

Textul și imaginile din acest document sunt licențiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



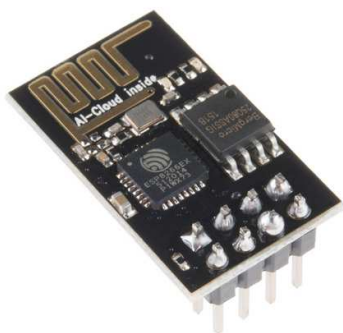
Codul sursă din acest document este licențiat

Public-Domain

Ești liber să distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, tipărire, sau orice alt mijloc), atât timp cât nu aduci nici un fel de modificări acestuia. Codul sursă din acest document poate fi utilizat în orice fel de scop, de natură comercială sau nu, fără nici un fel de limitări dar autorii nu își asumă nici o răspundere pentru pagubele pricinuite de implementările realizate de utilizatori. Schemele și codul sursă au un rol educativ și nu sunt gândite pentru a fi utilizate în mediu de producție (industrial, casnic sau comercial).

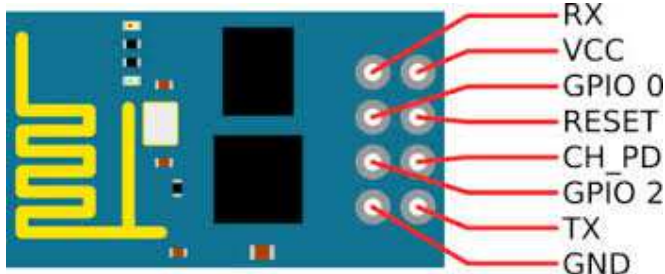
Buton Sonerie WiFi

Chiar dacă vechile sonerii (manuale sau cu fir) își au un farmec aparte, totuși, soneriile fără fir au devenit un lucru obișnuit la momentul actual datorită ușurinței de instalare. În cadrul lecției de față vă propunem realizarea unei sonerii formată doar din butonul de afară – avertizarea sonoră specifică va fi înlocuită, grație unei conexiuni WiFi, cu trimiterea unui email sau o avertizare pe telefonul mobil. Utilizând o astfel de sonerie puteți fi anunțat că este cineva la ușă chiar dacă nu sunteți acasă.



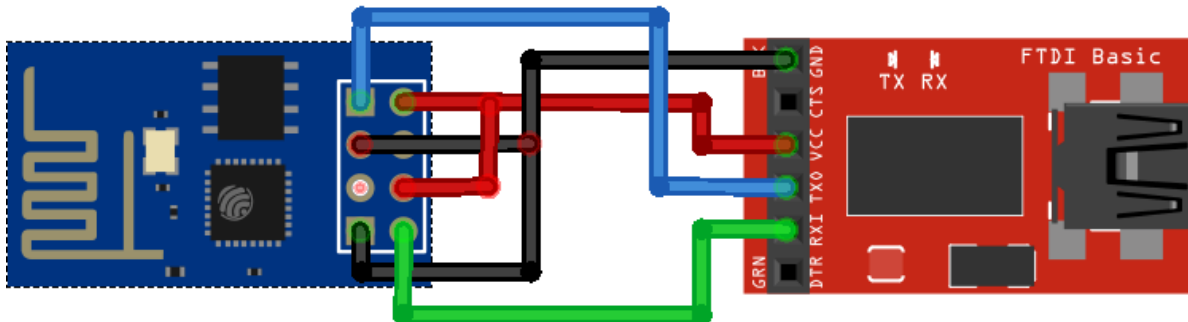
Pentru implementare vom utiliza un modul WiFi ESP8266-01S ([1](#)). Acest tip de modul bazat pe circuitul ESP8266 are un cost foarte mic dar este destul de dificil de utilizat spre deosebire de alte plăci de dezvoltare bazate pe același circuit. Modulul este gândit să fie utilizat ca modul WiFi serial pentru alte plăci de dezvoltare, vine preprogramat cu un firmware ce acceptă comenzi de tip AT. În plus, forma conectorului de pe acest modul nu dă posibilitatea de a fi folosit cu un breadboard. În cadrul lecției de față vom programa direct modulul fără a utiliza facilitatea de comunicație serială AT, vom utiliza modulul ca placă de dezvoltare. Pentru interconectarea modulului recomandăm utilizarea unor fire de interconectare de tip mamă-tată pentru a putea face legăturile cu un breadboard. Pentru mai multe informații legate de funcționarea modulului puteți consulta și materialul „ESP8266 WiFi Module for Dummies” ([2](#)).

