



LABORATORIJSKA VEŽBA BR. 4

Povezivanje sa Internetom

CILJ VEŽBE

- Upoznavanje sa načinima za povezivanje sa Internetom
- Upoznavanje sa različitim WI-Fi modulima za bežično povezivanje sa internetom
- Upoznavanje sa hardverskim zahtevima za povezivanje
- Upoznavanje sa softverskim kodom koji nam omogućava povezivanje sa Internetom
- Priključivanje Wi-Fi modula na Arduino UNO
- Hardversko povezivanje Wi-Fi modula
- Pisanje programskog koda koji će omogućiti bežičnu vezu
- Podešavanje Serial Monitora za prikaza podataka bežične veze
- Testirane aplikacije

POTREBNA OPREMA

- Računar sa instaliranim razvojnim softverom za Arduino
- Arduino UNO + Mega 3 komplet
- WiFi modul za povezivanje ESP8288
- Multimetar
- Komplet alata za montažu

TEORIJSKE OSNOVE

Povezivanje sa Internet-om

Svi IoT uređaji zahtevaju mehanizam za slanje ili primanje podataka. Postoje brojne opcije koje omogućuju da Arduino sistem povežete sa Internetom: putem žičane ili bežične Ethernet veze, Bluetooth veze, mobilne mreže i dr. Koju ćete opciju izabrati zavisi od mnogih faktora koji utiču na tu vezu:

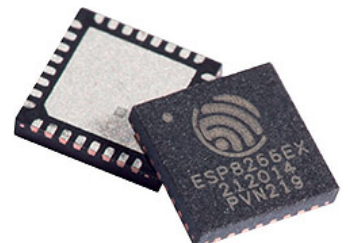
- veličina mreže u kojoj će se aplikacija raditi
- količina podataka koje treba obraditi i preneti
- fizička lokacija uređaja

Tabela 1. prikazuje neke od opcija za povezivanje sa Internet-om sa primerom gde se on koristi.

Tabela 1. Opcije povezivanja sa Internet-om kod IoT

Žičani (Ethernet)	Food storage temperature monitoring
Bežični (WiFi)	Praćenje vremenskih uslova u prirodi
Bluetooth	Praćenje osoblja
Cellular data	Praćenje životinjskog sveta
RFID (Radio Frequency Identification)	Roba u magacinu

Arduino Uno bežično povezivanje (WiFi)



Upoznaćemo se sada sa načinom kako možemo Arduino Uno ploču proširiti sa WiFi modulom (shield) preko koga ćemo ostvariti konekciju sa Internetom. Treba imati u vidu da pojedine Arduino ploče imaju već ugrađen WiFi modul koji nam omogućuje da se bežično povežemo sa Internetom (Arduino Yun). Takođe, postoje i veoma veliki broj različitih WiFi modula koji nam omogućuju da se povežemo sa Internetom. Većina njih bazirana je na integralnom kolu EPS8266. Čip *ESP8266EX* je zanimljiv i donekle tajanstven proizvod ne baš poznate kineske kompanije *Espressif Systems*. Izrađuje se na osnovu tehnologije američke firme *Tensillica*, koja je pre nekoliko godina postala deo kompanije *Cadence Design Systems*. Reč je o čipu baziranom na 32-bitnoj RISC arhitekturi pod nazivom *Xtensa LX106 Diamond* i radi na (za mikrokontrolere) vrlo pristojnih 80 MHz. Može da se overklokuje do 160 MHz, ali se stabilnost ne garantuje. Ima 96 kB radne memorije i 64 kB memorije za instrukcije. Šta je sada ovo?! Korišćena arhitektura mikrokontrolera je zanimljiva po tome što proizvođačima opreme dozvoljava kreiranje sopstvenih instrukcija u okviru mikrokoda. Uz mikrokontroler ide eksterni čip fleš memorije kapaciteta od 512 kB (retko se sreće), preko danas standardnog jednog megabajta, pa sve do 4 MB (teoretski maksimum 16 MB). Prvi modeli pojavili su se na tržištu 2014. godine uz reklamu da se radi o zaokruženom rešenju za Wi-Fi komunikaciju. To znači da moduli mogu da se koriste samostalno, ali i kao sredstvo za povezivanje mikrokontrolera bežičnom Wi-Fi vezom. S obzirom da je cena ovih čipova u odnosu na ostale „Wi-Fi capable” konkurente od samog početka bila veoma niska, ne čudi činjenica da su vrlo toplo prihvaćeni od strane mejkera. Osim što je dobro rasprostranjen i popularan, *ESP8266* je učinio mnogo za celokupno IoT tržište primoravajući ostale proizvođače da znatno snize cene.

