



UNIVERZITET U BANJOJ LUCI  
PRIRODNO-MATEMATIČKI FAKULTET



**SEMINARSKI RAD**  
**TEMA:Arduino tehnologija**

Predmet:Informacione tehnologije

Profesor:prof.dr Dragan Matić

Asistent:Viši asistent Milana Grbić

Maj,2020.god.

Radio:Nikola Đajić

## SADRŽAJ

<b>1. Uvod .....</b>	<b>3</b>
<b>2. Hardver .....</b>	<b>4</b>
<b>2.1. Mikrokontroleri .....</b>	<b>4</b>
<b>2.2. Najpopularniji Arduino hardver .....</b>	<b>5</b>
a) Arduino Uno.....	5
b) Arduino Mega 2560.....	6
<b>3. Softver .....</b>	<b>7</b>
<b>3.1. Osnovni prozor Arduino IDE programskog okruženja.....</b>	<b>8</b>
<b>3.2. Funkcije .....</b>	<b>11</b>
<b>4. Internet stvari (Internet of Things) .....</b>	<b>12</b>
<b>5. Zaključak.....</b>	<b>16</b>

## 1. Uvod

Arduino je osnovala italijanska kompanija SmartProjects 2005. koristeći 8-bitne mikrokontrolere Atmel AVR, da bi stvorili jednostavnu, malu i jeftinu platformu s kojom bi mogli lakše povezivati računare sa fizičkim svijetom.

Dizajneri Arduina su odlučili da ime dobije po imenu kafića u kojem su se sastajali kada su stvarali projekt.

Arduino je mala mikrokontrolerska pločica sa univerzalnom serijskom magistralom (engl. universal serial bus, USB) na koju povezujete svoj računar, i nizom utičnica u koje se mogu priključiti spoljne elektronske komponente kao što su motori, releji, svjetlosni senzori, laserske diode, zvučnici, mikrofoni i mnoge druge.

Kao i većina drugih elektrotehničkih komponenti, Arduino se sastoji od hardvera i softvera o kojima ćemo reći nešto više u nastavku rada.

Bitno je napomenuti da ukoliko je to potrebno, moguće je ugraditi i dodatni hardver odnosno zasebne štampane ploče (štitove) koji se montiraju na Arduino ploču i ostvaruju električni kontakt sa njom preko postojećih pinova-konektora. Ovi štitovi proširuju osnovne mogućnosti Arduino ploče.



*Logo Arduino tehnologije (slika 1.)*

## 2. Hardver

Trenutno ima više od 20 različitih verzija Arduino hardverskih sistema. Svaki od njih ima neke različite komponente, i samim tim imamo raznovrsnu primjenu Arduino tehnologija. Osnovni dio bilo kojeg Arduino hardverskog sistema je mikrokontroler.

### 2.1. Mikrokontroleri

Mikrokontroler je čip koji vrši obradu ulaznih informacija i daje izlazne signale na osnovu zadatih kriterijuma. Glavna uloga mikrokontrolera je upravljanje uređajima i procesima, a sastoji se od mikroprocesora, memorije, digitalnih i analognih ulaza i izlaza, digitalnih satova (tajmera), brojača (kauntera), oscilatora, komunikacionih sklopova (interfejsa).



*Mikrokontroler dsPIC30F4011 (slika 2.)*

Glavna razlika između mikroprocesora i mikrokontrolera je u tome što mikroprocesor da bi mogao da radi mora imati veze sa spoljašnjim svijetom (vanjskim komponentama), dok to mikrokontroleri imaju već integrisano u sebi.