Programarea modulului necesită un modul FTDI de 3.3V ([3](#)) și instalarea extensiei ESP8266 Community sub Arduino IDE. Pentru mai multe detalii puteți consulta și materialul „How to Install the ESP8266 Board in Arduino IDE” ([4](#)). Conexiunile dintre modulul ESP8266-01S și programatorul FTDI necesare încărcării programului sunt următoarele:



- Pinii GND și GPIO0 ai modulului ESP8266 se conectează la pinul GND al programatorului. Pinul GPIO0 este necesar să fie conectat la masă pentru programare.
- Pinii VCC și CH_PD ai modulului

ESP8266 se conectează la pinul VCC (adică 3.3V!!!) al programatorului. Pinul CH_PD se va menține conectat la VCC pe tot parcursul funcționării modulului – este pinul de activare al modulului.



- Pinul RX al modulului ESP8266 se conectează la pinul TX al programatorului.
- Pinul TX al modulului ESP8266 se conectează la pinul RX al programatorului.

Programul a fost dezvoltat și testat utilizând Arduino IDE 1.8.3 cu extensia ESP8266 Community 2.3.0 instalată. Pentru programare se selectează placa de dezvoltare "Generic ESP8266 Module". În cadrul programului trebuie personalizate datele de conectare WiFi (variabilele *ssid* și *pass*).

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>

char ssid[] = "...";
char pass[] = "...";
WiFiClient client;

void setup() {
  WiFi.begin(ssid, pass);
  delay(5000);
  if (WiFi.status() != WL_CONNECTED) {
    delay(60000);
    ESP.restart();
  }
}
```

```

    IFTTTpublish();
    ESP.deepSleep(0);
}

void loop() {
}

```

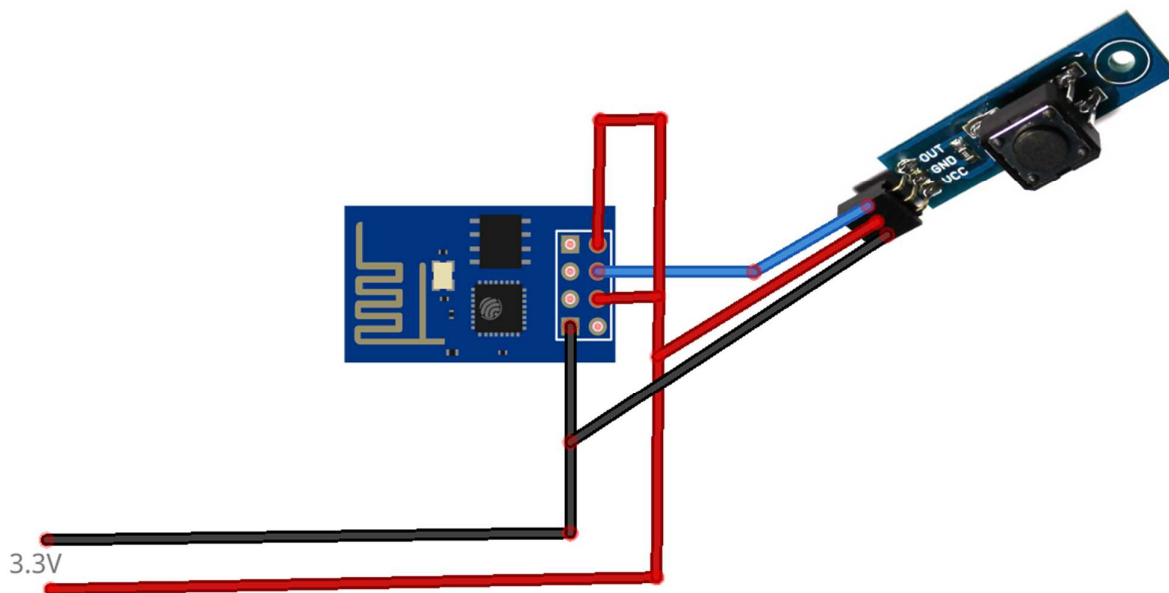
Programul transmite prin Internet către serviciul IFTTT ([5](#)) evenimentul de apăsare a butonului (procedura *IFTTTpublish*). Serviciul IFTTT permite redirectarea evenimentului către utilizator sub forma unui email, a unui avertizări pe telefonul mobil sau orice altă variantă dorită. În cadrul procedurii trebuie personalizate datele de identificare a evenimentului IFTTT Webhooks (variabilele *KEY* și *EVENT*) prin care se face legătura între sistemul nostru și platforma IFTTT.

