O autoru

Indira Knight je uspešni programer i dizajner pokretne grafike, a ekspert je i za računarsko programiranje i broadcast grafiku. U poslednjih šest godina se fokusirala na razvoj veb stranica i vizuelizaciju podataka. Trenutno je zaposlena u kompaniji „Future Cities Catapult“, u kojoj, kao kreativni veb programer, kreira prototipove i projekte koji obuhvataju interakciju sa Arduinom. Takođe je aktivna članica programerske zajednice i započela je organizaciju sastanka u Londonu na kome će se raspravljati o API-ju WebVR.

Magistrirala je 3D kompjutersku animaciju na Univerzitetu „Bournemouth“ i kompjuterske nauke na Univerzitetu „Birkbeck“ u Londonu. Kreirala je aplikacije, interaktivne umetničke instalacije, naslove popularnih igara i nezavisne filmove, a bila je zaposlena i na jednoj televizijskoj stanici.

O tehničkom recenzentu

Mark Furman je magistar softverskog inženjeringa, pisac, nastavnik i preduzetnik. U prethodnih 18 godina bio je zaposlen u nizu kompanija, uključujući „Host Gator“, „Interland“, Suntrust Bank, AT&T i „Winn-Dixie“, u kojima je izrađivao projekte iz oblasti informacionih tehnologija sa fokusom na Linux sisteme i programiranje u Pythonu. Trenutno je fokusirao karijeru na izradu sopstvenih projekata, a pokrenuo je i sajt Tech Forge (techforge.org). Magistrirao je poslovnu administraciju na Univerzitetu u Ohaju na temu poslovne inteligencije. Možete ga pratiti na Twitteru pod nadimkom @mfurman.

KRATAK SADRŽAJ

POGLAVLJE 1
Arduino, kola i komponente 1

POGLAVLJE 2
Kreiranje veb servera 31

POGLAVLJE 3
Arduino za Front End (Deo I) 71

POGLAVLJE 4
Uvod u kreiranje veb sadržaja 93

POGLAVLJE 5
Front End za Arduino 129

POGLAVLJE 6
Arduino za Front End (deo II) 163

POGLAVLJE 7
Vizuelizacija podataka 203

POGLAVLJE 8
Kreiranje veb komandne table 233

POGLAVLJE 9
Fizička vizuelizacija podataka pomoću Live Data prikaza 279

POGLAVLJE 10
Kreiranje upravljača za igre 309

POGLAVLJE A
Arduino zajednica i komponente 369

POGLAVLJE B
Još malo Front End programiranja 373

INDEKS 385

SADRŽAJ

POGLAVLJE 1

Arduino, kola i komponente 1

Arduino 1
Arduino hardver 2
Električna energija 3
Omov zakon 6
Otpornici 7
Električne šeme 8
Arduino softver 9
Preuzimanje i podešavanje Arduino IDE-a 10
Povezivanje Arduina sa računarom 11
Digitalni i analogni 17
Analogni izlaz 19
Digitalni ulaz 23
Analogni ulaz 27
Rezime 30

POGLAVLJE 2

Kreiranje veb servera 31

Šta je veb server? 31
Rutiranje 32
Šta je Node.js? 33
Upotreba interfejsa komandne linije 33
Podešavanje servera Node.js 37
Instaliranje servera Node.js 37
Kreiranje Node.js aplikacije 41
Struktura direktorijuma 42
Kreiranje veb stranice 51
Mehanizam za izradu šablona 51
Datoteka package.json i upravljanje verzijama 60
Kako funkcionišu socketi 66
Rezime 70

POGLAVLJE 3

Arduino za Front End (Deo I) 71

Uvod u serijski port....72
Pronalaženje serijskog porta....73
Serijski podaci i Arduino....73
Bodna brzina....74
Upotreba podataka na Front Endu....81
Biblioteka SerialPort....81
Preuzimanje biblioteke SerialPort....82
Rezime....91

POGLAVLJE 4

Uvod u kreiranje veb sadržaja 93

HTML....93
HTML elementi....94
HTML atributi....98
Ugnežđeni elementi....99
Objektni model dokumenta....101
CSS....102
CSS selektori....106
Pravila kaskadiranja....107
Model okvira....108
Raspored prikaza....109
Flexbox....110
Boja....116
RGB....116
HSL....116
Skalabilni grafički vektor (SVG)....117
SVG skaliranje....119
Viewbox....120
Računarsko programiranje....120
Promenljive....120
Operatori....121
Tipovi....121
Iskazi....123
Izrazi....123
Strukture podataka....123
Uslovni iskazi....124
Petlje....125
Funkcije....126
Oblast važenja....127
Rezime....127

POGLAVLJE 5

Front End za Arduino 129

Aplikacije....129
LED veb aplikacija....130

Još malo o Flexboxu 138
JPodešavanje LED-a 145
LCD veb aplikacija 149
Kreirajte server 150
Podešavanje LCD-a 157
Rezime 162

POGLAVLJE 6

Arduino za Front End (deo II) 163

Digitalni i analogni signali 163
Aplikacija 165
Node.js aplikacija 171
Proširenje aplikacije 189
Vizuelizacija podataka na Arduinu 195
Rezime 202

POGLAVLJE 7

Vizuelizacija podataka 203

Uvod u biblioteku D3.js 203
Kako funkcioniše biblioteka D3.js 204
Ulančavanje metoda 213
Vizuelizacija podataka iz Arduina pomoću biblioteke D3.js 213
Sređivanje koda 224
Šablon modula za prikazivanje 224
Razdvajanje vizuelizacije podataka 225
Rezime 232

POGLAVLJE 8

Kreiranje veb komandne table 233

Veb komandna tabla 233
Osnove vizuelizacije podataka 234
Tipovi podataka 236
Označavanje vizuelizacije 237
Boje 237
Senzori 237
Senzor za temperaturu i vlagu DHT11 238
Fotootpornik 238
Uvoz biblioteke 238
Dodavanje dnevnih vrednosti 266
Rezime 278

POGLAVLJE 9

Fizička vizuelizacija podataka pomoću Live Data prikaza 279

API 279
USGS API 280
Pribavljanje podataka sa spoljnog servera 282
Povratni pozivi i promisi 282

Statusni kodovi odziva zahteva ... 285
Node.JS aplikacija ... 286
Funkcije setTimeoute i setInterval ... 287
GeoJSON objekat ... 288
Arduino komponente ... 295
Piezo zujalica ... 295
Rezime ... 307

POGLAVLJE 10

Kreiranje upravljača za igre 309

Animacija ... 309
HTML5 elemenat canvas ... 310
CSS animacija ... 310
3D na Vebu ... 310
WebGL ... 311
3D prostor ... 311
3D mrežice ... 313
Programi za senčenje ... 313
Three.js ... 315
Three vektori ... 315
Igra ... 321
Veb aplikacija ... 325
Izrada igre ... 328
Rezime ... 367

DODATAK A

Arduino zajednica i komponente 369

Arduino zajednica ... 369
Arduino komponente ... 370

DODATAK B

Još malo Front End programiranja 373

JavaScript ... 374
ES6 i starije verzije ... 374
JavaScript radni okviri ... 377
Baze podataka ... 378
Node.js mehanizmi za izradu šablona ... 379
SerialPort ... 379
CSS ... 380
Flexbox ... 380
CSS mrežica ... 380
Vizuelizacija podataka ... 380
Biblioteke za vizuelizaciju podataka ... 381
Resursi za vizuelizaciju podataka ... 382
Mape ... 383
Boja ... 383

INDEKS 385

UVOD

Mogućnost kreiranja fizičkih interfejsa za veb stranice otvara nove oblasti za inovacije i kreativnost, pa možete da razmišljate o vašem projektu kao o dva različita medija. Možete da kreirate interaktivne displeje i igre, da podesite IoT komponente, a zatim da prikupljate i obrađujete podatke, tj. možete da izrazite vaše ideje na nove načine.

Pomoću Arduina možete na odličan način da kreirate fizičke interfejse. On je namenjen da bude jednostavna elektronska platforma koja omogućava da priključite elektronske komponente koju mogu da šalju i primaju podatke. Cenjena Arduino zajednica može vam pomoći i biti kreativna ukoliko vam je potrebna pomoć.

Elektronske komponente mogu biti ulazi ili izlazi. Arduino ploče se mogu koristiti za reprodukovanje muzike, kreiranje svetlosnih efekata, za pokretanje pećnica i robota, u umetnosti itd. Postoji širok spektar komponenata, uključujući tastere, motore, potenciometre, senzore i zujalice, koje se mogu priključiti na Arduino.

Arduino ploče mogu da šalju i primaju podatke sa veb servera. To znači da možete da kontrolišete elemente na veb stranicama pomoću fizičkih komponenata i da koristite te komponente za prikaz informacija sa vaših veb stranica ili iz podataka na Internetu.

Da biste sve ovo uradili, potrebno je da razumete elektroniku i programiranje. Ova knjiga nije uvod u programiranje, niti elektroniku, ali obezbediće vam informacije koje su potrebne da bi Arduino komunicirao sa veb serverom.

Koristićete dva programska jezika – JavaScript za veb programiranje i Arduino programski jezik, koji je skup C i C++ funkcija.