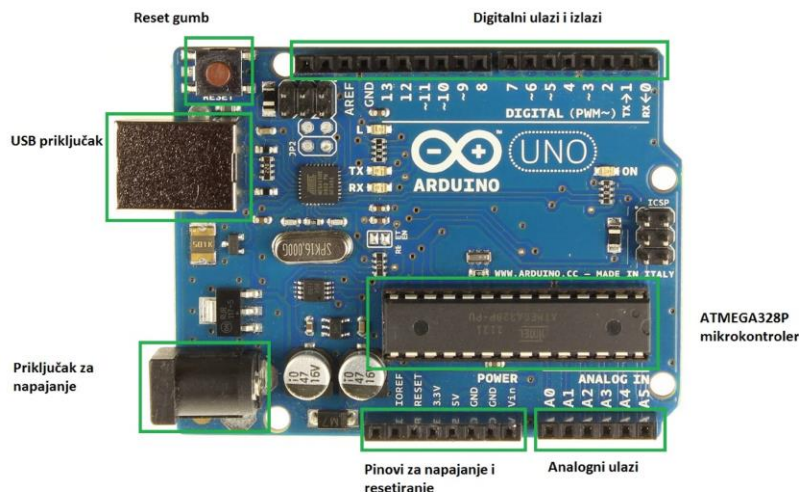
## 2.2. Najpopularniji Arduino hardver

### a) Arduino Uno

Arduino Uno predstavlja najbolji izbor za početnike zbog toga što se dosta informacija i pomoći o programiranju može naći na internetu, pored toga ova ploča je korištena više nego bilo koja druga.

Ovaj model sadrži 8-bitni ATmega328P mikrokontroler. Aktuelna verzija je opremljena USB interfejsom, zajedno sa 6 analognih ulaza i 14 digitalnih ulaza/izlaza opšte namjene.

Analogni i digitalni ulazno/izlazni priključci su izvedeni na konektore koji se nalaze na obodima pločice i preko kojih je moguće spajanje osnovne kontrolerske pločice sa različitim ekspanzionim pločicama, koje se nazivaju štitovi (engl. Shield).



*Arduino Uno ploča (slika 3.)*

Napajanje se vrši preko USB konektora ili nekog drugog izvora električne energije (baterije ili ispravljača napona 6-20V). USB konektor se takođe koristi za komunikaciju sa okolinom.

Ovaj model sadrži i reset dugme koje se koristi za pokretanje programa ispočetka ukoliko dođe do greške ili zastoja rada.

## **b) Arduino Mega 2560**

Arduino Mega2560 je mikrokontrolerska platforma bazirana na Atmel ATmega 2560 mikrokontroleru. Pored mikrokontrolera sadrži potrebno okruženje za njegov rad, komunikaciju i programiranje sa PC-om putem USB konektora.

Sastoji se od 54 digitalnih ulaz/izlaz pinova, 16 analognih ulaza. Mikrokontroler ATmega2560 ima 256 KB flash memorije koje se koristi za spremanje programskog koda.

Napajanje se vrši pomoću USB konektora ili nekog drugog izvora električne energije, kao i kod Arduino Uno modela.



*Arduino Mega 2560 pločica (slika 4.)*

Kompatibilan je sa većinom dodatnog hardvera (štitova) koji su dizajnirani za neke druge modele Arduina, uključujući i Arduino Uno.

Ovaj model takođe sadrži reset dugme.

### 3. Softver

Za početak je potrebno instalirati Arduino IDE softverski paket. Taj paket se nalazi na zvaničnoj stranici Arduina (<https://www.arduino.cc>) i jako je jednostavan za preuzeti i instalirati. Ovaj softver je open-source tipa i radi na svim platformama – Windows, Linux i Mac operativnom sistemu.

Budući da se koncept Arduina zasniva na otvorenom kodu (engl. open-source), kao i zahvaljujući činjenici da su razvojni alati jednostavni za upotrebu, ova platforma je vrlo brzo stekla široku popularnost i veliki broj korisnika širom svijeta. Vremenom, korisnici su razvili veliki broj primjera programa, kao i biblioteka korisnih funkcija, koje su stavili na raspolaganje drugim korisnicima, i samim tim olakšali početnicima rad.

Arduino IDE (Integrated Development Environment) je aplikacija koja je napisana u funkcijama iz programskih jezika C i C++. Koristi se za pisanje i učitavanje programa u kompatibilne Arduino pločice.

Kreatori Arduino platforme su programske aplikacije nazvali skicama (sketch), a pri snimanju na disk, automatski im se dodjeljuje ekstenzija **.ino**.