```

void IFTTTpublish() {
    String KEY = "...";
    String EVENT = "ButonSonerie";
    HTTPClient http;
    String data = String("http://maker.ifttt.com/trigger/" +
        EVENT + "/with/key/") + KEY ;
    http.begin(data);
    int httpCode = http.GET();
    delay(500);
    http.end();
}

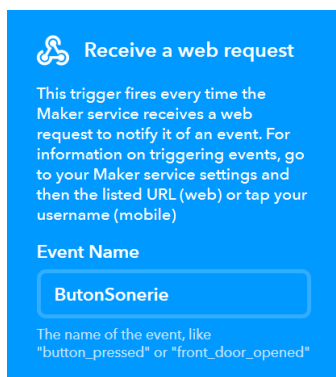
```

Sistemul nostru va funcționa în regim de consum redus în mod continuu (indus de instrucțiunea *ESP.deepSleep(0)* din secțiunea *setup*). Butonul va fi conectat pe pinul de RESET al modulului și va activa sistemul doar pentru a semnaliza evenimentul către platforma IFTTT după care va intra din nou în regim de consum redus. Acest lucru este extrem de util dacă dorim să alimentăm sistemul nostru de la o baterie sau un acumulator. Pentru buton vom utiliza un brick ([6](#)) pentru o interconectare mai ușoară. Brick-ul este gândit să tragă linia OUT la VCC dar pinul de RESET al modulului ESP8266 necesită tragerea la masă din acest motiv vom inversa conectarea brick-ului (vom conecta VCC la GND și GND la VCC):



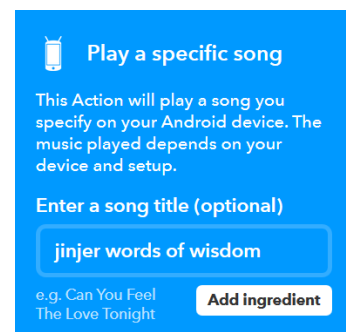
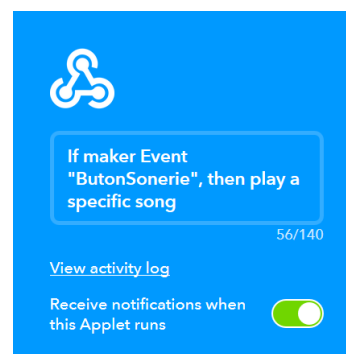
Atenție!!! Modulul ESP8266-01S nu are regulator de tensiune integrat. El trebuie alimentat fix la 3.3V. În plus necesită un curent destul de mare, nu poate fi alimentat din programatorul FTDI decât pe perioada programării. Pentru testare recomandăm o sursă de breadboard ([7](#)) și pentru funcționarea autonomă un regulator Step-Down de 3.3V (de exemplu [8](#)) pentru o alimentare de la un acumulator LiPo de 3.7V sau 7.2V.

Utilizarea serviciului IFTTT necesită înregistrare (gratuită). După înregistrare și conectarea la platforma IFTTT vom crea o nouă aplicație (*New Applet*) în care evenimentul *this* va fi de tipul *Webhooks* iar efectul *that* va fi de tipul *Android Device* / *play a specific song*. Bineînțeles, așa cum am



precizat anterior, efectul poate fi modificat după dorință în *Email*, *SMS*, postare web etc. sau ce vi se pare potrivit ca avertizare pentru butonul de sonerie. Este adevărat că sistemul propus în această lecție poate servi la fel de bine ca buton de panică, buton de recepție sau buton pentru servitori ☺. În cadrul efectului *Play a specific song* se va specifica denumirea unei piese aflată pe

mobilul care trebuie să aibă aplicație IFTTT instalată. Dacă melodia nu este găsită se va deschide în mod automat pagina Youtube și va fi căutată. Vă recomandăm să încercați ”jinjer words of wisdom”.



Referințe on-line

(1) ESP8266 WiFi

https://www.robofun.ro/wifi-module-esp8266?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(2) ESP8266 WiFi Module for Dummies

<http://www.instructables.com/id/ESP8266-WiFi-Module-for-Dummies/>

(3) Conector FTDI 3.3 V

https://www.robofun.ro/conector-ftdi-3V3?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(4) How to Install the ESP8266 Board in Arduino IDE

<http://randomnerdtutorials.com/how-to-install-esp8266-board-arduino-ide/>

(5) Learn how IFTTT works - IFTTT

<https://ifttt.com/>

(6) Buton Mare Brick

https://www.robofun.ro/buton-mare-brick?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(7) Breadboard Power Supply Stick - 5V/3.3V

https://www.robofun.ro/breadboard-power-supply-stick-5v-3-3v?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(8) Regulator Step-Down 3.3V, 500mA D24V5F3

https://www.robofun.ro/pololu-3-3v-500ma-step-down-voltage-regulator-d24v5f3?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL