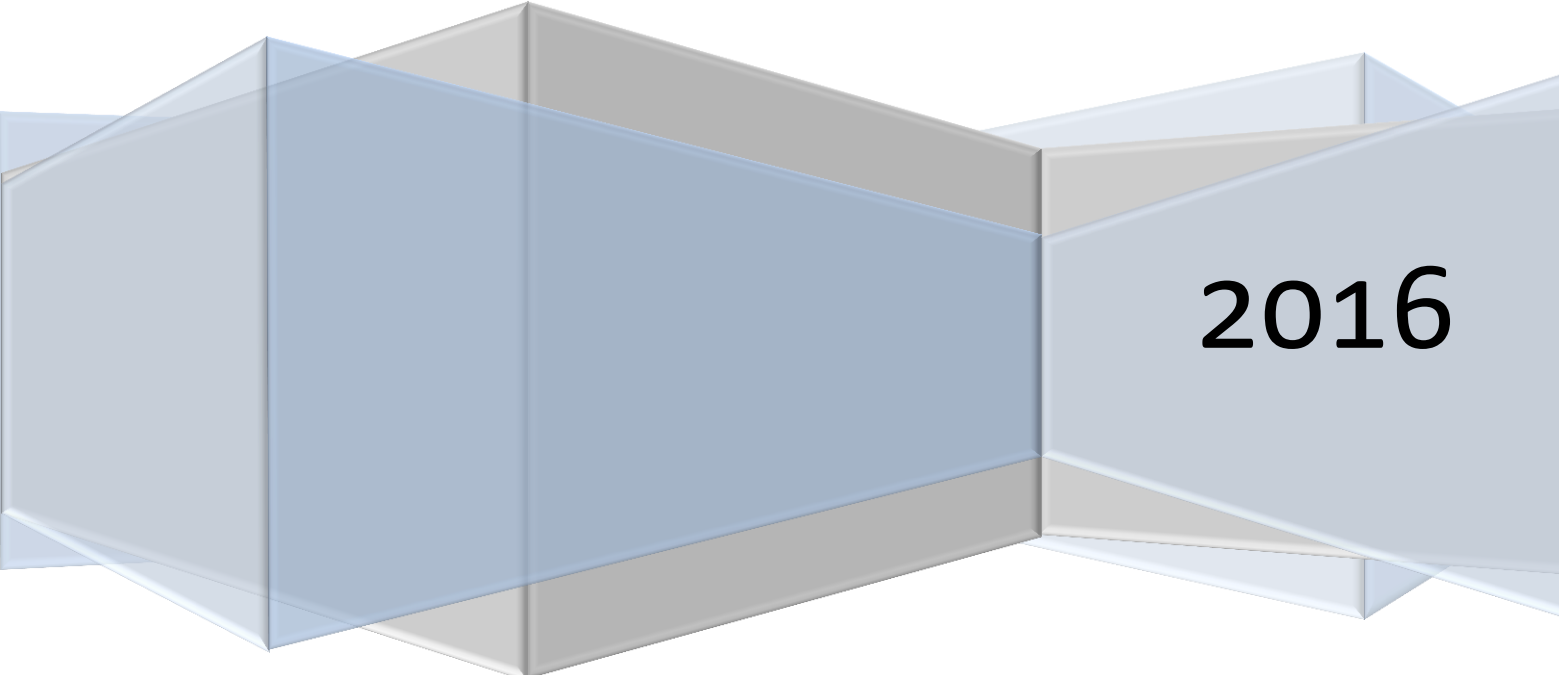


Elektrotehnički Fakultet, Univerzitet u Beogradu

Bežična temperaturna stanica

Mikroprocesorski sistemi

Vidor Gencel 502/12



2016

Sadržaj

Cilj projekta	2
Korišćene komponente	3
Povezivanje komponenti.....	4
Displej.....	5
Temperaturni senzor.....	6
Wi-Fi modul.....	7
Seriski kabl	8
Komunikacija sa komponentama.....	9
Displej.....	9
Temperaturni senzor.....	9
Wi-Fi modul.....	10
Dizajn softvera	11
Display.....	11
Temperature	11
Network	11
LED	12
Main	12
Uputstvo za prevođenje.....	13
Uputstvo za upotrebu	14