Najčešće korišćene biblioteke su uključene u samo razvojno okruženje, a ukoliko želimo da pristupimo nekim drugim bibliotekama sa već gotovim funkcijama to radimo korištenjem opcije *Sketch → Import Library*.

Programski jezik Arduino aplikacija koristi osnovne elemente jezika C i C++. Pisanje programa koji će se izvršavati na Arduino pločici podrazumijeva da korisnik mora da definiše dve funkcije, koje sačinjavaju izvršni program.

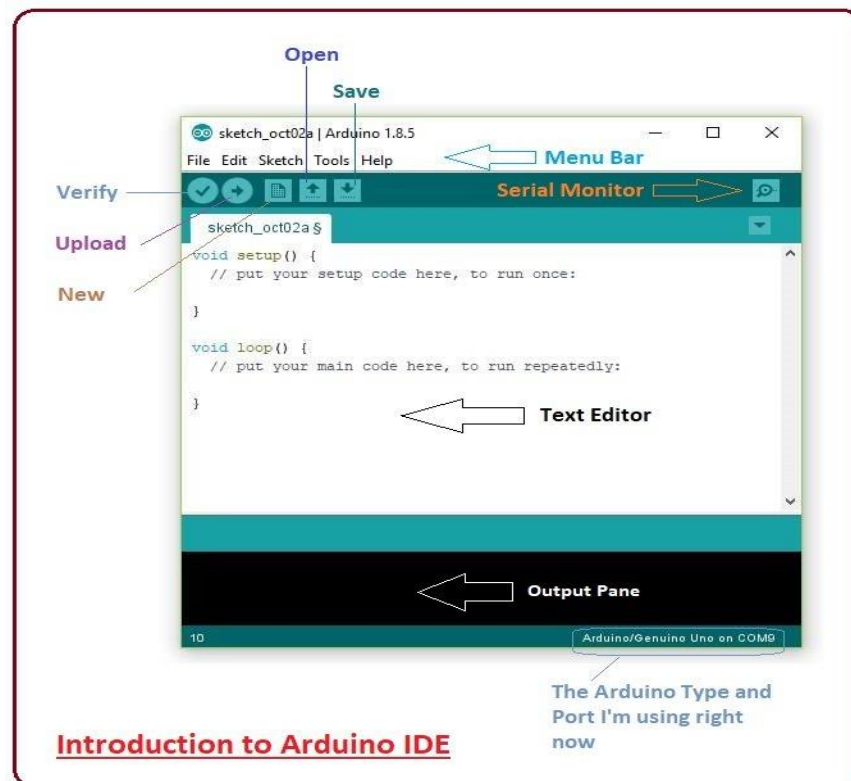
Te funkcije su:

- `setup( )` - funkcija koja se izvršava jednom na početku i služi za početna podešavanja (inicijalizaciju parametara)
- `loop( )` - funkcija koja se nakon inicijalizacije izvršava u beskonačnoj petlji i poziva se sve dok se ne isključi napajanje osnovne pločice, ili dok ne nastupi reset, koji dovodi do ponovnog izvršenja programa ispočetka.

### 3.1. Osnovni prozor Arduino IDE programskog okruženja

Na slici 5. je prikazano Arduino IDE okruženje. Ovaj prozor možemo podijeliti u 4 cijeline:

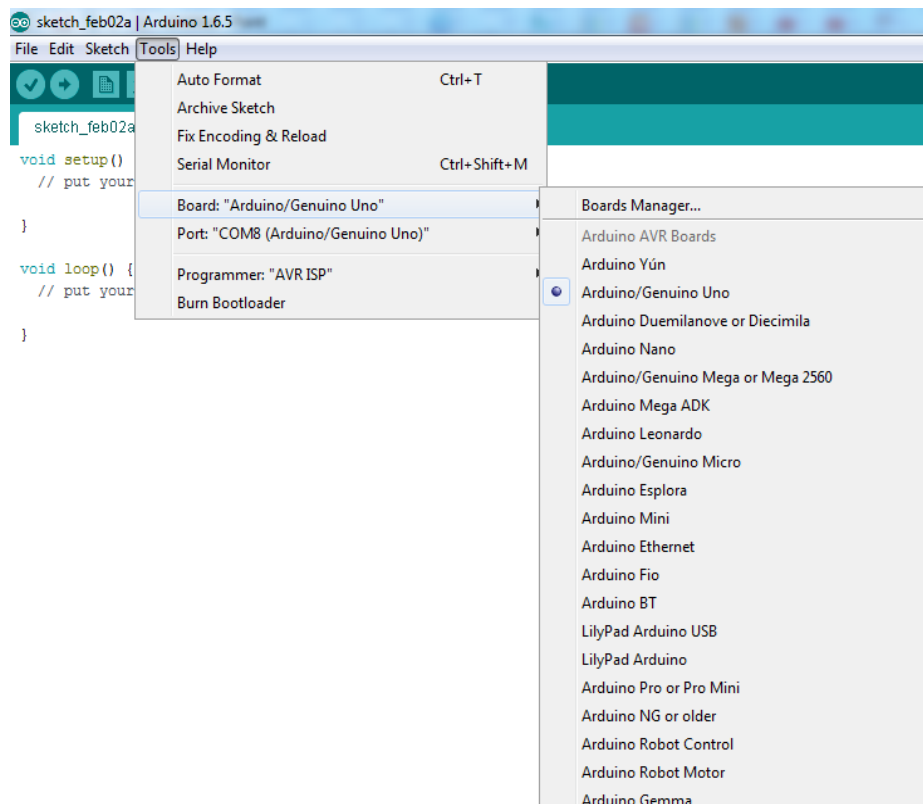
- Osnovni meni (*File, Edit, Sketch, Tools, Help*).
- Traka za brzi pristup:
  - ✓ Verify-Traži greške u kodu (*kompajler*).
  - ➕ Upload-Služi za uploudovanje koda u ploču.
  - 📄 New-Otvora novi prozor za editovanje teksta.
  - 📁 Open-Otvora skicu.
  - 💾 Save-Snimanje skice.
  - ✓ Serial monitor-Služi za komunikaciju programa sa hardverom i otkrivanje grešaka.
- Tekstualni editor.
- Konzola.



Izgled prozora Arduino IDE razvojnog okruženja (slika 5.)



Bitno je napomenuti da prije početka pisanja koda morama izabrati koju pločicu koristimo.



*Biranje pločice (slika 6.)*

Kao što je prikazano na slici biranje pločice se vrši pomoću opcije *Tools*→*Board*.

Takođe u opciji Tools možemo da vidimo i da li je USB kabal dobro povezan sa računarom, ukoliko jeste to ćemo moći vidjeti u opciji Port.

Na slici 5. možemo vidjeti da se isto to može vidjeti i u donjem desnom uglu prozora.

### Primjer

U ovom vrlo jednostavnom primjeru ćemo prikazati kod za program koji treba da pritiskom na taster pali ili gasi led diodu, zavisno u kojem stanju je bila prije pritiska tastera.

```
int led = 2;  // led dioda je povezana na 2. digitalni pin na Arduino ploči
int taster = 3;  // taster je povezan na 3. digitalni pin
void setup() {
    pinMode(led, OUTPUT);  // definišemo da je led dioda izlazni element
    pinMode(taster, INPUT);  // definišemo da je taster ulazni element
}
void loop() {
    if (digitalRead(taster) == 1) {  // uslov koji provjerava stanje tastera
        delay(1000);  // vremensko kašnjenje od 2 sekunde
        digitalWrite(led, HIGH);  // pali led diodu
    }
    else {
        delay(1000);  // vremensko kašnjenje od 2 sekunde
        digitalWrite(led, LOW);  // gasi led diodu
    }
}
```

## 3.2. Funkcije

Funkcije se koriste za kontrolu Arduino ploče, njegovih elemenata i izvršavanje složenijih računskih operacija.