Ova knjiga je kombinacija praktičnog i teorijskog. Ona „govori“ o tome kako neke komponente funkcionišu i čemu služe, a vežbe će vam pomoći da izgradite poverenje i naučite mnogo štošta. Upoznaćete različite tehnike programskog jezika, uključujući način izrade veb servera, vizuelizaciju podataka i 3D animaciju. Obezbedićete osnovu za razumevanje koncepata koji se koriste u ovoj knjizi i polaznu tačku za kreiranje sopstvenih projekata.

Ako ste zainteresovani za istraživanje beskonačnih mogućnosti fizičkog i digitalnog, ova knjiga je za vas. Ona će vam obezbediti uvid u način kako IoT i povezani uređaji

funkcionišu i omogućiće vam da pronađete nove načine za komunikaciju sa vašom publikom. Knjiga „pokriva“ elektroniku i programiranje koji će vam biti potrebni da biste počeli da izrađujete fizičke interfejse za veb stranice.

Opis knjige

U Poglavlju 1, „Arduino, kola i komponente“, predstavićemo elektronske osnove koje će vam biti potrebne u knjizi, uključujući Arduino hardver i softver, električnu energiju u kolima, povezivanje komponenata i četiri osnovna Arduino kola – analogni ulaz i izlaz i digitalni ulaz i izlaz.

U Poglavlju 2, „Kreiranje veb servera“, upoznaćete veb tehnologije, način podešavanja Node.js servera i slanje podataka na veb stranice pomoću API-ja WebSocket.

U Poglavlju 3, „Arduino za Front End (deo I)“, spojeno je znanje iz prva dva poglavlja, tako da možete da počnete da šaljete podatke sa Arduina na veb stranice.

U Poglavlju 4, „Uvod u kreiranje veb sadržaja“, naučićete kako se formiraju veb stranice pomoću HTML-a, CSS-a, SVG-a i osnovnog JavaScripta.

U Poglavlju 5, „Front End za Arduino“, predstavićemo tehnike koje su potrebne za slanje podataka sa Node.js servera na Arduino i načine za prikaz podataka pomoću elektronskih komponenata. Za prikaz podataka koristićete LED-ove i LCD-ove.

Poglavlje 6, „Arduino za Front End (deo II)“, je nastavak Poglavlja 3, uz detaljniji pregled načina interakcije komponenata sa elementima na veb stranicama. Prikazaćemo metričke jedinice koje je prikupio Arduino.

Poglavlje 7, „Vizuelizacija podataka“, je uvod u JavaScript biblioteku D3.js, koja omogućava da kreirate vizuelizaciju podataka na veb stranicama. Nastavićete da radite vežbe iz Poglavlja 6, tako što ćete dodavati trakasti grafikon na veb stranice.

U Poglavlju 8, „Kreiranje veb komandne table“, prvo razmatramo osnove vizuelizacije podataka, a zatim koristimo senzore da bismo prikupili podatke koji se prikazuju na veb komandnoj tabli.

Poglavlje 9, „Fizička vizuelizacija podataka pomoću Live Data prikaza“, je uvod u veb API-je i u način na koji se oni mogu koristiti da bismo pribavili podatke sa spoljnih veb servera. Kao primer su upotrebljeni podaci o zemljotresu, a za prikaz podataka korišćeni su LCD, LED i Piezo.

Poglavlje 10, „Izrada upravljača za igre“, omogućava da pomoću JavaScript biblioteke Three.js kreirate jednostavnu igru, koja se može kontrolisati pomoću fizičkog upravljača za igre. Ovo poglavlje „pokriva“ osnove 3D-a na Vebu i način na koji možete da kontrolišete 3D objekte pomoću džojstika koji je priključen na Arduinu.

Početak rada

Za ovu knjigu su potrebni softver i hardver. Pokušali smo da u celoj knjizi upotrebimo elektronske komponente koje su sadržane u Arduino početnom kompletu. Međutim, u nekoliko poglavlja upotreba tih komponenata nije bila moguća, pa smo upotrebili komponente koje su jeftinije i najčešće dostupne. U Dodatku A su navedeni neki dobavljači Arduino komponenata.

U ovoj knjizi ćete koristiti veliki broj JavaScript biblioteka koje se konstantno ažuriraju. Upotrebljen je isti broj verzija biblioteka u čitavoj knjizi i navedena je lista broja verzija koje će vam biti potrebne. Iako ovo nisu najnovije verzije biblioteka, one funkcionišu zajedno. Pošto različite osobe pišu biblioteke, ažuriranje jedne biblioteke može da utiče na rad druge.

Potreban vam je uređivač teksta da biste napisali kod, a sa Interneta možete preuzeti brojne uređivače, kao što su Sublime Text, Atom i Visual Studio. Različiti programeri koriste različite uređivače, a neki programeri su, pak, više usmereni ka određenim programskim jezicima.

Vežbe su testirane na Mac-u i PC-ju. One su u potpunosti testirane na operativnim sistemima MacOS Sierra verzija 10.12.5 i Windows 10 Home, a trebalo bi da funkcionišu i na operativnom sistemu Mac OS X verzija 10.8.5. Vežbe su testirane i na veb pregledačima Chrome i Firefox. Na Macu one su testirane u pregledaču Chrome 49.0.2623.112, a na PC-ju u pregledaču Chrome verzija 63.0.3239.132. Vežbe bi trebalo da funkcionišu i u pregledačima Firefox 57.0.4 i Firefox 45.9.0.

Verzija JavaScripta koja je objavljena 2015. godine sadrži osnovne izmene jezika, uključujući nove funkcije i promene u sintaksi. Verzija je prvo nazvana ECMAScript 6, odnosno ES6, a kasnije ES2015. Ova knjiga je zasnovana na ES5 verzijama JavaScripta i ne sadrži novu sintaksu ili funkcije iz verzije ES6 ili novije, zato što ne podržavaju svi pretraživači nove verzije JavaScripta na isti način, a potrebno je koristiti dodatne biblioteke. U ovoj knjizi ćete naučiti osnove JavaScripta i njegov način rada. U Dodatku B navedeni su neki detalji o funkcijama koje su dodate u novijim verzijama JavaScripta.

Sada je vreme da počnemo da koristimo Arduino. Poglavlje 1 će vas uvesti u Arduino i neka osnovna strujna kola koja se koriste u čitavoj knjizi.

POGLAVLJE **1**

ARDUINO, KOLA I KOMPONENTE

Ovo poglavlje je uvod u elektroniku za Arduino. U njemu je objašnjeno kako se podešava Arduino i kako struja teče kroz kola i komponente. Do kraja ovog poglavlja koristićete neke osnovne komponente i kreiraćete četiri osnovna kola – analogni ulaz i izlaz i digitalni ulaz i izlaz.

Arduino

Arduino omogućava da kreirate elektronske komponente. On je kolekcija hardvera i softvera otvorenog koda, koja omogućava da priključite i kontrolišete druge komponente da biste kreirali strujno kolo. Projekti, kao što su sistem za automatsko navodnjavanje, pećnica za pečenje pice ili autići kontrolisani daljinski, mogu se izraditi pomoću Arduina. Kada koristite Arduino za projekte, potrebno je da uradite sledeće:

- da povežete komponente sa Arduinom
- da napišete program za kontrolu komponenata
- da proverite da li je program pravilno napisan
- da otpremite program na Arduino

Arduino treba da povežete sa računarom pomoću USB priključka da biste otpremili program na njemu. Programi za Arduino se nazivaju skice. Kada se skica otpremi, skladišti se u mikrokontroleru i ostaje u njemu dok se ne otpremi druga skica. Kada je nova skica učitana, stara više nije dostupna.

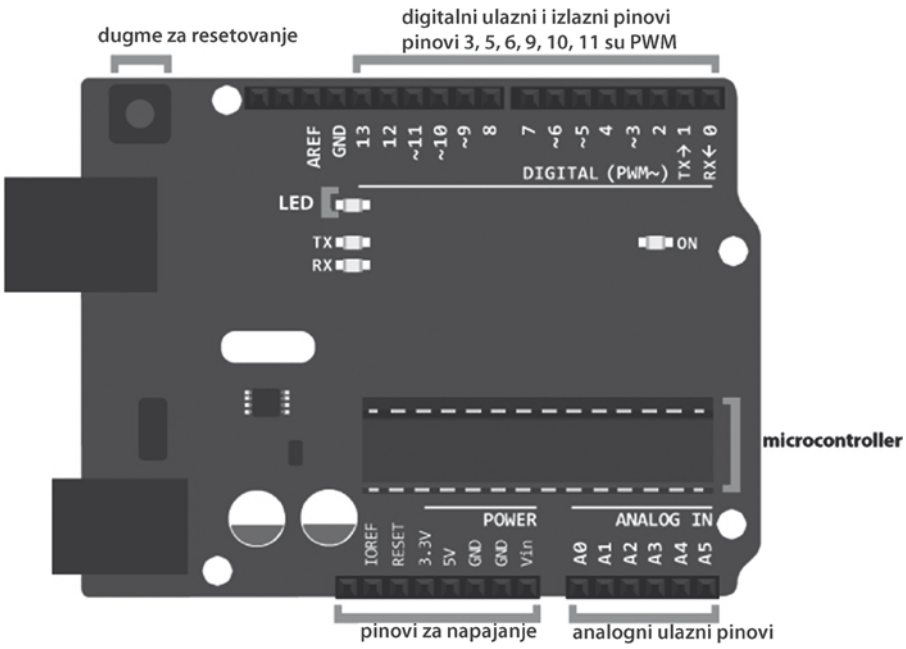
Nakon što otpremite skicu, možete da isključite Arduino iz računara. Ako je Arduino priključen na još jedan izvor napajanja, program se i dalje izvršava.

