

Prompter ESP

Introducere

Am folosit ESP8266 si o matrice led pentru a face un fel de Promoter. Trimit text de pe telefon si se afiseaza pe matricea led in derulare. Am pornit de la ideea de a face ceva apropiat de IoT prin care sa aduc mai aproape partea aceasta de microprocesoare de oamenii de rand care folosesc telefonul in fiecare zi. Cu mici ajustari acesta poate fi folosit pe post de prompter adevarat.

Descriere generală

Folosim o aplicatie de android pe telefon prin care scriem IP-ul dispozitivului ESP (trebuie conectate la aceeași rețea locală) și un text pe care vrem să-l afișăm și apăsăm pe butonul "Trimite". Aplicația de android va face un POST Request pe ip-ul ESP-ului, iar în body va introduce textul pe care dorim să-l afișăm. ESP-ul primește textul și îl va afișa derulându-se pe matrice până când cineva apasă pe buton pentru a opri derularea acestuia. În timp ce se derulează va fi aprins și un LED care semnifică că mesajul este în afișare. Când se apasă pe buton, derularea mesajului se oprește și se așteaptă al request. Și în timpul așteptării unui nou request, dar și în timpul afișării mesajului, putem rotiți de potenționemtru pentru a mări/micsora viteza de derulare a mesajului.

Hardware Design

Pentru parte de hardware am folosit:

1 x ESP8266

4 x Modul matrice LED FC-16 DIY

1 x Encoder rotativ hw-040 (folosit ca potenționemtru și buton)

Fire de legatură

1 x Mother board

Software Design

Pentru partea de android am folosit Android Studio. Pentru partea de ESP am folosit Arduino IDE.

Android:

Aplicatia porneste o activitate in care se afla 2 editor text si 1 buton. Pe un editor text trebuie sa trecem IP-ul ESP-ului, iar pe celalalt textul pe vrem sa-l trimitem. Pentru partea de request am folosit biblioteca Volley prin care cream un request de tip POST in care avem in body textul pe care vrem sa-l trimitem. Cand apasam pe butonul "Trimite" se trimite requestul catre ESP.

Biblioteci utilizate:

Volley

ESP:

In prima parte declaram pinii folositi, facem setarile matricei si ne conectam la retea. Dupa care in loop citim datele de pe potentiometru si in acelasi timp asteptam sa primim request-uri de la mobile. Cand primim un request, verificam daca textul este gol, iar daca nu este, afisam in continuu textul si se aprinde LED-ul pana la apasarea butonului. Pentru a nu folosii delay, in loc de acesta avem un do while in care citim potentiometrul pana trece un timp definit.

Biblioteci utilizate:

Max72xxPanel

Adafruit_GFX

ESP8266WebServer

ESP8266WiFi

Rezultate Obținute

Link rezultat video: https://youtube.com/shorts/ENO9x_B0sno?feature=share

Concluzii

Consider ca acest proiect m-a ajutat in intelegerea materiei, dar mai ales m-a apropiat de partea aceasta de hardware de care ma cam feream in trecut. Consider ca oportunitatile ce le ofera microprocesoarele sunt nelimitate si ca partea de IoT este cu siguranta printre domeniile viitorului.

Bibliografie/Resurse

https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf

https://majicdesigns.github.io/MD_MAX72XX/

<https://www.inro-electronics.com/hobby-robotic-parts/sensors/rotary-encoder-module-pulse-digital-hw-040?limit=100>

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/rtilimpea/prompter-esp>



Last update: **2022/06/02 14:31**