Vrste modula

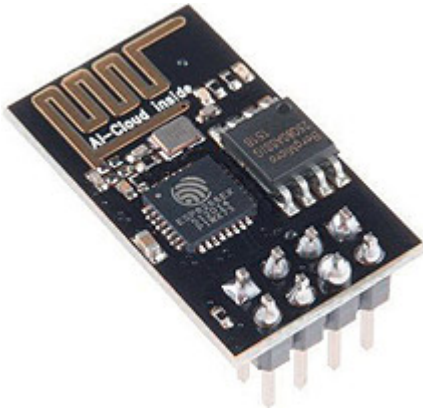
Ono što zbunjuje početnike u radu sa ovom platformom jeste činjenica da postoji ceo niz različitih modula zasnovanih na istom čipu (vidi sliku br.1). *AI-Thinker* je jedan od proizvođača koji su među prvima počeli sa kreiranjem modula na bazi čipa *ESP8266*, pa su sticajem okolnosti postali omiljeni kod hobista. Pa da pogledamo nekoliko njih koji bi nam potencijalno mogli biti zanimljivi.

Espressif ESP8266EX	
Mikrokontroler	Tensilica Xtensa LX106 Diamond, 80 MHz (160 MHz overklok)
Memorija	96 KB programski RAM, 64 KB instrukcijski RAM, eksterna fleš memorija (512 KB - 4 MB)
Wi-Fi	802,11b/g/n WPA/WPA2 (WEP/TKIP/AES)
Interfejsi	GPIO (x16), SPI, I2C, I2S, UART(x2), PWM (x4), SDIO, IR Remote, ADC (x1) 10-bit
Radni napon	3-3,6 V
Potrošnja struje	80 mA, maks, 320 mA, <10 uA sleep



Slika br.1 Različiti WI-Fi moduli ESP8266

ESP-01



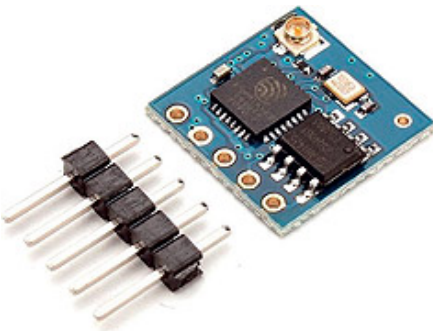
Prvi *AI-Thinker* modul sa čipom *ESP8266* koji se pojavio na tržištu, istovremeno je najpopularniji i najčešće korišćen, zahvaljujući ceni koja je niža od svih predstavnika u toj kategoriji. Omogućuje pristup do dva GPIO porta, što znači da može da se koristi za prikupljanje senzorskih podataka nezavisno od drugog mikrokontrolera. Dva reda sa po četiri pina nisu podesni za priključivanje na prototipsku pločicu, ali postoje jeftini dodatni moduli koji to omogućavaju. Dobar su izbor zbog toga što se često koriste u primerima koji se mogu naći na internetu, pa nije teško doći do gotovih rešenja. Međutim, nisu bez nedostataka. Pojedini izvodi glavnog čipa, zbog loših inženjerskih rešenja, nisu nigde priključeni, pa se nalaze u takozvanom „plutajućem” stanju, koje može prouzrokovati razne

nepravilnosti u radu.

Osim toga, ovi moduli nemaju mogućnost dovođenja elektronike u stanje takozvanog „dubokog sna”, koje je vrlo važno za ekonomiju električne energije. Najčešća veličina fleš memorije za module crne boje iznosi jedan megabajt.