NAPOMENA

Nakon što je skica otpremljena na Arduino, neće biti prikazana na način na koji ste je napisali. Ne možete da vratite skicu sa Arduina u obliku koji se može čitati kao originalna skica, pa memorišite originalni kod ako želite da zadržite originalnu skicu.

Arduino hardver

Arduino ploča je izrađena od velikog broja komponenata, uključujući mikrokontroler, digitalne i analogne pinove, pinove za napajanje, otpornike, diodu, kondenzator i LED. Na slici 1-1 prikazan je Arduino Uno.



Slika 1-1 Arduino Uno

Mikrokontroler sadrži centralnu procesorsku jedinicu (CPU – Central Processing Unit). On skladišti učitane skice i obrađuje i usmerava komande.

Digitalni i analogni pinovi se koriste za slanje i prijem digitalnih i analognih podataka.

Arduino sadrži i serijski interfejs koji mu omogućava da šalje podatke na računar pomoću serijskog porta – to je način na koji ćemo u ovoj knjizi da šaljemo podatke na računar i sa njega.

Električna energija

Pomoću Arduina možete da kreirate elektronska kola koja napajaju priključene komponente na njemu. Žice napravljene od provodnih materijala povezuju komponente koje omogućavaju da struja teče kroz te žice.

Električna energija je kretanje elektrona kroz provodni materijal. U provodnim materijalima elektroni mogu lako da se kreću između atoma, ali ne i u neprovodnim materijalima.

Atomi se sastoje od protona, neurona i elektrona. U centru atoma su nukleus, protoni i neutroni, a elektroni se nalaze sa spoljašnje strane. Protoni imaju pozitivan naboj, a elektroni negativan. Ova dva naboja se privlače. Elektroni su u „orbiti“ oko nukleusa. U neprovodnim materijalima, kao što su drvo ili porculan, elektroni se teško kreću - čvrsto su „vezani“ za atom. U provodnim materijalima, kao što su bakar i drugi metali, postoje elektroni koji su sasvim labavo „vezani“ za atome, tako da mogu lako da se kreću. Ovi elektroni se nazivaju valentni, a nalaze se na spoljnoj ivici atoma.

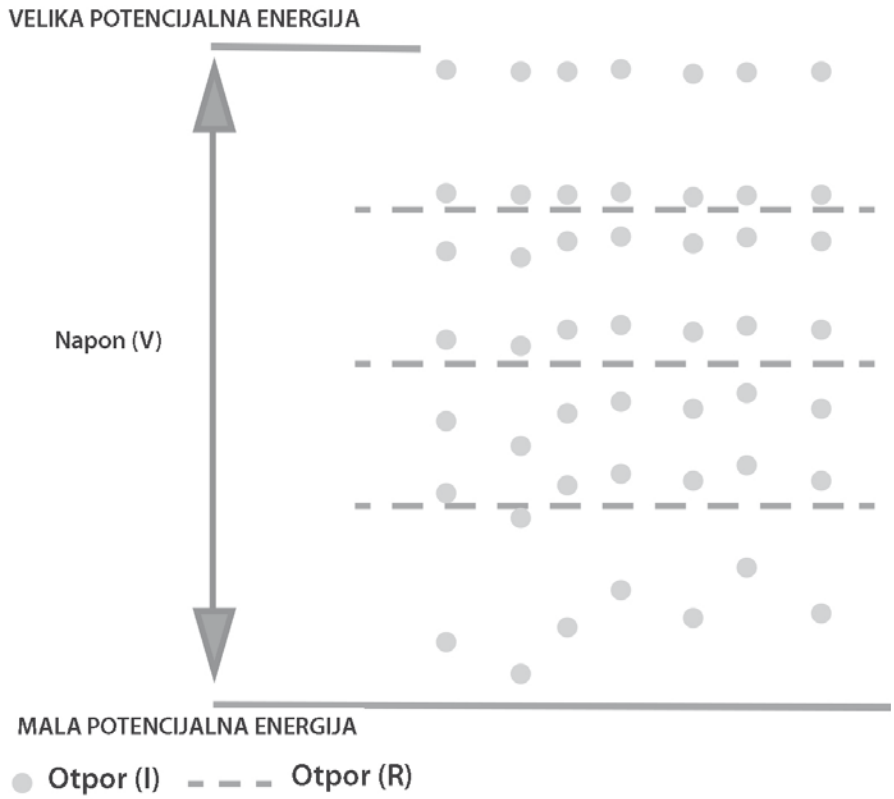
Elektroni se kreću oko strujnog kola, od negativnog prema pozitivnom polu. Kada je električna energija otkrivena, smatralo se da se elektroni kreću od pozitivnog prema negativnom polu, tako da se, po konvenciji, strujna kola često iscrtavaju od pozitivnog prema negativnom polu – od pozitivnog pola do uzemljenja (GND). U kolima koja su predstavljena u ovoj knjizi struja teče u jednom smeru, pa se naziva jednosmerna (DC), a u naizmeničnoj struji (AC) smer se menja određeni broj puta u sekundi.

Da bi se elektroni u provodnom materijalu počeli kretati, potreban im je napon. Napon je razlika između veće i manje potencijalne energije u kolu. Elektroni se kreću od veće potencijalne energije prema manjoj – od pozitivnog pola do zemlje.

Postoji nekoliko načina na koji se proizvodi napon. U bateriji se proizvodi hemijskom reakcijom. Elektroni se nakupljaju na negativni kraj baterije. Kada se poveže pozitivan kraj baterije, negativni elektroni su privučeni pozitivnim polom, od veće potencijalne energije prema manjoj. To uzrokuje „guranje“ elektrona na žicu - elektroni se kreću duž žice.

Struja je količina elektrona u sekundi koja prolazi kroz određenu tačku. Ona se izražava u amperima. Svaka komponenta koja je priključena na kolo koristi deo struje, a zatim pretvara tu struju u drugi oblik energije, kao što su svetlost ili zvuk. Komponente u kolu koriste svu energiju iz tog kola.

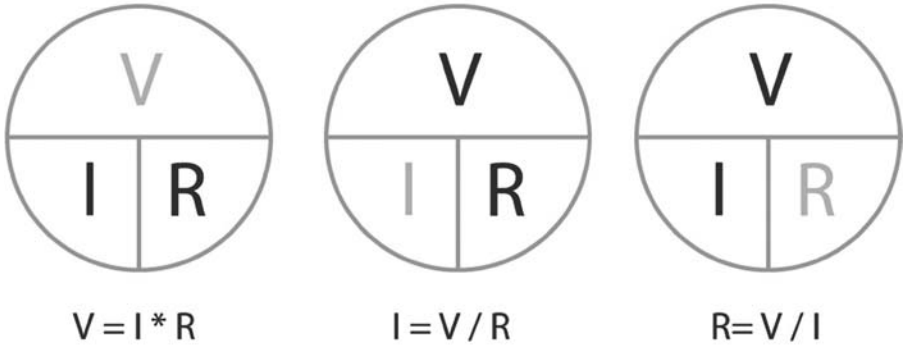
U strujnom kolu postoji otpor. Otpor je mera kojom se meri stepen otpora nekog materijala prolasku struje. On je poput prepreke na „putanji" struje. Izražava se u omima i označava se simbolom Ω. Električna energija će uvek „izabrati" najjednostavniji način za protok – putanju najmanjeg otpora. Na slici 1-2 je prikazana veza između napona, struje i otpora.



Slika 1-2 Interpretacija napona, struje i otpora

Omov zakon

Fizičar i matematičar Georg Simon Om je otkrio vezu između napona, struje i otpora, koja se naziva Omov zakon. Prema ovom zakonu, napon je jednak amperima pomnoženim sa otporom, a piše se kao V = I * R, gde je V volt, I je struja, a R je otpor. Pomoću ove formule možete da izračunate i otpor R = V / I i struju I = V / R, kao što je prikazano na slici 1-3.



Slika 1-3 Omov zakon

Komponente koje su priključene na kolo koriste deo energije iz tog kola i pretvaraju je u drugu vrstu energije, poput svetlosti ili zvuka. Svu energiju iz kola treba da iskoristi samo kolo. Ako se sva energija ne iskoristi, ona mora negde da se potroši da ne bi došlo do pregrevanja ili požara. Na primer, ako je na kolo priključen LED koji prima previše energije, svetlost će biti veoma jaka i LED može da pregori. Pomoću Omovog zakona možete izračunati otpor, ali umesto formule V = I * R, upotrebićete formulu R = V / I, koja označava da je otpor jednak naponu koji je podeljen sa strujom.

