# **Cuprins**

# 1. Introducere

## 1.1 Memoriu justificativ

## 1.2 Descriere generală a aplicației

## 1.3 Schema bloc

## 1.4 Descriere generală a blocurilor

## 1.5 Implementare practică

## 1.6 Aplicații utilizate

# 2. Module

## 2.1 Interfața cu utilizatorul

### 2.1.1 Generalități

### 2.1.2 Display

### 2.1.3 Buton meniu

### 2.1.4 Joystick

## 2.2 Control încărcare

### 2.2.1 Generalități

### 2.2.2 Panou

### 2.2.3 Convertorul

### 2.2.4 Baterie

### 2.2.5 Dioda

### 2.2.6 Releul

# 3. Microcontrolerul

## 3.1 Generalități

## 3.2 ADC

# 4. Software

## 4.1 Generalități

## 4.2 Descriere sumară

# 5. Analize și rezultate

## 5.1 Rapoarte aplicație

# 6. Concluzie

## 

# Introducere

## 1.1 Memoriu justificativ

# Economia de energie este la ora actuală o prioritate mondială, protejarea planetei și a resurselor ei devenind obiective principale internaționale. Contextul energetic mondial, conduce către o preocupare intensă în domeniul energiilor neconvenționale.

Energiile regenerabile sunt considerate energii care provin din surse care se regenerează de la sine în scurt timp sau care sunt surse practic inepuizabile.

Printre sursele regenerabile de energie se enumeră:

1. Energia eoliană;
2. Energia solară;
3. Energia apei;
4. Energia geotermică;
5. Energia de biomasă;

Energia solară ocupă un loc important pe plan mondial. Aceasta poate fi captată și transformată fie în energie electrică prin utilizarea tehnologiilor fotovoltaice, fie în energie termică, prin utilizarea diferitelor tipuri de panouri solare termice. În acest context, lucrarea are la bază producerea energiei electrice cu ajutorul panourilor fotovoltaice și transferul termic ce are loc la nivelul acestora.

Panourile fotovoltaice sunt dispozitive ce transformă energia solară în energie electrică. O problemă existentă în producerea energiei este dată de fenomenologia ce decurge din caracterul sursei solare și al variațiilor meteorologice imprevizibile. Pe de altă parte sistemele fotovoltaice nu utilizează decât o mică parte din radiația solară și anumite lungimi de undă, pentru a produce energie electrică. Restul energiei primite este transformată în căldură, ce conduce la creșterea temperaturii celulelor componente și la scăderea randamentului lor.

În consecință, creșterea productivității energetice a acestor instalații presupune atât eficientizarea funcționării lor în domeniul electric, cât și studiul fenomenelor termice care au loc.

## Descriere generală a aplicației

Aplicația prezentată este un sistem autonom care convertește energia luminii solare în electricitate cu ajutorul unui panou solar.

Pentru a funcționa cu o eficiență maximă, sistemul determină prin intermediul senzorului de lumină poziția optimă a panoului solar astfel încât să capteze cât mai multă energie. Mișcarea de rotație si înclinare a panoului este realizată cu ajutorul unui motor pas cu pas și a unui motor de curent continuu.

Scopul acestei aplicații este de a încarcă eficient o baterie de 12 V dar și pentru a oferi protecție la supratensiune, supraîncărcare, supracurent și descărcare accidentală a bateriei.

1. Protecție la supratensiune: în momentul în care panoul produce o tensiune prea mare, aceasta este reglata la un nivel acceptabil pentru baterie prin intermediul unui convertor de tip coborâtor (convertor buck).



Figura 1.2.1

2. Protecție la supraîncărcare: daca tensiunea de la bornele bateriei depășește pragul maxim admis, se întrerupe procesul de încărcare prin acționarea unui releu ce realizează legătura electrică între baterie și convertor.



Figura 1.2.2

3. Protecție la supracurent: se utilizează un disjunctor montat la bornele bateriei cu pragul de activare setat la valoarea de 10 A;



Figura 1.2.3

4. Protecție împotriva descărcării accidentale a bateriei: se utilizează o diodă care împiedică trecerea curentului electric de la baterie către convertor;



Figura 1.2.4

Disjunctorul este un aparat electric care întrerupe pe cale mecanică, în mod automat, curentul, conform cu cerințele unei funcționări.

## Schema bloc

Schema bloc a aplicației:



Figura 1.3.1

## Descriere generală a blocurilor

**MICROCONTROLERUL**

Microcontrolerul are rolul de unitate principală de logică și control. Acesta preia datele de intrare necesare (în acest caz intensitatea luminoasă) și acționează corespunzător modului de funcționare calculat pe baza datelor: acționare motoare, pornire/oprire proces de încărcare a bateriei, interacțiune cu utilizatorul, etc.



Figura 1.4.1

**PANOUL SOLAR**

Panoul solar este blocul cheie al acestei aplicații. Rolul său este a prelua energia luminoasă de la soare și de a o transforma în energie electrică.



Figura 1.4.2

**SENZOR**

Blocul senzor, montat pe panoul solar, este format din patru senzori de citire a intensității luminoase (patru fotorezistențe) și are rolul de a oferi microcontrolerului datele de intrare necesare detectării poziției optime, astfel încât panoul solar să capteze cantitatea maximă de energie posibilă. Cele patru fotorezistențe sunt despărțite între ele printr-un scut opac de plastic ce formează patru zone de interes din care se pot citi intensitățile luminoase.



Figura 1.4.3

**MOTOARE ȘI DRIVERE**

Pentru a funcționa cu eficiență maximă, sistemul poate orienta panoul solar pe două axe astfel încât razele solare să ajungă perpendicular pe acesta. Blocul motoare și drivere este responsabil cu mișcările de rotație și inclinație a panoului. Mișcarea de rotație este asigurată de un motor de curent continuu, în timp ce mișcarea de înclinare se realizează cu ajutorul unui motor de tip pas cu pas. Motoarele sunt acționate cu ajutorul driverelor specifice, care la rândul lor sunt controlate de microcontroler.

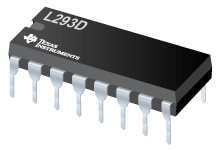


Figura 1.4.4

**BATERIE**

Bateria are rolul de a stoca energia generată de panoul solar pentru a putea fi utilizată ulterior. De asemenea, bateria reprezintă sursa principală de alimentare a sistemului.



Figura 1.4.5

**CONTROL ÎNCĂRCARE**

Elementele principale ce alcătuiesc blocul control încărcare sunt convertorul și releul. Cu ajutorul convertorului, tensiunea generată de panoul solar (aproximativ 18V) este adusă la un nivel potrivit încărcării bateriei (aproximativ 14.5V). Releul joacă rolul de întrerupător al alimentării bateriei de către panou.

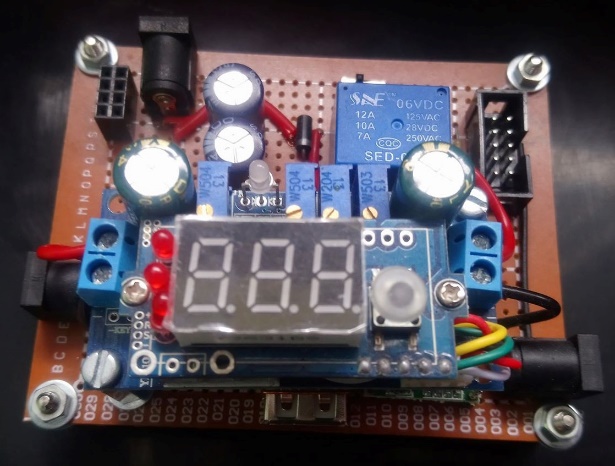


Figura 1.4.6

**INTERFȚĂ UTILIZATOR**

Blocul interfață utilizator permite comunicarea dintre sistem și utilizatorul uman. Elementul component display permite utilizatorul să vadă în timp real modul de funcționare al sistemului, datele procesate și alte informații utile (tensiune baterie, nivel de încărcare, tensiune convertor, etc. ). Prin intermediul butonului se poate schimba manual modul de funcționare al sistemului. Joystick-ul este utilizat în modul de funcționare control manual pentru a permite utilizatorului acționarea manuală a motoarelor.



Figura 1.4.7

## Implementare practică

****

Figura 1.5.1

## Aplicații utilizate

În procesul de realizare a proiectul, s-au folosit mai multe aplicații software. Acestea au avut numeroase roluri printre care se enumeră:

* scriere și generare cod sursă
* versionare evoluție cod aplicație
* încărcare cod mașină pe microcontroler
* design hardware: schematic & layout
* design model 3D
* printare model 3D
* design workflow & scheme bloc

**Atmel Studio 7.0**

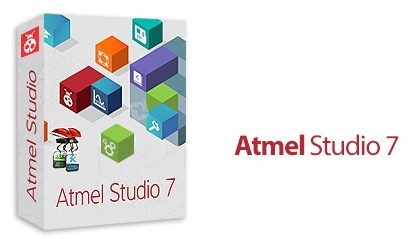


Figura 1.6.1

Atmel Studio 7 este o platformă de dezvoltare integrată (IDP) construită pentru dezvoltarea și depanarea tuturor aplicațiilor specifice microcontrolerelor de tip AVR și SAM.

Caracteristici principale:

* oferă suport pentru peste 500 de dispozitive AVR și SAM;
* oferă o vastă librărie de coduri sursă, inclusiv drivere, stive de comunicare, peste 1.600 de exemple de proiecte cu cod sursă, servicii grafice și funcționalitate prin atingere utilizând Advanced Software Framework (ASF);
* extensii IDE prin intermediul Atmel Gallery și magazin de aplicații online pentru instrumente de dezvoltare și software încorporat de la Microchip;
* monitorizare în timp real al consumului de energie și vizualizare de date în timp real cu posibilitatea de a genera grafice utilizând Atmel QTouch Composer;
* scriere și depanare cod C / C ++ și cod asamblare cu compilatorul integrat;
* funcții avansate de depanare incluzând întreruperea datelor, monitorizare întrerupere, urmărire a datelor interogate;
* editor integrat cu asistență vizuală;
* simulare completă a cipurilor pentru modele specifice de microcontrolere, întreruperi, periferice și stimuli externi

**GitHub Desktop**

****

Figura 1.6.2

GitHub Desktop este o aplicație de tip GUI (Graphical User Interface) ce permite utilizarea cu ușurință a funcționărilor de versionare Git. Git este un sistem de control al versiunii pentru urmărirea modificărilor în fișierele pe computer și coordonarea activității asupra acestor fișiere în rândul mai multor persoane. Acesta este utilizat în principal pentru gestionarea codului sursă în dezvoltarea de programe software, dar poate fi folosit pentru a urmări schimbările în orice set de fișiere. Ca sistem de control revizuit și distribuit, acesta vizează viteza, integritatea datelor și suport pentru fluxurile de lucru distribuite, neliniare.

Git a fost creat de Linus Torvalds în 2005 pentru dezvoltarea kernel-ului Linux, alți dezvoltatori de kernel-uri contribuind la dezvoltarea sa inițială.

Ca în majoritatea sistemelor de control distribuite și, spre deosebire de cele mai multe sisteme client-server, fiecare director Git de pe fiecare calculator este un depozit complet cu istorie completă și abilități de urmărire a versiunii complete, independent de accesul la rețea sau un server central.

Git este software gratuit și open-source, distribuit în termenii licenței GNU General Public Version 2.

Versionarea și codul sursă ale acestui proiect sunt valabile la următorul link:

https://github.com/scorpionipx/DD\_IPX/tree/develop/licenta/cod/SunTracking32A/SunTracking32A/SunTracking32A

**AVR Extreme Burner:**

****

Figura 1.6.3

AVR Extreme Burner este un program software de tip GUI (Graphical User Interface) dedicat programatoarelor pentru microcontrolere AVR și care folosesc la bază protocolul USB ASP (Universal Serial Bus – AVR Serial Programmer).

Aceasta diminuează efortul depus de utilizatorul în procesul de scriere a memoriilor EEPROM și flash pe microcontrolere.

Aplicația a fost utilizată pentru a încărca pe microcontroler codul sursă dezvoltat.

Pentru a flash-ui codul mașină pe microcontroler s-a utilizat un programator de tipul Atmel USBASP.



Figura 1.6.4

**EAGLE**

****

Figura 1.6.5

EAGLE este o aplicație software de automatizare a proiectării electronice, care permite creare schematicelor, layout-urilor pentru circuite imprimate (PCB), caracteristici de auto-router și asamblare automată. EAGLE (Easily Applicable Graphical Layout Editor) este un editor ușor de utilizat în format grafic și este dezvoltat de CadSoft Computer GmbH.

Caracteristici principale:

* conține un editor schematic, pentru proiectarea schemelor de circuit. Schematicul este stocat în fișiere cu extensie .SCH, iar componentele sunt definite în bibliotecile de dispozitive cu extensia .LBR. Componentele pot fi plasate pe mai multe coli și conectate împreună prin porturi;
* editorul de layout PCB stochează fișierele de tip board (plăcuță) cu extensia .BRD. Aceasta permite adnotarea înapoi la schema și rutarea automată pentru a conecta in mod automat componentele pe baza conexiunilor definite în schematic;
* EAGLE salvează fișierele de tip board generate sub forma Gerber și PostScript, iar fișierele de găurit sub format Excellon și Sieb & Meyer. Acestea sunt formatele de fișiere standard acceptate de companiile de fabricare a PCB-urilor, însă având în vedere faptul că cei mai mulți utilizatori EAGLE sunt reprezentați de firmele mici și pasionați, mulți fabricanți de PCB și magazine de asamblare acceptă și fișiere de bord EAGLE (cu extensia .BRD).
* EAGLE oferă o interfață grafică de tip windows și un sistem de meniuri pentru editare, management de proiect și pentru personalizarea interfeței și a parametrilor de proiectare. Sistemul poate fi controlat prin intermediul mouse-ului, scurtăturilor de tastatură sau prin introducerea unor comenzi specifice într-o linie de comandă încorporată. Comenzile repetate multiple pot fi combinate în fișierele de tip script (cu extensia de fișiere .SCR). Este, de asemenea, posibil să se exploreze fișiere de proiectare utilizând un limbaj de programare orientat pe obiecte specific EAGLE (cu extensia .ULP).

**VECTARY**

****

Figura 1.6.6

Vectary este un instrument 3D online, ușor de folosit, unde se pot realiza modele 3D. Aplicația permite personalizarea șabloanelor sau începerea propriului proiect prin asamblarea elementelor dintr-o imensă bibliotecă de materiale 2D și 3D de la Google Poly și alte arhive ce permit căutarea materialelor.

Cu ajutorul aplicației VECTARY s-a realizat designul 3D al scutului opac ce desparte cele patru fotorezistențe din alcătuirea senzorului de intensitate luminoasă.

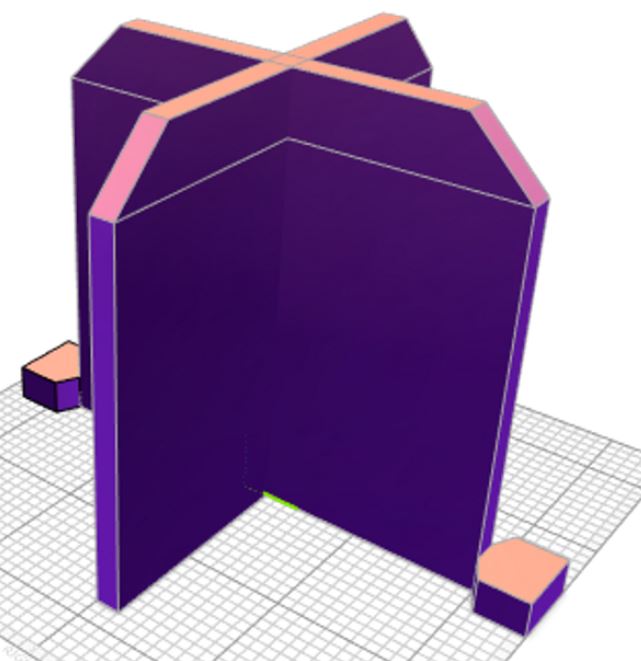


Figura 1.6.7

Modelul 3D poate fi găsit accesând link-ul următor:

<https://www.vectary.com/model3d/5dae01d1-ef5c-4821-adae-3c206c1e17d4>

**OctoPrint**

****

Figura 1.6.8

OctoPrint este o aplicație ce permite controlul imprimantelor 3D prin intermediul unui browser web. Această aplicație a fost folosită pentru a printa 3D modelul scutului creat.

Imprimanta 3D folosită este una modelul Anycubic 3D Printer I3 Mega.



Figura 1.6.9

**Microsoft Visio**

****

Figura 1.6.10

Visio este un program software ce ajută la crearea în mod simplu și intuitiv a schemelor logice, diagramelor de rețea, organigramelor, planurilor de nivel, proiectelor de inginerie și altele utilizând forme și șabloane moderne. A fost utilizat pentru crearea schemelor bloc ale proiectului.

# 2. Module

## 2.1 Interfața cu utilizatorul

### 2.1.1 Generalități

O interfață este o parte a unui sistem care servește comunicării. În tehnologia calculatoarelor, termenul de interfață se referă la un punct de interacțiune dintre două unități, dispozitive componente ale unui sistem, care este compatibil din punct de vedere hardware și software .

Interacțiunea si schimbul de informații printr-o interfață, între două unități de sistem, se realizează prin mărimi fizice precum tensiune, curent electric sau prin mărimi logice care se pot prezenta sub formă de semnale analogice (continue) sau semnale digitale (discontinue, discrete).

**Interfața cu utilizatorul**

Majoritatea sistemelor dețin o interfață prin intermediul căreia realizează comunicarea cu operatorul uman. În cazul de față, sistemul implementat dispune de o interfață cu utilizatorul atât hardware cât și grafică.

**Interfața hardware**

Exemplu: tastatura unui mic calculator de buzunar poate fi considerată o interfață hardware.

Interfața hardware a sistemului este formată din butoanele de meniu ce permit comutarea modului de funcționare și joystick-ul utilizat pentru comanda manuală a motoarelor.

**Interfața grafică**

Interfața grafică cu utilizatorul (GUI – Graphical User Interface) se situează din punct de vedere funcțional între utilizator și dispozitive electronice cum ar fi computere, dispozitive personale de tip hand-held (playere MP3, playere media portabile, dispozitive de jucat), aparate electrocasnice și unele echipamente de birou.

### 2.1.2 Display

Elementul principal al interfeței grafice a sistemului construit este display-ul LCD (Liquid Crystal Display) HX1230, o copie chinezească, dar îmbunătățită a modelului NOKIA 5110.

Afișajul cu cristale lichide sau prescurtat LCD este un [dispozitiv](https://ro.wikipedia.org/w/index.php?title=Dispozitiv&action=edit&redlink=1) de afișare pentru litere, cifre, grafică și imagini, fiind constituit dintr-o matrice de celule lichide care devin opace sau își schimbă culoarea sub influența unui curent sau câmp electric.

Un afișaj LCD se prezintă sub forma unui [ecran](https://ro.wikipedia.org/wiki/Ecran) afișor (display) care este comandat electronic printr-un [decodificator](https://ro.wikipedia.org/w/index.php?title=Decodificator&action=edit&redlink=1) de caractere numerice și alfabetice.

**Display-ul Nokia 5110**

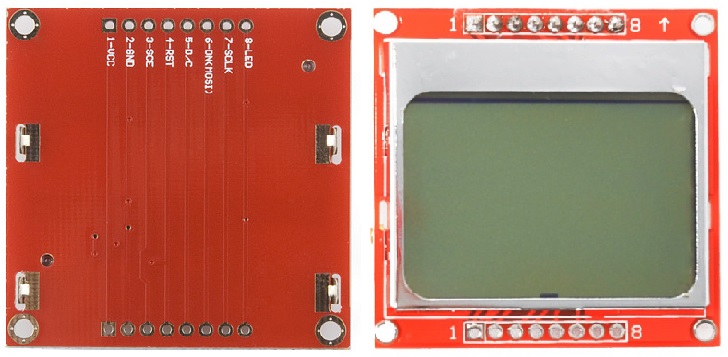
****

Figura 2.1.2.1

**Display-ul HX1230**

****

Figura 2.1.2.2

**Descriere configurație pini:**

1. Reset - acest pin resetează modulul în momentul în care i se aplică o tensiune de 0V, ducând sistemul într-o stare sigură, implicită și cunoscută;
2. Chip Enable (CE) – prin intermediul acestui pin se selectează display-ul ca dispozitiv comandat (de interes) pe magistrala de data și clock a interfeței SPI;
3. Data/Command (DC) - acest pin este utilizat pentru comutarea între modul de transmitere al datelor și modul transmitere al comenzilor; copia chinezeasca (HX1230) nu are implementat această funcționalitate. În schimb modulul său SPI este construit astfel încât să recepționeze date în pachete de 9 biți (față de pachetul standard de 8 biți). Astfel, primul bit transmis servește drept selecție al modului de interpretare a datelor receptate de display: 0 = comandă, 1 = date;
4. Serial Input (DIN) - acesta este pinul de intrare prin care sunt trimise datele seriale de către microcontroler;
5. Clock (CLK) – acestui pin i se furnizează un semnal sursă de ceas (clock) prin care se sincronizează dispozitivele SPI conectate pe magistrală;
6. Power (Vcc) – acest pin este folosit pentru alimentarea afișajului, tensiunea de alimentare poate varia între 2,7V și 3,3V;
7. Back Light (BL) - acest este folosit pentru acționarea luminii de fundal a afișajului ;
8. Ground (Gnd) – punctul de referință;

**Caracteristici:**

* Tensiunea de operare cuprinsă între 2.7V și 3.3V;
* Consumul de curent redus de aproximativ 6 mA;
* Constă din 68 de rânduri și 96 de coloane (96 x 68) pixeli monocromi (modelul Nokia 5110 dispunea de numai 84 × 48 pixeli);
* Funcționează utilizând interfața SPI;
* Controlat cu driver-ul Philips PCD8544 pentru o interfață ușoară;
* Poate fi ușor interfațat cu Arduino;
* Suporta grafica imaginilor în format bitmap;
* Interfață serială de maximum 8 Mb/s;
* Consum redus de energie, potrivit pentru funcționarea cu baterii;
* Interval de temperatură: -25 până la +70 ° C.

Display-ul este controlat de microcontroler prin intermediul unui driver dedicat. În domeniul electronicii, un driver de afișaj este de obicei un circuit integrat de tip semiconductor (dar poate cuprinde, în mod alternativ, o mașină de stări(state machine)) care oferă o interfață (legătură) între un microprocesor, microcontroler, ASIC sau o interfață periferică generală și un anumit tip de dispozitiv de afișare, de ex LCD, LED, OLED, ePaper, CRT, Fluorescent vid sau Nixie.

Driverul de afișare va accepta în mod obișnuit comenzi și date utilizând o interfață standard serială sau paralelă, cum ar fi TTL, CMOS, RS232, SPI, I2C etc. și generează semnale de tensiune, curent, temporizare și demultiplexare adecvate pentru a controla dispozitivul de afișare să expună textul sau imaginea dorită.

Driverul de afișare poate fi în sine un microcontroler specific aplicației și poate include memoria RAM, memoria flash, EEPROM și / sau ROM. Memoria ROM-ul poate conține firmware-ul și diferite fonturi de afișare.

Un exemplu notabil al unui IC (integrated circuit) driver de afișare este controlerul LCD Hitachi HD44780.

În cazul de față, display-ul HX1230 utilizează driver-ul PCD8544

**Driver display Philips PCD8544:**

PCD8544 este un controler / driver CMOS cu putere redusă, conceput pentru a conduce un afișaj grafic de 68 de rânduri și 96 coloane.

Funcțiile necesare pentru afișare sunt furnizate într-un singur chip, inclusiv generarea pe chip a alimentarii LCD-ului, astfel rezultând un minim de componente externe necesare și un consum redus de energie.

Pentru a interfera cu microcontrolerul, se folosește interfața serială SPI, cu lungimea cuvintelor de cod de 9 biți.

**Schema bloc driver PCD8544:**

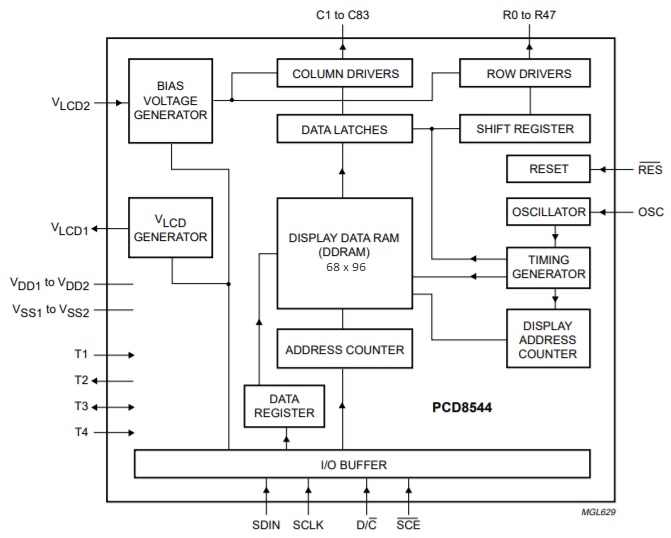
****

Figura 2.1.2.3

**Interfața seriala SPI**

Interfața seriala SPI (Serial Peripheral Interface) este o interfață sincronă standard de mare viteză, ce operează în mod full duplex.

Este folosită pentru transmiterea de date, unde circuitele sunt interconectate utilizând principiul master-slave. Modul master/slave semnifică faptul că circuitul digital master inițiază transmiterea de date. Este permisă conectarea mai multor dispozitive digitale slave cu selecție individuală.

SPI-ul are patru semnale logice specifice:

* **SCLK** - Ceas serial (ieșire din master).
* **MOSI/SIMO** - Master Output, Slave Input (ieșire master, intrare slave).
* **MISO/SOMI** - Master Input, Slave Output (intrare master, ieșire slave).
* **SS** - Slave Select (active low, ieșire din master).

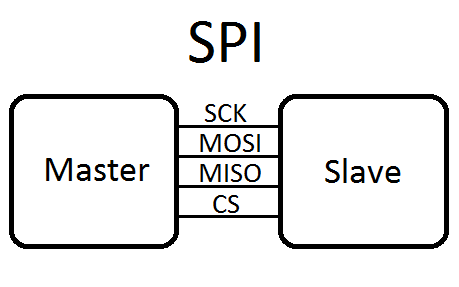


Figura 2.1.2.4

**Mod de funcționare:**

Ca mod de operare, interfața SPI utilizează un singur dispozitiv master și unul sau mai multe dispozitive slave.

Pentru a realiza transmisia, master-ul configurează mai întâi semnalul de ceas (clock) folosind o frecvență mai mică sau egală cu cea suportată de dispozitivele slave.

În timpul fiecărui ciclu de ceas SPI, apare o transmisie full duplex:

* master-ul trimite un bit pe linia MOSI; slave-ul îl citește de pe aceeași linie;
* slave-ul trimite un bit pe linia MISO; master-ul îl citește de pe aceeași linie.

Comunicația pe SPI implică de obicei existența a doi regiștri de shiftare (Shift Registers), unul aparținând dispozitivului master și unul aparținând dispozitivului slave, conectați în ciclu.

**Principiul de transfer al datelor specific comunicației SPI**

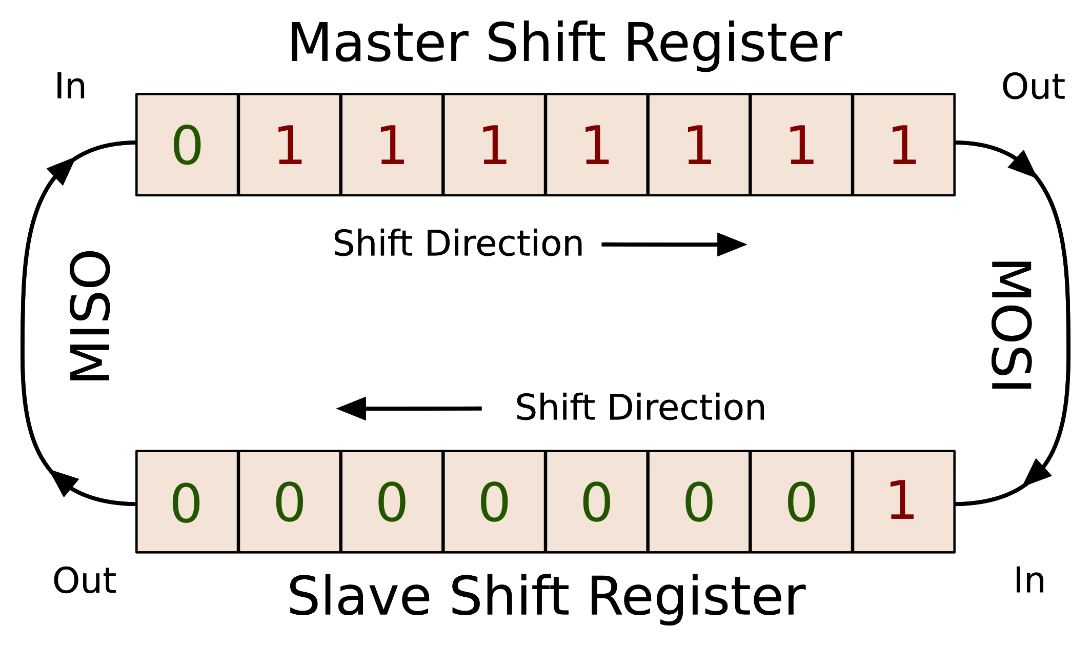
****

Figura 2.1.2.5

De obicei primul bit shiftat pe liniile de MISO/MOSI este bitul cel mai semnificativ, în timp ce un nou bit este adăugat pe poziția cea mai puțin semnificativă din registru. După ce întregul cuvânt a fost trimis, prin shiftare, masterul si slave-ul au interschimbat valorile din cei doi regiștri de shift. Dacă mai există date de transmis, procesul este reluat. Când nu mai există date de transmis, masterul întrerupe generarea semnalului de clock și, în general, pune valoarea de 1 logic pe linia de SS asociata dispozitivului slave. Cipurile slave care nu au fost selectate vor ignora semnalele de pe SCK și MOSI și nu vor genera nimic pe linia MISO.

**Control modul display**

Pentru a putea utiliza modulul display HX1230, acesta necesită o tensiune de alimentare de 3.3V. Microcontrolerul utilizat in cadrul proiectului funcționează la tensiunea nominală de 5V. Pentru a reduce nivelul de tensiune de 5V la cel util afișajului, 3V3, se utilizează regulatorul liniar de tip LDO (Low Dropout Regulators) KA78R33.

**KA78R33**

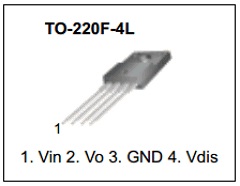
****

Figura 2.1.2.6

**Schema bloc internă a integratului KA78R33**

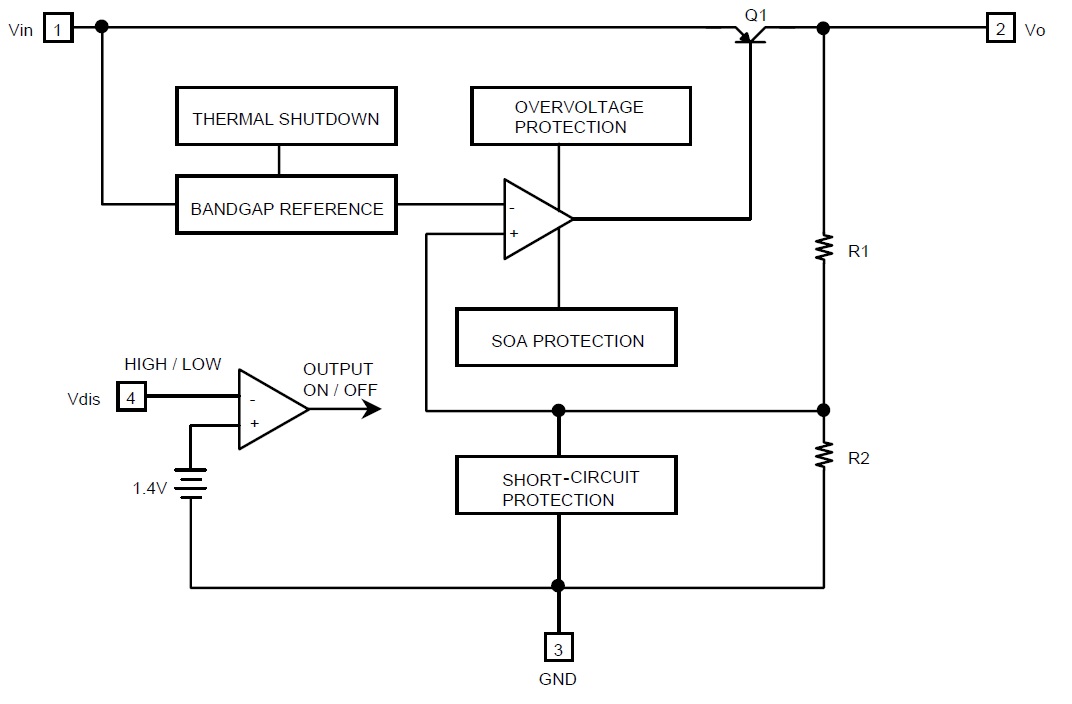
****

Figura 2.1.2.7

Din moment ce comunicația dintre microcontroler si display este unidirecțională (microcontrolerul doar trimite date, iar display-ul doar recepționează date), pentru a reduce nivelul de tensiune al semnalelor transmise de microcontroler (5V) către display (3V3) este suficientă utilizarea unor divizori rezistivi cu factorul de divizare 2/3.

**Principiu divizor rezistiv**

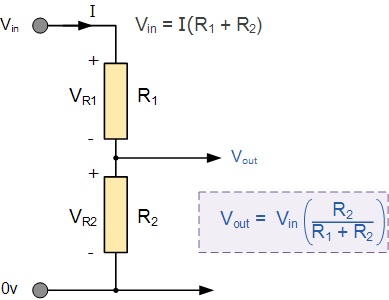
****

Figura 2.1.2.8

În dezvoltarea sistemului s-au utilizat rezistențe de valori 1K8Ω (R1) și 3K6Ω (R2).

**Calcul tensiune divizor**

Vin = 5V

R1 = 1K8

R2 = 3K6

**Implementare hardware – schematic**

****

Figura 2.1.2.9

Adaptorul pentru driver-ul display-ului a fost conceput pentru a putea controla cu ajutorul microcontrolerului alimentarea și lumina de fundal a afișorului, dar și pentru a permite funcționarea continuă a acestuia, fără nicio intervenție externă.

Tranzistorul T1 este folosit pentru a porni și pentru a opri alimentarea generală a afișajului. Un semnal de 0 logic aplicat acestuia reprezintă pornirea alimentării modului display prin activarea pinului Vdis (pin cu funcționalitate de enable) a regulatorului de tensiune KA78R33. Un semnal de 1 logic în baza tranzistorului T1 va furniza un semnal de 0 logic pinului Vdis, astfel întrerupând alimentarea display-ului.

Tranzistorul T2 permite pornirea și oprirea luminii de fundal a display-ului. Un semnal de 0 logic aplicat acestuia reprezintă aprinderea luminii de fundal, în timp ce un semnal de 1 logic ar cauza stingerea luminii de fundal. Pentru a regla intensitatea luminii de fundal a afișajului, baza tranzistorului T2 poate fi acționate de un semnal PWM (Pulse-width modulation = semnal modulat în durată). Astfel, intensitatea luminii de fundal va fi direct proporțională cu factorul de umplere al semnalului de comandă.

Conectorii JP1 și JP2 sunt utilizați pentru a selecta modul de comandă al alimentării display-ului și controlul luminii de fundal al acestuia. În implementarea practică se utilizează modul always-on (mereu pornit). Acest mod este selectat prin intermediul unor jumpere care conectează electric pinul Vdis al regulatorului KA78R33 la tensiunea de alimentare generală (Vcc, 5V), respectiv pinul BL (Back Light) al display-ului la tensiunea de alimentare de 3V3.

**Adaptor microcontroler – display**

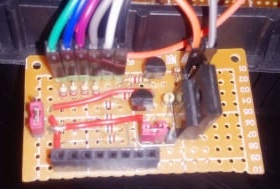


Figura 2.1.2.10

**Utilizare display:**

Cu ajutorul display-ul HX1230 se arată utilizatorului modul curent de funcționare al sistemului, dar și date interne în timp real. Tranziția de la un mediu de funcționare la altul se face prin intermediul butonului atașat.

**Modul inițializare**

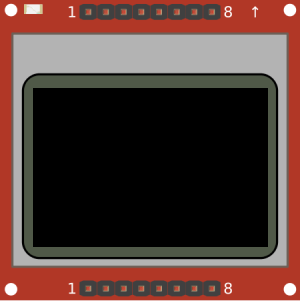
****

Figura 2.1.2.11

Modul de funcționare **inițializare** pregătește sistemul activând driverele interne și perifericele necesare funcționării și durează aproximativ 850 milisecunde de la pornirea alimentării. În tot acest timp, display-ul menține activați toți pixelii display-ului.

**Modul tracking / debugging**

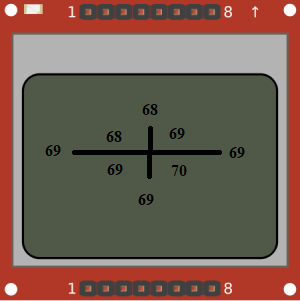
****

Figura 2.1.2.12

Modul de funcționare **tracking / debugging** activează funcționalitatea de găsire automată a poziției optime a panoului astfel încât acesta să capteze cantitatea maximă de energie posibilă. În plus, pe display se afișează datele preluat de microcontroler prin intermediul senzorului și informațiile calculate ulterior.

Desenul grafic de pe display ilustrează senzorul (format din patru fotorezistențe și un scut opac din plastic) atașat panoului solar și intensitatea luminoasă preluată de fiecare senzor în parte, sub formă procentuală (unde 0% înseamnă întuneric total, iar 100% înseamnă intensitate luminoasă maximă). Deasupra, dedesubtul și de o parte și de alta a reprezentării grafice a senzorului se afișează mediile intensităților luminoase calculate pentru fiecare zonă de interes (sus, jos, stânga, dreapta).

Pe baza informațiilor calculate, microcontrolerul va roti și / sau înclina panoul prin intermediul motoarelor controlate de drivere.

**Modul idle**

****

Figura 2.1.2.13

Modul idle dezactivează toate funcționalitățile și pregătește sistemul să intre în modul sleep.

**Modul sleep**

Modul sleep are rolul de a reduce la minim consumul sistemului. Toate funcționalitățile acestuia sunt dezactivate cu excepția celei de citire a butonului. Se așteaptă intervenția utilizatorului pentru a schimba modul de funcționare.

În acest timp, display-ul este dezactivat.

**Modul monitorizare**

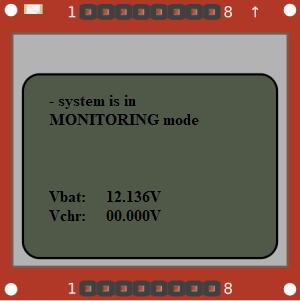
****

Figura 2.1.2.14

Modul monitorizare permite utilizatorului să urmărească în timp real tensiunea de la bornele bateriei și tensiunea de ieșire a convertorului buck. Datele sunt preluate prin intermediului modulului ADC al microcontrolerului, interpretate și apoi afișate pe display.

**Modul manual**

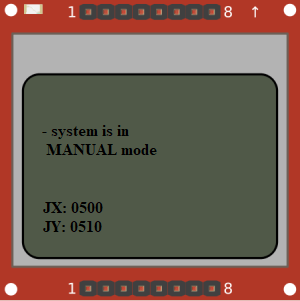


Figura 2.1.2.15

Modul manual permite utilizatorului control manual al motoarelor ce rotesc și inclină panoul solar. Pe display sunt afișate în timp real datele transmise microcontrolerului de către joystick.

**Implementare software – cod sursă**

Codul sursă utilizat pentru controlul display-ului este dezvoltat în fișierele hx1230.h și hx1230.c. În plus, în fișierul hx\_8x6\_characters.h sunt definite reprezentările simbolurilor tabelului ASCII (American Standard Code for Information Interchange) sub formă de vectori bidimensionali.

Pentru controlul display-ului HX1230 se utilizează numai patru pini ai microcontrolerului (PC0, PC1, PC6 și PC7) configurați astfel încât să îndeplinească funcțiile unui modul SPI.

**Configurare pini microcontroler**

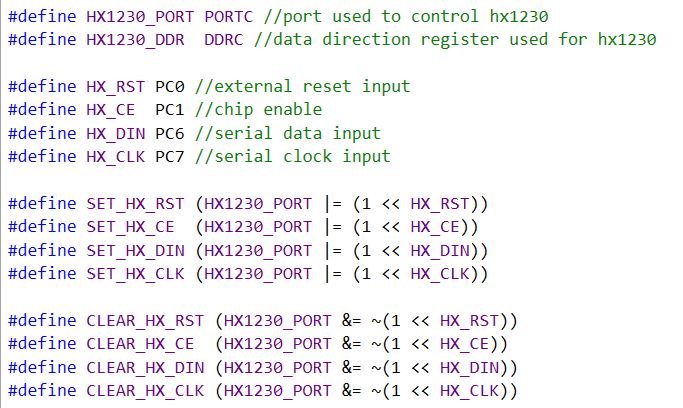


Figura 2.1.2.16

S-au dezvoltat un număr restrâns de funcții cu ajutorul cărora se controlează modulul display.

**Funcții de control al display-ului**

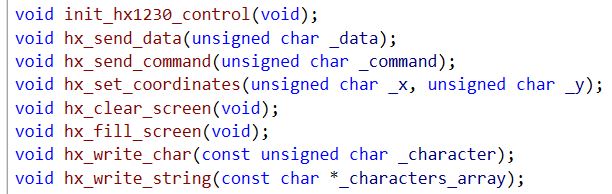


Figura 2.1.2.17

**Descriere funcții**

Funcția **void init\_hx1230\_control(void)** pregătește configurarea hardware a microcontrolerului necesară pentru a controla display-ul.

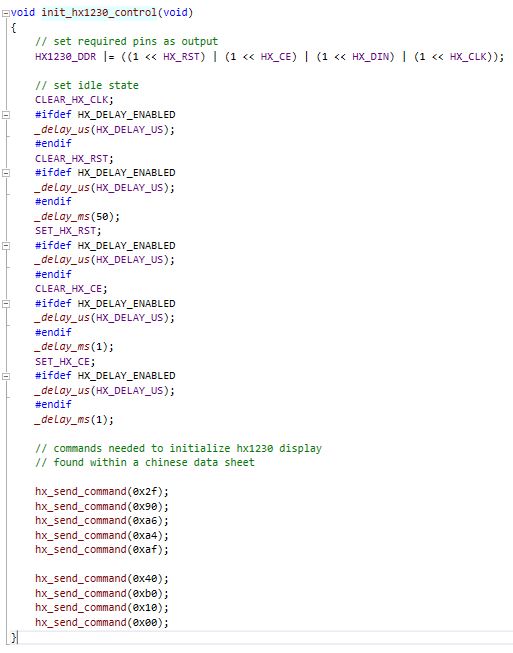


Figura 2.1.2.18

Seria de comenzi necesare inițializării modulului display HX1230 a fost preluată din interiorul datasheet-ului dispozitivului.

Link către datasheet: https://cdn.instructables.com/ORIG/F4Y/Z9ZU/J76GKXJX/F4YZ9ZUJ76GKXJX.pdf

Funcția **void hx\_send\_data(unsigned char \_data)** transmite către display un cuvânt de 8 biți (specificat prin intermediul parametrului \_data, de tip unsigned char) sub formă de date. În prealabil se transmite un bit de nivelul 1 logic, pentru a specifica transferul unui cuvânt de tip date.

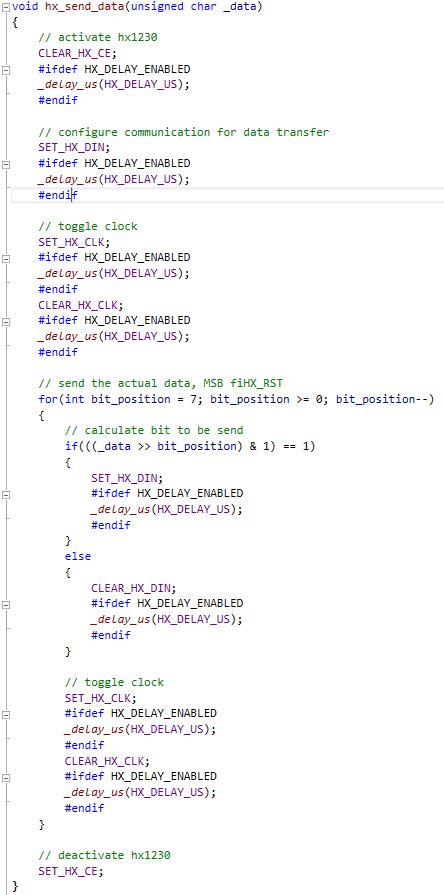


Figura 2.1.2.19

Funcția **void hx\_send\_command(unsigned char \_command)** transmite către display un cuvânt de 8 biți (specificat prin intermediul parametrului \_command, de tip unsigned char) sub formă de comandă. În prealabil se transmite un bit de nivelul 0 logic, pentru a specifica transferul unui cuvânt de tip comandă.

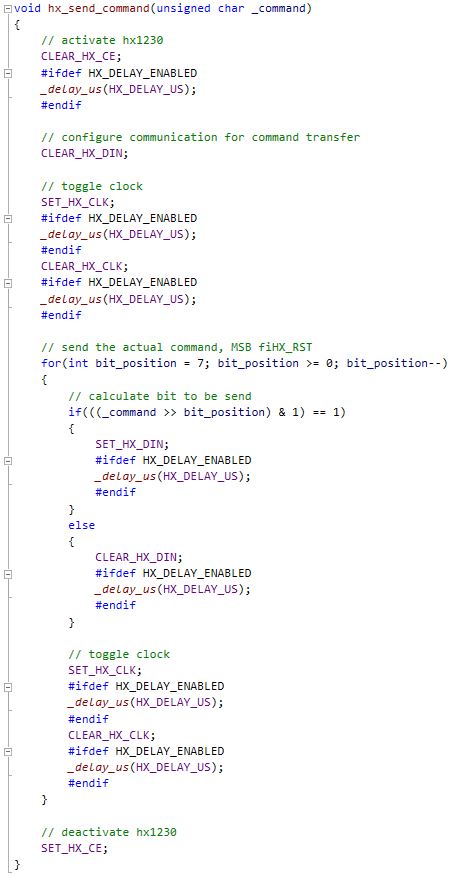


Figura 2.1.2.20

Funcția **void hx\_set\_coordinates(unsigned char \_x, unsigned char \_y)** transmite către display comanda de muta cursorul la coordonatele indicate. Coordonata carteziană X este specificată prin intermediul parametrului \_x de tip unsigned char, iar coordonată carteziană Y este specificată prin intermediul parametrului \_y de tip unsigned char.

Poziția cursorului reprezintă zona fizică unde va fi afișată următoarea data transmisă de microcontroler către display. Coordonata (0, 0) reprezintă pixelul aflat în colțul din stânga sus al ecranului.

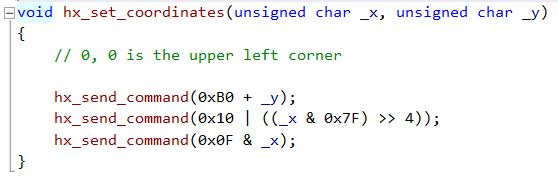


Figura 2.1.2.21

Funcția **void hx\_clear\_screen(void)** transmite comenzile și datele necesare ștergerii complete a ecranului (se sting toți pixelii).

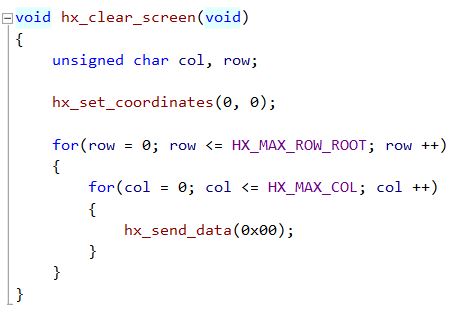


Figura 2.1.2.22

Funcția **void hx\_fill\_screen(void)** transmite comenzile și datele necesare umplerii complete a ecranului (se aprind toți pixelii).

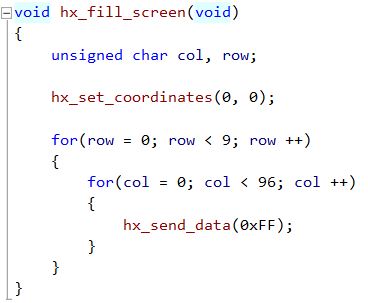


Figura 2.1.2.23

Funcția **void hx\_write\_char(const unsigned char \_character)** transmite datele și comenzile necesare afișării pe ecran a unui caracter ASCII. Caracterul este specificat prin intermediul parametrului **\_character** de tip const unsigned char.

Valoarea parametrului este codul ASCII corespunzător caracterului dorit. În realitate acesta reprezintă indexul vectorului de date necesare a fi transmise pentru a reprezenta caracterul pe ecran. Vectorii de date sunt definiți în fișierul hx\_8x6\_characters.h.

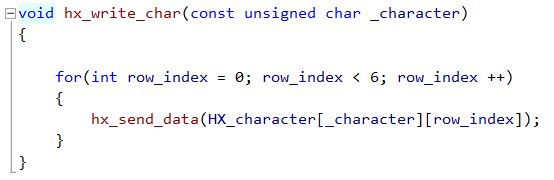


Figura 2.1.2.24

Vectorul bidimensional ce conține datele necesare afișării unui caracter ASCII:

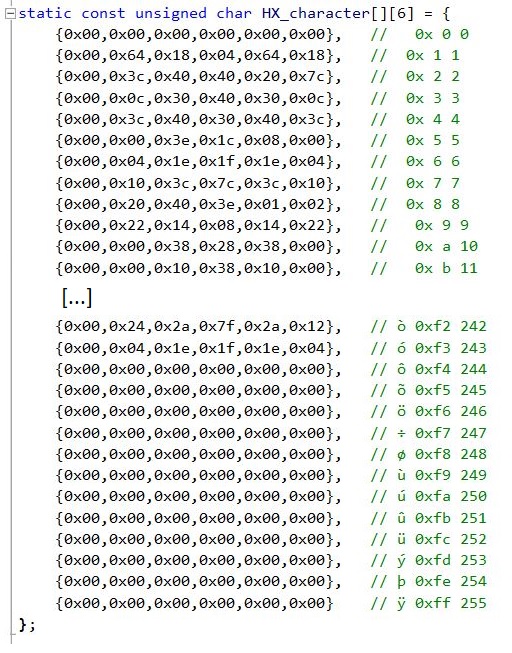


Figura 2.1.2.25

Fiecare caracter este reprezentat grafic pe display utilizând 48 de pixeli (8 rânduri și 6 coloane). Vectorul bidimensional **static const unsigned char HX\_character[ ][6]** conține valorile reprezentării grafice a tuturor caracterelor din tabelul ASCII de bază (256 de caractere).

În partea dreaptă a fiecărui set de valori este scris sub formă de comentariu caracterul descris și codul său ASCII sub forma hexazecimală și decimală.

Funcția **void hx\_write\_string(const unsigned char \*\_characters\_array)** transmite datele și comenzile necesare afișării pe ecran a unui șir de caracter ASCII (string). Șirul de caractere este specificat prin intermediul parametrului **\*\_characters\_array** de tip const unsigned char (pointer).

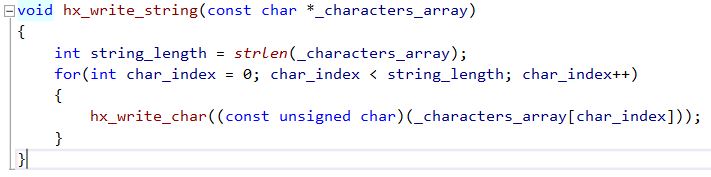


Figura 2.1.2.26

### 2.1.3 Buton meniu

În aplicația prezentata se folosește un buton tactil pentru a putea schimba starea de funcționare a sistemului și implicit setul de informații oferite utilizatorului prin intermediul ecranului.

Butonul este un tip de [comutator](https://ro.wikipedia.org/wiki/Comutator) utilizat pentru controlul unor aparate sau numai a unor funcții ale acestora. Butoanele sunt de diferite forme și dimensiuni și se găsesc în tot felul de dispozitive, deși în principal în echipamentele electrice și electronice.

Acestea sunt în general activate atunci când sunt apăsate și permit trecerea unui fluxul de curent. Când nu mai sunt apăsate, revin în poziția inițială.

Un buton poate fi, prin construcția sa mecanică, de tipul normal deschis - ND sau NO (Normally Open în limba engleză), sau normal închis – NI sau NC (Normally Closed în engleză).

Butonul unui dispozitiv electronic, de obicei, funcționează ca un [întrerupător](https://ro.wikipedia.org/wiki/%C3%8Entrerup%C4%83tor) electric. În interiorul său se află două contacte separate din punct de vedere electric ce se conectează în momentul acționării butonului.

În proiectul implementat se utilizează un buton de tipul **push switch** de pe un shield dedicat Arduino.



Figura 2.1.3.1

Funcționare:

Mecanismul acestuia constă din [butonul propriu-zis](https://ro.wikipedia.org/w/index.php?title=Butonul_propriu-zis&action=edit&redlink=1) acționat prin pulsare, o lamelă conductoare care stabilește contactele celor două terminale apăsând butonul propriu-zis, și un arc care face ca lamela să își recâștige poziția inițială după ce butonul nu mai este acționat.

Shield-ul nRF24L01 Nk 5110:



Figura 2.1.3.2

Din cadrul shield-ului anterior prezentat se utilizează buton albastru din partea dreaptă. Acesta este conectat la tensiunea de alimentare (VCC, 5V) prin intermediul unei rezistențe de pull-up. În momentul în care butonul este acționat (apăsat), terminalul ce realizează conexiunea electrică spre microcontroler se conectează la ground. Astfel, starea implicită a butonului este high.

**Implementare hardware – schematic echivalent**

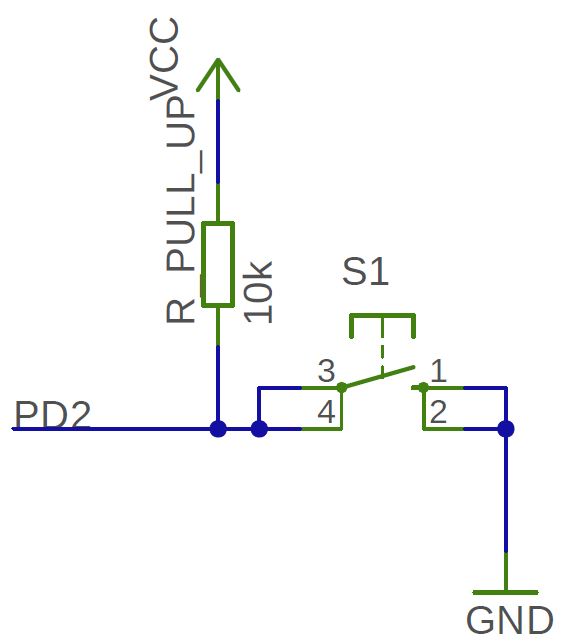
****

Figura 2.1.3.3

**Implementare software – cod sursă**

Codul sursă utilizat pentru interpretarea stării butonului este dezvoltat în fișierele user\_interface.h și user\_interface.c.

Pentru a detecta apăsarea butonului, s-a implementat o rutină de întrerupere ce se declanșează în momentul detecției unui front descrescător pe pinul PD2 (unde este conectat butonul) al microcontrolerului.

Rutina de întrerupere demarează procesele necesare schimbării modului de funcționare al sistemului. Pentru a elimina situațiile de acționare nedorită sau pentru a elimina riscul de a confunda o apăsare mai îndelungată cu apăsări multiple ale butonului, s-a implementat o metodă software de debouncing: se dezactivează temporar întreruperea specifică butonului, se face schimbarea modului de funcționare, se reactivează întreruperea și se așteaptă un interval de 250 milisecunde.

**Configurare microcontroler**

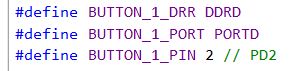
****

Figura 2.1.3.3

Funcția void **init\_next\_state\_button(void)** pregătește configurația hardware a microcontrolerului necesară interpretării stării butonului.

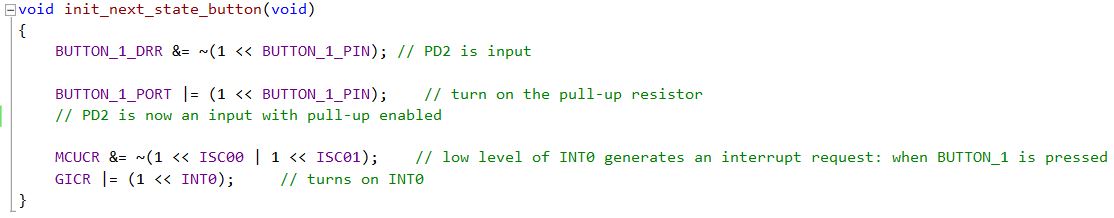


Figura 2.1.3.4

Rutina de întrerupere este definită in vectorul de întreruperi **ISR (INT0\_vect)**, specific pinului PD2, unde este conectat butonul.

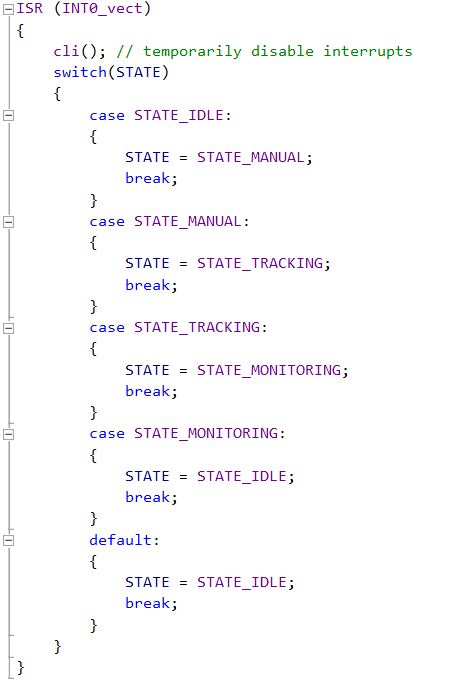


Figura 2.1.3.5

Variabila **STATE** indică modul de funcționare curent al sistemului (state machine). Schimbarea modului de funcționare se face succesiv, definit de ordinea: STATE\_IDLE, STATE\_MANUAL, STATE\_TRACKING, STATE\_MONITORING. Ca măsură de siguranță, în cazul în care printr-o eroare se cere trecerea la o stare care nu a fost definită în prealabil, sistemul va comuta pe starea IDLE.

**Definirea stărilor**

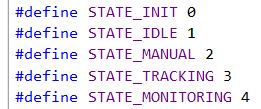


Figura 2.1.3.6

Cod:

|  |
| --- |
| switch(STATE) |
| { |
| case STATE\_IDLE: |
| { |
| STATE = STATE\_MANUAL; |
| break; |
| } |
| case STATE\_MANUAL: |
| { |
| STATE = STATE\_TRACKING; |
| break; |
| } |
| case STATE\_TRACKING: |
| { |
| STATE = STATE\_MONITORING; |
| break; |
| } |
| case STATE\_MONITORING: |
| { |
| STATE = STATE\_IDLE; |
| break; |
| } |
| default: |
| { |
| STATE = STATE\_IDLE; |
| break; |
| } |
| } |

**2.2.4 JoyStick**

JoyStick-ul este folosit pentru a controla panoul cand acesta se afla în modul IDLE.

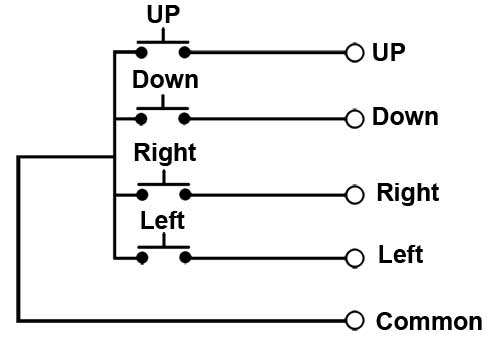
 Un joystick este un periferic al [computerului personal](https://ro.wikipedia.org/wiki/PC) sau un dispozitiv de comandă ce constă într-o manetă care pivotează și transmite apoi unghiul său în două sau trei dimensiuni unui computer.

Figura 2.11

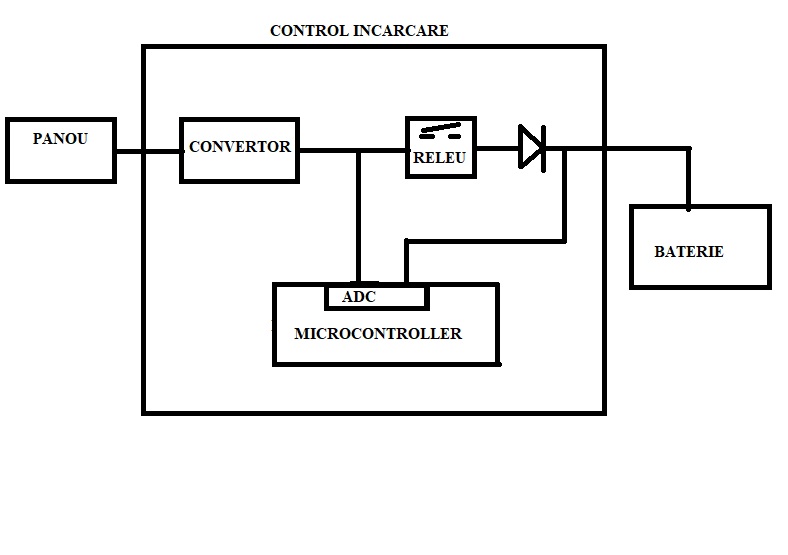
Cod:

|  |
| --- |
| display\_joystick\_data(x, y); |
|  |  | |
| if(y > (JOYSTICK\_IDLE\_VALUE + JOYSTICK\_DEAD\_ZONE)) | |
| { | |
| unipolar\_01\_step\_backward(UNIPOLLAR\_01\_CURRENT\_STEP); | |
| } | |
| else if(y < (JOYSTICK\_IDLE\_VALUE - JOYSTICK\_DEAD\_ZONE)) | |
| { | |
| unipolar\_01\_step\_forward(UNIPOLLAR\_01\_CURRENT\_STEP); | |
| } | |
| else | |
| { | |
| unipolar\_01\_clear\_steps(); | |
| } | |
|  | |
| if(x > (JOYSTICK\_IDLE\_VALUE + JOYSTICK\_DEAD\_ZONE)) | |
| { | |
| l293d\_hb2\_rotate\_right(); | |
| } | |
| else if(x < (JOYSTICK\_IDLE\_VALUE - JOYSTICK\_DEAD\_ZONE)) | |
| { | |
| l293d\_hb2\_rotate\_left(); | |
| } | |
| else | |
| { | |
| l293d\_hb2\_stop(); | |
| } | |
|  | |
| \_delay\_ms(25); | |
| } | |

**2.2 Control incarcare**

**2.2.1 Generalitati**

Schema bloc:

**** Figura 2.12

Schema bloc pentru control incarcare este alcatuita din convertor, releu, dioda iar prin canalele ADC conectam sistemul la microcontroler pentrua obtine tensiunea de la convertor implicit de la baterie.

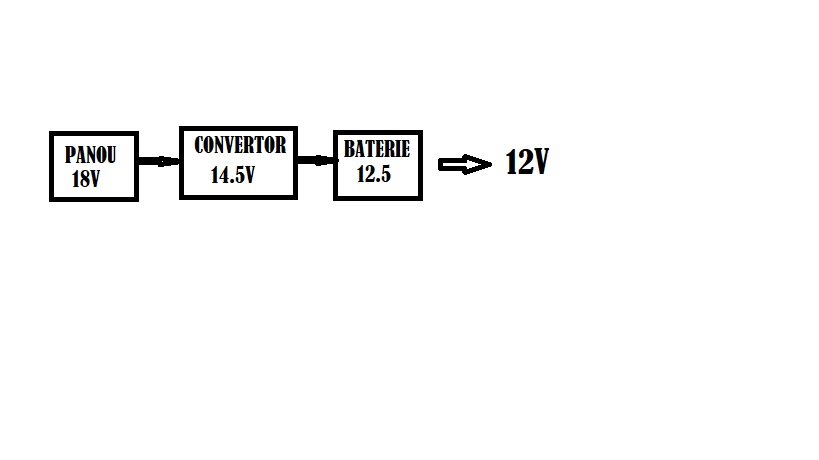


Figura 2.13

**2.2.2 Panou**

Panoul solar fotovoltaic transformă [energia luminoasă](https://ro.wikipedia.org/wiki/Energie_luminoas%C4%83) din razele solare direct în [energie electrică](https://ro.wikipedia.org/wiki/Energie_electric%C4%83) cu ajutorul [celulelor solare](https://ro.wikipedia.org/wiki/Celul%C4%83_solar%C4%83). Energia este stocata în baterii de acumulatori pentru a putea fi folosita atunci cand este necesar. Conversia este statica si nepoluanta, tocmai de aceea acest mod de producere a energiei electrice este unul ecologic.

Panourile solare fotovoltaice sunt produse în diferite dimensiuni avand puteri variate. Cele mai folosite panouri în gama rezidentiala sunt cele de 50 si 75 W, iar pentru centrale fotovoltaice de puteri mari, panouri solare de 220W.

Functionare: atunci cand lumina loveste un atom, acesta este absorbit de catre unul din electronii din jurul atomului, stimuland energia electronului. Pe unele materiale (cum ar fi metale sau siliciul), aceasta energie generata este suficienta pentru a separa electronul de atom, lasand electronul sa se deplaseze liber în structura cristalina a materialului.

**Clasificare:**

1. Panouri solare monocristaline;
2. Panoouri solare policristaline;
3. Panouri solare Thin Film;
4. Panouri solare amorfe;
5. Panouri solare tip tigla;

**Specificatii:**

- tensiune nominala 18V;

- putere maxima 4.5W;

- dimensiuni 170 x 170 mm;

Celulele solare:

Celulele solare sunt de mai multe tipuri: monocristaline, policristaline, amorfe, film subtire, CIS si CdTe, CIGS, etc. Diferenta intre aceste celule consta în structura si modul cum sunt aranjati atomii.

O celulă [solară](https://ro.wikipedia.org/wiki/Soare) este alcătuită din două sau mai multe straturi de material [semiconductor](https://ro.wikipedia.org/wiki/Semiconductor), cel mai întâlnit fiind [siliciul](https://ro.wikipedia.org/wiki/Siliciu).

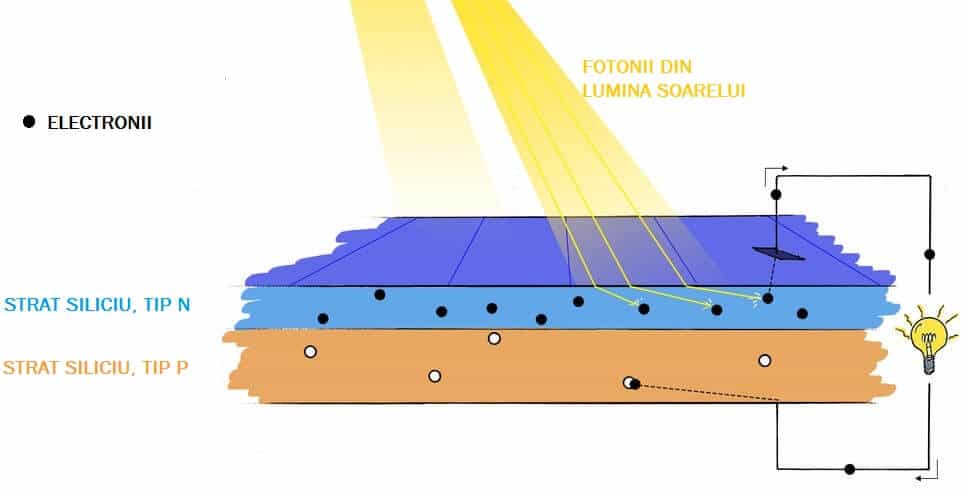


Figura 2.14

Un strat, denumit material de tip N, este contaminat cu o substanta chimica (oamenii de stiinta numesc acest proces: dopaj), avand astfel o multime de electroni în el. Celalalt strat este contaminat cu un alt material care il face capabil sa absoarba mai multi electroni, denumit material de tip P.

**2.2.3 Convertor**

**Generalitati:**

Deoarece panourile solare pot genera o tensiune limitata la iesire, aceasta trebuie adusa la un anumit nivel în functie de aplicatia ceruta. Astfel, tensiunea poate fi ridicata sau coborata la un anumit nivel.

Convertoarele de curent continuu care asigura conversia de c.c în c.c, au la intrare si la iesire, curenti si tensiuni continui, cu valori diferite. Pentru a imbunatati performantele convertoarelor de c.c, urmarim doua obiective:

1. Reducerea dimensiunilor de gabarit.

Pentru realizarea acestui obiectiv trebuie reduse dimensiunile dispozitivelor electronice de putere cu rol de comutator, dar si dimensiunile componentelor cu rol de stocare. Micsorarea dimensiunilor componentelor de stocare este permisa la utilizarea unor frecvente de lucru ridicate. Pe de alta parte, cresterea frecventei de lucru conduce la cresterea pierderilor în comutatie, iar pentru micsorarea lor s-a recurs la utilizarea comutatiilor soft care se realizeaza fie la curent zero, fie la tensiune zero.

1. Cresterea randamentului de conversie;

Realizarea convertoarelor de cc implica utilizarea unui comutator ca si componenta de baza. Acesta trebuie sa semene cat mai mult posibil cu un comutator ideal. Un comutator ideal este caracterizat prin curent nul de blocare, cadere nula de tensiune în comutatie, timpi nuli de comutatie.

**Convertorul coborator Buck**

În cadrul proiectului s-a folsit convertorul coborator buck.

Buck Converter sau Step-Down SMPS este o sursa în comutație de tip coborâtor, unde tensiunea nestabilizată de la intrare este micșorata pentru a produce o tensiune continua stabilizată la ieșire.

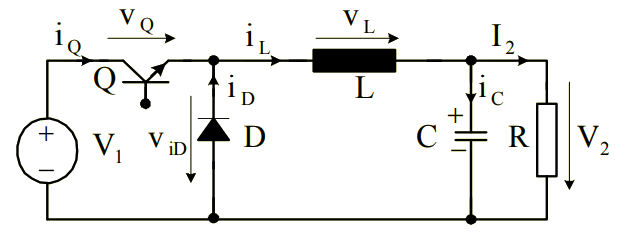


Figura 2.15

În general, o sursa de putere în comutație constă din cinci componente de bază:

1. Un circuit de control cu modulația impulsurilor în durata(PWM Controller);
2. Un tranzistor cu rol de comutator;
3. O inductanță;
4. O capacitate;
5. O dioda;

Circuitul de control cu modulatia impulsurilor în durata este un integrat specializat si are rolul de a furniza semnale de comanda adecvate în scopul reglarii si mentinerii tensiunii de iesire la o valoare constanta.

Tranzistorul, pe post de comutator,are roul de a controla puterea transmisa sarcinii. Acestea pot fi de tip MOS sau bipolare si alegerea lor se face în functie de putere si frecventa.

Inductorul este utilizat cu rol de filtru pentru a reduce riplul de curent. Reducere are loc deoarece curentul prin inductor nu poate fi schimbat instantaneu. Când curentul prin inductor tinde să scadă, inductorul tinde să-l mențină, având rolul de sursă de energie. Inductoarele utilizate în aceste convertoare sunt înfășurate de obicei pe miezuri toroidale, din ferita sau fier așchiat cu pierderi reduse la frecvențe înalte.

Capacitatea este utilizată cu rol de filtru pentru a reduce riplul de tensiune. Aceasta trebuie aleasă cu pierderi minime. Pierderile din capacitate sunt datorate rezistentei serie și inductanței proprii. Tipul capacității este ales după rezistenta serie efectiva(ESR). Cele mai indicate capacități sunt cele din tantal. Uneori, pentru creșterea performanței regulatorului, se leagă în paralel câteva capacități de valoare mai mica pentru a micșora rezistenta serie efectivă.

Dioda folosita este de circulație libera (free-wheeling). Aceasta nu are rol de redresor, ci are funcția de a direcționa corect calea de curent prin inductanță. Este important ca dioda să comute în starea de blocat foarte rapid, de aceea se vor folosi diode rapide de recuperare sau diode schottky, care sunt cele mai indicate.

Tranzistorul se comanda cu frecventa f=1/T, mentinandu-se saturat pe o durata dT si blocat pe o durata (1-d)T, unde d reprezinta factorul de umplere al semnalului de comanda.

Controlul convertorului buck poate fi făcut în două moduri:

1. Funcționarea la frecvență constantă, sau controlul prin modularea impulsurilor în durată(PWM) (cazul lucrarii noastre de laborator),
2. Funcționarea la frecvențe variabile, sau controlul prin modularea în frecvență.

Funcționarea convertorului trebuie analizată în două intervale distincte de timp:

1. intervalul I, în care tranzistorul Q conduce la satura în care tranzistorul Q conduce la saturație iar dioda D , iar dioda D este blocată, fiind polarizată invers. Considerând originea de timp în momentul comutației directe a lui Q, acest prim interval va fi: t ∈ [0, dT ] .

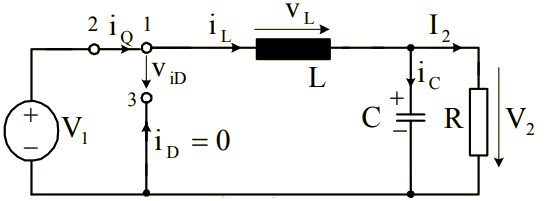
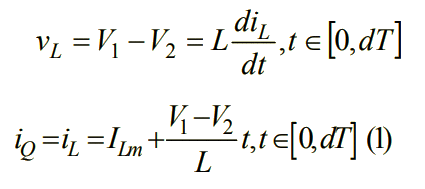


Figura 2.16



b) intervalul II, în care tranzistorul Q este blocat, iar dioda D conduce, asigurând închiderea curentului iL menținut de la inductanța L.

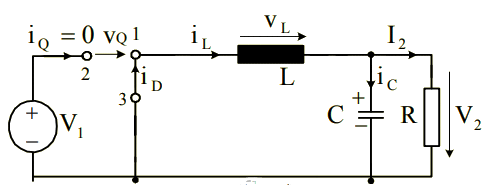
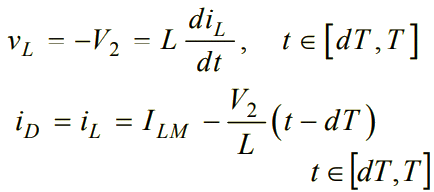


Figura 2.17



Pe baza relațiilor deduse au fost trasate formele de undă. Forma de undă a tensiunii vL ne permite să deducem caracteristica de reglaj a convertorului. Deoarece valoarea medie a tensiunii pe inductanța L este nulă (VLavr = 0 ).

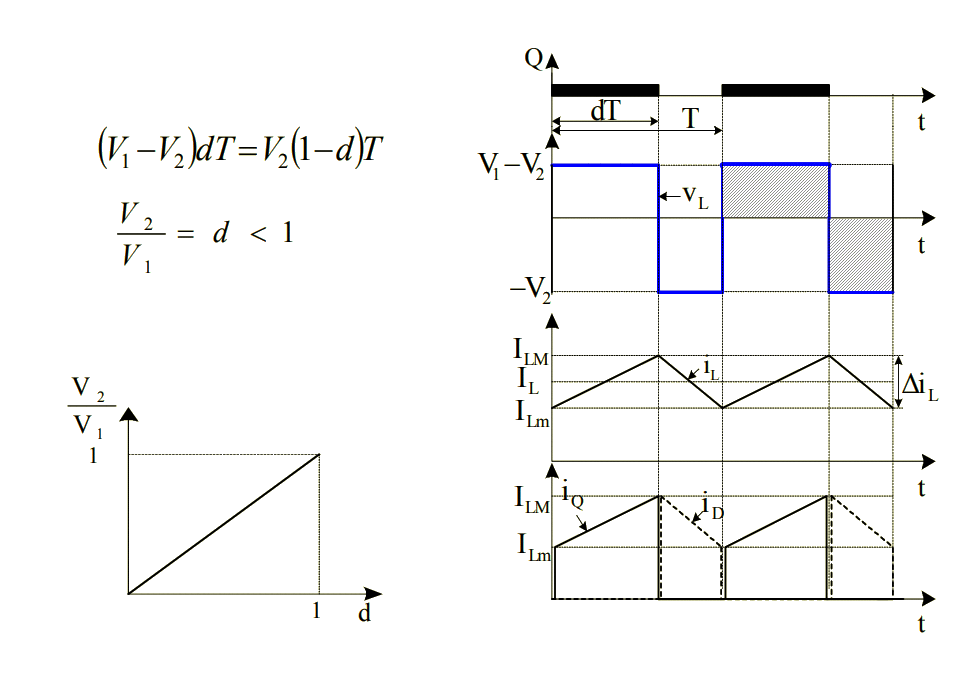


Figura 2.18

Specificatii convertor:

- tensiune de intrare 6-36 V;  
- tensiune de iesire 1.25-32 V;  
- curent de iesire 0.05-5 A;  
- frecventa de operare 180kHz;  
- eficienta maxima 95%;

În figura 2.19 este prezentat schematic control incarcare:

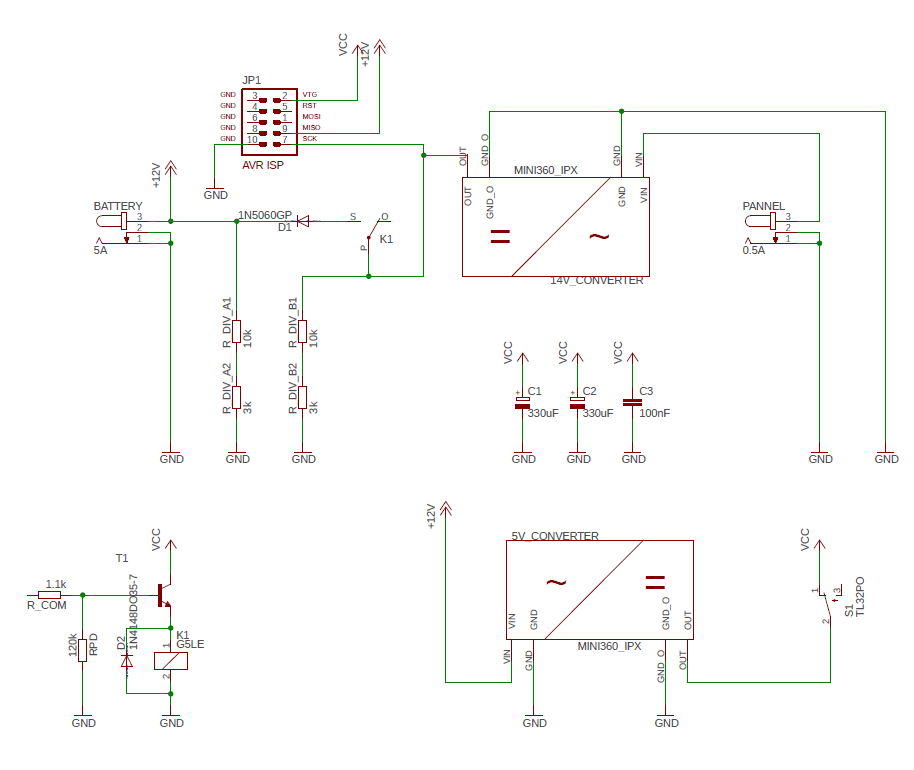


Figura 2.19

* + 1. **Bateria**

#### Bateria electrică este un dispozitiv de stocare a energiei electrice sub formă de energie chimică. Procesul este reversibil, astfel că, la conectarea unui consumator la bornele bateriei, energia chimică se eliberează sub formă de energie electrică. Bateria electrică primară este bateria de unică folosință și nu se poate reîncărca, fiind folosită până la epuizare. Bateria electrică secundară numită și acumulator este reîncărcabilă. Bateria poate avea structura bazată pe una sau mai multe celule. Bateriile mari sunt compuse din pachete de baterii mai mici care sunt conectate în serie pentru a obține o tensiune ridicată sau sunt conectate în paralel pentru a debita curent mare.

Functionare: substanțele chimice din interiorul bateriei produc electroni prin transformarea energiei chimice în energie electrică. Atunci când conectăm o baterie la un gadget, electronii se deplasează de la polul negativ al bateriei, prin dispozitivul pe care îl deservește, închizând circuitul prin polul pozitiv al bateriei.

Diferențele dintre diferitele tipuri de baterii sunt cauzate de substanțe chimice aflate în interiorul acestora, care disting între baterii și le fac bune pentru unele scopuri și mai puțin bune pentru altele.

Tipuri:

1. Baterii alcaline:

- sunt baterii clasice ce contin o cantitate rezonabila de energie .

- acestea se epuizeaza repede.

- nu sunt reincarcabile.

1. Bateriile cu oxid de argint:

- sunt baterii cu o viață lungă, dat fiind că acestea conțin o cantitate mare de energie.

- se gasesc în formă de nasturi, în ceasurile cu cuarț, dar și în submarine.

- contin mercur.

1. Bateriile alcaline reancarcabile:

- dupa epuizare, aceste baterii se pot incarca cu ajutorul incarcatorului .

- își pierd cu fiecare reîncărcare câte un pic din capacitatea de a stoca energie.

1. Bateriile reancarcabile nichel-cadamiu:

- sunt baterii reancarcabile.

- dacă sunt reancarcate înainte de descărcarea completă, se vor descărca din ce în ce mai repede pe viitor.

1. Bateriile nichel-metal hibride:

- inlocuiesc bateriile nichel-cadamiu care ajuta la stocarea în acelasi spatiu cu 40% mai multa energie.

1. Bateriile litiu-ion:

- constituie noul standard în ceea ce privește bateriile pentru gadgeturi, reprezentând un progres considerabil sub aspectul energiei stocate.

În figura 2.20 este prezentat graficul incarcarii unei baterii Pb-Acid în 3 etape:

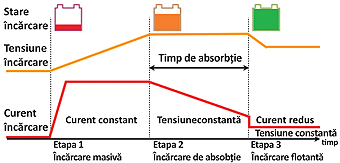


Figura 2.20

În funcție de nivelul tensiunii bateriei, se ajustează nivelul curentului de încărcare. Dacă bateria este descărcată, în Etapa 1 (bulk) se injectează masiv curent constant până se ajunge la încărcarea aprox. 80% și se trece în Etapa 2 când bateria este încărcată spre 99%, iar tensiunea bateriei cuplată la încărcător ajunge până la max. 14,4V. Etapa 3 menține flotant nivelul încărcării aproape de 100%, prin mici injectări de curent, dacă tensiunea scade la bornele bateriei.

Caracteristicile unui acumulator de calitate:

- curba de descarcare sa aiba un platou cat mai mare (sa poata furniza aceeasi tensiune de-a lungul intregii perioade de functionare);

- sa poata fi incarcat de cat mai multe ori;

- durata de incarcare cat mai scurta;

- sa nu se autodescarce în timp (daca nu este folosit);

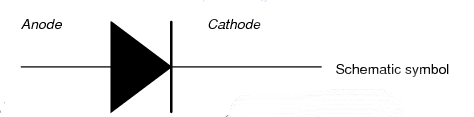
- sa fie cat mai usor;

- sa ocupe cat mai putin volum;

- sa contina cat mai putine substante toxice si sa fie reciclabil;

* + 1. **Dioda**

  Dioda este un dipol neliniar si polarizat (sau asimetric) ce permite trecerea curentului doar intr-o singura directie. Cea mai folosita dioda în circuitele electronice este cea semiconductoare.

 Figura 2.21

Clasificarea diodelor

a. Dupa natura materialului semiconductor folosit sunt:

- diode cu siliciu;

- diode cu germaniu.

b. Dupa tehnologia de fabricatie pot exista în principiu diode:

- cu contact punctiform;

- aliate;

- difuzate;

- epitaxiale.

c. Dupa utilizare se diferentiaza urmatoarele tipuri de diode:

- redresoare (rescurtat DR), utilizate pentru conversia de energie din curent alternativ în curent continuu;

- de comutatie (DC), care realizeaza trecerea rapida de la starea de conductie la cea de blocare;

- de semnal (DS), utilizate în circuite de extragere a informatiilor continute intr-un semnal electric, care variaza în timp, ca de exemplu de detectie si amestec;

- stabilizatoare de tensiune sau diode Zener (DZ), care asigura intre terminalele lor o tensiune constanta, intr-o gama de curenti specificata;

- varicap (DV), denumite si diode cu capacitate variabila, la care capacitatea variaza cu tensiunea aplicata;

- traductoare, care cuprind: fotodiode (F), diode electroluminiscente (LED);

- speciale, incluzand diode tunel, Schottkey, Gunn.

d. Dupa putere, diodele se pot clasifica în diode:

- de putere mica, pentru curenti medii redresati mai mici de 3A;

- de putere medie, pentru curenti cuprinsi intre 3A si 30A;

- de putere, pentru curenti cuprinsi intre 30A si 200A;

- de mare putere, pentru curenti peste 200A.

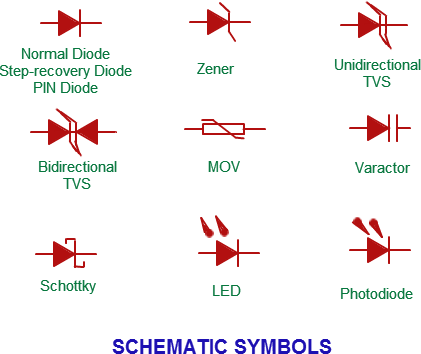


Figura 2.22

Dioda poate fi conectată într-un circuit în două moduri:

* polarizare directă (în care dacă legăm plusul la anod și minusul la catod, dioda conduce curentul electric);
* polarizare inversă (în care dacă legăm plusul la catod și minusul la anod, dioda nu conduce curentul electric).

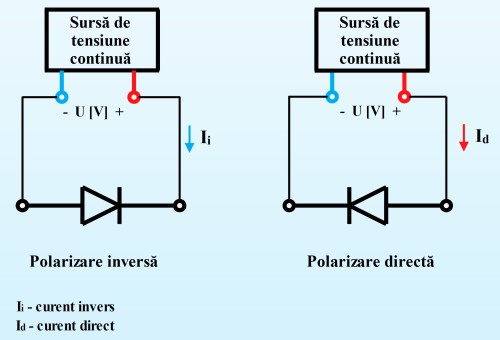


Figura 2.23

Mod de functionare:

Așa cum se vede în figura urmatoare, dioda lasă să treacă un curent direct (Id) atunci când este polarizată direct și un curent invers minuscul (Ii) atunci când este polarizată invers.

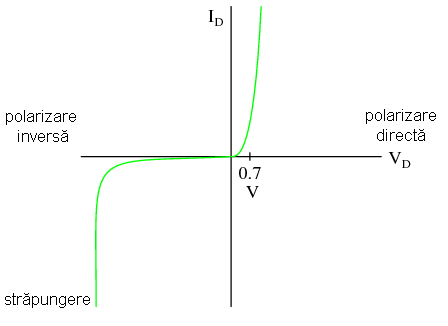


Figura 2.24

* + 1. **Releul**

Releul este un aparat care, sub actiunea marimii de intrare, realizeaza o variatie în salt a marimii de iesire, în scopul comenzii altor elemente.În acest proiect am folosit un releu electromecanic, cu rolul de a conecta/deconecta convertorul de baterie.

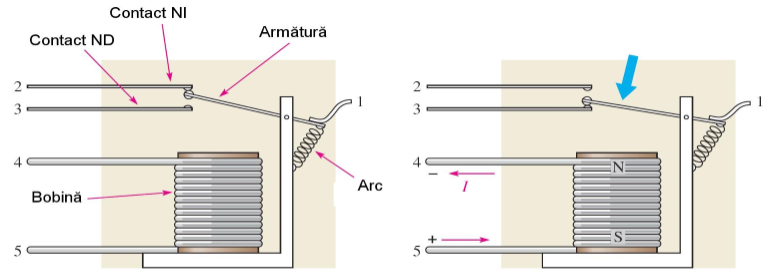
 Releele electromagnetice au ca organ principal de actionare un electromagnet cu armătură mobilă (ce poate fi basculantă, rotitoare sau cu mișcare de translație). Aceasta acționează asupra contactelor electrice, modificând circuitul electric în care releul este introdus.

Figura 2.25

Releu neenergizat – contact electric între terminalele 1 și 2;

Releu energizat – contact electric între terminalele 1 și 3;

Contactele unui releu electromecanic pot fi contacte normal deschise (ND) sau contacte normal închise (NI). Contactele sunt reținute în aceste poziții de arcuri sau uneori de forța gravitațională.

Brațul mobil, pârghia unui releu, se numește armătură. Armătura este realizată din material magnetic și este acționată de electromagnetul format de bobină. Prin acțiunea asupra armăturii se deplasează mecanic contactele releului, care se închid (ND) respectiv se deschid (NI) atunci când releul este alimentat. Se mai spune că releul este energizat.

Tipuri de relee:

* Relee electromecanice;
* Relee electromagnetice;
* Relee termice;
* Relee „Reed”;

## **Microcontroller-ul**

**3.1 Generalitati:**

**Caracteristici generale:**

Un microcontroller este un tip de circuit care integreaza un microprocesor si alte dispozitive periferice intr-un singur chip punandu-se accent pe un cost redus de productie si consum redus de energie electrica. Principala diferenta dintre un microcontroller (µC) si un microprocesor (µP) o constituie faptul ca un µC integreaza memoria de program, memoria de date si alte interfete de intrare-iesire sau periferice.

Cele mai intalnite structuri din circuitul integrat al unui µC sunt urmatoarele:

• Unitatea centrala de procesare (µP core) cu o arhitectura care poate fi pe 8, 16, 32 sau 64 de biti.

• Memorie de date volatila (RAM) sau nevolatila pentru date sau program (Flash sau EEPROM).

• Porturi digitale de intrare-iesire.

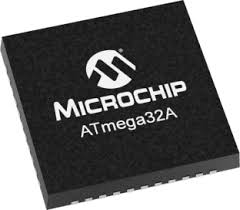
• Interfete seriale (RS232, SPI, I2C, CAN, RS485).

• Timere, generatoare de PWM sau watchdog.

• Convertoare analog-digitale.

• Suport pentru programare si debugging (ISP).

**µC Atmega32A:**



Microcontrolerul AVR are la bază un procesor RISC cu o arhitectură Harvard (adică unitatea centrală de procesare are memorie de program și memorie de date separate). Pe baza acestui nucleu RISC firma Atmel a dezvoltat mai multe familii de microcontrolere, cu diferite structuri de memorie și de interfețe I/O, destinate diferitelor clase de aplicații. Aceste microcontrolere sunt destinate aplicațiilor simple cum ar fi: controlul motoarelor, controlul fluxului de informație pe portul USB, controlul accesului de la distanță (Remote Access Control).

Figura 3.1

*Specificatii:*

• Arhitectura RISC avansată

- 131 Instrucțiuni ;

- 32 × 8 registre de lucru cu scop general;

- Operație complet statică;

- Până la 16MIPS la 16MHz;

• Segmente de memorie nevolatile de înaltă rezistență

- 32 Kbytes de memorie de program Flash auto-programabilă în sistem;

- 1024 biți EEPROM;

- 2Kbytes SRAM intern;

- Reținerea datelor: 20 de ani la 85°C / 100 de ani la 25°C;

• Operație True Read-While-Write

- Blocarea programării pentru securitatea software-ului;

• Interfață JTAG

- Programare Flash, EEPROM, siguranțe și biti de blocare prin interfața JTAG;

• Caracteristici periferice

- Două timere / contoare cu 8 biți;

- Un timer / contor de 16 biți cu Prescaler separat;

- Contor de timp real cu Oscilator separat;

- Patru canale PWM;

- ADC pe 8 canale și 10 biți;

• 8 canale cu un singur capăt

• 7 canale diferențiale numai în pachetul TQFP

• 2 canale diferențiale cu câștig programabil la 1x, 10x sau 200x

- interfață serială orientată pe două fire;

- Serial USART programabil;

- Interfață serială Master / Slave SPI;

- Timer programabil de supraveghere cu Oscilator separat pe cip;

- comparator analogic pe cip;

• Caracteristici speciale de microcontroler

- Resetare la pornire și detectare programabilă de marcare;

- Oscilatorul RC calibrat intern;

- Surse externe și interne de întrerupere;

- Șase moduri de repaus: inactiv, reducerea zgomotului ADC, economie de energie, pornire,

oprire și extindere;

• I / O și pachete

- 32 linii I / O programabile;

- PDIP cu 40 de pini, TQFP cu 44 de conductori și QFN / MLF cu 44 de cadre;

• Tensiuni de operare

- 2.7V - 5.5V;

• Graduri de viteză

- 0 - 16MHz;

• Consum de putere la 1MHz, 3V, 25°C

- Activ: 0.6mA;

- Mod inactiv: 0,2 mA;

- Mod de pornire: <1μA;

*Schematic microcontroler:*

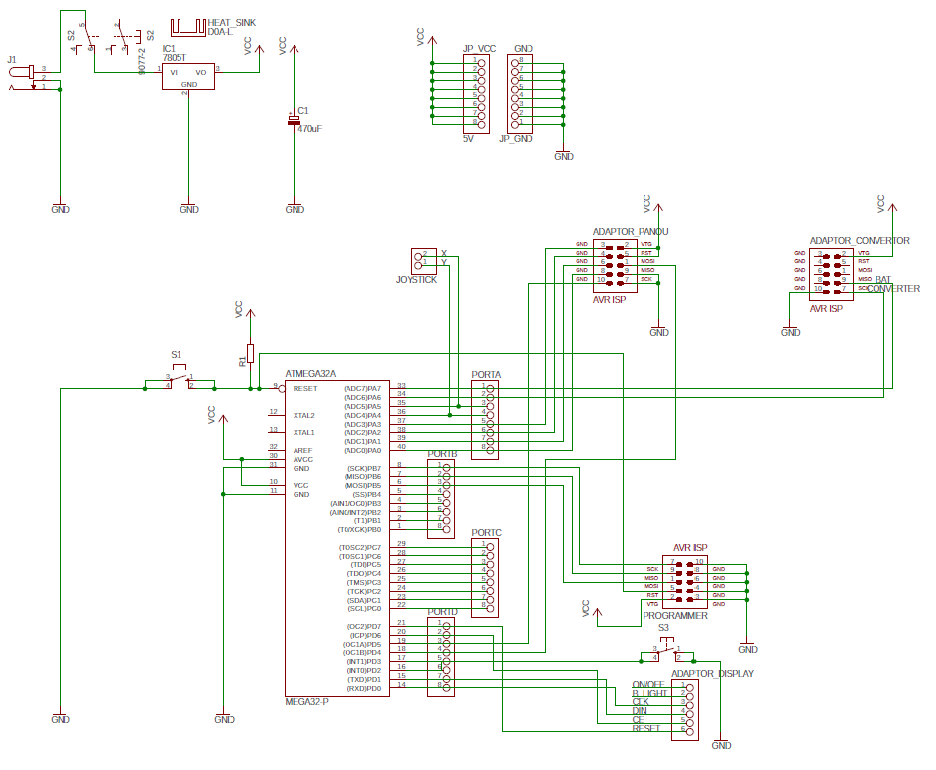
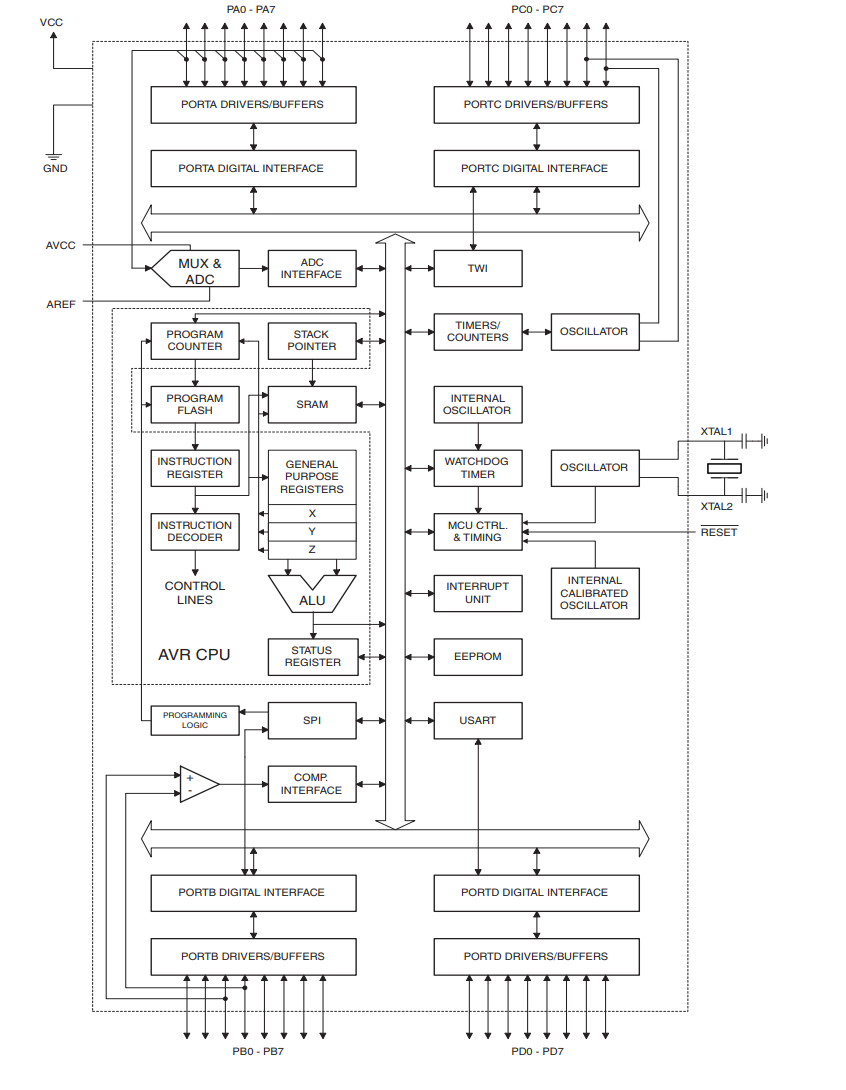
**

Figura 3.2

*Schema bloc microcontroler:*

**Figura 3.3

*Descriere pini:*

VCC: Tensiune de alimentare digitală;

GND: Masa;

Port A: Acest port servește ca intrări analogice la convertorul A / D. De asemenea, servește ca port I / O bidirecțional pe 8 biți dacă convertorul A / D nu este utilizat.

Port B: Este un port I / O bidirecțional pe 8 biți. Bufferele sale de ieșire au caracteristici simetrice de acționare, atât cu capacitate ridicată, cât și cu sursa.

Port C: Este un port I / O bidirecțional pe 8 biți. Dacă interfața JTAG este activată, rezistențele pull-up de la pinii PC5 (TDI), PC3 (TMS) și PC2 (TCK) vor fi activate.

În figura este reprezentata interfața JTAG folosind portul C:

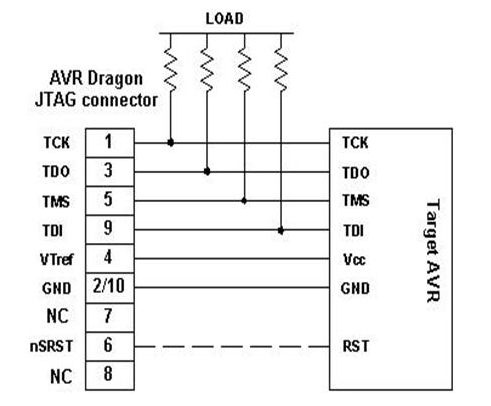


Figura 3.4

Reset: Resetare intrare.

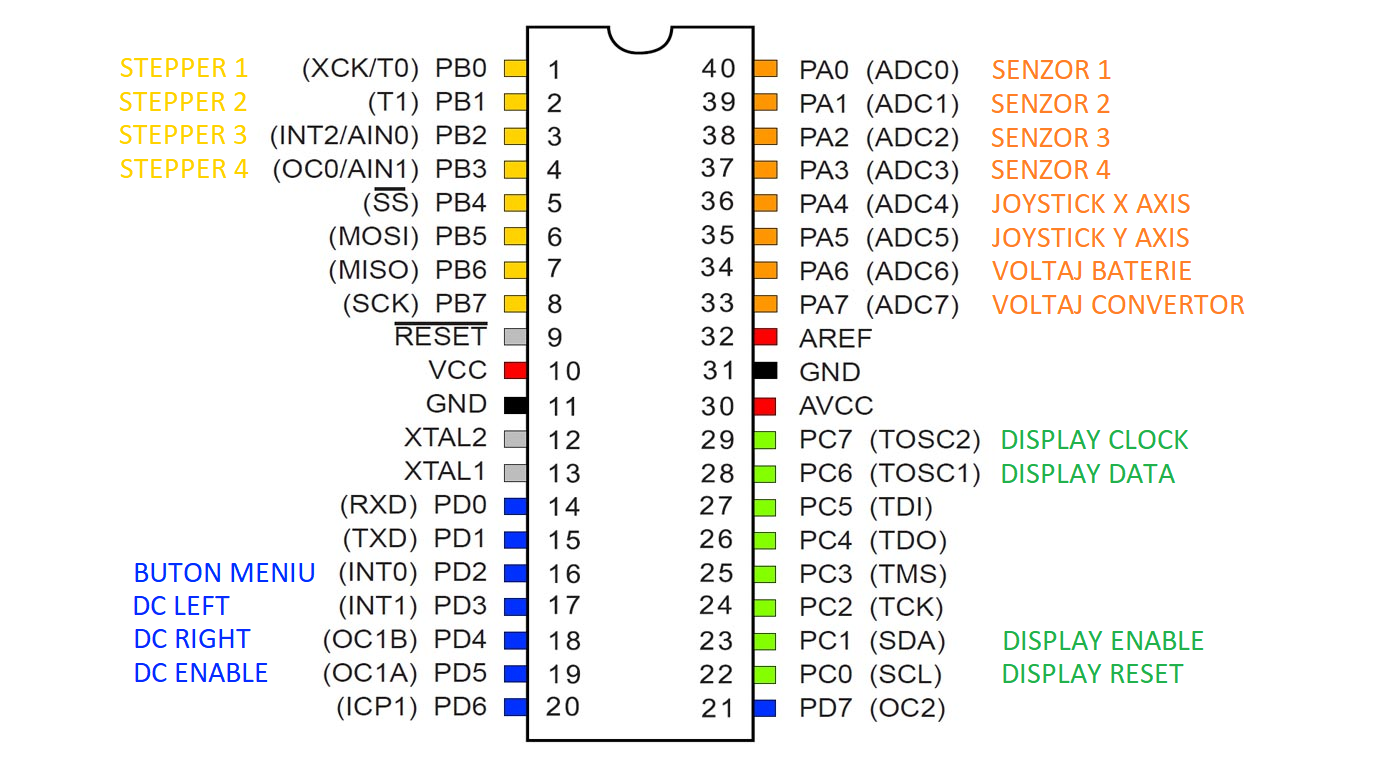
XTAL1: Este o intrare a amplificatorului de oscilator inversat și intrare în circuitul de funcționare a ceasului intern.

XTAL2: Este o ieșire de la amplificatorul oscilator inversor.

AVCC: Acesta este pinul de tensiune de alimentare pentru Port A și convertorul A / D. Ar trebui să fie conectat la VCC.

AREF: AREF este pinul analogic de referință pentru convertorul A / D.

*Functii pini:*



Pentru a converti informatii precum tensiunea bateriei, tensiunea convertorului si coordonatele joystick-ului s-au folosit pini de pe Port A, si anume pinii ADC4, ADC5, ADC6, ADC7.

* 1. **ADC**

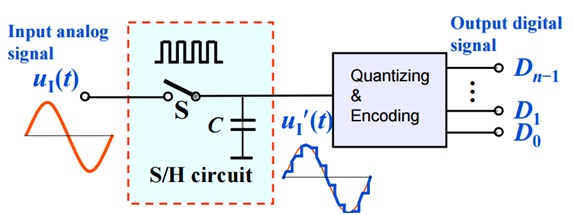
Convertorul analogic la digital (ADC) este un circuit electronic integrat folosit

pentru a converti semnalele analogice, cum ar fi tensiunile în formă digitală sau binară. Majoritatea ADC-urilor iau o tensiune de intrare de la 0 la 10V, -5V până la + 5V , etc și produce în mod corespunzător ieșire digitală ca un fel de număr binar.

În principal, există doi pași pentru conversia analog-digital:

S / H: Eșantionare și deținere

Q / E: cuantificarea și codificarea



S / H: Eșantionare și deținere:

Un semnal analog se schimbă continuu în timp, pentru a măsura semnalul pe care trebuie să-l păstrăm constant pentru o scurtă durată, astfel încât să poată fi eșantionată. Am putea măsura semnalul în mod repetat și foarte rapid și apoi am găsi scala de timp corectă sau am putea măsura semnalul la diferite sincronizări. Aceasta se face printr-un circuit de eșantionare și de așteptare.

Q / E: cuantificarea și codificarea:

Pe ieșirea (S / H), există un anumit nivel de tensiune. Noi îi atribuim o valoare numerică. Se caută cea mai apropiată valoare, în corespondență cu amplitudinea eșantionării și a semnalului de reținere. După identificarea celei mai apropiate valori, îi este atribuită o valoare numerică și este codificată sub forma unui număr binar. Numerele binare codificate generate sunt reprezentate de biți "n".

**4.Software**

**4.1 Generalități**

Codul sursă pentru această aplicație a fost scris în mediul de dezvoltare Atmel Studio 7, oferit de Microchip, utilizând ca limbaj de programare C/C++. Ca și complexitate, partea logică a aplicației se încadrează la un nivel mediu. Astfel a fost de preferat ca fiecare funcționalitate să fie implementată într-un grup de fișiere propriu; fiecare grup este constituit dintr-un fișier .c ce conține corpul funcțiilor dezvoltate și un fișier .h, (header) care cuprinde prototipurile funcțiilor definite în fișierul anterior menționat. Funcțiile implementate vor putea fi accesate în cod utilizând directiva de preprocesare **#include**.

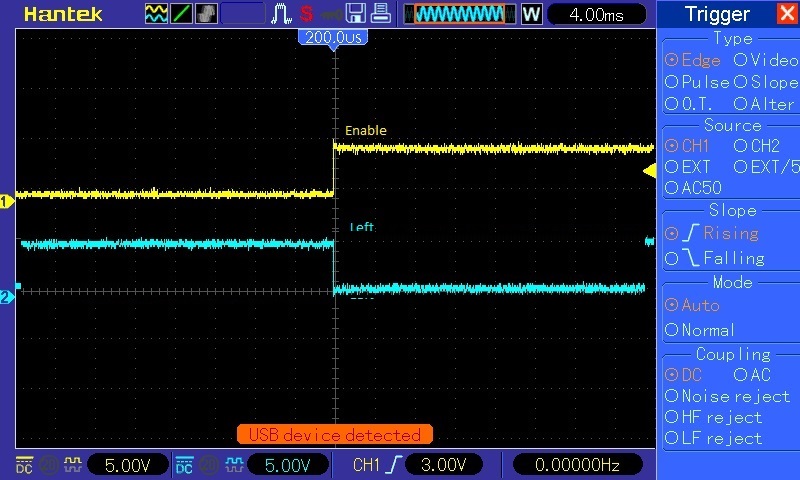
Lista modulelor software implementate:

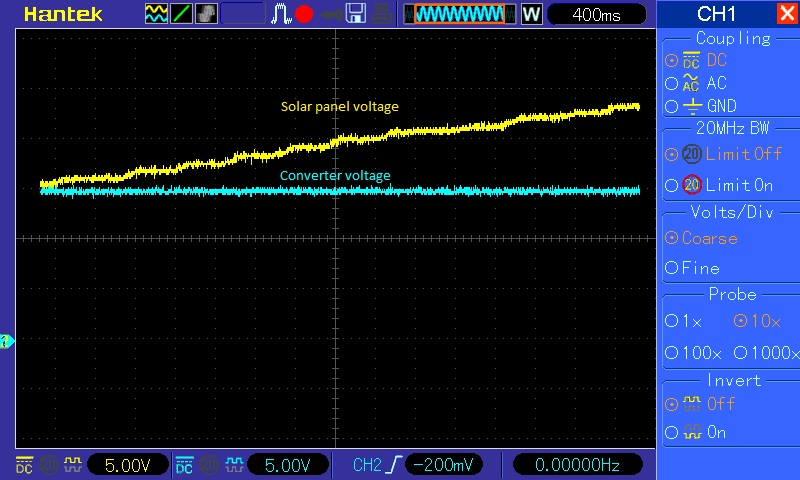


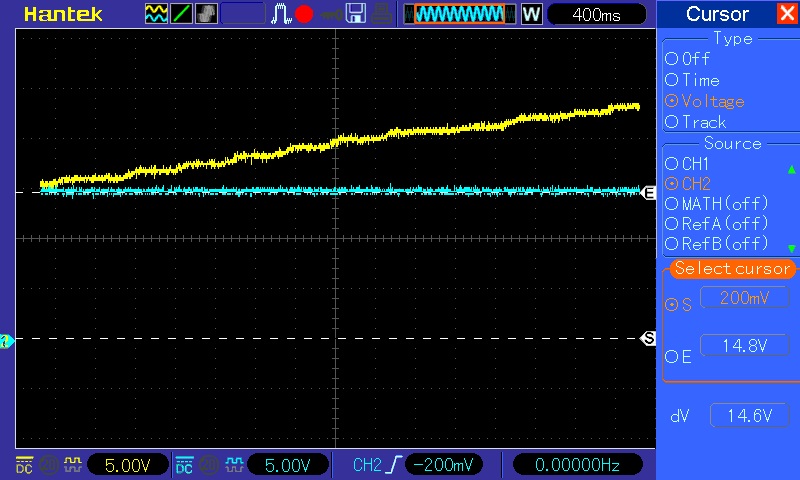
**5.Analize si rezultate**

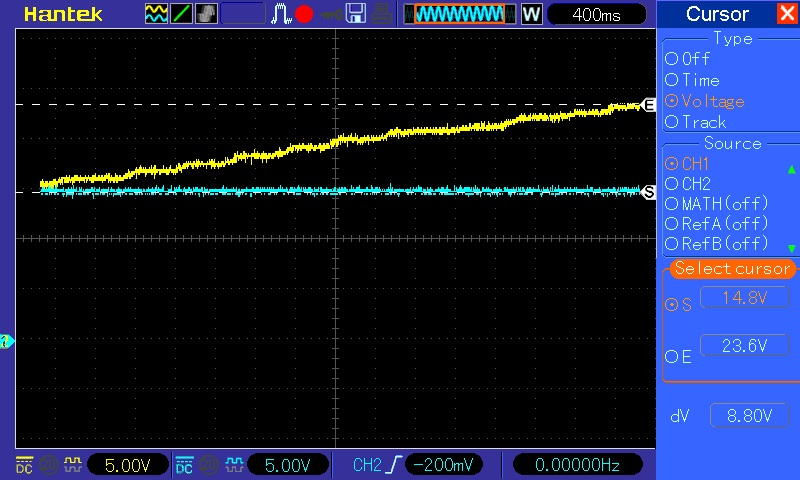
**5.1 Rapoarte aplicație**

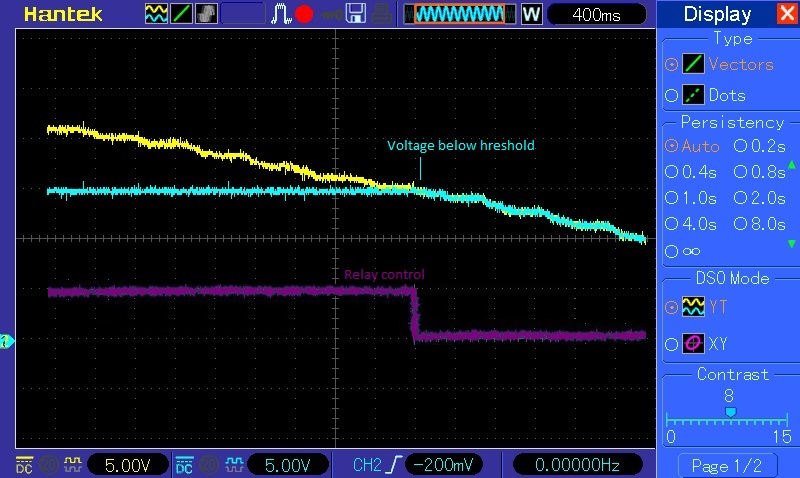
Convertor:



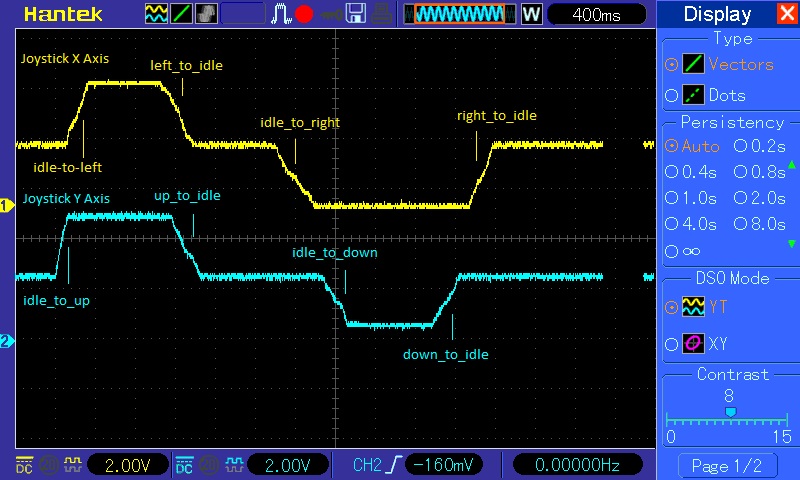
****

****

****

****

JoyStick:



**6.Concluzie:**

**Bibliografie**

<https://ro.wikipedia.org/wiki/Interfa_grafica>

<http://labs.cs.upt.ro/labs/so/html/so1.html>

[http://extremeelectronics.co.în/avr-tutorials/gui-software-for-usbasp-based-usb-avr-programmers/](http://extremeelectronics.co.in/avr-tutorials/gui-software-for-usbasp-based-usb-avr-programmers/)

<https://en.wikipedia.org/wiki/Display_driver>

<http://www.rasfoiesc.com/inginerie/electronica/SIGURANTE-FUZIBILE-Principiul-43.php>

<https://ro.wikipedia.org/wiki/Interfa%C8%9B%C4%83>

<https://ro.wikipedia.org/wiki/Afi%C8%99aj_cu_cristale_lichide>

<https://components101.com/nokia-5110-lcd>

<https://ro.wikipedia.org/wiki/Interfa%C8%9Ba_serial%C4%83_SPI>

<https://ro.wikipedia.org/wiki/Buton>

<http://alternativepureenergy.ro/despre/panouri-fotovoltaice/>

<http://energie-verde.ro/produse/panouri-fotovoltaice-2/>

<http://solarcenter.ro/blog/cum-functioneaza-panourile-solare-fotovoltaice/>

<http://www.esolar.ro/sfaturi-utile/toate-tipurile-de-panouri-solare-avantajele-si-dezavantajele-lor.html>

<http://ep.etc.tuiasi.ro/site/Electronica%20Industriala/referate%20laborator/L09%20-%20Convertor%20buck%202007.pdf>

<http://www.scientia.ro/tehnologie/gadgeturi/1091-scurt-ghid-despre-baterii.html>

<http://electronica-azi.ro/2013/12/11/bateriile-electrice-reincarcabile-pb-acid/>

<https://biblioteca.regielive.ro/referate/electronica/dioda-electrotehnica-fizica-220222.html>

<https://hobbytronica.ro/dioda/>

<http://www.ac.tuiasi.ro/~lmastacan/wp-content/uploads/L3-Relee-si-contactoare.pdf>

<http://microcontrollerslab.com/analog-to-digital-adc-converter-working/>

<https://www.elprocus.com/analog-to-digital-adc-converter/>

<https://www.elprocus.com/types-of-avr-microcontroller-atmega32-and-atmega8/>

<https://www.mouser.com/ds/2/268/atmel_atmel-8155-8-bit-microcontroller-avr-atmega3-1180575.pdf>

<http://andrei.clubcisco.ro/cursuri/3pm/lab1.pdf>

https://products.office.com/ro-ro/visio/flowchart-software