

TREBALL DE FI DE GRAU

Grau en Enginyeria Electrònica Industrial i Automàtica

**SISTEMA DE GESTIÓ DOMÒTICA PER A OPTIMITZAR
EL CONSUM ENERGÈTIC D'UN HABITATGE**



Volum IV

Annex

Autor: Hernández Fuster, Carles

Director: Manzanares Brotons, Manuel

Convocatòria: Maig 2018

Departament: Electrònica Industrial i Automàtica

Índex

1. Codi del programa.....	3
---------------------------	---

1. Codi del programa

```
#include <18F4520.h>                //Xip utilitzat
#device adc=10
#fuses XT,PUT,NOPROTECT,BROWNOUT,NOLVP,NOWDT
#use delay (clock=4M)              //Relotge de 4 MHz
#use i2c(master,sda=PIN_C4,scl=PIN_C3) //Configuració Master

#include <lcd.c>                    //Llibrebria LCD

#define EEPROM_ADDRESS long int     //Configuració Memòria EEPROM

#define RTC_SDA PIN_C4              //Configuració rellotge RTC
#define RTC_SCL PIN_C3
#include <ds1307.c>                 //Llibreria rellortge RTC

#use standard_io(A)                //Inicialització dels ports
#use fast_io(B)
#use standard_io(C)
#use standard_io(D)

#byte TRISA = 0xF92                //Configuració direccions memòria RAM
#byte PORTA = 0xF80
#byte TRISB = 0xF93
#byte PORTB = 0xF81
#byte TRISC = 0xF94
#byte PORTC = 0xF82
#byte TRISD = 0xF95
#byte PORTD = 0xF83

//-----
//-----

menu();                            //Menu principal

long menu_light(long item);        //Menu llum

long menu_blind(long item);        //Menu persiana
```

```
long menu_temperature(long item);           //Menu temperatura

long menu_consume(long item);              //Menu consum

long menu_savingplan(long item);           //Menu màquines

long menu_time(long item);                 //Menu endolls

float calcul_temperature();                 //Funció calcul temperatura (termometre)

acondicionament(float temp);               //Climatizador

open_window();                             //Funció finestra oberta

switch_state();                             //Funció estat dels interruptors de la llum de les
habitacions

presence_state();                          //Funció estat de presència de les habitacions

blind_control(float temp);                 //Funció control persiana

float calcul_light_bright();               //Funció lluminositat

float calcul_voltage();                    //Funció calcul voltatge

float calcul_current();                    //Funció calcul intensitat

float calcul_power(float voltage, float current); //Funció calcul potencia

overpower(float PW);                       //Funció excés de potencia

long int consume(byte sec);                //Funció registre consum

clock();                                   //Funció rellotge

saving_plan (byte sec);                    //Funció estalvi maquines
```

```
write_eeprom(long int address, int power_data);    //Funció escriptura EEPROM

int read_eeprom(long int address);                //Funció lectura EEPROM

int daily_consume(int cons1, cons2, cons3, cons4);

//-----
//-----

long int address;        //Direcció EEPROM on guardar
int cons1;               //Valor a guardar a EEPROM
int cons2;               //Valor a les 5h
int cons3;
int cons4;
int power_data;
int cons_total;         //Mitja diària

int i2c;                 //Dada a transmetre per i2c

char DR = "X";          //Llum Dining Room indica 'X' tancat 'O' obert
char WC = "X";          //Llum WC
char BR = "X";          //Llum Bedroom
char conm_DR= "X";      //Estat commutador
char conm_WC= "X";
char conm_BR= "X";

int auto_blind = "O";   //Persiana control automàtic

int FW = "X";           //Ventilador (Fan)
int HW = "X";           //Calefacció (Heater)
int auto_temp = "O";    //Estat automàtic temperatura

float temp;             //Temperatura
float light;            //Lluminositat
float voltage;          //Voltatge
float current;          //Intensitat
float power;            //Potència
```

```

float PW;           //Potencia (Voltatge x Intensitat)

char PL = "O";     //Endoll (Plug)

byte sec;          //Valors RTC
byte min;
byte hrs;
byte day;
byte month;
byte yr;
byte dow;
char sdow[11];

char save = "O";   //Pla d'estalvi energia

//-----

void envio_i2c()   //Funció enviament per i2c
{
    i2c_start();   //Inicialització de la transmissió per i2c
    i2c_write(0xB0); //Direcció del PIC esclau
    i2c_write(i2c); //Dada a enviar
    i2c_stop();    //Finalització de la transmissió
    delay_ms(200);
}

//-----

#int_ext           //Interrupció canvi d'estat RB0
int_open_window() //Interrupció finestra oberta
{

    while ((input(pin_B0)==1)&&(input(pin_B4)==1)) //La finestra esta oberta (pin_B0
    obert) i mode automàtic encès (pin_B4 tancat)
    {
        if (FW=='O' || HW=='O') //Ventilador o la calefacció estan encesos

```

```
{
  lcd_putc('\f');           //Borra pantalla
  lcd_gotoxy(1,1);         //Escriu segona linia
  printf(lcd_putc,"OPEN WINDOW"); //Missatge
  i2c=6;                   //Envia paràmetre per i2c
  envio_i2c();
}
}
FW='X';                   //Ventilador apagat
HW='X';                   //Calefacció apagat
lcd_putc('\f');
}
```

```
//-----
-----
```

```
menu()                   //Menu principal
{
  long item;              //Selecció moviment menú
  long n_menus=5;         //Número de funcions dins del menú

  item=1;

  while(1)
  {
    power_data=(int)power; //Agafa part entera
    cons_total=daily_consume(cons1, cons2, cons3, cons4);
    light = calcul_light_bright();
    blind_control(light);

    address = consume(sec);
    write_eeprom(address, power_data);
    if (sec==6)
    {
      cons1 = read_eeprom(address);
    }
    if (sec==11)
    {
```

```
    cons2 = read_eeprom(address);
}
if (sec==16)
{
    cons3 = read_eeprom(address);
}
if (sec==21)
{
    cons4 = read_eeprom(address);
}

switch_state();
presence_state();
temp = calcul_temperature();
acondicionament(temp);
saving_plan(sec);
voltage = calcul_voltage();
current = calcul_current();
power = calcul_power(voltage,current);
overpower(PW);

if (input(pin_A4)==1)           //Pulsa tecla MOV
{
    item ++;
    delay_ms(200);
    lcd_putc('\f');

    if (item>n_menus)
    {
        item=1;
    }
}

switch(item)
{
    case 1:                       //Menu llum
        lcd_gotoxy(1,1);
        printf(lcd_putc,"1.LIGHT");
```



```
    lcd_gotoxy(1,2);
    clock();
    break;
case 2:                //Menu persiana
    lcd_gotoxy(1,1);
    printf(lcd_putc,"2.BLIND");
    lcd_gotoxy(1,2);
    clock();
    break;
case 3:                //Menu temperatura
    lcd_gotoxy(1,1);
    printf(lcd_putc,"3.TEMPERATURE");
    lcd_gotoxy(1,2);
    clock();
    break;
case 4:                //Menu consum energètic
    lcd_gotoxy(1,1);
    printf(lcd_putc,"4.CONSUME");
    lcd_gotoxy(1,2);
    clock();
    break;
case 5:                //Menu màquines
    lcd_gotoxy(1,1);
    printf(lcd_putc,"5.SAVING PLAN");
    lcd_gotoxy(1,2);
    clock();
    break;

case 100:
    item = menu_light(item);
    break;
case 200:
    item = menu_blind(item);
    break;
case 300:
    item = menu_temperature(item);
    break;
case 400:
```

```
        item = menu_consume(item);
        break;
    case 500:
        item = menu_savingplan(item);
        break;
    }

    if(input(pin_A5)==1)          //Pulsa tecla OK
    {
        delay_ms(200);
        lcd_putc('\f');
        item=item*100;
    }
}
}
```

```
//-----
```

```
long menu_light(long item)
{
    long n_menus=6;

    lcd_putc('\f');

    while (1)
    {
        switch_state();
        presence_state();

        if (input(pin_A4)==1)
        {
            item ++;
            delay_ms(200);
            lcd_putc('\f');

            if (item>=(n_menus+100))
            {
```

```
        item=100;
    }
}

switch(item)
{
    case 100:                                     //Mostra estat dels interruptors de cada
habitació
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Status");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"BR:%c ",BR);
        lcd_gotoxy(7,2);
        printf(lcd_putc,"WC:%c ",WC);
        lcd_gotoxy(13,2);
        printf(lcd_putc,"DR:%c ",DR);
        break;
    case 101:                                     //Apaga llum dormitori
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Bedroom");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"BR: %c",BR);
        break;
    case 102:                                     //Apaga llum WC
        lcd_gotoxy(1,1);
        printf(lcd_putc,"WC");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"WC: %c",WC);
        break;
    case 103:                                     //Apaga llum menjador
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Dining R.");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"DR: %c",DR);
        break;
    case 104:                                     //Apaga totes les llums
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Lights off");
```

```
    lcd_gotoxy(1,2);
    printf(lcd_putc,"BR:%c ",BR);
    lcd_gotoxy(7,2);
    printf(lcd_putc,"WC:%c ",WC);
    lcd_gotoxy(13,2);
    printf(lcd_putc,"DR:%c ",DR);
    break;
case 105:                                //Torna al menu principal
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Go Back");
    break;
}

if(input(pin_A5)==1)
{
    delay_ms(200);

    switch(item)
    {
    case 101:
        if (BR=='O')                    //Llum dormitori esta oberta
        {
            lcd_putc('\f');
            lcd_gotoxy(1,1);
            printf(lcd_putc,"Bedroom");
            lcd_gotoxy(1,2);
            printf(lcd_putc,"BR: %c",BR);
            i2c=0;
            envio_i2c();
            BR='X';                       //Canvi estat a apagat
        }
        else                               //Llum dormitori esta apagada
        {
            lcd_gotoxy(9,1);
            printf(lcd_putc,"is close");
        }
        break;
    case 102:
```

```
if (WC=='O') //Llum WC esta oberta
{
  lcd_putc('\f');
  lcd_gotoxy(1,1);
  printf(lcd_putc,"WC");
  lcd_gotoxy(1,2);
  printf(lcd_putc,"WC: %c",WC);
  i2c=1;
  envio_i2c();
  WC='X'; //Canvi estat a apagat
}
else //Llum WCi esta apagada
{
  lcd_gotoxy(4,1);
  printf(lcd_putc,"is close");
}
break;
case 103:
if (DR=='O') //Llum menjador esta oberta
{
  lcd_putc('\f');
  lcd_gotoxy(1,1);
  printf(lcd_putc,"Dining R.");
  lcd_gotoxy(1,2);
  printf(lcd_putc,"DR: %c",DR);
  i2c=2;
  envio_i2c();
  DR='X'; //Canvi estat a apagat
}
else //Llum menjador esta apagada
{
  lcd_gotoxy(10,1);
  printf(lcd_putc,"is close");
}
break;
case 104:
if (BR=='O') //Llum dormitori esta oberta
{
```

```
        lcd_gotoxy(1,2);
        printf(lcd_putc,"BR:%c ",BR);
        i2c=0;
        envio_i2c();
        BR="X";                //Canvi estat a apagat
    }
    if (WC=='O')                //Llum WC esta oberta
    {
        lcd_gotoxy(7,2);
        printf(lcd_putc,"WC:%c ",WC);
        i2c=1;
        envio_i2c();
        WC="X";                //Canvi estat a apagat
    }
    if (DR=='O')                //Llum menjador esta oberta
    {
        lcd_gotoxy(13,2);
        printf(lcd_putc,"DR:%c ",DR);
        i2c=2;
        envio_i2c();
        DR="X";                //Canvi estat a apagat
    }
    break;
case 105:                        //Torna al menu principal
    lcd_putc('\f');
    delay_ms(200);
    return 1;
}
}
}
}

switch_state()
{
    if (input(pin_E0)==1)        //Detecta llum al dormitori (pin_E0 tancat)
    {
        BR = "O";
    }
}
```

```
else if (input(pin_E0)==0) //No detecta llum al Bedroom (pin_E0 obert)
{
  BR = "X";
}
if (input(pin_E1)==1) //Detecta llum al WC (pin_E1 tancat)
{
  WC = "O";
}
else if (input(pin_E1)==0) //No detecta llum al WC (pin_E1 obert)
{
  WC = "X";
}
if (input(pin_E2)==1) //Detecta llum al menjador (pin_E2 tancat)
{
  DR = "O";
}
else if (input(pin_E2)==0) //No detecta llum al menjador (pin_E2 obert)
{
  DR = "X";
}
}

presence_state()
{
  if ((input(pin_C0)==0)&&(input(pin_E0)==1)&&(conm_BR=='X')) //No detecta
presencia (pin_C0 obert) i esta oberta la llum (pin_E0 tancat) al dormitori
  {
    delay_ms(3000); //Espera 3s i s'apaga la llum
    i2c=0;
    envio_i2c();
    BR='X';
    conm_BR='O';
  }
  if ((input(pin_C0)==1)&&(input(pin_E0)==1)&&(conm_BR=='X'))
  {
    conm_BR='X';
  }
  if ((input(pin_C1)==0)&&(input(pin_E1)==1)&&(conm_WC=='X')) //No detecta
```

```

presencia (pin_C1 obert) i esta oberta la llum (pin_E1 tancat) al WC
{
    delay_ms(3000);                //Espera 3s i s'apaga la llum
    i2c=1;
    envio_i2c();
    WC='X';
    conm_WC='O';
}
if ((input(pin_C1)==1)&&(input(pin_E1)==1)&&(conm_WC=='O'))
{
    conm_WC='X';
}
if ((input(pin_C2)==0)&&(input(pin_E2)==1)&&(conm_DR=='X'))        //No detecta
presencia (pin_C2 obert) i esta oberta la llum (pin_E2 tancat) al menjador
{
    delay_ms(3000);                //Espera 3s i s'apaga la llum
    i2c=2;
    envio_i2c();
    DR='X';
    conm_DR='O';
}
if ((input(pin_C2)==1)&&(input(pin_E2)==1)&&(conm_DR=='O'))
{
    conm_DR='X';
}
}

//-----
-----

```

```

int16 menu_blind(long item)
{
    int16 n_menus=6;

    lcd_putc('\f');

    while (1)
    {

```



```
blind_control(light);
//light = calcul_light_bright();

if (input(pin_A4)==1)
{
  item++;
  delay_ms(200);
  lcd_putc('\f');

  if (item>=(n_menus+200))
  {
    item=200;
  }
}

switch(item)
{
  case 200: //Puja persiana
    light = calcul_light_bright();
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Blind Auto: %c",auto_blind);
    lcd_gotoxy(1,2);
    printf(lcd_putc,"Lightning: %f",light);
    break;
  case 201: //Puja persiana
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Blind Up");
    break;
  case 202: //Para persiana
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Blind Stop");
    break;
  case 203: //Baixa persiana
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Blind Down");
    break;
  case 204: //Para persiana
    lcd_gotoxy(1,1);
```

```
    printf(lcd_putc,"Blind Stop");
    break;
case 205:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Go Back");
    break;
}

if(input(pin_A5)==1)
{
    delay_ms(200);

    switch(item)
    {
        case 200:
            break;
        case 201:
            if (input(pin_B5)==0)                //Mode automàtic apagat (pin_B5 obert)
            {
                i2c=3;
                envio_i2c();
            }
            else
            {
                lcd_gotoxy(1,2);
                printf(lcd_putc,"Automatic is on");
            }
            break;
        case 202:                                //Mode automàtic apagat (pin_B5 obert)
            if (input(pin_B5)==0)
            {
                i2c=4;
                envio_i2c();
            }
            break;
        case 203:                                //Mode automàtic apagat (pin_B5 obert)
            if (input(pin_B5)==0)
            {
```

```
        i2c=5;
        envio_i2c();
    }
    else
    {
        lcd_gotoxy(1,2);
        printf(lcd_putc,"Automatic is on");
    }
    break;
case 204:                                     //Mode automàtic apagat (pin_B5 obert)
    if (input(pin_B5)==0)
    {
        i2c=4;
        envio_i2c();
    }
    break;
case 205:
    lcd_putc('\f');
    delay_ms(200);
    return 2;
}
}
}
```

```
float calcul_light_bright()
{
    int16 q;
    float p;

    port_b_pullups(true);
    set_tris_b(0x01);

    setup_adc(adc_clock_internal);
    set_adc_channel(0);           //Habilitació canal 0 analògic
    delay_us(20);
    q=read_adc();                //Lectura canal 0
    p=(5.0*q*100)/1024.0;        //Conversió a tensió
}
```

```

    return p;
}

blind_control(float light)
{
    if ((light>=350)&&(input(pin_B5)==1))                //Llumiositat major o igual a 3 i
mode automàtic ences (pin_B5 tancat)
    {
        i2c=5;                //Baixa
        envio_i2c();
        auto_blind='O';
    }
    else if ((150<light)&&(light<350)&&(input(pin_B5)==1))    //Llumiositat entre 2 i 3 i
mode automàtic ences (pin_B5 tancat)
    {
        i2c=4;                //Para
        envio_i2c();
        auto_blind='O';
    }
    else if ((light<=150)&&(input(pin_B5)==1))                //Llumiositat menor o igual a 2 i
mode automàtic ences (pin_B5 tancat)
    {
        i2c=3;                //Puja
        envio_i2c();
        auto_blind='O';
    }
    if ((input(pin_B5)==0)&&(auto_blind=='O'))                //Mode automàtic persiana
apagat (pin_B5 obert)
    {
        i2c=4;
        envio_i2c();
        auto_blind='X';                //Canvi estat persiana manual
    }
}

//-----
-----

```

```
int16 menu_temperature(long item)
{
    int16 n_menus=5;

    lcd_putc("\f");

    open_window();                //Interrupció finestra oberta

    while (1)
    {
        temp = calcul_temperature();
        acondicionament(temp);

        if (input(pin_A4)==1)
        {
            item ++;
            delay_ms(200);
            lcd_putc("\f");

            if (item>=(n_menus+300))
            {
                item=300;
            }
        }

        switch(item)
        {
            case 300:                //Mostra temperatura i estat del ventilador i
calefacció
                lcd_gotoxy(1,1);
                printf(lcd_putc,"Temperature:");
                lcd_gotoxy(1,2);
                printf(lcd_putc,"%1.1f ",temp);
                lcd_gotoxy(7,2);
                printf(lcd_putc,"F:%c ",FW);
                lcd_gotoxy(13,2);
                printf(lcd_putc,"H:%c ",HW);
```

```
        break;
    case 301:                                //Encen el ventilador
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Fan:%c ",FW);
        temp = calcul_temperature();
        lcd_gotoxy(8,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        break;
    case 302:                                //Encen la calefacció
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Heater:%c ",HW);
        temp = calcul_temperature();
        lcd_gotoxy(11,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        break;
    case 303:                                //Apaga ventilador i calefacció
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Stop temperature");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"%1.1f ",temp);
        lcd_gotoxy(7,2);
        printf(lcd_putc,"F:%c ",FW);
        lcd_gotoxy(13,2);
        printf(lcd_putc,"H:%c ",HW);
        temp = calcul_temperature();
        break;
    case 304:
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Go Back");
        break;
}

if(input(pin_A5)==1)
{
    delay_ms(200);

    switch(item)
    {
```

```
case 301:
    if ((input(pin_B0)==1)&&(input(pin_B4)==0))           //Finestra oberta (pin_B0
obert) i mode automàtic apagat (pin_B4 obert)
    {
        lcd_putc('\f');
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Fan:%c ",FW);
        temp = calcul_temperature();
        lcd_gotoxy(8,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        lcd_gotoxy(1,2);
        printf(lcd_putc,"Window is open");
        i2c=7;
        envio_i2c();
        FW='O';                                     //Estat del ventilador
        HW='X';                                     //Estat de la calefacció
    }
    if ((input(pin_B0)==0)&&(input(pin_B4)==0))           //Finestra tancada (pin_B0
tancat) i mode automàtic apagat (pin_B4 obert)
    {
        lcd_putc('\f');
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Fan:%c ",FW);
        temp = calcul_temperature();
        lcd_gotoxy(8,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        i2c=7;
        envio_i2c();
        FW='O';
        HW='X';
    }
    else if (input(pin_B4)==1)
    {
        lcd_gotoxy(1,2);
        printf(lcd_putc,"Automatic is on");
    }
    break;
case 302:
```

```

        if ((input(pin_B0)==1)&&(input(pin_B4)==0))           //Finestra oberta (pin_B0
obert) i mode automàtic apagat (pin_B4 obert)
    {
        lcd_putc('\f');
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Heater:%c ",HW);
        temp = calcul_temperature();
        lcd_gotoxy(11,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        lcd_gotoxy(1,2);
        printf(lcd_putc,"Window is open");
        i2c=8;
        envio_i2c();
        HW='O';           //Estat de la calefacció
        FW='X';           //Estat del ventilador
    }
    if ((input(pin_B0)==0)&&(input(pin_B4)==0))           //Finestra tancada (pin_B0
tancat) i mode automàtic apagat (pin_B4 obert)
    {
        lcd_putc('\f');
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Heater:%c ",HW);
        temp = calcul_temperature();
        lcd_gotoxy(11,1);
        printf(lcd_putc,"T=%1.1f ",temp);
        i2c=8;
        envio_i2c();
        HW='O';
        FW='X';
    }
    else if (input(pin_B4)==1)
    {
        lcd_gotoxy(1,2);
        printf(lcd_putc,"Automatic is on");
    }
    break;
case 303:
    if (input(pin_B4)==0)

