

SISTEM PENTRU STUDIUL RENTABILITAȚII IMPLEMENTARII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE ÎN SISTEMELE FOTOVOLTAICE

Candidat: Arnăutu Dumitru-Florin

Coordonator științific: Conf.dr.ing. Adrian Korodi

Sesiunea: Septembrie 2023

Internal

CUPRINS

CUPRINS.....	2
REZUMAT	3
1 Produse existente în domeniu	4
2. Fundamentele teoretice	5
2.1 Limbajul de programare C++ în Arduino IDE.....	5
2.2 Protocolul de comunicare I2C	7
2.2.1 Introducere.....	7
3. Componentele hardware.....	8
3.1 Placa de dezvoltare ESP32	8
3.2 Cloona a plăcii de dezvoltare Arduino Mega2560	9
3.3 Senzorii de temperatura BMP280 și AHT21	10
3.5 Punctele H	13
3.6 Coborât de tensiune XL4015	14
3.7 Modul microSD	15
3.8 Module de control încărcare solară PWM , RBL-30A	15
3.9 Panouri solare 20W	16
3.10 Actuatore liniare.....	17
3.11 Modul SENZOR 3 AXE ADXL345	18
3.13 TRANSLATOR NIVEL LOGIC I2C IIC BIDIRECTIONAL 8 CANALE 3.3V 5V TXS0108E.....	18
4. Codul de la baza proiectului	20
4.1 Mega2560:	20
5. Implementarea proiectului	27
6. Testarea proiectului	28
7. Concluzii și direcții de dezvoltare	29
7.1 Concluzii.....	29
7.2 Direcții de dezvoltare	29
FIGURI și TABELE	30
BIBLIOGRAFIE.....	31

REZUMAT

Această lucrare de licență are drept scop crearea unui sistem ce poate ajuta la determinarea eficienței și rentabilității instalării diverselor echipamente în instalațiile fotovoltaice pentru a spori eficiența acestora.

Pentru aceasta am creat un ansamblu format din:

- 2 panouri fotovoltaice de 20W, unul montat într-un cadru fix iar cel de-al doilea panou pe un cadru mobil pe 2 axe, acesta ținând să creeze un unghi de 90 de grade între panou și razele soarelui constant.
- 2 acumulatori de tip AGM, 12V 9.1Ah 109Wh, aceștia vor ajuta sistemul să facă față cererilor de energie ce depășesc capacitatea panourilor și vor alimenta sistemul când energia solară este indisponibilă.
- 2 controlere PWM pentru încărcarea acumulatorilor și protecția acestora
- Arduino Mega 2560 pentru colectarea de date de la diverși senzori analogici și digitali, controlul deciziilor logice din cadrul ansamblului și acționarea diverselor actuatori.
- un modul ESP32 pentru funcționalitățile de colectare de date și conectarea la internet a proiectului (îmbunătățirea conectivității proiectului folosind acest modul)
- coborâtor de tensiune liniar reglabil (pentru a nu suprasolicita regulatoarele de tensiune de pe arduino și ESP, acestea având pierderi semnificative de energie la tensiuni ridicate)
- 2 punți H de mare putere pentru controlul motoarelor liniare de pe cadrul mobil.
- 2 senzori de temperatură, BMP280 (temperatură + presiune atmosferică) și AHT21 (temperatură + umiditate aer)
- un stabilizator de tensiune bazat pe LM7805 (5V, 1.5A) folosit pentru alimentarea cu 5V a componentelor sensibile și ce pot induce perturbații ale tensiunii în sistem, ex.: releu, leduri)
- 2 ventilatoare cu turbină (12V, 130mA) pentru răcirea componentelor (în mod special punțile H și stabilizatorul de tensiune LM7805)
- 2 module pentru măsurarea cantității de energie produse de panouri
- un modul microSD pentru colectarea de date.

Drept urmare am obținut un sistem ce poate fi adaptat, dimensionat conform nevoilor, ușor de menținut și îmbunătățit cu piese ușor de obținut.

1 Produse existente în domeniu

În momentul de față există pe piața diverse sisteme ce satisfac o parte din funcțiile implementate în acest proiect, acestea sunt totuși costisitoare și încorporează componente greu de înlocuit, ce nu permit reparații și necesită înlocuirea unor ansambluri complexe în cazul unor defecțiuni.



Figura 1. Tracker solar pentru panouri solare fotovoltaice [1]

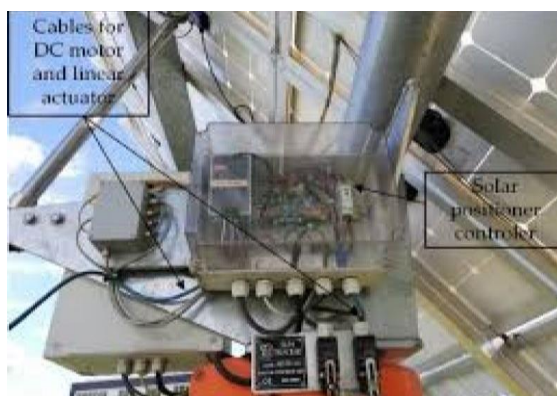


Figura 2. Micro calculator tracker complet poziționar dual motor [2]

Produsele de mai sus asigură funcționalitatea de urmărire a soarelui, acestea nu dispun de alte funcționalități.

2. Fundamentele teoretice

2.1 Limbajul de programare C++ în Arduino IDE

Limbajul C++ este creația lui Bjarne Stroustrup și reprezintă o extensie a limbajului C care permite programarea pe obiecte. Dezvoltarea acestuia a început în 1979, cu denumirea inițială de "C cu obiecte". Acesta a ajuns să capete mult mai multe funcționalități pe lângă programarea orientată pe obiecte, ajungând să fie redenumit ulterior C++ făcând referire la operatorul de incrementare ++. Acesta este un mod unic de a spune că C++ este o revizie incrementală a lui C.

Structura generală a unui program C++:

- un program C++ este constituit din funcții, una dintre aceste funcții este funcția principală, denumită main()

```
int main()
{
    cout<<"Hello World";

    return 0;
}
```

- main() este o funcție specială, care trebuie să apară obligatoriu o singură dată în orice program C++
- execuția oricărui program începe cu funcția main()
- o funcție este constituită din antet și corp. Antetul funcției conține numele funcției, tipul rezultatului pe care îl calculează funcția și o listă de parametri prin care funcția comunică cu exteriorul ei, încadrată între paranteze rotunde
- corpul funcției conține declarații și instrucțiuni care specifică prelucrările realizate de funcția respectivă

Instrucțiunea return este utilizată pentru a încheia execuția unei funcții și a returna valoarea expresiei specificate în instrucțiunea return ca valoare a funcției.

Vocabularul limbajului C++ este format din:

- setul de caractere
- identificatori
- cuvinte cheie
- comentarii
- separatori

Programarea C++ în Arduino Studio IDE este puțin diferită față de programarea C++ clasică. Diferența între cele două este că în Arduino Studio IDE nu mai avem funcția main, aceasta fiind înlocuită de două funcții: setup și loop.

Funcția `setup` se execută mereu prima și aceasta se execută doar o dată, la începutul programului, în schimb funcția `loop` se execută după funcția `setup`, însă aceasta se reapelează pe ea însăși la infinit.

De obicei în funcția `setup` se fac inițializările de variabile, clase, componente, etc., iar în funcția `loop` se execută codul care se dorește să fie executat mai mult de o singură dată.[3]

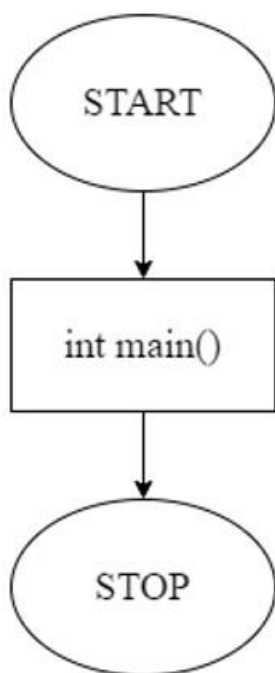


Figura 3. Structura programului C++

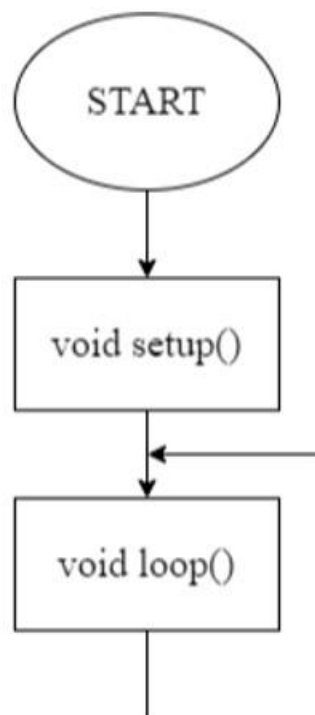


Figura 4. Structura programului în Arduino IDE

2.2 Protocolul de comunicare I2C

2.2.1 Introducere

Comunicarea prin interfața I2C este o opțiune eficientă pentru transmiterea de date între 2 microcontrolere, aceasta este și foarte practică punând la dispoziție o lățime de bandă de până la 3.4mb/s în modul de mare viteză, în cadrul acestui proiect s-a evitat folosirea unei viteze de transfer aproape de plafonul maxim pentru sporirea distanței la care este posibilă realizarea unei conexiuni stabile pe linia de comunicație.

Din cauza conductorilor folosiți în cadrul liniei de I2C odată cu creșterea lungimii acestora apare și efectul de inductor ce se opune schimbărilor ce țin de câmpul electromagnetic, în mod normal semnalele digitale pot călători pe distanțe de ordinul zecilor de metri dar în cadrul unei linii I2C starea acestei linii se schimbă de la HIGH la LOW chiar și de 400.000 de ori pe secundă astfel chiar și cea mai mică forță reactivă (în cazul acesta inducția din cabluri) poate perturba interpretarea semnalelor între dispozitivul Master și Slave-urile acestuia.

Protocolul de comunicare I2C (Inter-Integrated Circuit) este un protocol serial sincron folosit pentru comunicația între circuite integrate (ICs) pe o placă de bază (board) sau între plăci de bază. Protocolul I2C utilizează două linii pentru comunicație:

- SDA (Serial Data): Aceasta este linia de date serială bidirecțională pe care informațiile sunt transmise în forma serială între dispozitivele conectate. Linia SDA este controlată de dispozitivul master, care inițiază și controlează comunicația.
- SCL (Serial Clock): Aceasta este linia de ceas folosită pentru a sincroniza transferul datelor între dispozitivele conectate. Semnalele de pe linia SCL sunt controlate de dispozitivul master și sunt utilizate pentru a sincroniza transferul de biți pe linia SDA.

Datorită faptului că adresele I2C sunt în mod normal pe 7 biți, într-o rețea se pot conecta până la 128 de dispozitive, dar unele adrese sunt rezervate coborând numărul de adrese disponibile sub 128.

O mențiune importantă este existența integratelor cu adrese I2C pe 10 biți, ducând astfel numărul de componente suportate într-o rețea aproape de 1024, teoretic.

3. Componentele hardware

3.1 Placa de dezvoltare ESP32

Placa de dezvoltare WiFi bazată pe ESP32, un circuit integrat care conține module GPIO, I²C, I²S, PWM, SDIO, SPI, UART și ADC toate pe o singură placă și facil de utilizat cu sintaxa gen Arduino pentru acces la hardware.

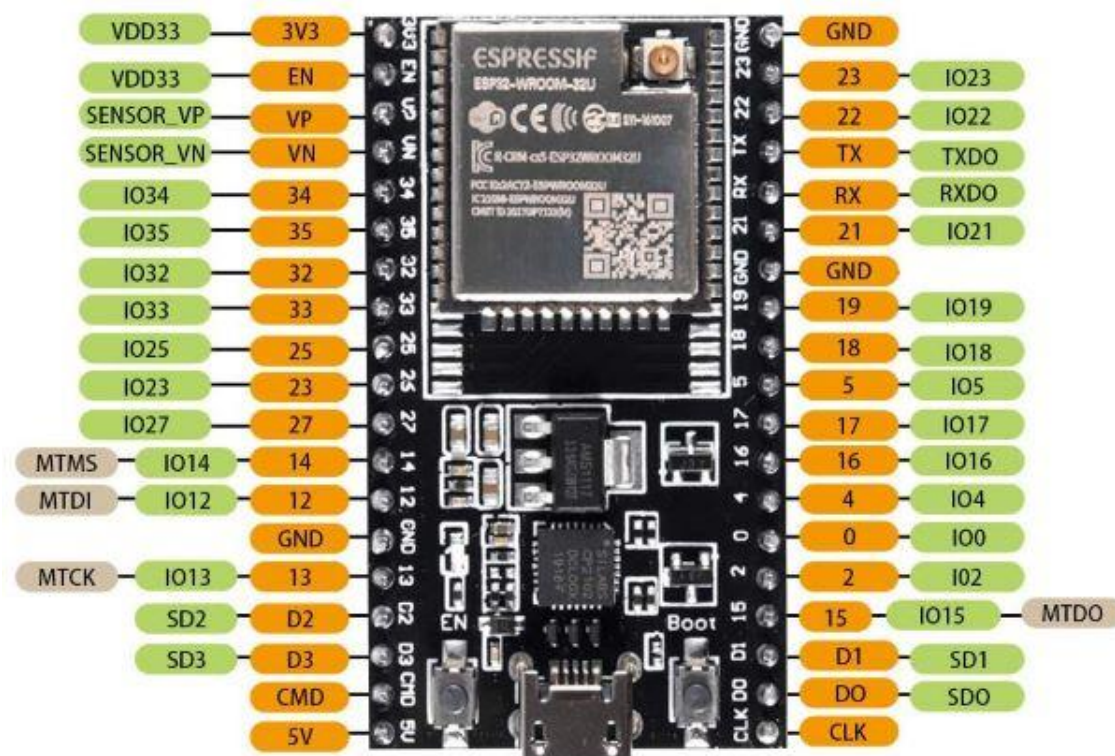


Figura 5. Pinout-ul microcontroller-ului esp32

Caracteristici tehnice:

- Tensiune de alimentare: 3.3V
- Driver: CP2102
- Wi-Fi Direct (P2P), soft-AP , Bluetooth
- Consum current: 10uA – 170 mA
- Memorie flash: 4 MB
- Protocol TCP/IP
- Procesor: dual-core Xtensa® 32-bit LX6
- Wireless model: 802.11 b/g/n (pana la 150Mb/s) [3]

3.2 Cloona a plăcii de dezvoltare Arduino Mega2560

Placa de dezvoltare construită după schemele publice ale celebrei plăci ArduinoMega , fundația ce se ocupă de dezvoltarea platformei Arduino încurajează crearea și comercializarea plăcilor după propriile scheme , aceștia le fac publice în mod intenționat pentru a spori contribuția utilizatorilor în dezvoltarea viitoarelor modele , în prețul unei plăci originale se regăsește și o “taxă” ce merge către dezvoltarea proiectului Arduino , “misiunea” fundației este : ” enable anyone to enhance their lives through accessible electronics and digital technologies” (să pună la dispoziție oricui mijloacele electronice necesare pentru a-și îmbunătăți viața) .

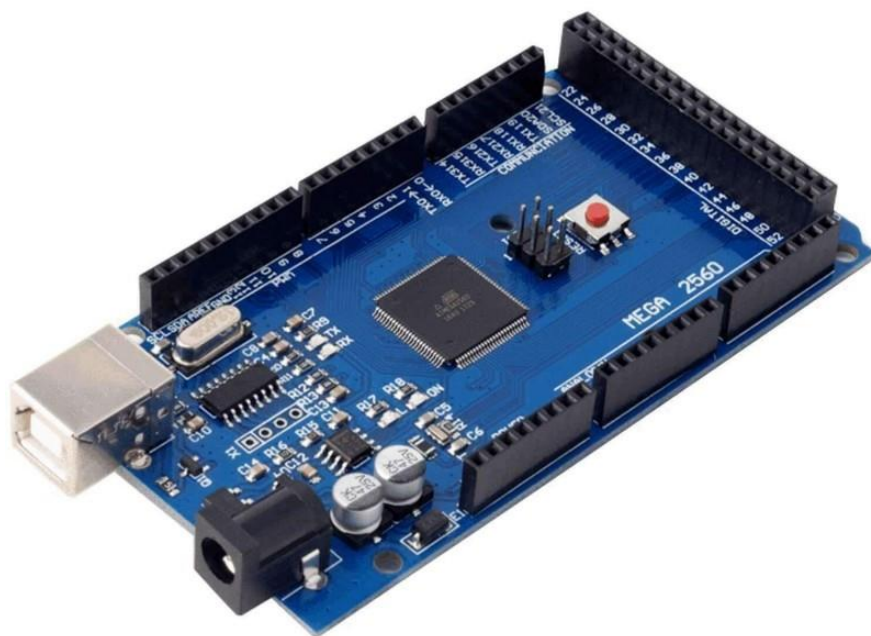


Figura 6. ATmega2560

- Driver: CH340
- Tensiune de lucru: 5V
- Tensiune de intrare (recomandat): 7-12V
- Pini digitali: 54 (14 PWM output)
- Pini analogici: 16
- Curent de ieșire: 40 mA
- Flash Memory: 256 KB, 8 KB pentru bootloader
- SRAM: 8 KB
- EEPROM: 4 KB
- Clock: 16 MHz [4]

3.3 Senzorii de temperatura BMP280 si AHT21

Cei 2 senzori au fost adaugați în proiect pentru a extinde gama de date colectate și pentru a monitoriza temperatura din cutia de control ,aceștia folosesc tot magistrala de date I2C , adăugarea acestora presupunând efort minim , iar alegerea celor 2 modele diferite oferă o redundanță sporită în colectarea datelor.

Optându-se pentru o cutie din plastic cu capac transparent , aceasta fiind umplută cu componente ce disipă căldura am constatat că este absolut necesară o îmbunătățire a răcirii sistemului , temperatura maximă măsurată cu senzorii fiind de 75°C , prag la care presupunem că sistemul a suferit avarii ale modulelor de măsură a curentului INA219 (2 din 4 module au fost avariate iremediabil).



Figura 7. BMP280

Tensiune de alimentare: 3.3V DC

Protocol de comunicare: I²C

Adrese I²C: 0x77(default - jumper deschis) sau 0x76(jumper închis)

Tip conectori I²C 3V3: SH 4P

Pitch conectori I²C 3V3: 1mm [5]



Figura 8. AHT21

Tensiune de alimentare: 3.3V DC

Temperatura măsurată: -40°C ~ 120°C

Umiditate: 0~100 %RH

Protocol de comunicare: I²C

Adresa I²C: 0x38

Tip conectori I²C 3V3: SH 4P

Pitch conectori I²C 3V3: 1mm [6]

3.4 Senzorii pentru măsurarea curentului INA219

4 senzori cu adresa configurabilă prin lipirea a 2 perechi de pini așezați pe aceștia, așezați chiar datorită faptului că suporta legarea a 4 senzori cu 4 adrese diferite pe aceeași magistrală I2C aceștia au generat un număr de probleme și obstacole ce au necesitat modificări majore asupra sistemului, plecând de la rezistența shunt de $.1\Omega$ ce este subdimensionată pentru puterea disipată în aceasta, conform specificațiilor acest senzor poate măsura o intensitate a curentului de până la $3.2A$, în cadrul rezistenței de $.1\Omega$, această intensitate va genera o cădere de tensiune conform legii $I=U/R$ de $.32V$ ($3.2A=U/.1$), prin rezistor trecând $3.2A$ cu o cădere de $.32V$ va rezulta o putere disipată în energie termică de $1.024W$ ($P=U \cdot I \rightarrow P=.32V \cdot 3.2A$), după atingerea unei temperaturi excesive rezistorii de $.1\Omega$ au capatat valori noi, necesitând înlocuirea, am optat pentru rezistori de 1Ω dar capabili să disipeze $5W$ (mult peste cei $\sim 1W$ ce îi va atinge sistemul).

O nouă problemă ridicată de acești senzori, în urma unei supraîncălziri a sistemului însuși integratele INA219 au fost afectate, din 4 numai 2 mai raportează valori corecte, unul fiind complet inaccesibil pe linia de I2C, iar celălalt raportând valori constante de $320mV$ pe rezistența de shunt cu tensiunea circuitului la $0V$, caracteristicile senzorilor detaliază temperaturi de operare cuprinse între -40 și $125^\circ C$, și de stocare până la $150^\circ C$, există posibilitatea avarierii acestora în timpul înlocuirii rezistorilor de $.1\Omega$, însă defectul nu a fost sesizabil imediat după reparație.

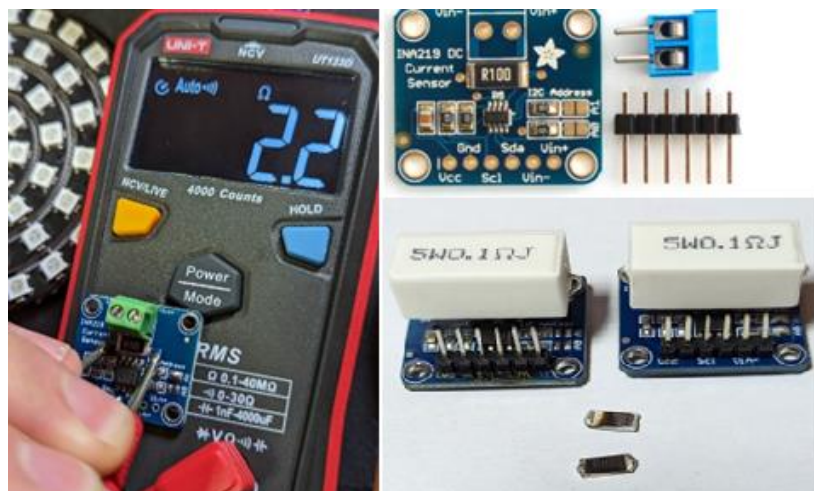


Figura 9. Senzor de curent bidirecțional INA219

Rezistor pentru măsurarea curentului de 0.1Ω , $2W$, 1% ;

Poate măsura curenți ce dau o cădere de tensiune de maxim $26V$;

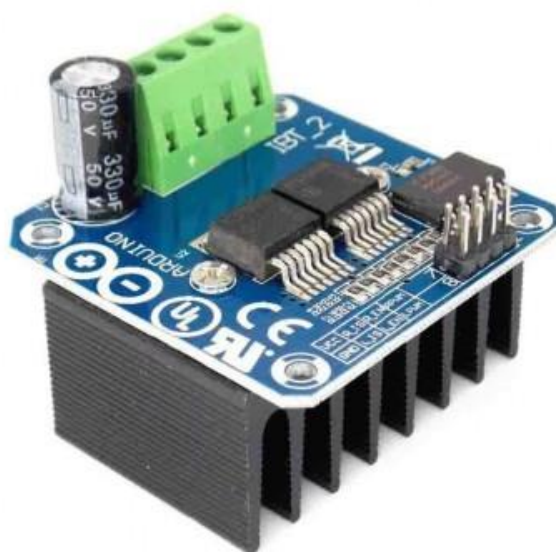
Poate măsura curenți de $\pm 3.2A$ cu rezoluție de $\pm 0.8mA$;

Folosește adresele I2C $0x40$, $0x41$, $0x44$, $0x45$ selectabile prin jumperi;

Tensiune de alimentare: $3V - 5.5V$. [7]

3.5 Puntile H

2 punți H de mare putere, tip BTS7960 au fost folosite pentru a controla cele 2 motoare liniare ce directionează cadrul mobil prezent în proiect, acestea se controlează în mod clasic folosind semnale digitale și PWM. Acestea nu au prezentat dificultăți de integrare în sistem, acest fapt se atribuie simplității



ce stă la baza operării unor astfel de module.

Figura 10. Punte H de putere BTS7960

Tensiune alimentare: 5.5 - 27V

Tensiune logică: 5V

Protecție temperatură: Da

Protecție supra-tensiune: Da

Protecție curent: Da

Protecție scurt-circuit

Semnal: PWM sau 5V

Frecvență PWM: până la 25KHz

Dimensiuni mm: 40 x 50 x 12mm [8]

3.6 Coborator de tensiune XL4015

Un modul step down pentru coborârea tensiunii de la 12V(uneori chiar și 14V) provenite din circuitul realizat de panoul fotovoltaic la valori mai ușor de gestionat , la bordul plăcilor de dezvoltare (Mega și ESP32) se regăsesc reglatoare de tipul AMS1117 de 5V respectiv 3.3V , acestea disipând o cantitate considerabilă de energie pentru a reduce tensiunea la valorile specificate, am optat în coborârea mai eficientă într-un pas intermediar a tensiunii folosind acest modul din 2 motive :

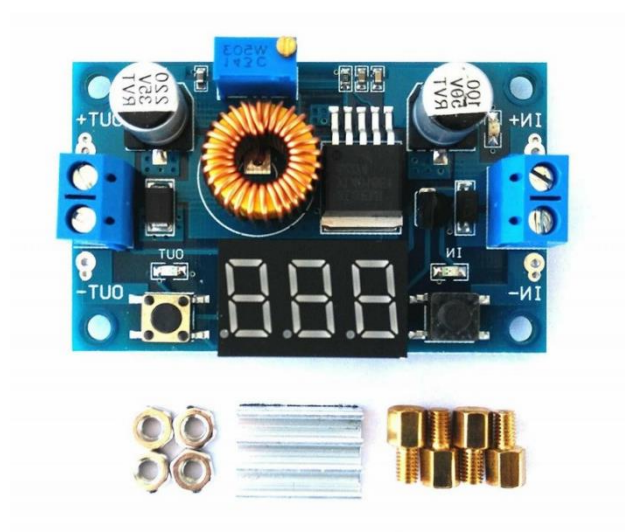


Figura 11. Coborator de tensiune XL4015

- 1) Reglatoarele AMS1117 disipă multă căldură punându-le în pericol atât funcționalitatea acestora cât și componentele ce depind de tensiunile stabile generate de ele
- 2) Pentru a beneficia de puterea maximă la tensiunile stabile de 5V , respectiv 3.3 V , acestea se vor alimenta cât mai aproape de tensiunea finală+ căderea de tensiune din acestea (1V, maxim 1.3V). astfel XL4015 este setat să convertească cei 11V din acumulator (tensiunea minimă, sub această tensiune acumulatorul poate fi avariat dacă se continuă descărcarea) în 6V.

Tensiune intrare: 5 - 36VDC

Tensiune iesire: 1.25 - 32VDC, reglabila

Curent: între 0 și 5 Ah [9]

3.7 Modul microSD

Modul folosit pentru colectarea datelor preluate de sistem :cantitatea de energie produsa , temperatura si umiditatea din cutia de control.

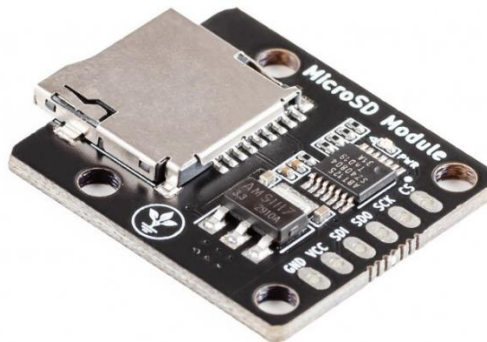


Figura 12. Modul microSD

Dimensiuni card microSD acceptate: 2GB ~ 32GB

Tensiune de alimentare: 3.3V sau 5V DC

Protocol de comunicare: SPI [10]

3.8 Module de control incarcare solara PWM , RBL-30A

Module folosite pentru controlul incarcarii acumulatorilor Plumb acid AGM folositi in proiect , acest modul a fost ales datorita costurilor reduse si functionalitatilor multiple.

Un dezavantaj al acestor module fata de un modul ce foloseste MPPT (maximum power point tracking) este ca nu poate profita de toata energia produsa de panou , pentru functionarea acestuia el necesita o tensiune de intrare de la panou mai mare decat cea a acumulatorului, un modul MPPT dispune de un convertor DC-DC de tip boost ce ridica in mod dinamic tensiunea livrata de panou peste cea a acumulatorului.



Figura 13. Modul de control incarcare solara PWM

SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE
ÎN SISTEMELE FOTOVOLTAICE

Model: RBL-30A

Tensiune baterie: 12 -24VDC

Curent descarcare: 10A maxim

Putere maxima: 360W la 12V, 720W la 24V

Tensiune panou maxima: 48V pentru baterie de 24V, 24V pentru baterie de 12V

Egalizare: 14.4V (Plumb Acid Sigilata), 14.2V (Baterie cu gel), 14.6V (baterie Plumb Acid nesigilata)

Tensiune absorbtie: 13.7V default, reglabila

Oprire descarcare: 10.7V default, reglabila

Reconectare la incarcare: 13V

Consum curent propriu modul: <10mA

Temperatura operare: -35 - +60 grade C

Protectii: Scurt circuit, circuit deschis, protectie inversa, protectie supra-sarcina

Protectie Curent Invers: Mosfet Dual

Dimensiuni mm: 133.5 x 70 x 35mm [11]

3.9 Panouri solare 20W

Panouri fotovoltaice monocristaline de 20W, rama de aluminiu , sticla de protectie securizata si cabluri de conectare atasate.



Figura 14. panou solar

Putere maxima:20W

Voltaj circuit deschis:22.3V

Voltaj maxim circuit inchis:18V

SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE
ÎN SISTEMELE FOTOVOLTAICE

Curent de scurtcircuit: 1.2A

Curent maxim de lucru: 1.11A

Dimensiuni: 450*340*20 mm

Greutate 1.55Kg [12]

3.10 Actuatore liniare

2 actuatore liniare sunt folosite pentru a facilita mișcarea de rotație în jurul axelor X și Y.



Figura 15. Actuatore liniare

Cursa actuatorului: 100mm

Tensiune alimentare: 12V

Forță maximă: 1100N

Viteza de acțiune: 13mm/s

Regim utilizare recomandat : 10% (10 secunde utilizare , 90 secunde idle) [13]

3.11 Modul SENZOR 3 AXE ADXL345

Modul accelerometru pe 3 axe pentru determinarea poziției panoului solar, aceasta abordare pune bazele unei implementări mult mai tehnice a poziționării panoului, în cazul actual va fi folosit pentru a reseta poziția panoului de la o zi la alta și protecția împotriva grindinei.

Dat fiind faptul că modulul va îndeplini funcții critice s-a ales protejarea lui prin acoperirea cu lipici termică în lipsa unei metode de protecție mai adecvate, motoarele liniare nu au puncte de oprire reglabile, cadrul mobil își va atinge limitele de mișcare înaintea motoarelor, valori hardcodate ale înclinării panoului vor fi stabilite ca măsură de protecție pentru a preveni avariile motoarelor sau a cadrului.

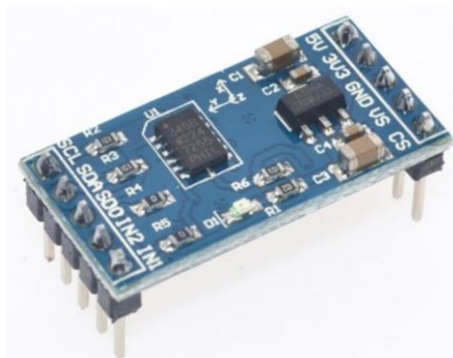


Figura 16. Senzor ADXL345

Tensiune alimentare: 3.3 - 5VDC

Integrat: ADXL345

Integrat putere: RT9161

Interfata: I2C, SPI

Dimensiuni mm: 28 x 14mm [14]

3.13 TRANSLATOR NIVEL LOGIC I2C IIC BIDIRECTIONAL 8 CANALE 3.3V 5V TXS0108E

Modul ce permite trecerea de la nivelul logic de 3.3V la 5V, folosit în principal pentru protejarea componentelor mai sensibile aflate pe magistrala de I2C, în proiect se regăsesc atât componente ce funcționează la nivel logic de 5V cum ar fi: ADXL345 și ADS1015, acestea beneficiind chiar de voltajul mai mare ele aflându-se la o distanță mai mare de controller.

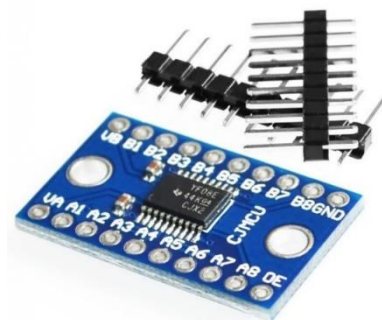


Figura 17. Translator nivel logic

-nivelul logic de 5V: INA219,ADXL345,Mega2560

-nivel logic de 3.3V : AHT21,BMP280,ecranul OLED, ESP32

VCCA se conectează la sursa 3.3V

VCCB se conectează la sursa 5V

GND se conectează în comun la cele 2 surse de 3.3 și 5V

Când Ax are input TTL 3.3V, Bx are output TTL 5V

Când Bx are input TTL 5V, Ax are output TTL 3.3V

NU necesită control de direcție [15]

4. Codul de la baza proiectului

4.1 Mega2560:

Pentru implementarea sistemului , am fractionat codul in mai multe fisiere, la baza regulii de impartire a codului fiind programarea in timp real , intreg sistemul bazandu-se pe acest concept prin care nu exista timpi in care microcontroller-ul sa astepte rezultate.

```
#include <Wire.h> // librarie ce faciliteaza comunicarea prin I2C
#include <Adafruit_INA219.h>
#include <Adafruit_Sensor.h>
#include "Adafruit_BMP280.h"
#include <SPI.h>
#include <SD.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_AHTX0.h>
#include <Adafruit_ADXL345_U.h>
```

Wire.h este o librarie ce faciliteaza comunicarea prin I2C

Adafruit_INA219.h pentru colectarea datelor de la senzorii de intensitate

Adafruit_Sensor.h folosita pentru senzorii AHT21/BMP280

Adafruit_BMP280.h folosita pentru controlul BMP280

SPI.h protocolul SPI este folosit doar pentru modulul microSD

SD.h librarie folosita pentru a scrie pe cardul MicroSD

Adafruit_GFX.h ajuta la pozitionarea textului pe ecranul OLED

Adafruit_SSD1306.h pentru controlul ecranului OLED

Adafruit_AHTX0.h pentru prelucrarea datelor provenite de la AHT21

Adafruit_ADXL345_U.h pentru prelucrarea datelor provenite de la accelerometru

În interiorul buclei ce rulează în mod repetat după inițializarea, microcontroller-ului se află numai apelurile către funcțiile recurente și codul aferent apelării acestora la intervale regulate de timp , există posibilitatea ca o sarcină de lucru să interfereze cu apelul acesteia (ex. în interiorul funcției cu recurență la 10ms , codul rulat să țină microcontroller-ul ocupat pentru 15ms, caz în care apelul următoarei funcții se va face o dată la 15ms în loc de 10ms) , acest comportament a fost realizat prin plasarea codului ce este costisitor din punct de vedere al timpului de execuție în funcții cu recurență mai scăzută.

```
void loop(void)
{
    actual_time=millis();
    if(actual_time-prev_time_10ms>=task_10ms){      func_10ms();
        prev_time_10ms=actual_time;
    }
```

SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE
IN SISTEMLILE FOTOVOLTAICE

```
    if(actual_time-prev_time_50ms>=task_50ms){      func_50ms();
        prev_time_50ms=actual_time;
    }
    if(actual_time-
prev_time_100ms>=task_100ms){      func_100ms();
        prev_time_100ms=actual_time;
    }
    if(actual_time-prev_time_1s>=task_1s){      func_1s();
        prev_time_1s=actual_time;
    }
    if(actual_time-prev_time_5s>=task_5s){      func_5s();
        prev_time_5s=actual_time;
    }
    if(actual_time-prev_time_1m>=task_1m){      func_1m();
        prev_time_1m=actual_time;
    }
    if(actual_time-prev_time_10m>=task_10m){      func_10m();
        prev_time_10m=actual_time;
    }
    if(actual_time-prev_time_1h>=task_1h){      func_1h();
        prev_time_1h=actual_time;
    }
}
```

Un prim exemplu de cod folosit in evenimente recurente este cel pentru determinarea impactului in cazul grindinei ce poate distruge panourile , folosind datele de la senzorul ADXL345 sistemul poate detrmna in cazul unui impact schimbarea brusca a valorilor vectorilor de acceleratie masurati de senzori , caz in care un indicator (impactDetectat) va fi activat , punand sistemul intr-o stare de alarma , repositionand panoul astfel in cat orice impact cu bucatile de gheata sa fie la un unghi cat mai mic , micșorand astfel sansele avarierii panourilor.

```
void func_10ms()
{
total_X=0;
total_Y=0;
total_Z=0;
for(int i=0;i<10;i++){
    total_X=total_X+achizitii_X[i];
    total_Y=total_Y+achizitii_Y[i];
    total_Z=total_Z+achizitii_Z[i];
}

sensors_event_t event;
accel.getEvent(&event);
X_out = event.acceleration.x;
Y_out = event.acceleration.y;
Z_out = event.acceleration.z;
achizitii_X[indexSample_ADXL]=X_out;
achizitii_Y[indexSample_ADXL]=Y_out;
achizitii_Z[indexSample_ADXL]=Z_out;
if(pozFinala){
    if(X_out-2.0>total_X/10.0 || X_out+2.0<total_X/10.0 ||
    Y_out-2.0>total_Y/10.0 || Y_out+2.0<total_Y/10.0 ||
```

```
        Z_out-2.0>total_Z/10.0 || Z_out+2.0<total_Z/10.0)
        {
            impactDetectat=true;
        }
    }
    if(indexSample_ADXL<9){
        indexSample_ADXL=indexSample_ADXL
+1;
    }else{
        indexSample_ADXL=0;
    }
    if(impactDetectat){
        digitalWrite(motor1Ren,HIGH);
        analogWrite(motor1R_PWM,0);
        digitalWrite(motor1Len,HIGH);
        analogWrite(motor1L_PWM,225);
    }
}
```

Datele colectate sunt prelucrate ținând cont de abaterea de la medie a ultimelor 10 valori citite , în cazul unei erori mai mare de $2m/s^2$, se vor lua măsurile necesare protejării panourilor.

Datorită importanței protejării panourilor, această bucată de cod beneficiază de cel mai mult timp de execuție , ea fiind rulată de 100 de ori pe secundă pentru a acționa la primul impact .

Următoarea rutină ce se execută este cea de 50ms , aceasta are ca rol poziționarea panoului mobil , ținând crearea unui unghi de 90 de grade cu razele soarelui , folosind valori provenite de la un ansamblu de fotorezistori în cadrul rutinei motoarele liniare sunt controlate cu ajutorul punctelor H existente în proiect , ce la rândul lor sunt controlate folosind semnale digitale și PWM , pentru o mișcare mai lină și controlată, factorul de umplere nu este setat la 100% (255) , am ales valoarea de 225 (~88%) pentru a alimenta motoarele cu aproximativ 10.5V , ele fiind supradimensionate pentru dimensiunile ansamblului).

```
void func_50ms()
{
    //=====urmarire solara
    punct1=analogRead(adc1); punct2=analogRead(adc2); punct3=a
nalogRead(adc3);
    punct4=analogRead(adc4);
    if(!pozFinala && !impactDetectat){
        //control M2
        if(punct1>900 ){
            digitalWrite(motor2Ren,HIGH);          analogWrite(motor2R_P
WM,0);
            digitalWrite(motor2Len,HIGH);          analogWrite(motor2L_P
WM,0);
            pozFinala=true;
        }else if(punct4<900){
            digitalWrite(motor2Ren,HIGH);          analogWrite(motor2R_P
WM,225);
            digitalWrite(motor2Len,HIGH);          analogWrite(motor2L_P
WM,0);
        }else if (punct3<900) {
            digitalWrite(motor2Ren,HIGH);          analogWrite(motor2R_P
WM,0);
            digitalWrite(motor2Len,HIGH);          analogWrite(motor2L_P
WM,225);
        }else{

```

SISTEM PENTRU STUDIUL RENTABILITATII IMPLEMENTARII METODELOR DE EFICIENTIZARE A PRODUCTIEI DE ENERGIE

IN SISTEMLILE FOTOVOLTAICE

```
        digitalWrite (motor2Ren,HIGH);        analogWrite (motor2R_P
WM, 0);
        digitalWrite (motor2Len,HIGH);        analogWrite (motor2L_P
WM, 0);
    }

    if (punct1>900 ) {
        digitalWrite (motor1Ren,HIGH);        analogWrite (motor1R_P
WM, 0);
        digitalWrite (motor1Len,HIGH);        analogWrite (motor1L_P
WM, 0);
        pozFinala=true;
    } else if (punct3<900) {
        digitalWrite (motor1Ren,HIGH);        analogWrite (motor1R_P
WM, 225);
        digitalWrite (motor1Len,HIGH);        analogWrite (motor1L_P
WM, 0);
    } else if (punct2<900) {
        digitalWrite (motor1Ren,HIGH);        analogWrite (motor1R_P
WM, 0);
        digitalWrite (motor1Len,HIGH);        analogWrite (motor1L_P
WM, 225);
    } else {
        digitalWrite (motor1Ren,HIGH);        analogWrite (motor1R_P
WM, 0);
        digitalWrite (motor1Len,HIGH);        analogWrite (motor1L_P
WM, 0);
    }
}

}
```

Rutina de 100ms, in cadrul acesteia se abordeaza masurarea productiei de energie de la cele 2 panouri solare , aici se face colectarea datelor neprelucrate , intensitatea si tensiunea curentului.

In aceasta functie se mai creeaza si o animatie in coltul din dreapta al ecranului OLED, aceasta servind ca indicator pentru functionarea corecta a codului .

```
void func_100ms ()
{

    //=====
    =====INA219
    shuntvoltage_0x40 = ina219_0x40.getShuntVoltage_mV();
    busvoltage_0x40 = ina219_0x40.getBusVoltage_V();
    current_mA_0x40 = ina219_0x40.getCurrent_mA();
    power_mW_0x40 =
current_mA_0x40*busvoltage_0x40;//ina219_0x40.getPower_mW();
    loadvoltage_0x40 = busvoltage_0x40 + (shuntvoltage_0x40 /
1000);

    shuntvoltage_0x44 = ina219_0x44.getShuntVoltage_mV();
```

SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE

IN SISTEMELE FOTOVOLTAICE

```
busvoltage_0x44 = ina219_0x44.getBusVoltage_V();
current_mA_0x44 = ina219_0x44.getCurrent_mA();
power_mW_0x44 =
busvoltage_0x44*current_mA_0x44;//ina219_0x44.getPower_mW();
loadvoltage_0x44 = busvoltage_0x44 + (shuntvoltage_0x44 /
1000);

achizitii_0x44_mA[indexSample]=fabs(current_mA_0x44);
achizitii_0x44_mW[indexSample]=power_mW_0x44;
achizitii_0x44_V[indexSample] =busvoltage_0x44;
achizitii_0x40_mA[indexSample]=fabs(current_mA_0x40);
achizitii_0x40_mW[indexSample]=power_mW_0x40;
achizitii_0x40_V[indexSample] =busvoltage_0x40;

if(indexSample<9){
    indexSample=indexSample+1;
}else{
    indexSample=0;
}
//=====
=====INA219
//=====desenez un cerc
pe OLED , feedback la rularea codului
display.fillCircle(62+cursor,60,2,WHITE);
display.display();
if(cursor<64){
    cursor=cursor+8;
}else{
    cursor=0;
}
}
```

În cadrul rutinei de 1s au loc atât filtrarea datelor provenite de la senzorii INA219, preluarea datelor de la senzorii de temperatură AHT21 și BMP280, cât și afișarea parametrilor sistemului pe ecranul OLED.

Filtrarea datelor se face prin obținerea unei medii a 10 valori colectate în rutina de 100ms, pe ecranul OLED afișându-se valorile filtrate, afișarea fiecărei valori colectate făcând urmărirea ecranului foarte dificilă, astfel afișând mediile se păstrează o rată de improspătare a datelor pe ecran de o secundă.

```
void func_1s()
{
    pressure = bmp.readPressure();
    temperature = bmp.readTemperature()-1.0;
    aht.getEvent(&humidity, &temp);

    total_0x40_mA=0;
    total_0x40_mW=0;
    total_0x40_V =0;
    total_0x44_mA=0;
    total_0x44_mW=0;
    total_0x44_V =0;
    for(int i=0;i<10;i++){
        total_0x40_mA=total_0x40_mA+achizitii_0x40_mA[i];
```


SISTEM PENTRU STUDIUL RENTABILITATII IMPLEMENTARII METODELOR DE EFICIENTIZARE A PRODUCTIEI DE ENERGIE
IN SISTEMLILE FOTOVOLTAICE

```
        total_0x40_mW=total_0x40_mW+achizitii_0x40_mW[i];
        total_0x40_V =total_0x40_V+achizitii_0x40_V[i];
        total_0x44_mA=total_0x44_mA+achizitii_0x44_mA[i];
        total_0x44_mW=total_0x44_mW+achizitii_0x44_mW[i];
        total_0x44_V =total_0x44_V+achizitii_0x44_V[i];
    }
    display.clearDisplay();
    display.setTextSize(1);
    display.setCursor(0 ,0 );display.print("MCU_0x44:");
    display.setCursor(0 ,8 );display.print(round(total_0x44_
mW/10));
    display.setCursor(40 ,8 );display.print("mW");
    display.setCursor(0 ,16
);display.print(total_0x44_mA/10.0);
    display.setCursor(40 ,16 );display.print("mA");
    display.setCursor(0 ,24
);display.print(total_0x44_V/10.0);
    display.setCursor(40 ,24 );display.print("V");

    display.setCursor(0 ,32 );display.print("X.Y_0x40:");
    display.setCursor(0 ,40
);display.print(round(total_0x40_mW/10));
    display.setCursor(40 ,40 );display.print("mW");
    display.setCursor(0 ,48
);display.print(total_0x40_mA/10.0);
    display.setCursor(40 ,48 );display.print("mA");
    display.setCursor(0 ,56
);display.print(total_0x40_V/10.0);
    display.setCursor(40 ,56 );display.print("V");

    display.setCursor(60 ,0 );display.print("Presiune:");
    display.setCursor(60 ,8 );display.print(pressure);
    display.setCursor(115,8 );display.print("Pa");

    display.setCursor(60 ,16 );display.print("Temp:");
    display.setCursor(60 ,24 );display.print("T1:");
    display.setCursor(80 ,24 );display.print(temperature);
    display.setCursor(118,24 );display.print("C");
    display.setCursor(60 ,32 );display.print("T2:");
    display.setCursor(80 ,32 );display.print(temp.temperature);
    display.setCursor(118,32 );display.print("C");

    display.setCursor(60 ,40 );display.print("Umiditate:");
    display.setCursor(60 ,48
);display.print(humidity.relative_humidity);
    display.setCursor(118,48 );display.print("%");

    display.display();
    cursor=0;
}
```

SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE EFICIENTIZARE A PRODUCȚIEI DE ENERGIE
IN SISTEMELE FOTOVOLTAICE

În cadrul rutinei de 5 secunde se colectează datele sistemului pe cardul microSD , pentru a face mai ușoară prelucrarea acestora sunt scrise fără unități de măsură, aranjate în grupuri cu linii goale între ele , pot fi prelucrate ulterior pentru a obține grafice ale producției de energie corespunzătoare fiecărui panou, variației de temperatură ,presiune și umiditate din cutia de control .

```
void func_5s()
{

    if (!SD.begin(pinCS))
    {
        //Serial.println("SD card initialization failed");
    }

    myfile = SD.open("Date.txt", FILE_WRITE);
    if (myfile)
    {
        myfile.println(round(total_0x44_mW/10));
        myfile.println(total_0x44_mA/10.0);
        myfile.println(total_0x44_V/10.0);
        myfile.println(round(total_0x40_mW/10));
        myfile.println(total_0x40_mA/10.0);
        myfile.println(total_0x40_V/10.0);
        myfile.println(pressure);
        myfile.println(temperature);
        myfile.println(temp.temperature);
        myfile.println(humidity.relative_humidity);
        myfile.println();
        myfile.println();
        myfile.close();
    } else
    {
        //Serial.println("error opening the text file!");
    }
}
```

O dată la un minut temperatura măsurată este folosită pentru a porni sau opri ventilatoarele.

```
void func_1m()
{
    if(temperature>40.00){
        digitalWrite(pinReleuVentIna219,HIGH);
        digitalWrite(pinReleuVentESP,HIGH);
    }else {
        digitalWrite(pinReleuVentIna219,LOW);
        digitalWrite(pinReleuVentESP,LOW);
    }
}
```

Ultima rutină folosită este cea de 10 minute, în cadrul acesteia se reactivează urmărirea solară pentru a repositiona panoul , soarele avansează pe cer cu viteză de 2.5 grade o dată la 10 minute , am considerat că o corecție a poziției panoului în acest interval ar fi potrivită .

```
void func_10m()
{
    pozFinala=false;// soarele își schimbă unghiul cu 15 grade pe
    ora, deci 2.5 grade o dată la 10 minute
}
```

5. Implementarea proiectului

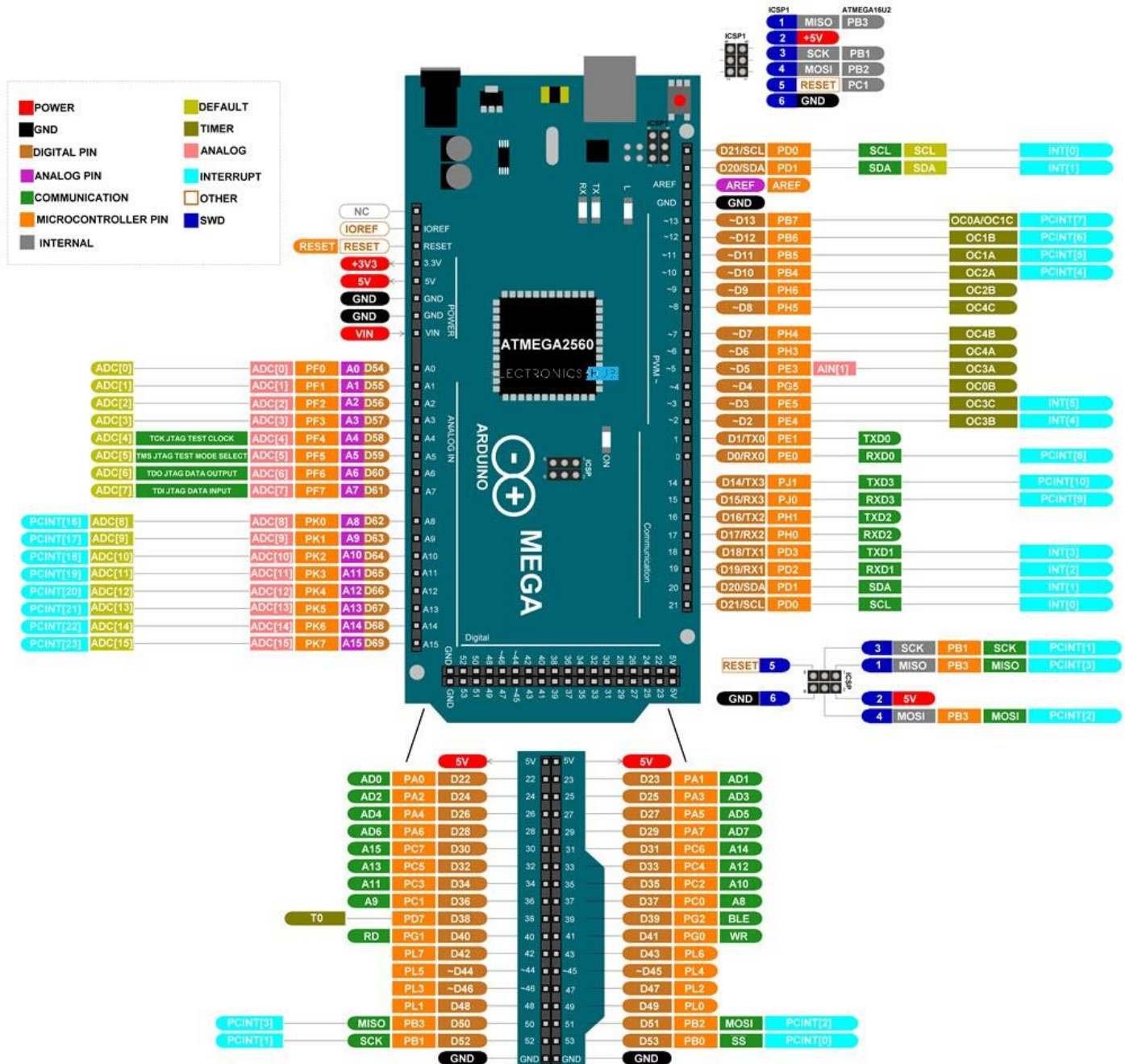


Figura 18. Pinout Mega2560

O schema electrica completa va fi realizata in etapele viitoare de dezvoltare ale proiectului , acesta fiind un prototip o schema electrica finala este aproape imposibil de realizat si ar ajunge la o complexitate si un nivel de detaliu greu de cuprins intr-un document, necesitand aplicatii specializate pentru vizualizarea acesteia, in continuare voi merge pe ideea de mici scheme de conectare .

6. Testarea proiectului

Folosind metode de testare manuala proiectului i-a fost testata functionalitatea atat in conditii normale, cat si in situatii mai speciale.

Testarea manuala este unul din procesele esentiale in dezvoltarea software, desi in momentul de fata tehnicile de testare automatizata au luat avant , testarea manuala ramane esentiala in descoperirea erorilor din cod, putem efectua orice tip de testare atat manuale cat si automatizate dar testarea 100% automata nu este pozibila sau nu este rentabila , timpul necesar dezvoltarii unor teste automate pentru particularitati depasind timpul necesar testarii manuale pentru testarea manuala a acelorasi particularitati ale sistemului. [99]

In cadrul proiectului am folosit mai multe teste pentru a usura integrarea modulelor individuale, fiecare componenta a proiectului a fost testata individual pentru a-i observa comportamentul si a crea scenariile potrivite pentru a rezolva problemele ce pot aparea prin nefunctionarea acestora.

Ecranul Oled, in mod normal exemplele pentru folosirea acestuia trateaza cazul in care acesta este deconectat sau nu raspunde in cadrul initializarii prin introducerea microcontroller-ului intr-o bucla infinita in care se incearca conectarea la acesta, desigur, acest comportament duce la oprirea intregului sistem , prin simpla indisponibilitate a modulului OLED tot restul functiilor sistemului nefiind disponibile, rolul ecranului fiind pentru diagnoza rapida si de a indeplini rolul de interfata cu utilizatorul el nu indeplineste functii critice , astfel lipsa acestuia este tratata prin executia in continuare a codului.

Senzorii INA219, acestia indeplinesc un rol important in cadrul proiectului, scopul principal al proiectului fiind colectarea de date de la acesti senzori pentru stabilirea diferentei de cantitate de energie electrica generata de diverse sisteme fotovoltaice fie acestea mobile sau fixe, s-a optat pentru introducerea unui blocaj in rularea codului in cazul nefunctionarii acestora pentru a preveni colectarea de date irelevante dupa cum urmeaza:

```
if (! ina219_0x40.begin() || ! ina219_0x44.begin()) {  
    while (1) { delay(10); }  
}
```

Prin testarea senzorilor de temperatura am sesizat o deviatie de temperatura de 1 grad la nivelul senzorului BMP280 , prin compararea rezultatelor de pe acesta , a celor furnizate de AHT21 si a celor de la clima interioara.

Astfel o corectie de 1 grad a fost adaugata direct la citirea senzorului.

```
temperature = bmp.readTemperature()-1.0;
```

Tot in cadrul senzorului BMP s-au depistat aparitia sporadica a unor erori de masurare a presiunii atmosferice , datorita aparitiei sporadice si cu deviatii minore am renuntat la incercarea de corectare a acestora. Alti utilizatori au mai observat date eronate provenite de la acest senzor [100]

7. Concluzii și direcții de dezvoltare

7.1 Concluzii

Sistemul realizat a trecut prin mai multe implementări decât am prevăzut, pornindu-se de la ideea de bază ce presupunea urmărirea soarelui folosind un panou fotovoltaic s-a ajuns la forma actuală ce nu se limitează la o singură funcționalitate, plecând de la o idee cu scopul implementării acestora pe instalațiile fotovoltaice de mici dimensiuni am ajuns la un spațiu de testare pentru funcționalități noi, cu direcții de dezvoltare nelimitate, făcând posibilă testarea și compararea rezultatelor obținute într-o multitudine de situații plecând de la simpla comparare simultană a producției energetice a unui panou fix și unul mobil, diferite tipuri de panouri, diferite tipuri de controlere pentru sistemele fotovoltaice, etc.

La realizarea proiectului am întâmpinat și unele probleme, care totuși au fost rezolvate. Câteva dintre problemele întâmpinate sunt:

- senzorii INA219 necesită înlocuirea cu componente mai robuste, sunt predispuși defectiunilor
- lipsa de experiență de lucru cu ESP32 ce a introdus delay-uri și întârzierea livrării unui proiect funcțional, livrarea funcționalităților bazate pe acesta fiind amânate

7.2 Direcții de dezvoltare

Există diverse direcții de dezvoltare care pot fi aplicate acestui proiect. Cele mai consistente ar fi:

1) Înlocuirea modulelor de control al încărcării PWM cu unele MPPT ce permit o analiză mai detaliată a graficului de energie produsă și studiul producției de energie chiar și pe timpul nopții (luna poate servi ca un reflector), modulele actuale făcând imposibilă extragerea de energie din panou când aceasta se apropie de limita inferioară.

2) Renunțarea la ADS1015 și folosirea fotorezistorilor, implementarea unui algoritm de poziționare bazat pe data, ora, orientare, latitudine și longitudine cu ajutorul ADXL345 și a unor senzori adiționali, aceasta îmbunătățire ar putea facilita montarea panourilor pe platforme tractabile pentru rulote spre exemplu, făcând operațiunea de poziționare 100% automatizată după parcarea vehiculului.

FIGURI ȘI TABELE

Figura 1. Tracker solar pentru panouri solare fotovoltaice [1].....	4
Figura 2. Micro calculator tracker complet poziționar dual motor [2].....	4
Figura 3. Structura programului C++.....	6
Figura 4. Structura programului în Arduino IDE.....	6
Figura 5. Pinout-ul microcontroller-ului esp32.....	8
Figura 6. ATmega2560.....	9
Figura 7. BMP280.....	10
Figura 8. AHT21.....	11
Figura 9. Senzor de curent bidirecțional INA219.....	12
Figura 10. Punte H de putere BTS7960.....	13
Figura 11. Coborător de tensiune XL4015.....	14
Figura 12. Modul microSD.....	15
Figura 13. Modul de control încărcare solară PWM.....	15
Figura 14. Panou solar.....	16
Figura 15. Actuator liniar.....	17
Figura 16. Senzor ADXL345.....	18
Figura 17. Translator nivel logic.....	19
Figura 18. Pinout Mega2560.....	27

BIBLIOGRAFIE

- [1] https://shop.ecosolaris.ro/tracker-orientare-solara/tracker_solar_cu_doua_axe
Accesat în 20 August 2023
- [2] <https://shop.ecosolaris.ro/tracker-orientare-solara/micro-calculator-tracker-complet-pozitioner-dual-motor> Accesat în 20 August 2023
- [3] <https://www.sigmanortec.ro/placa-dezvoltare-esp32-cu-wifi-si-bluetooth> Accesat în 20 August 2023
- [4] <https://www.sigmanortec.ro/Placa-dezvoltare-Mega-2560-R3-cablu-usb-compatibil-ARDUINO-p125149866> Accesat în 20 August 2023
- [5] https://www.sigmanortec.ro/Senzor-BMP280-presiune-atmosferica-temperatura-p125704124?fast_search=fs
- [6] <https://ardushop.ro/ro/home/2510-modul-senzor-umiditate-si-temperatura-aht21-groundstudio.html>
- [7] <https://www.sigmanortec.ro/Senzor-monitor-curent-tensiune-bidirectional-INA219-p136254418>
- [8] <https://www.sigmanortec.ro/punte-h-de-putere-driver-motor-bts7960-43a>
- [9] https://www.sigmanortec.ro/Modul-coborator-tensiune-XL4015-5-36VDC-5A-75W-cu-display-p158469512?fast_search=fs
- [10] https://www.sigmanortec.ro/Modul-MicroSD-p126079625?fast_search=fs
- [11] https://www.sigmanortec.ro/modul-control-incarcare-solara-pwm-rbl-30a-versiune-hq?fast_search=fs
- [12] <https://brecknergermany.ro/panouri-solare/panou-solar-20w-460x350x40mm-monocristalin-cu-regulator-de-incarcare-10a-12-24v-cablu-1900-mm-breckner-germany.html>
- [13] https://www.amazon.com/gp/product/B0B1P88CJB/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1
- [14] <https://www.sigmanortec.ro/modul-accelerometru-adxl345-i2cspi-33-5v>
- [15] https://www.sigmanortec.ro/Modul-Translator-nivel-logic-I2C-IIC-bidirectional-8-canale-3-3V-5V-TXS0108E-p136254032?fast_search=fs
- [99] <https://www.guru99.com/manual-testing.html> Accesat în 1 Septembrie 2023
- [100] <https://github.com/dotnet/iot/issues/1027> Accesat în 26 August 2023

**DECLARAȚIE DE AUTENTICITATE A
LUCRĂRII DE FINALIZARE A STUDIILOR***

Subsemnatul Arnautu Dumitru-Florin

legitimat cu CARTEA DE IDENTITATE seria GZ nr. 836580,

CNP 5000722181091,

autorul lucrării SISTEM PENTRU STUDIUL RENTABILITĂȚII IMPLEMENTĂRII METODELOR DE
EFICIENTIZAREA PRODUCȚIEI DE ENERGIE IN SISTEMLILE FOTOVOLTAICE

elaborată în vederea susținerii examenului de finalizare a studiilor de
LICENTA organizat de către Facultatea DE
AUTOMATICA SI CALCULATOARE din cadrul Universității Politehnica Timișoara, sesiunea
SEPTEMBRIE a anului universitar 2022-2023, coordonator Conf.dr.ing. Adrian Korodi,
luând în considerare art. 34 din *Regulamentul privind organizarea și desfășurarea examenelor de
licență/diplomă și disertație*, aprobat prin HS nr. 109/14.05.2020 și cunoscând faptul că în cazul
constatării ulterioare a unor declarații false, voi suporta sancțiunea administrativă prevăzută de art. 146
din Legea nr. 1/2011 – legea educației naționale și anume anularea diplomei de studii, declar pe proprie
răspundere, că:

- această lucrare este rezultatul propriei activități intelectuale;
- lucrarea nu conține texte, date sau elemente de grafică din alte lucrări sau din alte surse fără ca acestea să nu fie citate, inclusiv situația în care sursa o reprezintă o altă lucrare/alte lucrări ale subsemnatului;
- sursele bibliografice au fost folosite cu respectarea legislației române și a convențiilor internaționale privind drepturile de autor;
- această lucrare nu a mai fost prezentată în fața unei alte comisii de examen/prezentată public/publicată de licență/diplomă/disertație;
- În elaborarea lucrării nu am utilizat instrumente specifice inteligenței artificiale (IA)¹.

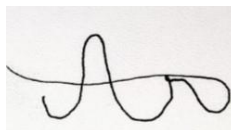
Declar că sunt de acord ca lucrarea să fie verificată prin orice modalitate legală pentru confirmarea originalității, consimțind inclusiv la introducerea conținutului său într-o bază de date în acest scop.

Timișoara,

Data

03/09/2023

Semnătura



*Declarația se completează de student, se semnează olograf de acesta și se inserează în lucrarea de finalizare a studiilor, la sfârșitul lucrării, ca parte integrantă.

¹ Se va păstra una dintre variante: 1 - s-a utilizat IA și se menționează sursa 2 – nu s-a utilizat IA