R.b.	Pin	Funkcija
1.	GND	uzemljenje
2.	TX	UART slanje podataka
3.	GPIO2	GPIO port 2
4.	CH_PD	aktiviranje modula
5.	GPIO0	GPIO port 0
6.	RST	eksterni reset signal
7.	RX	UART prijem podataka
8.	VCC	konektor napajanja

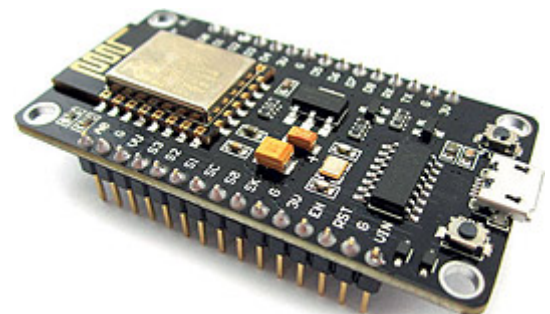
ESP-05



U pitanju je mali, pojednostavljeni modul koji ima svega pet nožica za povezivanje, kao i konektor po IPX standardu za priključivanje spoljašnje antene. Njegov najveći plus predstavlja mogućnost jednostavnog korišćenja na prototipskoj pločici, dok mu se kao minus računa to što ne obezbeđuje nikakav pristup ka GPIO portovima mikrokontrolera. Sa *Arduinom* se povezuje putem UART interfejsa i služi za prenos podataka putem Wi-Fi mreže.

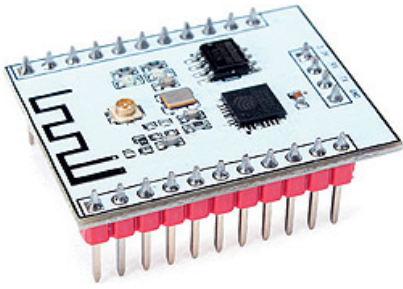
ESP-12E (NodeMCU 1.0)

Ova vrsta modula, zbog rastojanja među konektorima od dva milimetra i nemogućnosti lemljenja pinova za priključivanje na prototipsku pločicu, nije naročito podesna za rad sa *Arduinom*. Međutim, postoje i moduli koji na sebi imaju zalemljen *ESP-12* i istovremeno sadrže konektore sa iglicama razmaka 2,5 milimetara. Njihov plus u odnosu na ostale predstavljene module odnosi se na prisustvo 4096 kilobajta fleš memorije. Uz njih dolazi firmver pod nazivom *NodeMCU*, koji podržava skript programski jezik *Lua*, pa kao proizvod dobijamo funkcionalno i zaokruženo IoT razvojno okruženje. *Lua* je moćan i dosta jednostavan programski jezik, donekle sličan *JavaScriptu*, ali fleksibilniji od njega i znatno uprošćava programiranje, kako funkcija vezanih za Wi-Fi komunikaciju, tako i za rad sa priključenim sensorima. Ovi moduli obično na sebi imaju i čip *CH340G* koji omogućava USB<>UART povezivanje, što pojednostavljuje proces zapisivanja



podataka u integrisanu fleš memoriju. Uz *ESP-07*, reč je verovatno o inženjerski najbolje rešenim modulima ove vrste, a sa cenom od približno 3,5 evra, predstavljaju odličan izbor.

ESP-201



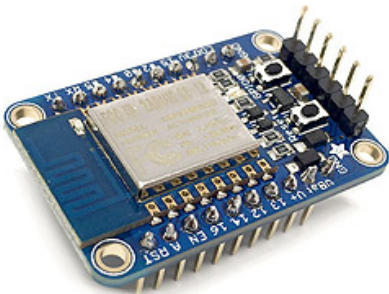
Ovaj modul osim antene na pločici, pruža mogućnost priključenja eksterne antene. U standardnom pakovanju obično dolazi sa jednostavnom antenom koja ima IPX interfejs. Inicijalno se koristi spoljašnja antena, a u slučaju da želimo da koristimo integrisanu, potrebno je prelemiti SMD otpornik veličine 0 oma.