Otpornici

Otpornici su ključne komponente za kola, jer ograničavaju količinu struje na kolu. Otpornik ima određeni otpor na tok struje. Svaka komponenta ima maksimalnu količinu struje, izmerenu u amperima, koja se može bezbedno koristiti. Na primer, ako komponenta može da koristi maksimalno 0,023 ampera, što je 23 miliampera, a vaše kolo prima 5 V (volti), onda će biti potrebno da na kolo dodate otpornik od 220 oma da biste mogli bezbedno da koristite LED. Elektronske komponente će koristiti malo snage iz kola - ovo se naziva pad napona, koji treba da uzmete u obzir pri izračunavanju otpora. Na slici 1-4 prikazan je primer izračunavanja otpora pomoću Omovog zakona, sa različitim naponima i strujom.

otpor = napon / struja

otpor za komponentu koja koristi maksimalnih 0,023 ampera na kolu od 5 V

$$\text{otpor} = \frac{5\ \text{V}}{0{,}023\ \text{ampera}} = 217\ \text{oma}$$

koristi se otpornik od 220 oma, koji je najbliži otporniku od 217 oma

otpor za komponentu na kolu od 5 V, koja koristi maksimalnih 0,02 ampera i 2 V (pad napona od 2 volta)

$$\text{otpor} = \frac{5\ \text{V} - 2\text{V}}{0{,}02\ \text{ampera}} = 150\ \text{oma}$$

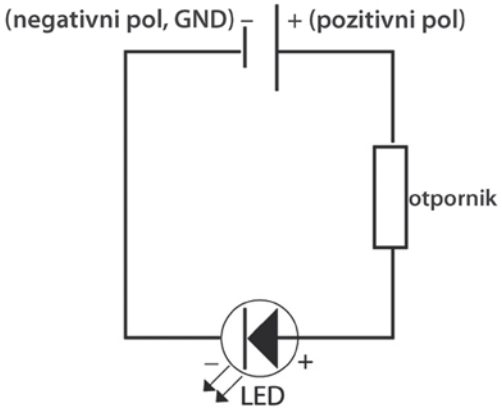
Slika 1-4 Formula za izračunavanje otpora koji je potreban za kolo

Za komponentu bi trebalo da postoji i tabela sa podacima, koja se može nalaziti na Internetu. U ovoj tabeli ćete pronaći informacije o padu napona i maksimalnim amperima, pa ćete moći da izračunate otpor koji je potreban za vaše kolo.

Otpornici imaju vrednost izmerenu u omima i kodne boje koje označavaju tu vrednost. Vrednost otpornika ukazuje na količinu struje koja se rasipa.

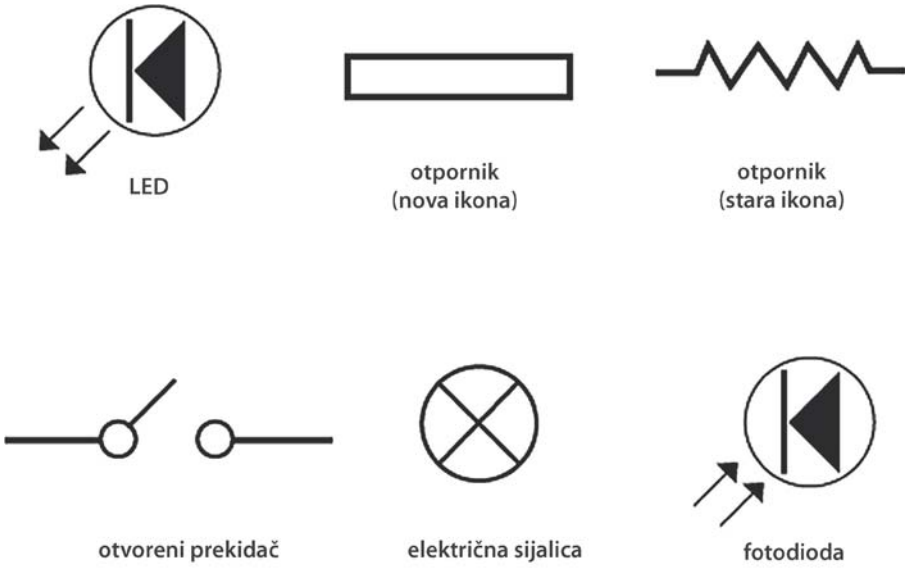
Električne šeme

Pomoću električnih šema prikazujemo kolo. Postoje skupovi ikona koje se koriste za prikazivanje komponenata na elektronskim kolima. U ovoj knjizi neće biti korišćene električne šeme za prikaz kola, već slike Arduina i komponenata. Na slici 1-5 možete da vidite kako izgleda električna šema. Ovo je električna šema za LED.



Slika 1-5 Električna šema za LED sa otpornikom

Na slici 1-6 su prikazane neke od ikona koje se mogu koristiti na električnoj šemi.



Slika 1-6 Neke ikone električne šeme

Arduino softver

Arduino ima ugrađeni programski jezik, koji je skup C i C++ funkcija. Arduino programi se nazivaju skice i imaju ekstenziju .ion. Arduino ima integrisano razvojno okruženje (IDE – Integrated Development Enviroment), koje sadrži uređivač i druge alatke koje pomažu pri pisanju i čitanju koda.

Preuzimanje i podešavanje Arduino IDE-a

Možete da instalirate Arduino IDE na vaš računar. Arduino IDE je dostupan na Internetu i lako ga možete preuzeti i instalirati. Pratite sledeće instrukcije u nastavku:

1. Pristupite stranici https://www.arduino.cc/en/Main/Software.
2. Odeljak „Download the Arduino IDE“ sadrži linkove za Mac i PC.

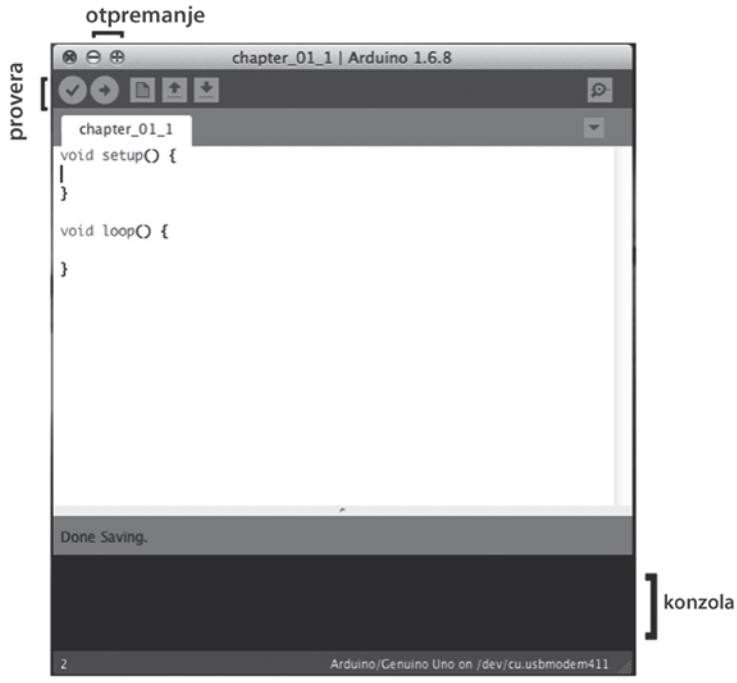
za Mac:

1. Kliknite na link OS X 10.7 Lion ili na noviju verziju i izaberite dugme JUST DOWNLOAD ili CONTRIBUTE & DOWNLOAD. Oba dugmeta se nalaze ispod slike.
2. Otpakujte preuzetu datoteku.
3. Pojaviće se Arduino ikona – samo kliknite na nju da biste pristupili IDE-u.

za PC:

1. Kliknite na link Windows Installer ili na Windows ZIP datoteku, u zavisnosti od administratorskih ovlašćenja koja imate na svom računaru. Izaberite JUST DOWNLOAD ili CONTRIBUTE & DOWNLOAD.
2. Otpakujte preuzetu datoteku.
3. Trebalo bi da pristupite IDE-u pomoću ikone.

Kada pristupite IDE-u, treba da se otvori nova skica. Na slici 1-7 je primer prozora za uređivanje.



Slika 1-7 Prozor za uređivanje Arduino IDE-a

Skica sadrži dve osnovne funkcije – setup i loop. Ikona znaka potvrde se nalazi na vrhu stranice. Na nju ćete kliknuti da biste proverili da li je vaš kod pravilno napisan. Ako se pojavi neki problem, biće prikazana poruka u konzoli u crvenoj boji. Na ikonu strelice ćete kliknuti kada želite da otpremite skicu na Arduino. U konzoli će biti prikazane poruke koje se odnose na vašu skicu. Kada verifikujete i otpremite skicu, biće prikazane sve greške u kodu i informacije o njima.

Povezivanje Arduina sa računarom

Potreban vam je USB 2.0 kabl B tipa da biste povezali Arduino Uno sa računarom. USB ćete koristiti za slanje podataka sa Arduina i na njega, ali i za napajanje. Za različite tipove Arduina se koriste različiti tipovi kablova.

Nakon što otvorite IDE dvostrukom klikom na ikonu i povežete Arduino sa računarom pomoću USB priključka, potrebno je da proverite meni sa alatkama da biste videli da li je Arduino naveden kao ploča i na koji port je priključen. U okviru tog menija izaberite Tools/Board, a zatim proverite da li je navedena ploča „Arduino/ Genuino Uno“. Ako nije navedena, izaberite Uno iz padajućeg menija.

Priključci

Povezaćete Arduino sa računarom pomoću jednog od USB priključka. Ovi priključci imaju broj, a u meniju sa alatkama Arduino IDE-a potrebno je da proverite padajući meni da biste izabrali priključak.