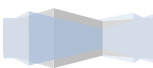


Cilj projekta

Cilj projekta je bio realizovati temperaturnu mernu stanicu koja, pomoću Wi-Fi modula, ima mogućnost da se konektuje na bežičnu mrežu i da preko iste pruža interfejs korisniku za iščitavanje senzora.

Naizgled jednostavan projekat se ispostavio kao pravi izazov zbog potrebe dobrog poznavanja kako hardvera, tako i softvera zajedno sa raznim komunikacionim protokolima.

Uprkos svemu, smatram da je ovo bila izvanredna prilika za sticanje praktičnog znanja iz oblasti mikroprocesorskih sistema i dobra osnova za dalje istraživanje.



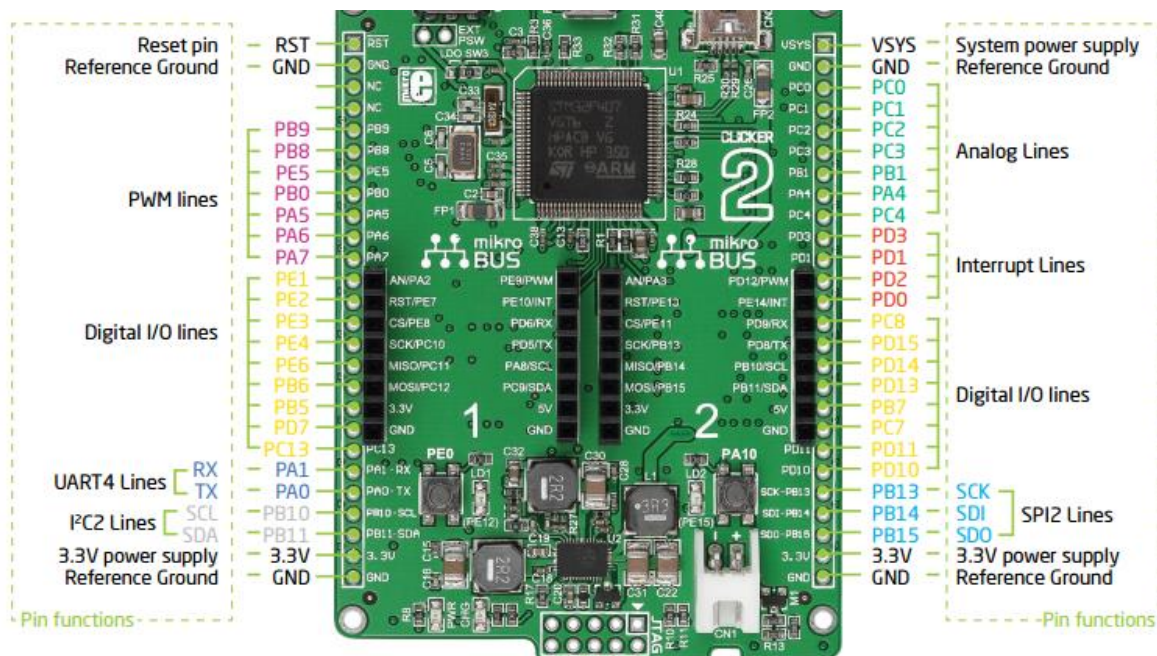
Korišćene komponente

- MikroElektronika Clicker2 sa ARM mikrokontrolerom (STM32F407VG)
- Karakterni 16x2 LCD displej sa HD44780 kontrolerom
 - Promenljivi (varijabilni) otpornik od 10K
- DS1820 temperaturni senzor
 - 4.7K otpornik
- ESP8266 Wi-Fi modul
- USB to Serial kabl



Povezivanje komponenti

Dat je raspored pinova mikrokontrolera i njihovi nazivi kako će biti referisani u daljem tekstu.



