

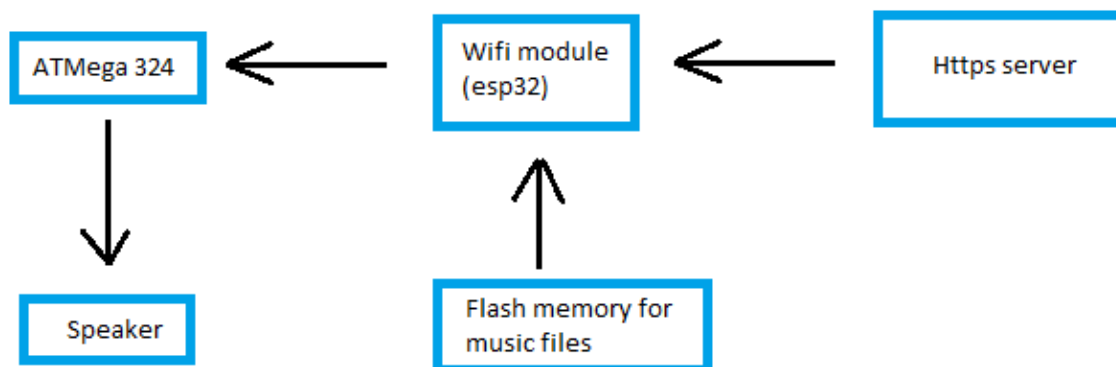
# Andrei LUPAȘCU (67047) - Wifi music player

Autorul poate fi contactat la adresa: **Login pentru adresa**

## Introducere

Vreau sa implementez un music player care poate fi accesat prin wifi. Consider ca este mult mai comod sa poti sa schimbi muzica din casa de pe orice device, remote.

## Descriere generală



## Hardware Design

Lista piese:

Componenta	Cantitate
Placa de bază	1
Buzzer pasiv	1
Esp32	1
Leduri	2
Fire mama-mama	5

## Software Design

Mediu de dezvoltare :

1. Arduino IDE
2. Programmer's notepad

Biblioteci :

1. util/delay
2. avr/io
3. avr/interrupt

Biblioteci esp32:

1. app <https://github.com/eerimoq/pumbaa>
2. 2nd stage bootloader  
[https://github.com/espressif/esp-idf/blob/master/components/bootloader/src/main/bootloader\\_start.c](https://github.com/espressif/esp-idf/blob/master/components/bootloader/src/main/bootloader_start.c)

Algoritmi folositi:

1. Crearea unui server http pe esp32, redirectarea requesturilor catre atmega

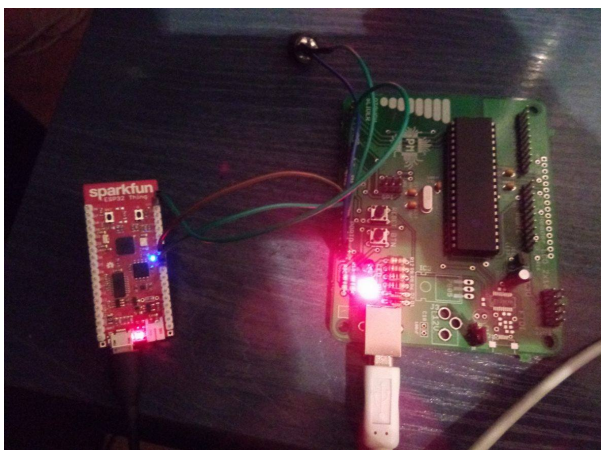
## Rezultate Obținute

Proiectul este gata. ESP32 deschide un host de wifi. Conectandu-ne la el, avem acces la un server http deschis pe aceeași placuță.



Cele 2 butoane trimit requesturi REST către placuță; În funcție de request-ul primit, esp32 semnalează către atmega operația de "play/stop" sau operația de "next song". Atmega așteaptă întreruperi asincrone și redă melodii pe buzzer-ul pasiv.

În poza de mai jos se observă atmega, modulul wifi ( esp32 ) și buzzerul conectate prin cabluri mama-mama.



## Concluzii

Aceasta a fost prima data cand am construit o componenta hardware si am fost incantat ca am putut pune in practica cunostintele de la electronica. Implementarea este usoara, am gresit la implementarea spi-ului si nu am putut face stream-ul melodiilor de pe esp pe atmega, pierzandu-se astfel o functionalitate pe care voiam s-o implementez ( upload-ul melodiilor din serverul http ) . M-au ajutat foarte mult cunostintele dobandite la laboratoare si am fost profund incantat de asistenta oferita de laboranti pentru acest proiect, chiar si in afara programului scolar.

## Download

Am atasat mai jos codul de pe esp32, cel de pe atmega si un foarte scurt ghid pentru compilarea codului pe esp32

[111iotmusicplayer.zip](#)

## Jurnal

04.04.2017: Am comandat pentru piesele hardware necesare

12.04.2017: Am luat piesele, am inceput sa lipesc placa de baza

14.04.2017: Am luat esp32, am scris serverul http pe esp32

15.04.2017: Am scris logica de redare a muzicii pentru atmega

23.04.2017: Atmega este montat, functioneaza, am pus bootloaderul la laborator si am urcat codul scris in data de 15.04

24.04.2017: Am implementat trimiterea de date prin SPI intre atmega si esp32. Am vazut ca nu merge. Am rescris trimiterea de semnale prin intreruperi asincrone.

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

- Documentația în format [PDF](#)

Last update: 2021/04/14 15:07 pm:prj2017:mandrei:spercanuamaialesnimeniasta <http://ocw.cs.pub.ro/courses/pm/prj2017/mandrei/spercanuamaialesnimeniasta>

---

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2017/mandrei/spercanuamaialesnimeniasta>



Last update: **2021/04/14 15:07**