U meniju Tools/Port proverite da li je izabran USB priključak. Naziv USB priključka u meniju će se malo razlikovati, u zavisnosti od USB priključka koji ste priključili na Arduino i od Maca ili PC-ja. Na Macu priključak je naveden kao „dev/cu.usbmodem621 (Arduino/ Genuino Uno)“, a na PC-ju kao „COM4 (Arduino/Genuino Uno)“.

NAPIŠITE SKICU

Arduino ima ugrađeni LED, pa je najjednostavnija skica koju možete da napišete ona koja kontroliše i uključuje LED. Ovo je prva skica koju će napisati većina programera, a toliko je uobičajena da može biti već instalirana na Arduinu.

Arduino IDE sadrži veliki broj primera skica, a blink je jedan od njih. Ako u IDE-u izaberete File/Examples/01. Basics, videćete skicu blink. Možete da otvorite skicu iz IDE-a ili da je kopirate iz koda u listingu 1-1. Potrebno je da memorišete skicu pre nego što je otpremite.

Listing 1-1 blink.ino

```
 void setup() {
   pinMode(13, OUTPUT);


 }
void loop() { digitalWrite(13, HIGH);
   delay(1000); digitalWrite(13, LOW);
   delay(1000);
 }
```

Objašnjenje koda

Kad god otvorite novu skicu, uvek će biti prikazane funkcije setup i loop. Funkcija setup se poziva jednom kada se program prvi put pokrene. Funkcija loop će praviti petlju i izvršavati komande u programu, dok se Arduino napaja. U skici blink.ino se koristi LED na Arduinu. Ovaj LED se kontroliše pomoću te skice. On je digitalni pin 13 na Arduinu, pa u podešavanju koristite funkciju pinMode da biste ukazali da se pin 13 koristi kao izlaz, a zatim će se LED uključiti.

Petlja ima funkciju digitalWrite koja utvrđuje da li pin 13 treba da bude „visok“ ili „nizak“ (tj. uključen ili isključen). Funkcija delay zaustavlja petlju nakratko. Ako se ne koristi ta funkcija, petlja jednostavno prelazi na sledeću liniju koda, nakon što se završi u prethodnoj liniji. Kašnjenje se izražava u milisekundama. U tabeli 1-1 je detaljnije objašnjena skica blink.ino.

TABELA 1-1 OBJAŠNJENJE SKICE BLINK.INO

`void setup() {`
`  pinMode(13,`
`  OUTPUT);`
`}`
	setup je funkcija. Funkcije mogu da kao rezultat vrate vrednost, a u ovom programskom jeziku morate prilikom pisanja funkcije da deklarišete šta želite da ona vrati. Funkcije setup() i loop() ne vraćaju ništa, tako da se pre njih stavlja rezervisana reč void. Svaka funkcija se sastoji od naziva, iza kojeg slede zagrade. Zagrade mogu da budu prazne ili da sadrže argumente. Argumenat se prosleđuje funkciji kada se ta funkcija pozove. Vitičaste zagrade sadrže kod za funkciju.

`pinMode(13,` `OUTPUT);`	Arduino biblioteka sadrži veliki broj funkcija koje možete da koristite. Jedna od njih je pinMode, a druga je digitalWrite. Funkciju koristite tako što je pozivate. Da biste je pozvali, napišite njen naziv, iza kojeg slede zagrade. Ako funkcija očekuje argumente, stavite ih unutar zagrada. Ako ne očekuje argumente, zagrade ostavite prazne. Poziv funkciji se završava znakom tačka-zarez, koji omogućava da kompajleru programa ukažete da je kraj poziva ili komande.

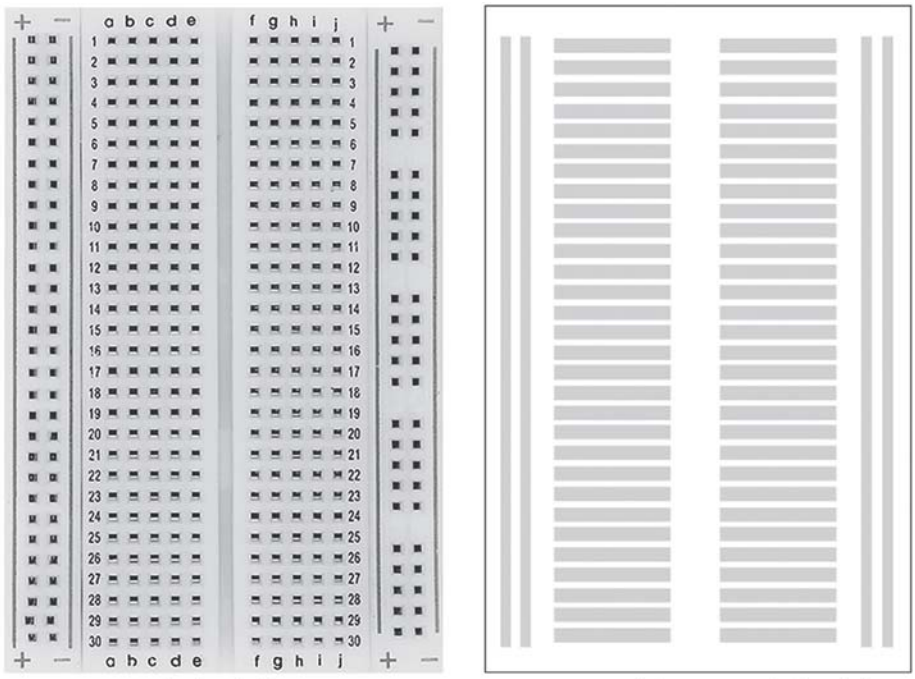
Možete da proverite da li je kod sintaksički ispravan, a zatim da ga otpremite na Arduino. Da biste to proverili, pritisnite ikonu znaka potvrde na vrhu koda, a zatim otpremite kod klikom na ikonu strelice. Obe ikone su prikazane na slici 1-7.

NAPOMENA

Arduino IDE je veoma osetljiv na sintaksičke greške; ako zaboravite znak „;“ na kraju komande ili napišete nešto malim slovima, a treba velikim, pojaviće se greška. IDE prilično dobro ukazuje gde se nalaze greške, koje se, obično, mogu lako korigovati.

Prototipska ploča

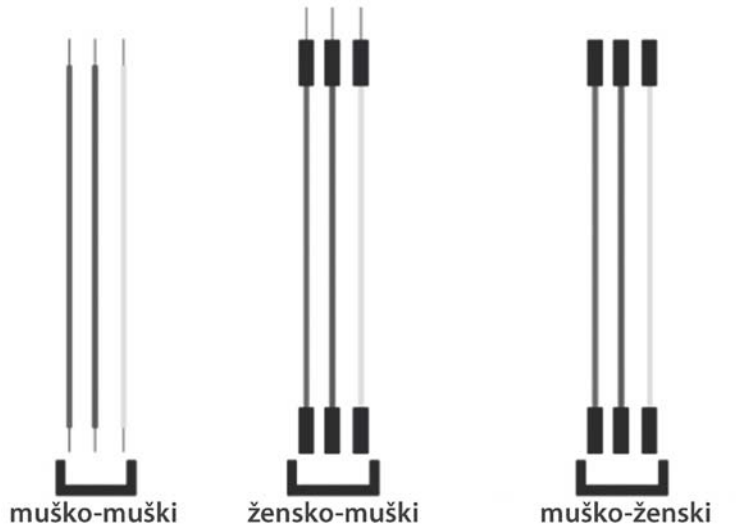
Prototipska ploča se koristi u elektronici za izradu prototipa. Pomoću nje možete da priključite komponente na Arduino bez lemljenja. Ona je napravljena od plastike i ima niz rupica za pinove komponenata i žice. Obično ima dve trake rupica sa obe strane za napajanje i uzemljenje. Unutar prototipske ploče su metalne trake koje su provodne. Žice i pinove povezujete sa ovim metalnim trakama da biste napravili strujno kolo. Prototipske ploče se prodaju u različitim veličinama. Na slici 1-8 prikazana je prototipska ploča.



Slika 1-8 Prototipska ploča

Kablovi

Arduino početni komplet se isporučuje sa veliki brojem kablova koje možete da koristite u većini projekata da biste napravili kolo između Arduina i komponenata, koristeći prototipsku ploču. Postoje neke komponente sa pinovima koje je teško uklopiti na prototipsku ploču. U Poglavlju 10 ćete praviti upravljač za igre, pa će vam možda biti potrebni kablovi sa različitom „glavom“. Postoje tri vrste kablova – muško-muški, žensko-muški i muško-ženski. Oni su prikazani na slici 1-9.



Slika 1-9 Kablovi

Digitalni i analogni

Na Arduinu možete da koristite digitalni ulaz i izlaz i analogni ulaz i izlaz. Digitalni ulaz ili izlaz mogu da imaju dva stanja – uključeno ili isključeno (visoko ili nisko). Analogni ulaz ili izlaz mogu da budu između 0 i 1023 kada se koristi 5 V. U sledećim vežbama u ovom poglavlju prikazani su primeri digitalnog izlaza i ulaza i analognog ulaza i izlaza.