■ Digital lines
 ■ Analog Lines
 ■ Interrupt Lines
 ■ SPI Lines
 ■ I2C Lines
 ■ UART lines
 ■ PWM lines



Displej

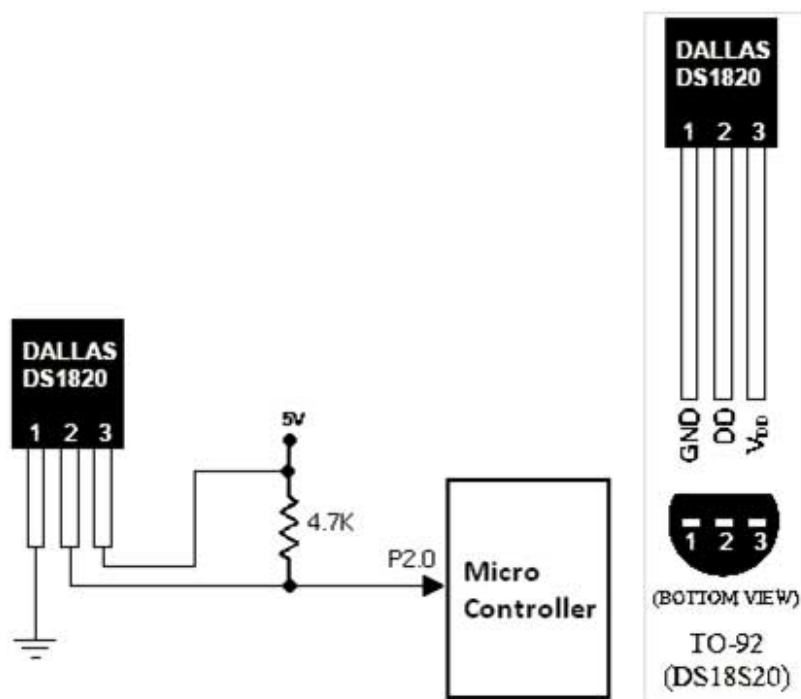
Raspored pinova displeja je sledeći:

Pin No.	Symbol	Level	Description
1	V _{SS}	0V	Ground
2	V _{DD}	5.0V	Supply Voltage for logic
3	VO	(Variable)	Operating voltage for LCD
4	RS	H/L	H: DATA, L: Instruction code
5	R/W	H/L	H: Read(MPU→ Module) L: Write(MPU→ Module)
6	E	H,H→ L	Chip enable signal
7	DB0	H/L	Data bit 0
8	DB1	H/L	Data bit 1
9	DB2	H/L	Data bit 2
10	DB3	H/L	Data bit 3
11	DB4	H/L	Data bit 4
12	DB5	H/L	Data bit 5
13	DB6	H/L	Data bit 6
14	DB7	H/L	Data bit 7
15	A	-	LED +
16	K	-	LED -

Displej	Mikrokontroler
PIN1	GND
PIN2	5V iz clicker socket-a, može i 3.3V
PIN3	Izlaz varijabilnog otpornika Na ulaz varijabilnog otpornika treba dovesti GND i 5V
PIN4	PD7
PIN5	GND
PIN6	PD10
PIN7	N/C
PIN8	
PIN9	
PIN10	
PIN11	PD11
PIN12	PD13
PIN13	PD14
PIN14	PD15
PIN15	3.3V
PIN16	GND



