

Mikroprocesorski sistemi

Dokumentacija – Tim 2



Profesor:

Aleksandar Peulić

Članovi tima:

Aleksandar Milutinović

63/2020

Maksim Šapić 58/2021

Nikola Lalić 52/2021

Nevena Tomić

89/2021

Luka Ivanović 93/2021

1. Uvod

Ovaj projekat prikazuje implementaciju komunikacionog sistema baziranog na ESP32 mikrokontroleru, LoRa bežičnoj komunikaciji i MQTT protokolu. Sistem se sastoji iz:

- LoRa čvora (Node) – uređaj koji periodično šalje podatke putem LoRa protokola.
- LoRa prijemnika (Receiver) – uređaj koji prima pakete od čvora.
- LoRa gateway-a – uređaj koji povezuje LoRa mrežu sa WiFi mrežom i prosleđuje podatke ka MQTT brokeru.

Ovim pristupom omogućeno je da se podaci sa senzora (ili generisane poruke) prenesu na udaljeni server i dalje obrađuju.

2. Korišćene biblioteke i softverski alati

- SPI.h – omogućava komunikaciju ESP32 mikrokontrolera sa LoRa modulom putem SPI magistrale.
- LoRa.h – biblioteka za upravljanje SX1276/78 LoRa transiverom.
- WiFi.h – omogućava povezivanje ESP32 na WiFi mrežu.
- PubSubClient.h – implementacija MQTT klijenta za ESP32.

CP210x Universal Windows Driver

CP210x je drajver za USB–UART most koji razvija kompanija **Silicon Labs**.

- Koristi se za komunikaciju između računara i mikrokontrolera preko USB porta.
- Mnogi razvojni moduli (uključujući i ESP32 dev board) imaju ugrađen **CP2102/CP2104 čip**, koji omogućava da se mikrokontroler prepozna kao serijski (COM) port na Windows sistemu.
- Instalacijom **CP210x Universal Windows Driver-a** obezbeđuje se stabilno prepoznavanje uređaja i mogućnost programiranja ESP32 preko Arduino IDE-a ili drugih razvojnih okruženja.

ESP32 by Espressif

ESP32 by Espressif Systems predstavlja razvojni softverski paket i hardversku platformu za rad sa ESP32 mikrokontrolerima.

- Espressif obezbeđuje:
 - **ESP-IDF (IoT Development Framework)** – zvanično razvojno okruženje za programiranje ESP32 u C/C++,
 - **Arduino Core for ESP32** – omogućava korišćenje Arduino IDE-a i poznatih Arduino biblioteka za rad sa ESP32,
 - biblioteke i drajvere za periferije (SPI, I2C, UART, PWM, ADC, DAC, WiFi, Bluetooth, LoRa preko dodatnih modula).
 - ESP32 by Espressif paketi omogućavaju lako integrisanje uređaja u IoT sisteme putem različitih mrežnih protokola (MQTT, HTTP, WebSocket itd.).

4. Povezivanje LoRa modula i ESP32

LoRa modul koristi SPI komunikaciju. Pinovi se definišu pomoću #define direktiva:

```
#define LORA_SCK 5  
#define LORA_MISO 18  
#define LORA_MOSI 23  
#define LORA_SS 17  
#define LORA_RST 14  
#define LORA_DIO0 26
```

- LORA_SCK, LORA_MISO, LORA_MOSI – SPI linije.
- LORA_SS – Chip Select pin.
- LORA_RST – reset pin LoRa modula.
- LORA_DIO0 – prekidni pin (signalizuje dolazak paketa).

