Robotics with Scratch in Primary Education, (2011), http://www.formatex.info/ict/book/356-363.pdf, pristupljeno 20.08.2018.
6. J. Patterson, House of Robots: Robot Revolution, Hachette Book Group, 2017.
7. Code with Mu, https://codewith.mu/, pristupljeno 15.09.2018.
8. Croatian Makers, Edukacija i Makers projekti, URL: http://izradi.croatianmakers.hr/, pristupljeno 20.12.2018.
9. Micro:bit, https://microbit.org/, pristupljeno 20.12.2018.
10. Sratch, https://scratch.mit.edu/, pristupljeno 15.09.2018.
11. Lego education, https://education.lego.com/en-us, pristupljeno 20.09.2018.
12. STARTIT, https://startit.rs/, pristupljeno 15.09.2018.
13. Sparkfun, https://www.sparkfun.com/, pristupljeno 16.09.2018.
14. Pico Cricket, https://www.picocricket.com/, pristupljeno 15.09.2018.
15. Arduino, https://www.arduino.cc/, pristupljeno 15.09.2018.
16. Kitronik, https://www.kitronik.co.uk, pristupljeno 3.1.2019.
17. MakeBlock, https://www.makeblock.com/, pristupljeno 21.12.2018.
18. Arduino Project Hub, https://create.arduino.cc/projecthub, pristupljeno 21.09.2018.
19. E-radionica, https://e-radionica.com/, pristupljeno 21.09.2018.
20. Kurikulum nastavnog predmeta Informatika za osnovne i srednje škole, 2018., https://mzo.hr/sites/default/files/dokumenti/2018/OBRAZOVANJE/Nacionalni-kurikulumi/informatika-6-3-2018.pdf, pristupljeno 2.1.2019.
21. Nacionalni okvirni kurikulum, 2011., https://www.azoo.hr/images/stories/dokumenti/Nacionalni_okvirni_kurikulum.pdf, pristupljeno 2.1.2019.
22. CSTA K-12 Computer Science Standards, 2011., http://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/Docs/Standards/CSTA_K-12_CSS.pdf, pristupljeno 2.1.2019.

Sažetak

U ovom diplomskom radu opisani su neki fizički objekti i načini na koji se oni programiraju. Dane su i aktivnosti za učenike koje učitelji u školama mogu provoditi. Aktivnosti su prilagođene učenicima osnovnih i srednjih škola.

Diplomski rad sastoji se od 9 poglavlja. U prvom poglavlju opisani su programski jezici Logo i Scratch. U drugom poglavlju opisana je pločica Pico i način na koji se pločica programira. U trećem poglavlju dan je kratki opis edukacijskog paketa Lego WeDo. U četvrtom i petom poglavlju opisane su pločice Arduino i Orion i dane su aktivnosti koje se u školi mogu provoditi. U šestom poglavlju opisan je edukacijski robot mBot i kako se može koristiti u nastavi. U sedmom poglavlju opisano je mikroračunalo micro:bit i nekoliko aktivnosti koje se mogu provesti s učenicima. U osmom poglavlju istaknute su prednosti i nedostatci programiranja fizičkih objekata i potencijalni problem s kojima se učitelji i učenici mogu susresti. U devetom poglavlju promatrani su hrvatski i američki kurikulum i uspoređivani su ishodi s ishodima aktivnosti programiranja fizičkih objekata. Na kraju rada je dan kratki zaključak o programiranju fizičkih objekata.

Summary

In this thesis, we have described some physical objects and the ways how they could be programmed. Also, we have provided activities for students that can be implemented in classrooms. All activities are suitable for classes from elementary to high school level.

The thesis consists of nine chapters. The first chapter presents two programming languages, Logo and Scratch. In second chapter there is a description of the board Pico and the way how it can be programmed. In the third chapter a short description of education set Lego WeDo is given. In the fourth and fifth chapter there is a description of boards Arduino and Orion. Also, few activities are described in which boards Arduino and Orion are being programmed. Sixth chapter presents education robot mBot and how it can be used in the classrooms. In the seventh chapter we have described a small microcomputer called micro:bit and few activities for the students. In the eighth chapter we discuss about the advantages and disadvantages of programming physical objects previously described. In the ninth chapter the comparison of the learning outcomes of the curriculum of Croatian and American schools is given. At the end of the thesis there is a conclusion about programming physical objects.

Životopis

Eni Katanić rođena je 16. travnja 1994. godine u Požegi. Osnovnu školu (OŠ fra Kaje Adžića) pohađala je u Pleternici, a srednju školu (Prirodoslovno-matematička gimnazija) u Požegi. Nakon završetka srednje škole, 2013. godine upisuje preddiplomski sveučilišni studij Matematika; smjer: nastavnički na Matematičkom odsjeku Prirodoslovno-matematičkog fakulteta Sveučilišta u Zagrebu. Nakon završetka preddiplomskog studija 2016. godine upisuje diplomski sveučilišni studij Matematika i informatika, smjer: nastavnički na istom fakultetu. Za vrijeme studiranja radila je u informatičkoj tvrtki.