OPREZ

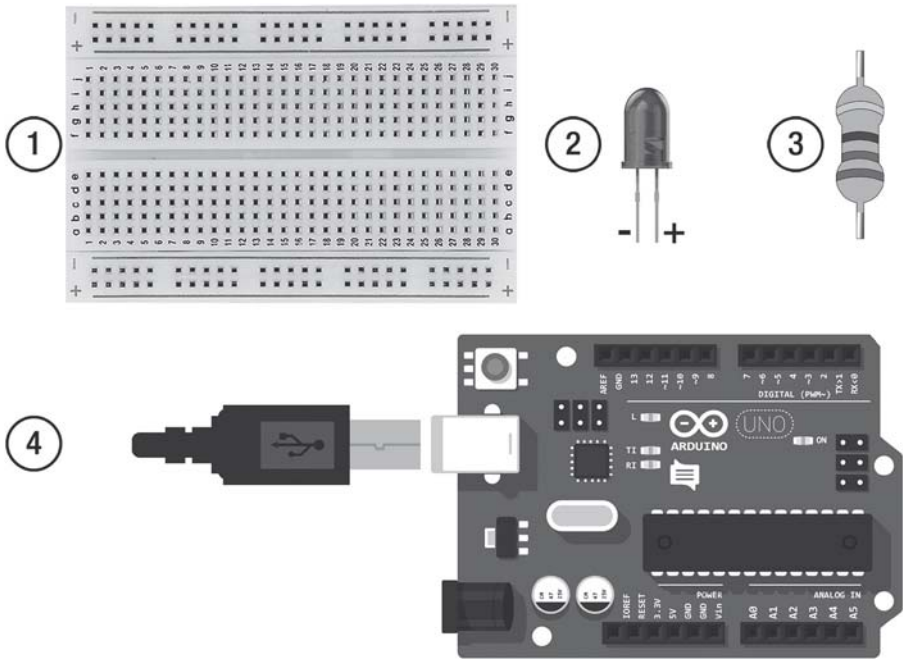
Morate da isključite Arduino iz računara kada povezujete komponente. Dok je Arduino povezan sa računarom, u njemu teče struja, pa može doći do električnog udara. Komponenta može iskočiti ili eksplodirati ukoliko je napajanje prejako, pa nećete želeti da budete suviše blizu ako se to dogodi.

DIGITALNI IZLAZ

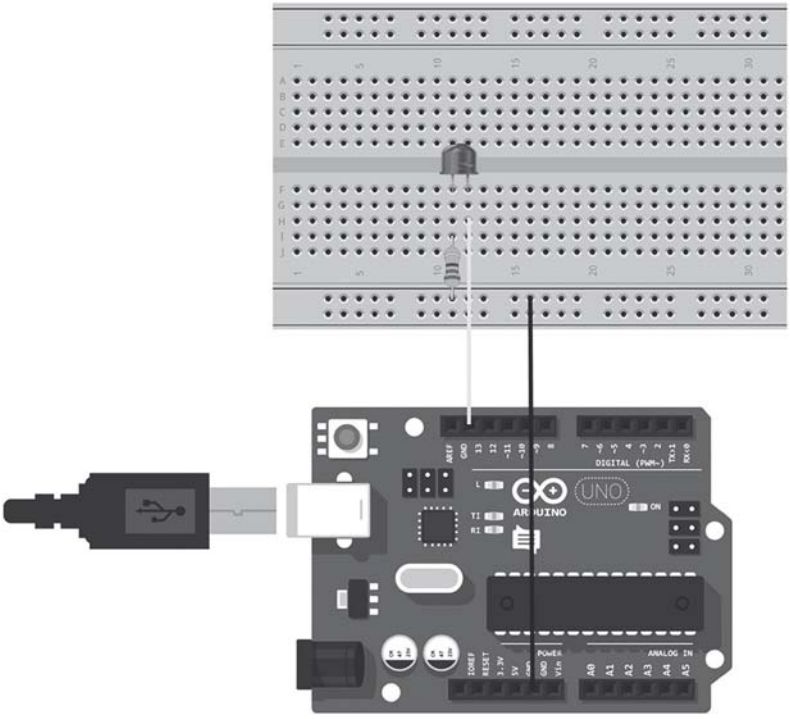
Komponenta koja se koristi za digitalni izlaz prima signal HIGH za uključeno i LOW za isključeno. U kodu u listingu 1-1 koristimo digitalni izlaz. U ovoj vežbi ćete koristiti isti kod kao iz listinga 1-1 blink.ino, koji je LED priključen na Arduino. Za ovu vežbu će vam biti potrebni:

- 1 Arduino Uno
- 1 LED
- 1 otpornik od 220 oma

Komponente su prikazane na slici 1-10, a podešavanje komponenata na slici 1-11. Pre nego što priključite komponente, isključite Arduino iz računara ili sve izvore napajanja. Dugačka „nogica“ Arduina je pozitivna, a kratka „nogica“ je negativna.



Slika 1-10 za vežbu digitalnog izlaza: 1. prototipska ploča, 2. LED, 3. otpornik od 220 oma, 4. Arduino Uno



Slika 1-11 Podešavanje komponenata za vežbu digitalnog izlaza

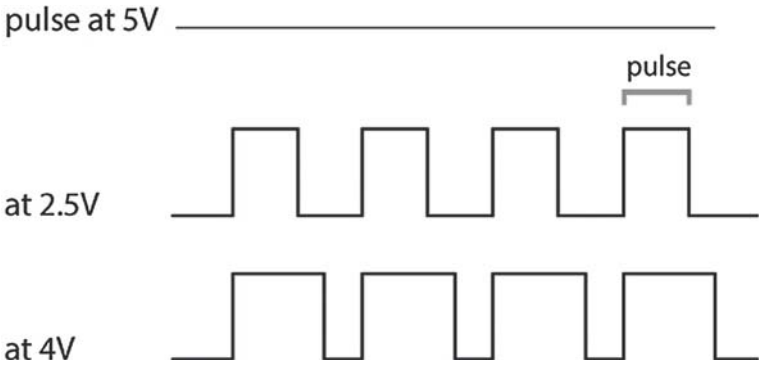
Ponovo priključite Arduino na računar pomoću USB-a. Ako je poslednji program koji ste otpremili blink.ino, treba da se vidi LED koji trepće. Ako poslednji program koji ste otpremili nije blink.ino, ponovo ga učitajte.

Analogni izlaz

Analogni izlaz i ulaz „proizvode“ niz brojeva koji se kreću gore-dole. Na Arduinu neki digitalni pinovi imaju simbol „~“ pored tih brojeva. Ovi pinovi se koriste za analogni izlaz, a na njima se primenjuje PWM (modulisanje širine impulsa).

Modulisanje širine impulsa

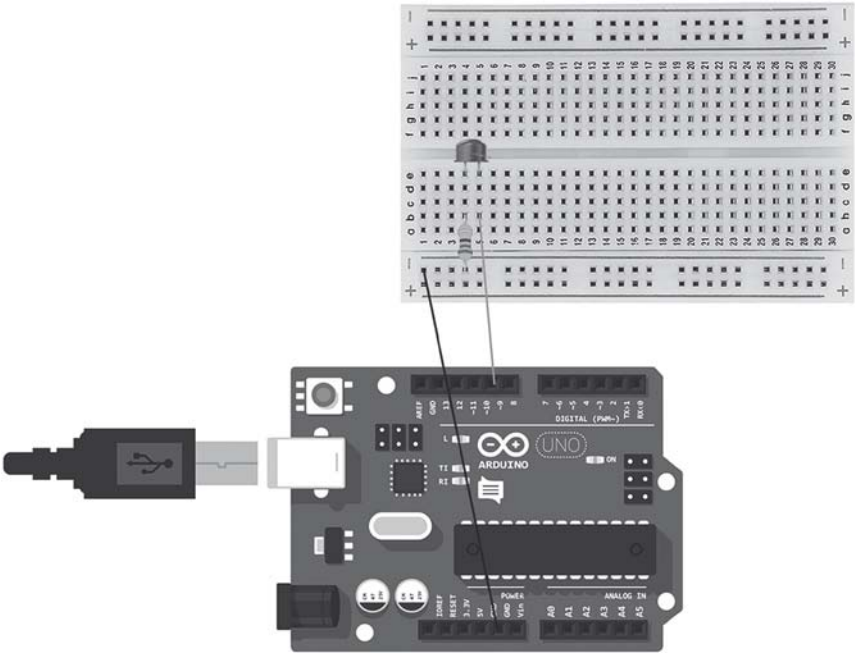
PWM se koristi za simulaciju analognog izlaza pomoću digitalnih pinova. Digitalni signal može da bude uključen ili isključen, a zatim se šalje impuls za uključenje. PWM simulira analogni sistem pomoću digitalnog signala, tako što menja dužinu impulsa – simulira impuls između 5 V i 0 V. Na slici 1-12 je prikazana širina impulsa za simulaciju različitih napona.



Slika 4-1 Modulisanje širine impulsa

ANALOGNI IZLAZ

Pošto analogni signal može da „proizvede“ niz brojeva, možete da izvršavate sve radnje postepeno. U ovoj vežbi LED će se postepeno uključivati pre nego što se isključi, a zatim će se ponovo postepeno uključivati. Komponente za ovu vežbu su iste kao na slici 1-10. Isključite Arduino, a zatim podesite komponente, kao što je prikazano na slici 1-13. LED se povezuje sa digitalnim pinom 9, pored kojeg se nalazi znak ~. Kreirajte novu skicu koja će imati naziv chapter_01_1.ino, a zatim kopirajte kod iz listinga 1-2 u tu skicu.