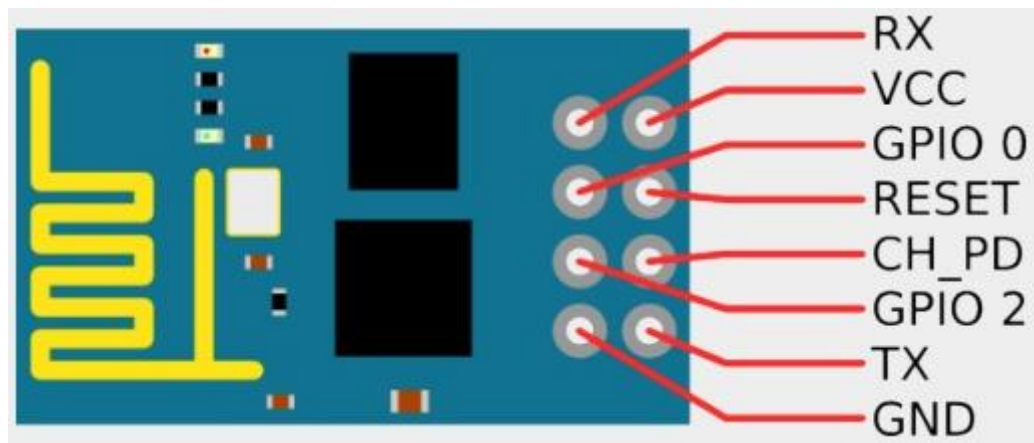
Temperaturni senzor



DS1820	Mikrokontroler
PIN1	GND
PIN2	PB5
PIN3	Jedna nogica otpornika od 4.7K, druga nogica otpornika ide na izvor napajanja od 5V



Wi-Fi modul



ESP8266	Mikrokontroler
RX	PD8
VCC	3.3V
GPIO 0	N/C
RESET	3.3V
GPIO 2	N/C
CH_PD	3.3V
TX	PD9
GND	GND



Serijski kabl

U svrhu debugovanja moguće je povezati serijski kabl na sledeći način:

Serijski kabl	Mikrokontroler
Zelena žica	PA1
Bela žica	PA0
Crvena žica	N/C
Crna žica	GND



Komunikacija sa komponentama

Displej

Komunikacija sa displejom je realizovana upotrebom Mikroelektronikine LCD biblioteke.

Dokumentacija za ovu biblioteku se može naći na adresi:

http://www.mikroe.com/download/eng/documents/compilers/mikroc/pro/arm/help/lcd_library.htm

Korišćene funkcije:

- Lcd_Init – inicijalizuje displej
- Lcd_Out – štampa zadati tekst na zadatoj liniji
- Lcd_Cmd – šalje komande displeju
 - _LCD_CLEAR
 - _LCD_CURSOR_OFF

Temperaturni senzor

Komunikacija sa temperaturnim senzorom je realizovana upotrebom Mikroelektronikine OneWire biblioteke.

Dokumentacija za ovu biblioteku se može naći na adresi:

http://www.mikroe.com/download/eng/documents/compilers/mikroc/pro/arm/help/onewire_library.htm

Treba obratiti pažnju da ova biblioteka funkcioniše samo ukoliko su prekidi onemogućeni!

Korišćene funkcije:

- Ow_Reset
- Ow_Read
- Ow_Write

Da bi se uspostavila komunikacija sa senzorom potrebno je utvrditi odgovarajuće poruke koje moraju biti razmenjene u svrhu dohvaćanja temperature.

Iz data sheet-a senzora (<http://www.systronix.com/Resource/ds1820.pdf>) utvđuje se da je najjednostavnija sekvenca instrukcija za očitavanje sledeća:

- **RESET**
- **WRITE 0xCC** (Skip ROM)
- **WRITE 0x44** (Convert Temperature)
- U ovom trenutku je zatražena konverzija temperature, a kako po dokumentaciji ovaj proces može da traje i do 500ms, čeka se upravo toliko
- **RESET**



- **WRITE 0xCC** (Skip ROM)
- **WRITE 0xBE** (Read Scratchpad)

Sada kada imamo potrebnu biblioteku i znamo potreban set komandi sa lakoćom možemo iščitavati vrednost senzora.