5. LoRa čvor (Node)

Sender (pošiljalac) periodično šalje poruke putem LoRa modula. Svaka poruka sadrži indeks paketa i tekstualnu poruku. Koristi SPI interfejs da šalje podatke preko RFM95 modula. Inicijalizacija:

```
SPI.begin(LORA_SCK, LORA_MISO, LORA_MOSI, LORA_SS);  
LoRa.setPins(LORA_SS, LORA_RST, LORA_DIO0);  
LoRa.begin(868E6); // frekvencija 868 MHz (za Evropu)
```

Slanje paketa:

```
String message = "Node1 Packet#" + String(packetIndex++);  
LoRa.beginPacket();  
LoRa.print(message);  
LoRa.endPacket();
```

6. LoRa prijemnik (Receiver)

Receiver (prijemnik) služi za prijem LoRa paketa. Povezuje se na RFM95 modul i koristi SPI interfejs za komunikaciju. Kada stigne paket, ispisuje ga u serijski monitor zajedno sa RSSI vrednošću (jačinom signala) :

```
int packetSize = LoRa.parsePacket();  
if (packetSize) {  
    while (LoRa.available()) {  
        Serial.print((char)LoRa.read());  
    }  
    Serial.print(" with RSSI ");  
    Serial.println(LoRa.packetRssi());  
}
```

7. LoRa Gateway (ESP32 sa WiFi + MQTT)

Gateway je centralna tačka sistema. Prima podatke od LoRa čvorova putem RFM95 modula, a zatim ih prosleđuje na MQTT broker (HiveMQ) putem WiFi mreže. Time omogućava integraciju LoRa mreže sa internetom i drugim aplikacijama koje koriste MQTT protokol.

Povezivanje na WiFi:

```
WiFi.begin(ssid, password);  
while (WiFi.status() != WL_CONNECTED) {  
    delay(500);  
}  
Serial.println(WiFi.localIP());
```

Konfiguracija MQTT klijenta:

```
client.setServer(mqtt_server, mqtt_port);  
client.connect("ESP32Gateway");
```

Primanje i slanje paketa na MQTT:

```
int packetSize = LoRa.parsePacket();  
if(packetSize) {  
    String message = "";  
    while(LoRa.available()) {  
        message += (char)LoRa.read();  
    }  
    client.publish(mqtt_topic, message.c_str());  
}
```

8. Rezultati rada

1. Čvor periodično šalje pakete putem RFM95 modula..
2. Prijemnik ili gateway prima pakete i prikazuje ih preko serijskog monitora.
3. Gateway prosleđuje pakete na MQTT broker (broker.hivemq.com).
4. Na ovaj način, podaci sa senzora ili poruke sa čvorova dostupni su bilo kojoj aplikaciji koja je povezana na MQTT server.

9. Zaključak

Implementacijom ovog sistema uspešno je realizovan komunikacioni lanac: LoRa Node → LoRa Gateway (ESP32) → MQTT broker → korisničke aplikacije.

Sistem se može proširiti dodavanjem senzora na LoRa čvorove (temperatura, vlagasvetlost), čime se dobija kompletno rešenje za pametne gradove, poljoprivredu ili kućnu automatizaciju.

10. Hardverske komponente

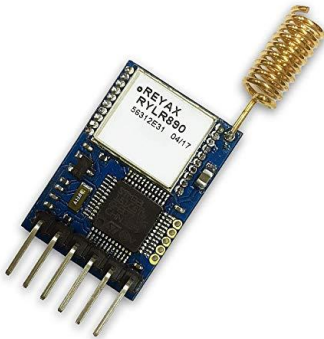
1. ESP32 x2

- Glavni mikrokontroler, koristi se za slanje i prijem LoRa paketa, kao i za gateway funkcionalnost (WiFi + MQTT).



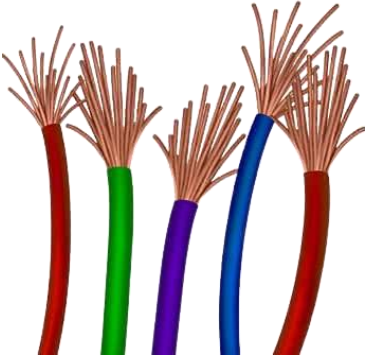
2. LoRa modul (SX1276/78)

- LoRa modul omogućava dalekosežnu komunikaciju sa niskom potrošnjom energije na frekvenciji 868 MHz.



