

Dispozitiv Multifunctional

Introducere

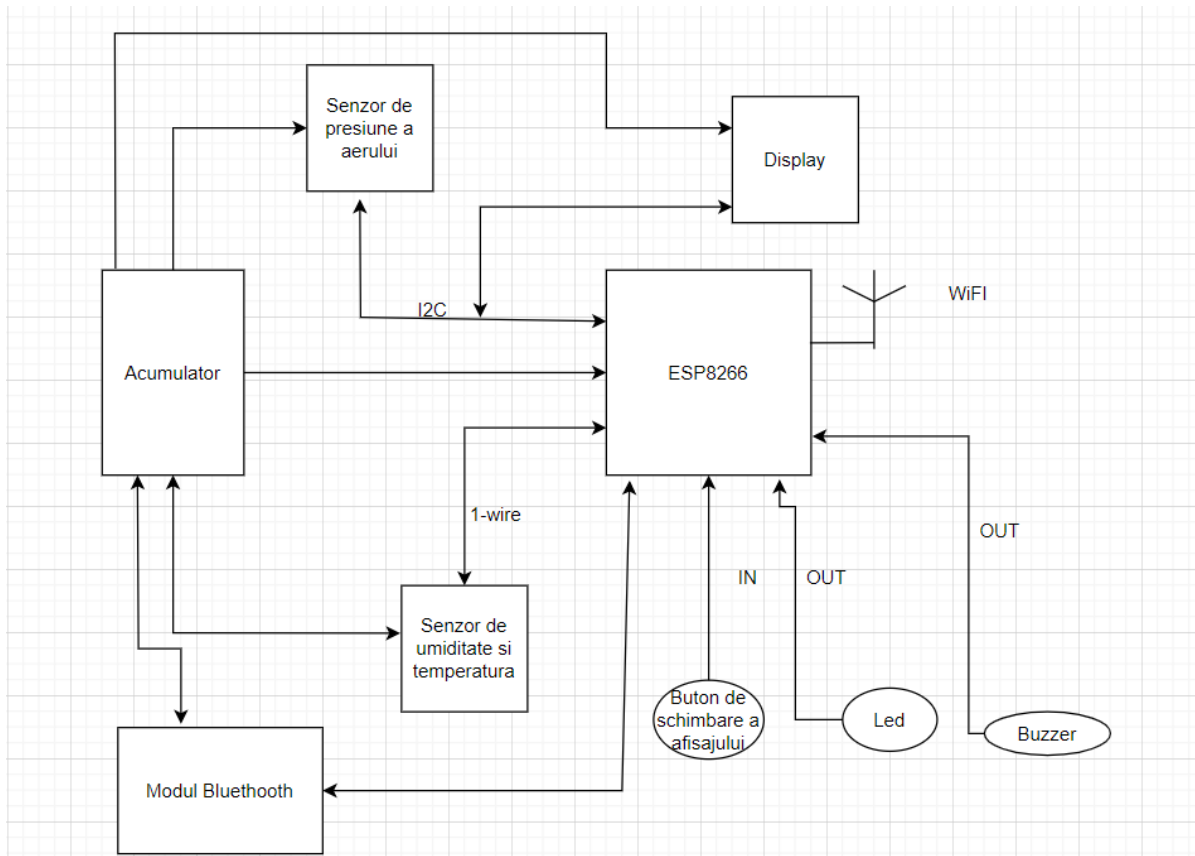
Proiectul consta in proiectarea unui Dispozitiv Multifunctional care va afisa ora si data curenta in functie de localizare, informatii pe care le va accesa de pe internet. In cazul in care nu are acces la internet, acesta va avea si un ceas de timp real. Sincronizarea se face automat la ora 12 noaptea.

Mai mult decat atat, ceasul va afisa o data la cateva secunde pe ecran un ceas analog, unul digital si informatiile citite de senzorul de presiune atmosferica, cel de temperatura si cel de umiditate. Acest afisaj se poate schimba si prin apasarea unui buton. Datele intoarse de senzori vor fi puse pe un server web la fiecare minut (ThingsSpeak) unde vor fi desenate si grafice pentru date, astfel avand acces si remote la acestea. Prin intermediul modulului bluetooth o sa trimit datele si pe telefon.