Slika 1-13 Podešavanje analognog ulaza

Listing 1-2 chapter_01_1.ino

```
int analogOutPin = 9;
int outputValue = 0;

void setup() {
  pinMode(analogOutPin, OUTPUT);
}

void loop() {
  if (outputValue >= 40){
     outputValue = 0;
  } else {
     outputValue = outputValue + 1;
  }
  analogWrite(analogOutPin, outputValue);
  delay(200);
}
```

Objašnjenje koda

Da biste postepeno uključili LED, potrebno je da dodelite vrednost koja povećava svaku petlju. Postoji veliki broj novih programskih koncepata u ovom kodu. Ne brinite mnogo ukoliko još uvek ne razumete sve koncepte, jer će u nekoliko sledećih poglavlja programiranje biti detaljnije objašnjeno. U tabeli 1-2 je dato detaljnije objašnjenje koda u skici chapter_01_1.ino.

TABELA 1-2 OBJAŠNJENA SKICA CHAPTER_01_1.INO

`int analogOutPin = 9;`	Pin 9 će se koristiti za LED. Uobičajena praksa je da se uskladišti ovaj broj u promenljivu koja se koristi u čitavom programu. To će vam olakšati da vidite šta broj predstavlja u čitavom programu, a moći ćete i da promenite broj pina u kodu ukoliko odlučite da koristite neki drugi broj.
`int outputValue = 0;`	Promenljiva sadrži vrednost za LED.

`if (outputValue >= 40){`
`  outputValue = 0;`
`} else {`
`  outputValue = output`
`  Value + 1;`
`}`
	Iskaz else proverava da li nešto ima vrednost true - ako je vrednost true, ovaj iskaz izvršava jedno, a ako nije, izvršava nešto drugo. U našem primeru proverava da li je vrednost promenljive outPutValue veća od 40 ili jednaka tom broju. Ako je veća, promenljiva sadrži vrednost 0 koja isključuje LED, a ako je jednaka tom broju, povećava se za 1, pa se osvetljenje LED-a pojačava.

`analogWrite(analogOutPin,` `outputValue);`	Funkcija analogWrite ima dva argumenta – broj pina i vrednost, a u našem primeru vrednost iz funkcije outputValue se šalje u komponentu koja je priključena na pin 9. Ovde se koristi komponenta LED koja će promeniti jačinu osvetljenja.

Otpremite skicu na Arduino – trebalo bi da vidite da se osvetljenje LED-a pojačava, a zatim da se isključuje.

Digitalni ulaz

Dobro kolo za prikaz digitalnog ulaza je dugme prekidača. Dugme prekidača je gore ili dole, tj. u jednom od dva stanja – pritisnuto ili nije pritisnuto. Ovo je još jedan koncept pod nazivom Input Pullup.

Postoji problem na Arduinu koji se odnosi na prekidač. Kada je prekidač otvoren, on ne zatvara kolo i ne postoji napon, pa Arduino ne zna ulaznu vrednost. Ta vrednost može da bude 0 ili 1. Pošto Arduino ne zna ulaznu vrednost, može da dâ neobične rezultate i da stvara buku, jer je ulazna vrednost nepoznata, a pokušaće i da doda neku vrednost. Ovaj problem se rešava pomoću pullup otpornika, koji podešavaju napon kada je prekidač otvoren.

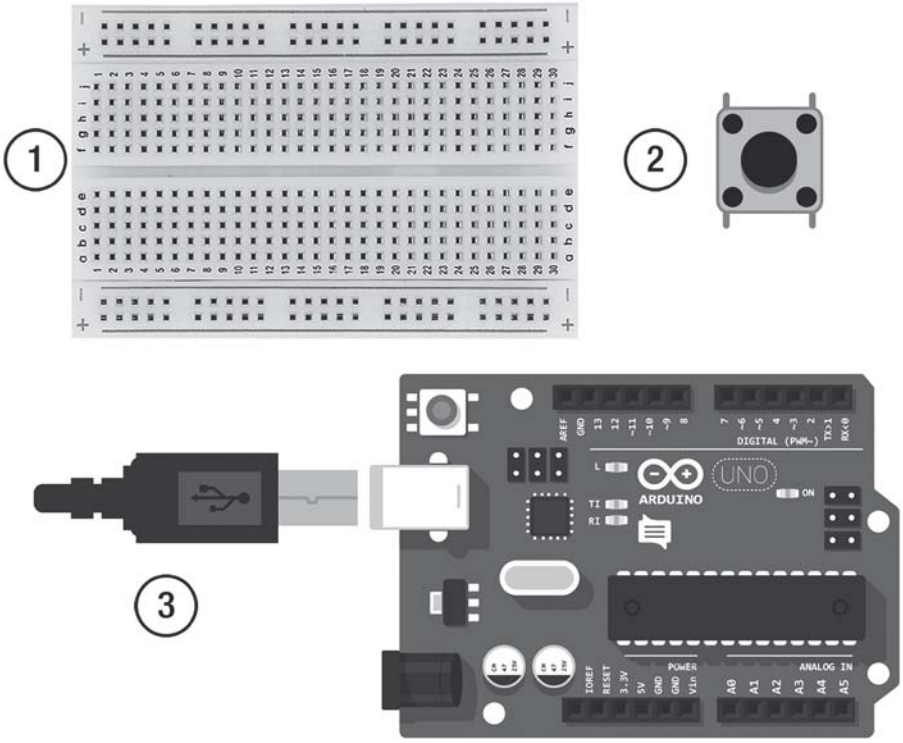
Pullup otpornici su ugrađeni u Arduino i možete im pristupiti pomoću funkcije pinMode(), koju ćete postaviti na vrednost INPUT_PULLUP, umesto samo na INPUT. Pin će očitati vrednost HIGH kada je prekidač otvoren, a LOW kada je pritisnut.

DIGITIALNI ULAZ

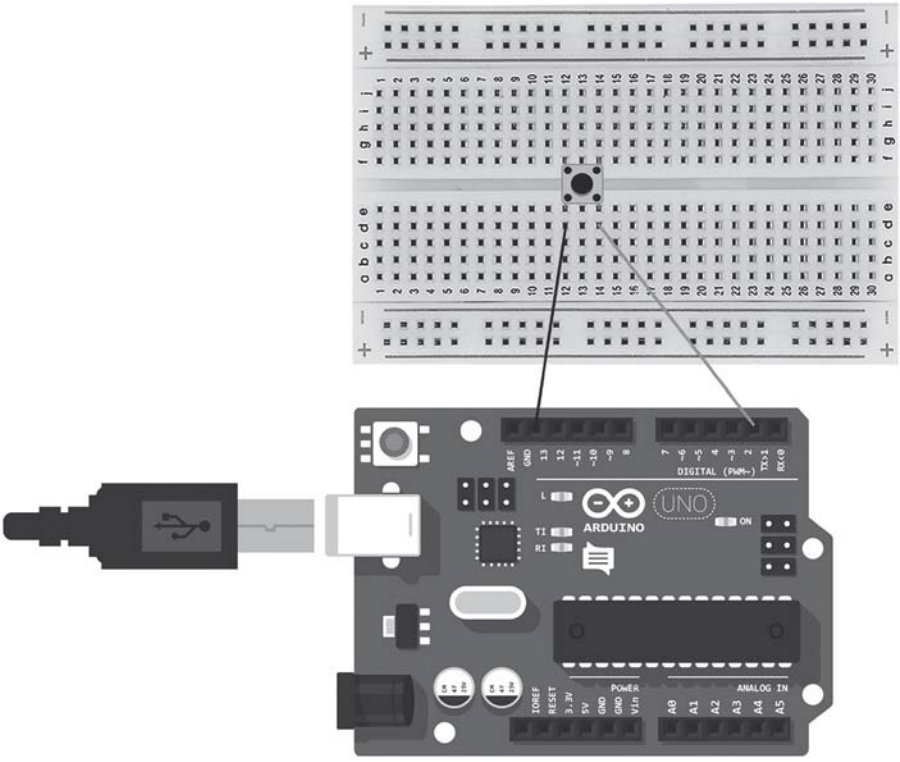
Za ovu vežbu koristićete dugme za uključivanje i isključivanje LED-a na Arduinu. Biće vam potrebni:

- 1 Arduino Uno
- 1 dugme prekidača

Nemojte zaboraviti da isključite Arduino iz računara kada menjate komponente. Prekidač se priključuje na prototipsku ploču. Jedan pin se priključuje na pin za uzemljenje, a drugi na digitalni pin 2. Na slici 1-14 su prikazane komponente koje su vam potrebne, a na slici 1-15 podešavanje Arduina.



Slika 1-14 Komponente za vežbu digitalnog ulaza: 1. prototipska ploča, 2. prekidač, 3. Arduino



Slika 1-15 Podeša*vanje komponenata za vežbu digitalnog ulaza*

Kreirajte novu skicu u Arduino IDE-u kojoj smo dodelili naziv chapter_01_2, a zatim je kopirajte u kod iz listinga 1-3. Verifikujte i otpremite kod na Arduino. Kada pritisnite dugme, LED na Arduinu bi trebalo da zasvetli.

Listing 1-3 chapter_01_2.ino

```
int buttonInput = 2;
int LEDOutput = 13;

void setup() {
pinMode(buttonInput, INPUT_PULLUP);
  pinMode(LEDOutput, OUTPUT);
}
```

```
void loop() {
     int sensorVal = digitalRead(buttonInput);
     if (sensorVal == HIGH) {
        digitalWrite(13, LOW);
     } else {
        digitalWrite(13, HIGH);
     }
   }
```

Objašnjenje koda

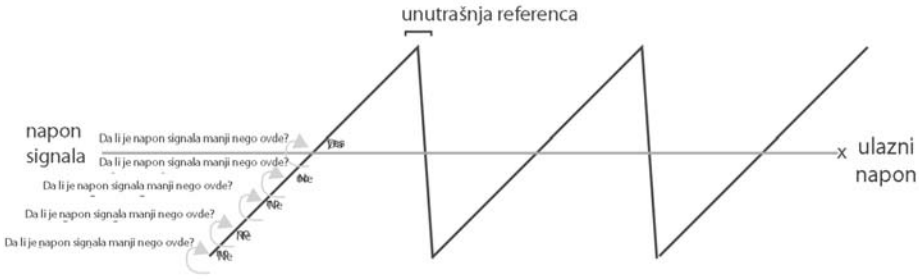
U tabeli 1-3 je detaljnije objašnjen kod za skicu chapter_01_2.ino.

TABELA 1-3 KOD ZA SKICU CHAPTER_01_2.INO

`pinMode(buttonInput, INPUT_PULLUP);`	Za dugme prekidača treba da koristite INPUT_PULLUP kao drugi argumenat u funkciji pinMode().
`int sensorVal = digitalRead(buttonInput);`	Vrednost prekidača se očitava u promenljivoj na svakoj petlji.
`if (sensorVal == HIGH) { digitalWrite(LEDOutput, LOW); } else { digitalWrite(LEDOutput, HIGH); }`	Iskaz if/else proverava da li je prekidač HIGH ili LOW. Pomoću pullup otpornika logika dugmeta će biti obrnuta. Ako je logika HIGH, znači da prekidač nije pritisnut, pa je LED isključen. Ako je LOW, prekidač je pritisnut, pa je LED uključen.

Analogni ulaz

Analogni ulaz se koristi sa komponentama, kao što su fotootpornici i potenciometri, koje daju različite vrednosti. Arduino Uno može da registruje vrednosti između 0 i 1023. Analogni ulaz šalje napon signala. Kada je napon signala primljen, on se upoređuje sa unutrašnjom referencom. Primer analognog ulaza je prikazan na slici 1-16.



Slika 1-16 Prikaz procesa analognog ulaza

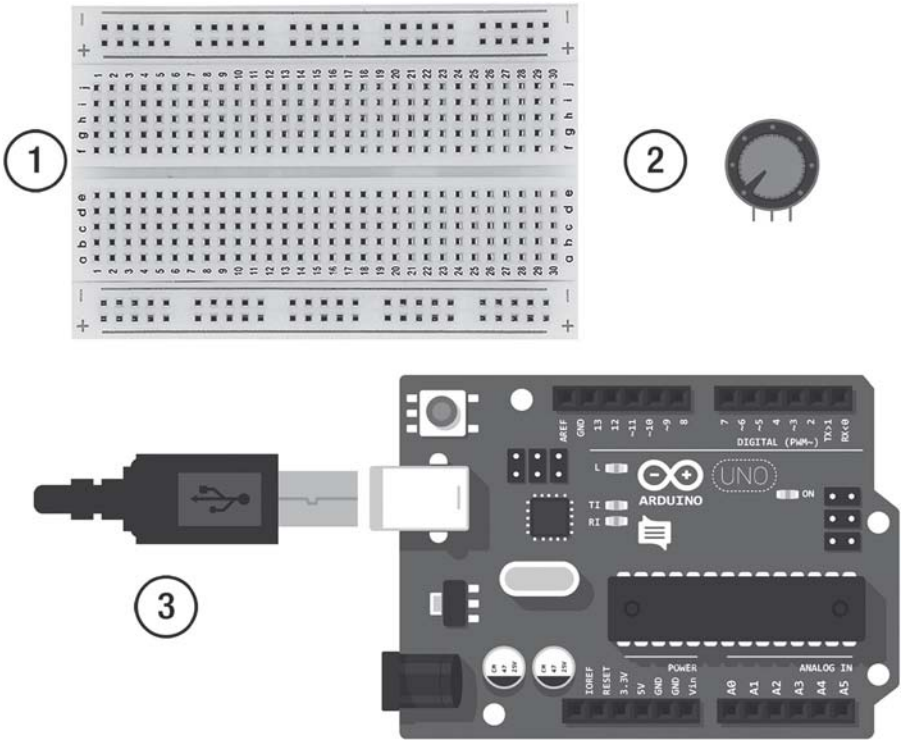
Kada je napon signala primljen, upoređuje se sa unutrašnjom referencom na višestrukim tačkama na liniji. Na primer, proverava se da li je ulaz veći od 0. Ako nije veći, proverava se da li je veći od sledećeg broja na referenci, a zatim se provera nastavlja, dok ulaz ne bude veći od 0. Tačka na kojoj je ulaz veći od 0 postaje broj ulaza.

ANALOGNI ULAZ

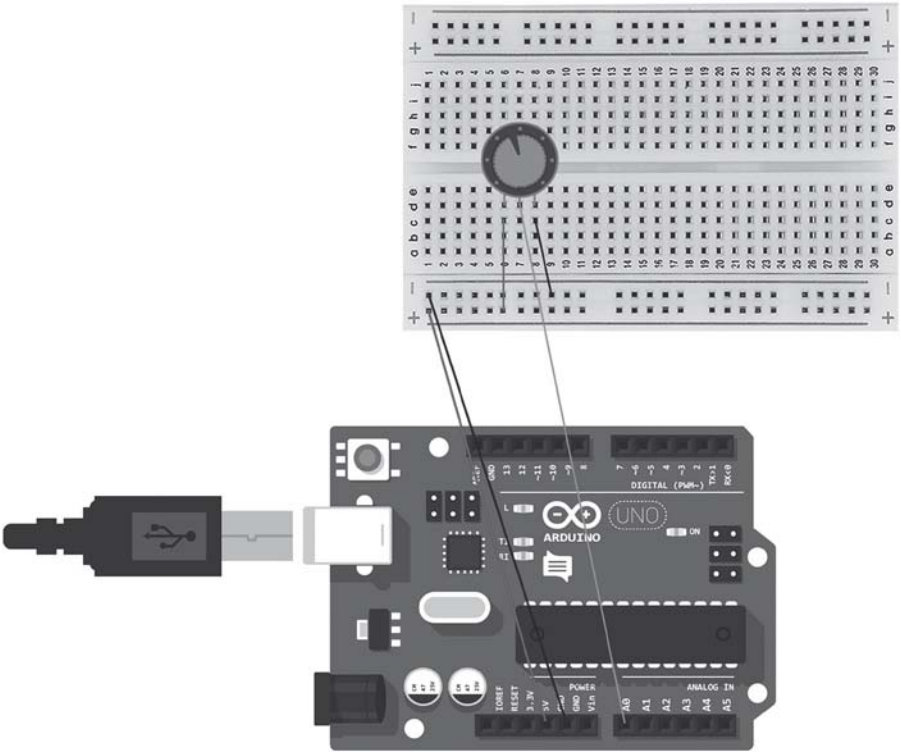
U ovoj vežbi se koristi potenciometar kao analogni ulaz. Kada se potenciometar okrene do pola, LED na Arduinu se uključuje i isključuje. Za ovu vežbu potrebne su sledeće komponente:

- 1 Arduino Uno
- 1 potenciometar

Na slici 1-17 prikazane su komponente, a na slici 1-18 podešavanje za Arduino.



Slika 1-17 Komponente za vežbu analognog ulaza: 1. prototipska ploča, 2. potenciometar, 3. Arduino Uno



Slika 1-18 Podešavanje komponenata za vežbu analognog ulaza

Otvorite novu skicu kojoj smo dodelili naziv chapter_01_3, a zatim u nju kopirajte kod iz listinga 1-4.

Listing 1-4 chapter_01_03.ino

```
int pinAnalogInput = A0;
int LEDOutput = 13;
int valueLight =  0;

void setup() {
  pinMode(LEDOutput, OUTPUT);
}
```

```
void loop() {
  valueLight = analogRead(pinAnalogInput);
  if (valueLight < 500) {
    digitalWrite(LEDOutput, LOW);
  } else {
    digitalWrite(LEDOutput, HIGH);
  }
  delay(500);
}
```

Verifikujte skicu i priključite USB nazad u računar da biste otpremili skicu na Arduino. Sada, kada uključite potenciometar, samo ga okrenite do pola, pa bi trebalo da se LED na Arduinu uključuje i isključuje.

Taj kod je veoma sličan kodovima u prethodnim skicama iz ovog poglavlja. Razlikuje se samo po tome što je promenljiva za analogni pin int pinAnalogInput = A0. Analogni ulaz „prolazi“ kroz pin A0.

Rezime

Ovo poglavlje je osnovni uvod u Arduino. Videli ste kako funkcionišu strujno kolo i analogni i digitalni ulaz i izlaz. Ovo su osnovni blokovi za Arduino, koji će biti nadgrađivani u ovoj knjizi. U sledećem poglavlju ćete da koristite JavaScript i da izradite veb server koji će moći da prima podatke poslate sa Arduina.