Modul na sebi ima 26 pinova od kojih su četiri namenjena povezivanju sa UART interfejsom. Nije baš najjasnije šta je konstruktore motivisalo da ta četiri pina okrenu prema dole i onemogućuje postavljanje na prototipsku pločicu, ali je za veliki broj korisnika prva stvar koju urade njihovo odlemljivanje i postavljanje na vrh. Nazivi pinova su štampani sa donje strane pločice, ali je to od male koristi, pošto za vreme rada oni nisu vidljivi.

Modul sadrži šest GPIO portova (GPIO6-GPIO11) koje nije moguće praktično koristiti. Uz hardversku modifikaciju na čipu sa fleš memorijom moguće je osloboditi GPIO portove 9 i 10. Prilikom flešovanja pin IO15 mora biti u stanju logičke nule.

Iako *ESP-201* na prvi pogled izgleda kao sjajan modul, on ima dosta slabosti konstruktorske prirode. Cena od 2,4 evra ga čini privlačnim za kupovinu, ali uprkos konektorima koji se mogu priključiti na prototipsku ploču, znatan broj konstrukcijskih grešaka uveliko smanjuje njegov potencijal.

Adafruit Huzzah

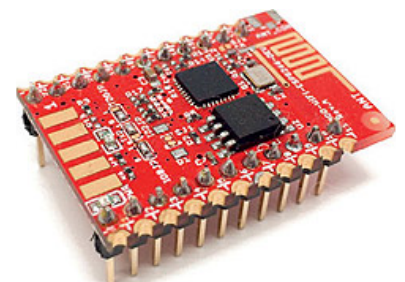


U pitanju je modul renomiranog proizvođača u svetu *Arduino* tehnike i baziran je na modulu *ESP-12*. Uz nešto višu cenu (ispod 10 dolara) nudi dosta jednostavnije upravljanje Wi-Fi modulom. Od ostalih modela se razlikuje postojanjem dva mala tastera od kojih je jedan namenjen ulasku u *bootloading* režim, dok drugi funkcioniše kao *resettaster*. Korisniku je na raspolaganju devet GPIO portova (0, 2, 4, 5, 12, 13, 14, 15, 16), između ostalog, i sa funkcijama SPI i I2C interfejsa, dva UART interfejsa i jednim analognim ulazom (1,8 volti maksimalno). Na pločici se nalazi i regulator struje koji omogućuje napajanje do 500 miliampera.

I ovde se koristi programski jezik *Lua* kao osnova za programiranje, ali je moguće koristiti i programiranje putem *Arduino IDE* i raznih drugih programskih jezika.

Olimex ESP8266-DEV

Zanimljivo je da se sedište kompanije Olimex nalazi u nama susednoj Bugarskoj i da već duži niz godina uspešno opstaje u teškoj tržišnoj borbi sa dalekoistočnim proizvođačima elektronike (po svedočenju vlasnika firme Cvetana Usunova, dobrim delom zahvaljujući stimulaciji iz fondova EU). Ovo je jedan od prvih Olimexovih modula koji se pojavio još 2014. godine i u međuvremenu je stekao dosta veliku popularnost.

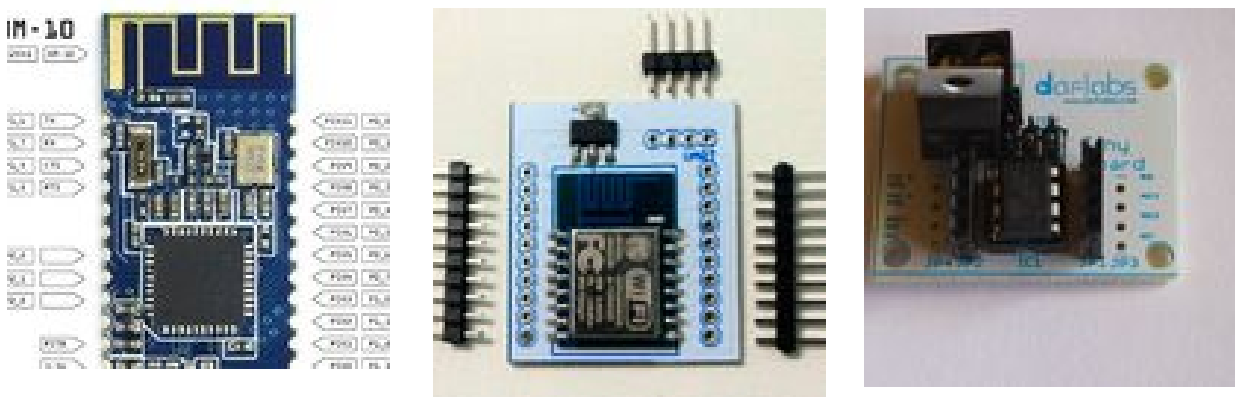


Na pločici se nalaze tzv. UEXT (*Universal EXTension*) konektori za priključivanje na UART interfejs. Posедуje dva megabajta fleš memorije,

što omogućava upotrebu u kompleksnijim projektima. Cena uređaja je oko pet evra, što se može smatrati vrlo prihvatljivim. Isti proizvođač po ceni od devet evra nudi razvojnu pločicu *ESP8266-EVB* koja već sadrži *ESP8266-DEV* modul i umnogome pojednostavljuje komunikaciju s njim.

Tu je i cela linija drugih modula (*ESP-02, ESP-03, ESP-04, ESP-06, ESP-08, ESP-09, ESP-10...*) koji su namenjeni ugrađivanju na druge uređaje i nisu pogodni za direktni rad sa *Arduinom*, pa im zbog toga nećemo posvetiti više prostora. Naravno, i oni se, slično modulu *ESP-12E*, mogu zalemiti na dodatnu pločicu, ali to je već van opsega našeg teksta. Zbog ograničenosti prostora mesto u opisu nisu našli ni neki drugi moduli, uključujući jedan od najboljih (ali i nekoliko puta skuplji) *Sparkfun Thing*, koji ima konektor za eksternu Li-Po bateriju, koja mu uz ispravan dizajn projekta omogućava višegodišnje autonomno funkcionisanje.

Mi ćemo koristiti WiFi modul pod oznakom WiFi 8266 ESP-12 koji je prikazan na donjoj slici.



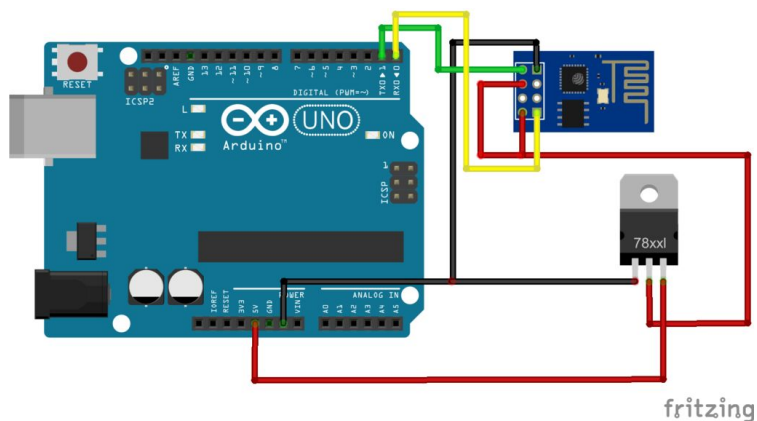
WiFi modul 8266 ESP-12

Tehničke karakteristike korišćenog modula:

- Mala potrošnja, za uređaje koji se napajaju iz baterija
- Podesiva snaga signala, 0dBm do +12dBm
- hardverski UART interfejs
- podržava Adhock i infrastrukturne mrežne modove
- programabilni GPIO i ADC ulazi/izlazi
- Real-time sat za vremensko označavanje, auto sleep i auto wakeup modove
- može da koristi napajanje iz baterija ili regulisanog napajanja
- konfigurabilan preko serijskog ili bežičnog interfejsa koristeći jednostavne ASCII komande
- bežični firmware upgrade
- Wi-Fi autentifikacija (WEP/WPA/WPA2)
- pun TCP/IP stack (nisu potrebni dodatni drajveri)

Hardverski deo

Modul *ESP-12* ima rastojanje između pinova od 2mm, što znači da ne možemo direktno da ga montiramo na perforiranu/tačkastoj matrici, koja ima 2,54mm razmak (udaljenost između susednih pinova). Zbog toga je potrebno da se taj modul montira na *XBee* adaptersku ploču kako bi ovaj modul bio kompatibilan sa perforiranom pločom. Treba voditi računa i o napajanju ovog modula jer on radi na 3,3V. Svako povezivanje na veći napon (5V) može



dovesti do uništenja ovog modula. Takođe je potrebno povezati još nekoliko pinova kako bi se uspostavila veza između WI-Fi modula i Arduino UNO ploče. Te veze su sledeće:

- VCC --> 3.3V napajanje (Vout iz regulatorskog čipa)
- GND --> GND iz izvora napajanja
- CH_PD --> HIGH (3.3V)
- GPIO2 --> HIGH (3.3V)
- GPIO15 --> LOW (GND)
- GPIO0 --> HIGH ili ostaviti nepovezano za AT Mode (3.3V) [ukoliko želimo da izvršimo ubacivanje novog firmware (flash-ovanje modula) obavezno je potrebno da ovaj pin bude povezan na GND]

Softverski deo

Nakon što hardverski osposobimo WiFi modul (shield), potrebno je napisati programski kod koji će omogućiti bežično povezivanje našeg Arduino Uno uređaja sa nekom od dostupnih bežičnih mreža koje su povezane sa Internetom. Na taj način omogućićemo da Arduino uređaj može da šalje i prima podatke preko Interneta. Pokrenite Arduino IDE i unesite sledeći programski kod koji je dat u nastavku ove vežbe. Celokuna programska podrška može se podeliti na sledeća tri dela:

1. **Spoljne biblioteke** – uključuje definisanje svih spoljnih biblioteka koje su potrebne da bi programski kod za priključivanje sa Internetom nesmetano radio. Kako se vrši bežično povezivanje potrebno je instalirati WiFi biblioteku WiFi.h. Vaš Arduino IDE trebalo bi da već ima instaliranu navedenu biblioteku Wifi.H, ali ako iz bilo kog razloga ona nedostaje, možete je preuzeti sa sledećeg sajta: <https://github.com/arduino/Arduino/tree/master/libraries/WiFi>
2. **Povezivanje sa Internetom** (bežično) - Drugi deo koda definiše promenljive, konstante i funkcije koje se koriste za povezivanje sa Internetom. Da biste povezali Arduino uređaj sa bežičnom mrežom, kao i bilo koji drugi uređaj, potrebno je prvo da znate ime mreže (ssid) kao i lozinku (password) koja će vam omogućiti pristup toj mreži. Takođe, u okviru ovog dela programskog koda kreiraćemo i WiFiClient deo (funkcija connectToInternet()) koja nam služi da se povežemo sa Internetom. Ova funkcija prvo proverava da li postoji WiFi modul u okviru našeg Arduino uređaja i da li on pravilno funkcioniše. Ako je sve u redu tada na osnovu zadatih parametara (SSID i password) program u programskoj petlji pokušava da se poveže sa željenom mrežom. Iz petlje se izlazi nakon što se uspostavi uspešna konekcija sa bežičnim ruterom tj. mrežom. Nakon povezivanja Arduino uređaja sa bežičnom mrežom, izvršava se f-ja printConnectionInformation(). Njen zadatak je da ime mreže SSID, MAC adresu rutera, jačinu bežičnog signala (RSSI), IP adresu našeg Arduino uređaja kao i njegovu MAC adresu pošalje Serial Monitoru kako bi u okviru njegovog prozora mogli da očitame sve te parametre bežične konekcije.
3. **Standardne funkcije** – Zadnji deo programskog koda implementira Arduino-ove standardne funkcije setup() i loop(). Za ovu laboratorijsku vežbu sve što radite je povezivanje Arduina na Internet i nakon toga nema nikakve obrade podataka, tako da petlja loop() ostaje prazna jer nema ništa da obrađuje.

I Spoljne biblioteke

```
#include <SPI.h>
#include <WiFi.h>
```

II Povezivanje sa Internetom

```
char ssid[] = " "; // SSID mreže na koju se povezujemo
char pass[] = " "; // PASSWORD za pristup mrežnim resursima
int keyIndex = 0;
int status = WL_IDLE_STATUS;
```

void connectToInternet()

```

{
status = WiFi.status();
// Proveravamo da li postoji Wi-Fi modul(Wi-Fi shield)
if (status == WL_NO_SHIELD)
{
Serial.println("[ERROR] WiFi Shield Not Present");
while (true);
}
// Pokušavamo da se povežemo sa WPA/WPA2 Wi-Fi mrežom
while ( status != WL_CONNECTED)
{
Serial.print("[INFO] Attempting Connection - WPA SSID: ");
Serial.println(ssid);
status = WiFi.begin(ssid, pass);
}

// Povezivanje je uspelo
Serial.print("[INFO] Connection Successful");
Serial.print("");
printConnectionInformation();
Serial.println("-----");
Serial.println("");
}

void printConnectionInformation()
{
// Print Network SSID
Serial.print("[INFO] SSID: ");
Serial.println(WiFi.SSID());
// Štampanje Ruterove MAC adrese
byte bssid[6];
WiFi.BSSID(bssid);
Serial.print("[INFO] BSSID: ");
Serial.print(bssid[5], HEX);
Serial.print(":");
Serial.print(bssid[4], HEX);
Serial.print(":");
Serial.print(bssid[3], HEX);
Serial.print(":");
Serial.print(bssid[2], HEX);
Serial.print(":");
Serial.print(bssid[1], HEX);
Serial.print(":");
Serial.println(bssid[0], HEX);
// Štampanje jačine prijemnog signala (RSSI - received signal strength)
long rssi = WiFi.RSSI();
Serial.print("[INFO] Signal Strength (RSSI): ");
Serial.println(rssi);
// Štampanje vrste enkripcije
byte encryption = WiFi.encryptionType();
Serial.print("[INFO] Encryption Type: ");
Serial.println(encryption, HEX);

// Štampanje IP adrese Arduino uređaja (Wi-Fi shield)

```

```

IPAddress ip = WiFi.localIP();
Serial.print("[INFO] IP Address: ");
Serial.println(ip);
// Štampanje MAC adrese Arduino uređaja
byte mac[6];
WiFi.macAddress(mac);
Serial.print("[INFO] MAC Address: ");
Serial.print(mac[5], HEX);
Serial.print(":");
Serial.print(mac[4], HEX);
Serial.print(":");
Serial.print(mac[3], HEX);
Serial.print(":");
Serial.print(mac[2], HEX);
Serial.print(":");
Serial.print(mac[1], HEX);
Serial.print(":");
Serial.println(mac[0], HEX);
}

```

III Standardne funkcije

void setup()

```

{
// Inicializacija serijskog porta za slanje informacija prema Serial Monitoru
Serial.begin(9600);
// Povezivanje Arduino uređaja sa Internetom
connectToInternet();
}

```

void loop()

```

{
// Nema nikakvih zadataka za izvršavanje
}

```

Arduino programski kod za povezivanje sa Internetom je sada završen. Da biste testirali vašu aplikaciju, potrebno je proveriti i iskompajlirati aplikaciju i prebaciti je na Arduino. Nakon što se učita, otvorite prozor Serijal monitor na kome bi trebalo da se prikažu log podaci koji su prikazani na donjoj slici. Ukoliko se u prozoru Serial Monitor ne pojavljuju ovi podaci već se ispisuje neki nerazumni tekst proverite da li ste dobro podesili brzinu prenosa (prozor u donjem desnom uglu u okviru Serial Monitor-a – 9600 baud).