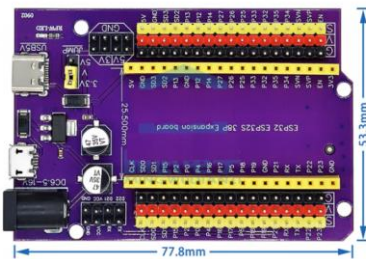
3. Žice za povezivanje (Dupont kablovi)

Koriste se za povezivanje ESP32 ploče sa LoRa modulom preko SPI interfejsa.



4. ESP32 Base Board (podloga) x1

- Podloga/adapterska ploča za ESP32 modul.
- Omogućava:
 - USB povezivanje sa računarom radi programiranja i napajanja,
 - pretvaranje pinova modula u praktične header pinove,
 - korišćenje dugmadi RESET i BOOT,
 - stabilizaciju napajanja (5V → 3.3V).
- U projektu služi da olakša rad i povezivanje ESP32 modula sa RFM95 i proto pločom.



11. Povezane komponente

Reciver(stm32 ploča, lora modul, stm32 base):



Sender(stm32 ploča, lora modul):



12. Tok razvoja projekta

Razvoj projekta tekao je kroz nekoliko logičkih faza:

1. Istraživanje i analiza

U prvoj fazi istraživanja su mogućnosti korišćenja LoRa tehnologije na mikrokontrolerima tipa ESP32.

- Upoređivane su različite biblioteke (Arduino LoRa, PubSubClient za MQTT, SPI komunikacija).
- Analizirano je kako se LoRa čvorovi mogu povezati sa MQTT brokerom i kako ESP32 može raditi kao gateway.

2. Pisanje i priprema koda

Na osnovu istraživanja pripremljeni su različiti fajlovi za pojedinačne uloge sistema:

- **Sender (čvor)** – slanje poruka preko LoRa modula
- **Receiver** – prijem poruka i ispis na serijski monitor
- **Gateway** – prijem LoRa poruka i prosleđivanje na MQTT broker putem WiFi mreže
- **Node** – dodatni čvor koji šalje telemetriju

Kod je pisan i testiran modularno – svaka komponenta sistema razvijana je odvojeno kako bi se lakše uočile greške.

3. Povezivanje hardverskih komponenti

Nakon validacije koda, ESP32 moduli su povezani sa RFM95 LoRa modulima pomoću lemljenih provodnika.

- Korišćen je ESP32 base board koji omogućava stabilno napajanje, lako povezivanje pinova i jednostavno programiranje preko USB-a.
- Posebna pažnja posvećena je povezivanju SPI pinova (MOSI, MISO, SCK, CS) i kontrolnih pinova (RST, DIO0).

4. Testiranje funkcionalnosti

- Prvobitno je testirana osnovna LoRa komunikacija između jednog sender-a i receiver-a.
- Nakon uspešnog prijema paketa, sistem je proširen gateway-jem, koji je preko WiFi mreže povezan sa MQTT brokerom (HiveMQ).

- Testirano je da li se poruke sa čvorova ispravno prenose kroz **LoRa → Gateway → MQTT** lanac.
- Na kraju je izvršeno stres testiranje sa više poslatih paketa i praćenje RSSI vrednosti za procenu jačine signala.

5. Iteracije i optimizacija

- Prilagođavana je učestalost slanja paketa i format poruka
- Dodata je serijska dijagnostika za lakše praćenje grešaka
- Proverena je stabilnost konekcije sa WiFi mrežom i MQTT brokerom

6. Finalizacija sistema

- Sistem je uspešno integrisan tako da više čvorova može slati podatke, koje gateway prosleđuje na centralni MQTT server
- Dokumentovana je hardverska šema i funkcionalna podela koda