Wi-Fi modul

Komunikacija sa temperaturnim senzorom je realizovana upotrebom Mikroelektronikine UART biblioteke.

Dokumentacija za ovu biblioteku se može naći na adresi:

http://www.mikroe.com/download/eng/documents/compilers/mikroc/pro/arm/help/uart_library.htm

Korišćene funkcije:

- UARTx_Init_Advanced
- UARTx_Read
- UARTx_Read_Text
- UARTx_Write
- UARTx_Write_Text

Da bi se uspostavila komunikacija sa modulom potrebno je utvrditi odgovarajuće poruke koje moraju biti razmenjene u svrhu konektovanja na mrežu i odgovaranja na zahteve klijenta.

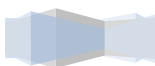
Za ovu komponentu ne postoji zvaničan data-sheet ali ona implementira set AT instrukcija pa ćemo koristiti sledeću dokumentaciju za instrukcije:

https://cdn.sparkfun.com/assets/learn_tutorials/4/0/3/4A-ESP8266_AT_Instruction_Set_EN_v0.30.pdf

Utvrdjemo potreban set komandi:

- **AT+RESTORE** - resetuje sva podešavanja
- **AT+CWMODE_CUR=1** – setuje modul u „Station mode“
- **AT+CWJAP_CUR="AP Name","Password"** – Konektuje se na zadatu bežičnu mrežu
- **AT+CIFSR** – Dohvata dodeljenu ip adresu
- **AT+CIPMUX=1** – Omogućava više konekcija na modul
- **AT+CIPSERVER=1,80** – Otvara TCP server na zadatom portu
- **+IPD** – Označava dolazak zahteva
- **AT+CIPSEND=connection,length**– Slanje poruke određenom klijentu, određene dužine
- **AT+CIPCLOSE=x** – Zatvara zadatu konekciju

Sada kada imamo potrebnu biblioteku i znamo potreban set komandi sa lakoćom možemo implementirati zahtevanu komunikaciju.



Dizajn softvera

Softver je pisan modularno tako da svaka hardverska komponenta ima svoj softverski modul.

Postojeći moduli su:

- Display
- Temperature
- Network
- LED
- Main

Opis pojedinih modula sledi u nastavku.

Display

Ovaj modul obuhvata fajlove display.c i display.h. U njima su deklarirani pinovi na koje je nakačen displej kao i pomoćna funkcija za ispis poruke čiji potpis sledi.

```
void displayLoadingMessage(int row, char * message, unsigned long sleep_time_ms);
```

Ova funkcija u redu **row** ispisuje poruku **message** a potom „spava“ **sleep_time_ms** milisekundi.

Temperature

Ovaj modul obuhvata fajlove temperature.c i temperature.h.

U njemu su deklarirane sledeće promenljive:

- **high_val**, **low_val** i **current_val** koje sadrže vrednosti sirovih očitavanja senzora. **current_val** sadrži poslednju iščitanu vrednost dok **low_val** i **high_val** čuvaju najnižu i najvišu temperaturu, respektivno
- **high_val_str**, **low_val_str** i **current_val_str** sadrže tekstualnu reprezentaciju u stepenima Celzijusove skale

Funkcije:

- **unsigned int fetchRawMeasurement()** funkcija služi za iščitavanje sirove vrednosti temperature sa senzora
- **float convertRawToFloat(unsigned int raw)** – konvertuje sirovu vrednost u vrednost u pokretnom zarezu
- **void renderTemperatureStrings()** – konvertuje sirove vrednosti u stringove
- **void displayTemperature()** – prikazuje temperature na displeju

Network

Ovaj modul obuhvata fajlove network.c i network.h.



U njemu su deklarisanе sledeće promenljive:

- **char buffer[4096]** i **volatile unsigned int position = 0** – bafer i njegov brojač su zaduženi da smeštaju podatke pristigle od Wi-Fi modula
- **char dynamic_ip[16]** smešta dobijenu ip adresu od strane mreže

Funkcije:

- **void initNetwork()** – inicijalizuje bafer i serijsku komunikaciju sa modulom
- **void initNetworkTransferInterrupt()** – omogućava prekide tako da se prenos podataka vrši asinhrono
- **void enableNetworkTransferInterrupt()** i **void disableNetworkTransferInterrupt()** omogućavaju i onemogućavaju pristizanje podataka prekidom kada je to potrebno (npr. očitavanje temperature)
- **void UART3_Interrupt()** i **IVT_INT_USART3 ics ICS_AUTO** – sama prekidna rutina koja se poziva kada god postoji podatak na serijskoj vezi, prima se i čuva u bafer
- **void sendCmd(char * cmd, unsigned long sleep_time_ms)** i **void sendCmdAndPrintResponse(char * cmd, unsigned long sleep_time_ms)** – služe za slanje poruke na modul, čekanje i štampanje odgovora u svrhu debugovanja i logovanja
- **int extractIp()** – parsira primljene podatke i pokušava da dohvati dodeljenu adresu, po uspešnom pronalasku vraća 1 u suprotnom 0

LED

Ovaj modul je mali omotač koji olakšava paljenje i gašenje dve LE diode na razvojnoj ploči.

Moguće je pozivanje funkcije:

`void setLed(int ledId, int value)` gde je `ledId` 1 ili 2, a `value` je 0 ili 1.

Takođe je moguće bilo gde u kodu koristiti promenljive `LED1` i `LED2` i dodeljivati im vrednosti 0/1 i time se postiže paljenje i gašenje dioda.

Main

U ovom delu je realizovana kontrolna logika aplikacije. Odavde se pozivaju funkcije za podešavanje Wi-Fi modula i štampaju statusne poruke na LCD displeju.

Postoji beskonačna petlja koja čeka pristigle zahteve, meri temperaturu u tom trenutku i šalje korisniku stranicu sa ažurnim informacijama.



Uputstvo za prevođenje

Ceo projekat je realizovan u **mikroC PRO for ARM v4.7.0** razvojnom okruženju.

Korišćen je **mikroCARM.exe** kompajler.

Olakšano korisničko iskustvo se može dobiti tako što se koristi isto razvojno okruženje i tada je potrebno samo učitati .mcpa fajl i razvojno okruženje će se pobrinuti za ostalo.

Ishod prevođenja je .hex fajl koji se može pomoću mikroElektronika USB HID Bootloader-a (v2.5.0.0) „spustiti“ na razvojnu ploču.



Uputstvo za upotrebu

1. Potrebno je kompajlirati projekat sa odgovarajućim kredencijalima za lokalnu mrežu i spustiti .hex fajl na razvojnu pločicu
2. Resetovanjem razvojne pločice se započinje rad
3. Na displeju će se smenjivati statusne poruke
 - a. Starting App...
 - b. Setup as client
 - c. Connecting to AP
 - d. Couldn't join AP
 - e. Connected, [IP address]

U slučaju situacije 3.d. potrebno je proveriti podatke bežične mreže i ispraviti potencijalnu grešku. Ukoliko nema greške, treba proveriti LED1 na razvojnoj pločici. Ukoliko ne svetli, znači da nije uspela komunikacija sa Wi-Fi modulom i treba proveriti konekcije i po potrebi restartovati pločicu.

4. Nakon ispisane IP adrese (3.e.) moguće joj je se pristupiti iz okvira iste mreže preko internet browsera
5. Ukoliko se ne učita stranice iz prve treba osvežiti stranicu, ali ne previše puta jer ovaj Wi-Fi modul nije toliko snažan da može da podrži puno konekcija
6. Na stranici će biti prikazane trenutna temperatura kao i najniža i najviša izmerena