Ceasul poate avea si o alarma setata de pe telefon. Acest lucru se realizeaza prin intermediul modulului bluetooth. Cand ora si minutul trimise de pe telefon reprezinta timpul curent, se va aprinde un led si va suna un buzzer timp de 5 secunde.

Descriere generală

Schema bloc:

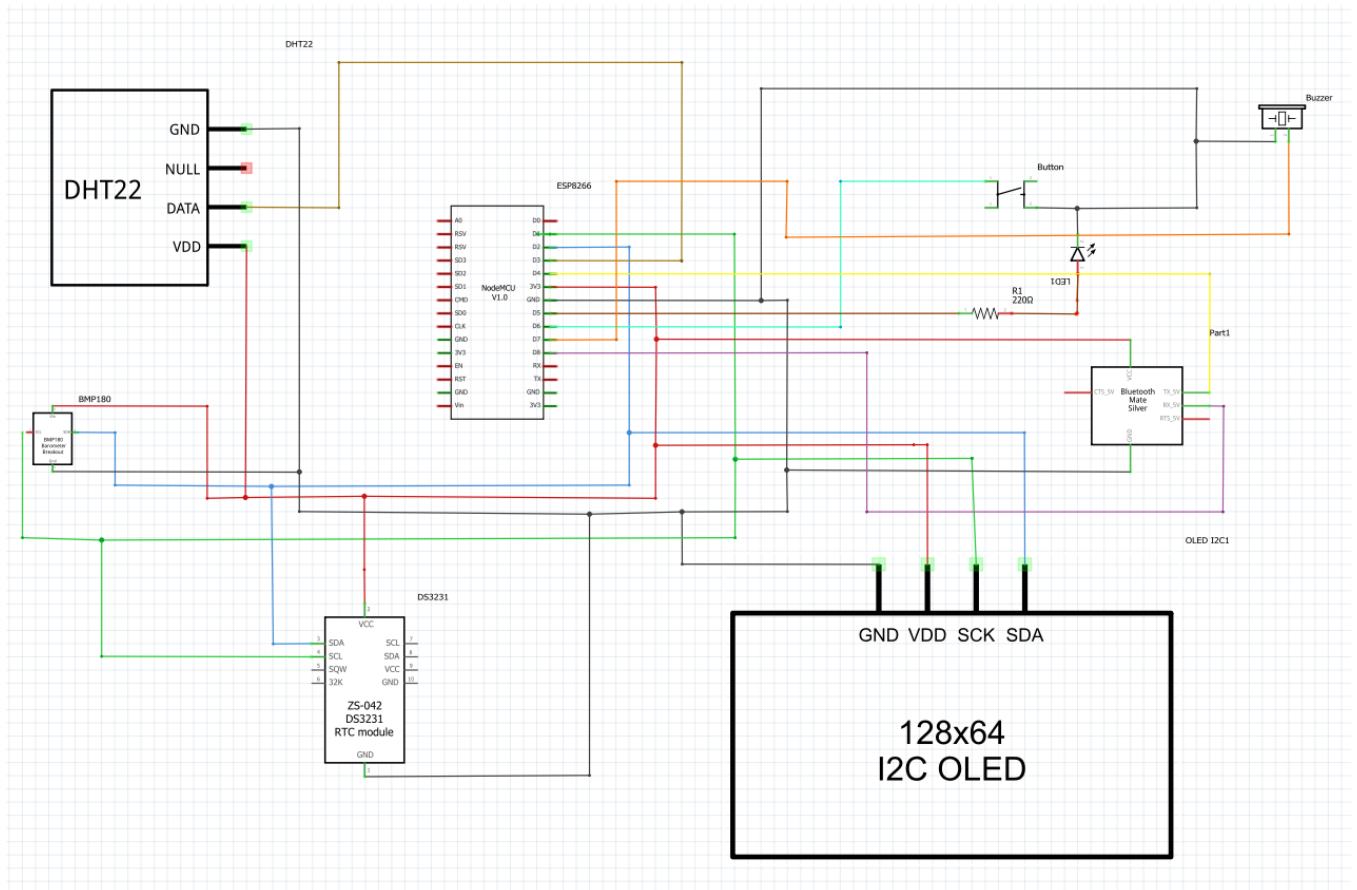


Mod de functionare:

ESP-ul interogheaza senzorul de temperatura si umiditate si cel de presiune atmosferica doar cand e in ecranul care le afiseaza. Definesc 3 ecrane, 1 pentru afisarea unui ceas analog, 1 pentru afisarea cesului digital impreuna cu data si locatia si 1 pentru temperatura, presiunea aerului si umiditate. Acestea sunt schimbate la un interval de 5 secunde. Sincronizarea timpului se realizeaza la ora 12 seara si la pornirea ceasului. Datele colectate de senzori sunt transmise pe server o data pe minut. Prin modulul bluetooth se poate seta de pe telefon o ora la care ceasul sa simuleze o alarma prinzand un led si sunand dintr-un buzzer.

Schema a fost realizata in Draw.io

Hardware Design



Lista de piese:

1. placuta ESP8266
2. breadboard
3. senzor de umiditate si temperatura DHT22
4. senzor de presiune atmosferica si temperatura BMP180
5. display oled SH1106
6. ceas de timp real DS3231
7. modul bluethooth HC05
8. acumulator
9. buton
10. rezistenta
11. buzzer
12. led

Schema a fost realizata in Fritzing

Software Design

Mediu de dezvoltare: Arduino IDE

Biblioteci:

- NTPClient.h
- IPGeolocation.h
- RtcDS1307.h
- OLEDDisplayUi.h
- SH1106Wire.h
- BMP180I2C.h
- DHT22.h
- ThingSpeak.h
- SoftwareSerial.h

Structura folosita de IPGeolocation: struct IPGeo {

```
String tz;  
int offset;  
String city;  
String country;  
String country_code;  
bool is_dst;  
double latitude;  
double longitude;
```

```
};
```

Explicarea codului:

1. Mi-am setat datele precum numele serverului, portul etc. pentru a putea realiza conexiunea la ThingSpeak, WiFi si bluetooth.
2. Am 3 functii pentru afisarea a 3 ecrane diferite: un ceas analog, unul digital unde se afiseaza si data si locatia si ultimul afiseaza datele citite de senzori. Pentru ceasul analog voi pune codul mai jos.
3. Pentru realizarea ceasului analog am folosit functiile puse la dispozitie de clasa OLEDDisplay(drawLine, drawString). In functie secunda / ora / minut am calculat unghiul la care ar trebui sa se afle limba ceasului, am calculat coordonatele x si y ale varfului si am apelat functia drawLine pentru a desena limba ceasului.
4. Functii ajutatoare:
 - twoDigits ma ajuta sa imi afiseze 0-urile din fata. De exemplu eu nu vreau sa afisez 9:00, ci 09:00
 - isStringDigitsOnly imi verifica ca stringul primit sa contina doar cifre sau caractere terminatoare precum '\0', '\n', '\r'
 - getLunaFromString imi intoarce luna in
 - wasError imi afiseaza prin bluetooth ce eroare am primit de la RTC
 - printDateTime imi afiseaza data si timpul pe telefon
5. In setup() mi-am setat buzzer-ul, led-ul si butonul, am setat display-ul specificand numarul de FPS-uri, indicatorul, frame-urile, animatia de slide etc, am realizat conexiunea la WiFi si la bluetooth apoi am facut un request pentru a afla datele geografice ale retelei mele WiFi la care m-am conectat. Parsand raspunsul primit, am aflat tara, orasul, offset-ul pentru timp (+3) pentru Romania pe timp de vara si data. Am facut si un request la NTP pentru a citi timpul de pe net si a sincroniza ceasul de timp real cu ce am citit de pe net.
6. In functia loop() actualizez ora, minutul, secunda, data la fiecare secunda. La ora 12 se face sincronizarea timpului si a datii cu cele citite de pe net. O data pe minut trimit datele catre ThingSpeak, iar inainte de asta verific sa am conexiune la internet. In cazul in care nu am, voi incerca sa fac o reconectare. Daca m-am reconectat, voi face si resincronizare intre RTC si timpul

citit de pe net. Tot in loop() verific daca am setat o alarma de pe telefon. Voi primi un string de forma HH:MM care reprezinta ora la care sa sune alarma. Daca stringul nu respecta formatul, voi trimite un mesaj in care spun ca alarma nu s-a putut seta din cauza formatului gresit al stringului. Daca timpul nu este unu valid (de exemplu 12:72), voi afisa un mesaj care imi spune sa aleg un timp valid, iar daca totul este ok voi afisa un mesaj in care spune ca alarma a fost setata pentru ora introdusa. La alarma voi aprinde un led si va suna buzzer-ul timp de 5 secunde(buzzerul va suna de 3 ori in acest interval).

Folosind functiile implementate in clasa OLEDDisplay am reusit sa afisez un ceas analog:

Code snippet:

```
void analogClockFrame(OLEDDisplay *display, OLEDDisplayUiState* state, int16_t x, int16_t y) {
```

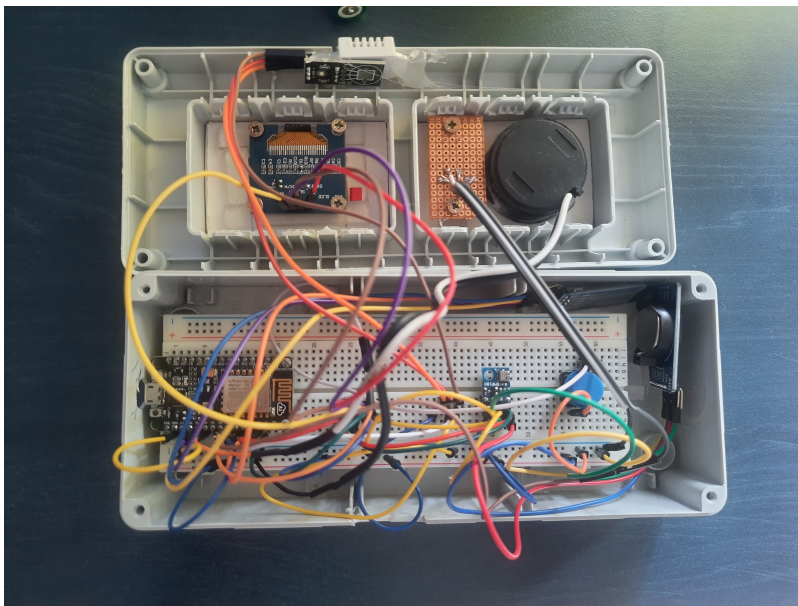
```
    display->drawCircle(clockCenterX + x, clockCenterY + y, 2); // cercul mic
    din care pleaca limbile ceasului
    for (int h = 0; h < 12; h++) {
        float angle = h * 30;
        angle = angle * PI / 180; // grade in radiansi
        int x1 = clockCenterX + x + (int)(sin(angle) * (clockRadius - 2));
        int y1 = clockCenterY + y - (int)(cos(angle) * (clockRadius - 2));
        int x2 = clockCenterX + x + (int)(sin(angle) * clockRadius);
        int y2 = clockCenterY + y - (int)(cos(angle) * clockRadius);
        // afisez ora doar pentru 3 6 9 12, altfel pun o liniuta
        if (h == 0)
            display->drawString(x1, y1 - 2, "12");
        else if (h == 3)
            display->drawString(x1 + 2, y1, "3");
        else if (h == 6)
            display->drawString(x1, y1 + 2, "6");
        else if (h == 9)
            display->drawString(x1 - 2, y1, "9");
        else
            display->drawLine(x1, y1, x2, y2);
    }
    // limba ceasului pentru secunde
    float angle = secunda * 6 ;
    angle = angle * PI / 180; ; // grade in radiansi
    int x3 = ( clockCenterX + ( sin(angle) * ( clockRadius - 5 ) ) );
    int y3 = ( clockCenterY - ( cos(angle) * ( clockRadius - 5 ) ) );
    display->drawLine( clockCenterX + x , clockCenterY + y , x3 + x , y3 + y);
    // limba ceasului pentru minute
    angle = minut * 6 ;
    angle = angle * PI / 180 ;
    x3 = ( clockCenterX + ( sin(angle) * ( clockRadius - 8 ) ) );
    y3 = ( clockCenterY - ( cos(angle) * ( clockRadius - 8 ) ) );
    display->drawLine( clockCenterX + x , clockCenterY + y , x3 + x , y3 + y);
    // limba ceasului pentru ora
    angle = ora * 30 + (int)(minut/2);
    angle = angle * PI / 180 ;
    x3 = ( clockCenterX + ( sin(angle) * ( clockRadius - 11 ) ) );
```

```
y3 = ( clockCenterY - ( cos(angle) * ( clockRadius - 11 ) ) );  
display->drawLine( clockCenterX + x , clockCenterY + y , x3 + x , y3 + y);  
  
}
```

Codul complet poate fi descarcat de aici: [ceas_multifunctional.zip](#)

Rezultate Obținute

In final, proiectul arata asa fara carcasa:



In final, proiectul arata asa cu carcasa:



Aici se poate gasi un scurt demo [demopm.zip](#). Vizionare placuta!

Concluzii

Dispozitivul Multifunctional este un obiect destul de folositor. Pe langa timp, dispozitivul afiseaza si data, tara si orasul in care te afli, dar si date utile despre mediul din camera in care stai precum temperatura, umiditatea si presiunea aerului. Datele acestea sunt trimise pe un server web unde se deseneaza grafice pentru a vedea cum evolueaza/ fluctueaza aceste informatii. Pentru un confort adus utilizatorului, daca acesta vrea sa isi seteze o alarma, acest lucru se poate realiza foarte usor si rapid cu ajutorul telefonului si a conexiunii bluetooth. Datele pot fi acesate remote si de pe telefon(unde sunt trimise prin bluetooth), dar si online accesand serverul web unde sunt trimise.

Proiectul a fost interesant de implementat si m-a ajutat sa imi fixeze notiunile pe care le-am dobandit la laboratorul de PM mai bine. De la simpla idee de la care am plecat, am ajuns sa imi adaug noi functionalitati pe parcurs, deoarece chiar mi-a placut sa fac acest proiect.

Download

Codul se poate descarca de aici [ceas_multifunctional.zip](#) sau din sectiunea software, la notes

Jurnal

- 7 Mai - am creat pagina de Wiki si completat-o cu introducerea si descrierea generala
- 15 Mai - am inceput sa lucrez la partea hardware. Am testat toate piesele sa vad ca sunt functionale.
- 17 Mai - m-am apucat de scris codul pentru Dispozitivul Multifunctional. Am reusit sa afisez datele citite de pe net.
- 18 Mai - am afisat si datele citite de la senzori.
- 20 Mai - am completat pagina de Wiki cu design-ul Hardware. Aici am adaugat schema si lista de piese folosite.
- 22 Mai - am adaugat un buton pentru a schimba ce se afiseaza pe display.
- 24 Mai - am facut conexiunea bluetooth si am setat dispozitivul sa trimita datele la ThingSpeak. Am adaugat si functionalitatea de alarma
- 27 Mai - am completat pagina de Wiki cu design-ul Software. Am specificat ce biblioteci am utilizat, IDE-ul folosit pentru a scrie cod, o explicare a codului in mare si descrierea functiilor utilizate. Am pus si un code snippet pentru functia de ceas analog si putine explicatii. Am adaugat si codul sursa si am completat pagina de Wiki cu ce mai lipsea.
- 28 Mai - am terminat de completat pagina Wiki cu demo si tot

Bibliografie/Resurse

Resurse Hardware:

- [nodemcu-esp8266-pinout-features-and-datasheet](#)
- [esp8266 and HC05](#)
- [DHT22 pinout and info](#)
- [BMP180 pinout and info](#)
- [DS3231 datasheet](#)
- laboratoare PM

Resurse software:

- [esp8266 and HC05](#)
- [IPGeolocation library](#)
- [Tutorial about how to connect esp8266 to ThingSpeak](#)
- laboratoare PM
- documentatiile bibliotecilor folosite

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/daniel/ceasmultifunctional>



Last update: **2023/05/28 17:44**