Neke od osnovnih funkcije su:

- **Digitalne funkcije U/I**

- **digitalRead()** – Čita vrijednost za određeni pin, može biti HIGH ili LOW;
- **digitalWrite()** – Piše HIGH ili LOW vrijednost za određeni pin;
- **pinMode()** – Postavlja određeni pin kao izlaz ili kao ulaz;

- **Napredne funkcije U/I**

- **noTone()** – Prekida impulsni signal aktiviran funkcijom **tone()** ;
- **pulseIn()** – Mjeri dužinu impulsnog signala i vraća vrijednost u  $\mu\text{s}$ ;
- **pulseInLong()** – Slična prethodnoj funkciji ali bolje savladava duže signale;
- **shiftIn()** – Prima jedan bajt podataka bit po bit. Počinje ili od najvećeg ili najmanjeg bita (prvi ili zadnji).
- **shiftOut()** – Šalje jedan bajt podataka bit po bit. Počinje ili od najvećeg ili najmanjeg bita (prvi ili zadnji).
- **tone()** – Generiše impulsni talas zadate frekvencije (radnog ciklusa 50%) na pinu. Trajanje funkcije se može odrediti, u protivnom signal se generiše dok se ne pozove funkcija **noTone()**. Ovi pinovi mogu biti spojeni na zujalicu ili zvučnik da proizvode zvukove.

- **Vremenske funkcije**

- **delay()** – Zaustavlja program na određeno vrijeme (u milisekundama  $1\text{s} = 1000\text{ms}$ );
- **delayMicroseconds()** – Zaustavlja program na određeno vrijeme (u mikrosekundama  $1\text{s} = 10^3\text{ms} = 10^6\mu\text{s}$ );

- **micros()** – Vraća za vrijednost broj mikrosekundi koje su prošle od kada je pokrenut program na Arduino ploči. Nakon otprilike 70 minuta ovaj broj će preći svoj maksimum i vratiti se na 0;
- **millis()** – Slična funkcija prethodnoj daje vrijednost u milisekundama i dostiže maksimum otprilike za 50 dana.

## 4. Internet stvari (Internet of Things)

S obzirom da je primjena Arduina velika i raznovrsna, u nekim projektima primjenu je pronašao i IoT.

Internet of Things, odnosno Internet stvari je tehnologija koja omogućava povezivanje tehničkih uređaja putem interneta kako bi se omogućila njihova međusobna interakcija.

Mogućnosti i upotreba IoT tehnologije su vrlo velike, a danas svjedočimo pojavi IoT tehnologije u svakodnevnoj elektronici poput frižidera, mašine za pranje veša, pećnica, ali i u drugim granama tehnike – automobilima, satovima, raznim senzorima, itd.

Postoje različite platforme koje omogućavaju interakciju Arduina i mobilnih uređaja. Jedna od njih je i **Blynk** platforma. Blynk aplikaciji omogućava direktnu komunikaciju aplikacije i Arduino hardvera putem Blynkovog cloud sistema.

Upotrebu Internet stvari preko aplikacije Blynk prikazacemo kroz sljedeći primjer.

### Primjer.

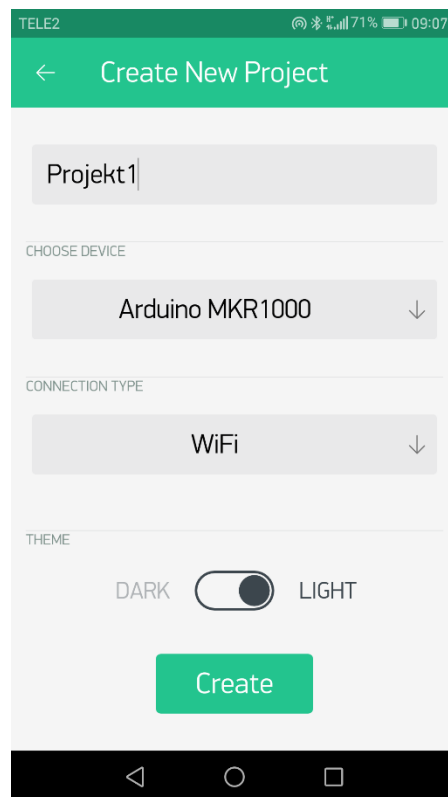
Izraditi projekat gdje će se pomoću widgeta Slider unutar aplikacije Blynk upravljati jačinom svijetleće diode koja je spojena na izvod broj 2 na Arduino mikrokontroleru.

Da bismo mogli početi raditi ovog zadatak prvo moramo preuzeti i instalirati aplikaciju Blynk. Aplikacija se može preuzeti unutar Google trgovine (Play Store) ukoliko radite na Android operativnom sistemu odnosno preko Trgovine aplikacija (App store) ako radite na iOS operativnom sistemu.

Nakon instaliranja aplikacije idući korak je registrovanje. Ono je potrebno kako bi se naši projekti mogli spremati na Blynk serveru.

Kada smo se registrovali možemo da započnemo sa radom i idemo na New Project.

Tu biramo ime projekta, hardver koji koristimo, vrstu povezivanja i temu projekta.



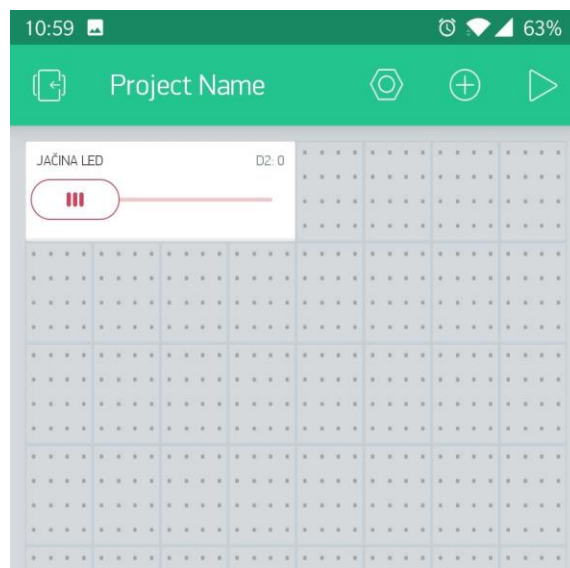
*Kreiranje novog projekta u Blynk aplikaciji (slika 7.)*

U ovom projektu korišćićemo **Arduino MKR1000** koji već u sebi ima integrisanu mogućnost povezivanja na WiFi nije nam potreban nikakav dodatni hardver (shield).

Nakon kreiranja projekta prikazuje se obavještenje da je stvoren **autentifikacijski token** i da je poslan na e-mail koji smo koristili prilikom registrovanja. Ovaj token

je jedinstven za svaki projekat i služi za povezivanje hardvera s Blynk aplikacijom. Iskoristićemo ga kasnije u programskom kodu za Arduino.

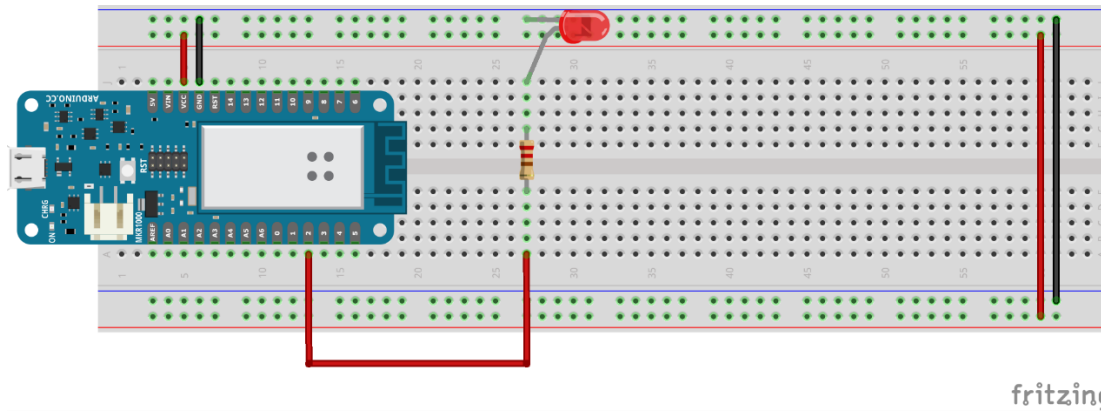
Što se tiče aplikacije ostao nam je još samo jedan korak. Potrebno je izraditi novi projekt, odaberati dodavanje novog widgeta te odabrati Slider. Ovaj widget omogućava nam upravljanje izvodom pomoću pulsno širinske modulacije (PWM), u ovom slučaju izvoda broj 2. Na ovaj način, upravljaćemo jačinom kojom će svjetleća dioda svijetliti.



*Widget slide u aplikaciji Blynk (slika 8.)*

Kako bi naš Arduino MKR1000 povezali sa internetom i Blynk aplikacijom instaliraćemo Blynk biblioteku. Instalaciju biblioteka obavljamo koristeći Library manager unutar Arduino programskog okruženja. Kliknite na Sketch, potom

odaberite Include Library i kliknite na Manage Libraries. Nakon otvaranja Library managera u Search polje upisaćemo Blynk potom kliknuti na Blynk rezultat i za instalaciju kliknuti na Instal. Nakon nekoliko trenutaka biblioteka će biti instalirana.



Šema koja prikazuje način spajanja elemenata (slika 9.)

Instalirali smo aplikaciju, ubacili smo Blynk biblioteku u Arduino IDE i imamo šemu povezivanja električnih elemenata, jedina stvar koja je potrebna za završavanje ovog zadatka je kod.

```
#define BLYNK_PRINT SerialUSB
#include <SPI.h>    // Služi za povezivanje mikrokontrolera sa perifernim uređajima
#include <WiFi101.h>    // Služi za povezivanje sa bežičnim internetom
#include <BlynkSimpleWiFiShield101.h>
char token[] = "Token"; // Upišite vlastiti token za autentifikaciju
char ime[] = "ImeOdBezicneMreze"; // Upišite ime vaše bežične mreže
char sifra[] = "SifraOdBezicneMreze"; // Upišite šifru vaše bežične mreže
void setup()
{
  SerialUSB.begin(9600);
  Blynk.begin(token, ime, sifra);
}
void loop()
{
  Blynk.run();
}
```

## 5. Zaključak

Primjena Arduino je veoma raznovrsna, a može se koristiti kao sigurnosni sistem (alarm), kao upravljač, u raznim segmentima robotike, kao razne vrste senzora (temperature, vlažnosti, svjetlosti, rastojanja), kao sistem za navigaciju (GPS), za kontrolisanje određenih električnih uređaja (šporeta, frižidera, svjetla, grijanja), za pokretanje i upravljanje radom elektromotora...

Pored njegove raznovrsne primjene, prednost mu je i dostupljenost zadataka i primjera preko interneta. Samim tim predstavlja i dobar način za učenje i spajanje softverskog dijela sa hardverskim.

Međutim zbog otvorenog koda i relativno visokih cijena, postoji i dosta firmi koje proizvode i prodaju slične uređaje po nižoj cijeni. Naravno, ti uređaji nisu toliko kvalitetni i ne mogu da vrše neke funkcije koje Arduino može.

Još jedna bitna funkcija Arduina je i Internet stvari, u kojem dosta ljudi vidi budućnost tehnologije. Pravljenje “pametnih” stvari od svakodnevnih fizičkih objekata koje će moći da komuniciraju i razmjenjuju informacije o sebi ili svome okruženju, izvršava se ugrađivanjem određenih mikrokontrolera ili procesora i možda baš tu leži budućnost Arduina.



## **LITERATURA:**

1. [http://www.mikroknjiga.rs/Knjige/ARD/Uvod\\_ARD.pdf](http://www.mikroknjiga.rs/Knjige/ARD/Uvod_ARD.pdf) (21.04.2020.god.)
2. <https://www.hobbytronics.co.uk/arduino-summary> (21.04.2020.god.)
3. <https://hr.if-koubou.com/articles/how-to/what-is-arduino-learn-about-this-open-source-electronics-platform.html> (23.04.2020.god.)
4. <https://www.elektronika.ftn.uns.ac.rs/elektronika-e2/wp-content/uploads/sites/87/2018/03/Arduino-uputstvo.pdf> (25.04.2020.god.)
5. <https://www.arduino.cc/en/main/software> (28.04.2020.god.)
6. <https://izradi.croatianmakers.hr/project/instalacija-blynk-aplikacije-i-prvi-iot-projekt/> (28.04.2020.god.)