```



```
        {
            i2c=9;
            envio_i2c();
            HW='X';
            FW='X';
        }
        break;
    case 304:
        lcd_putc('\f');
        delay_ms(200);
        return 3;
    }
}
}
```

float calcul_temperature()

```
{
    int16 q;
    float p;

    port_b_pullups(true);
    set_tris_b(0x01);

    setup_adc(adc_clock_internal);
    set_adc_channel(1);
    delay_us(20);
    q=read_adc();
    p=(5.0*q*10)/1024.0;

    return p;
}
```

acondicionament(float temp)

```
{
    if ((temp>=25)&&(input(pin_B0)==0)&&(input(pin_B4)==1)) //Temperatura major
    o igual a 2,5 i finestra tancada (pin_B0 tancat) i mode automàtic encens (pin_B4 tancat)
    {
```

```

    open_window();
    i2c=7;
    envio_i2c();
    FW='O';
    HW='X';
}
else if ((temp<=17)&&(input(pin_B0)==0)&&(input(pin_B4)==1))           //Temperatura
menor o igual a 2 i finestra tancada (pin_B0 tancat) i mode automàtic encès (pin_B4
tancat)
{
    open_window();
    i2c=8;
    envio_i2c();
    HW='O';
    FW='X';
}
else if ((temp>=25)&&(input(pin_B0)==1)&&(input(pin_B4)==1))           //Temperatura
major o igual a 2,5 i finestra oberta (pin_B0 obert) i mode automàtic encès (pin_B4 tancat)
{
    lcd_gotoxy(1,2);
    printf(lcd_putc,"Window is open");
}
else if ((temp<=17)&&(input(pin_B0)==1)&&(input(pin_B4)==1))           //Temperatura
menor o igual a 2 i finestra oberta (pin_B0 obert) i mode automàtic encès (pin_B4 tancat)
{
    lcd_gotoxy(1,2);
    printf(lcd_putc,"Window is open");
}
else if ((temp>17)&&(temp<25)&&(input(pin_B4)==1))                     //Temperature entre 2 i
2,5 i mode automàtic encès (pin_B4 tancat)
{
    i2c=9;
    envio_i2c();
    HW='X';
    FW='X';
}
//else if ((input(pin_B4)==0)&&((HW=='O')||(FW=='O')))                 //Mode
automàtic apagat (pin_B4 obert)

```

```
//{
// i2c=9;
// envio_i2c();
// HW='X';
// FW='X';
//}
}

open_window()
{
    bit_set(trisb,0);          //B0 com entrada
    port_b_pullups(true);
    enable_interrupts(int_ext);
    ext_int_edge(l_to_h);      //Canvi d'estat de baix a alt
    enable_interrupts(global);
}

//-----
//-----

int16 menu_consume(long item)
{
    int16 n_menus=6;

    lcd_putc('\f');

    while (1)
    {
        voltage = calcul_voltage();
        current = calcul_current();
        power = calcul_power(voltage,current);
        overpower(PW);

        if (input(pin_A4)==1)
        {
            item ++;          //Si pulsa augmenta la variable
            delay_ms(200);
            lcd_putc('\f');
        }
    }
}
```

```
if (item>=(n_menus+400))
{
    item=400;
}
}

switch(item)
{
    case 400:
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Actual Consume");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"PW:%1.1f kW",power);
        break;
    case 401:
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Voltage");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"V:%1.1f V",voltage);
        break;
    case 402:
        lcd_gotoxy(1,1);
        printf(lcd_putc,"Current");
        lcd_gotoxy(1,2);
        printf(lcd_putc,"C:%1.1f A",current);
        break;
    case 403:
        lcd_gotoxy(1,1);
        printf(lcd_putc,"5h:%u", cons1);
        lcd_gotoxy(10,1);
        printf(lcd_putc,"10h:%u", cons2);
        lcd_gotoxy(1,2);
        printf(lcd_putc,"15h:%u", cons3);
        lcd_gotoxy(10,2);
        printf(lcd_putc,"20h:%u", cons4);
        break;
    case 404:
```

```
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Daily consume");
    lcd_gotoxy(1,2);
    printf(lcd_putc,"PW:%u", cons_total);
    break;
case 405:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Go Back");
    break;
}
```

```
if(input(pin_A5)==1)
{
    delay_ms(200);
```

```
    switch(item)
    {
        case 400:
            break;
        case 401:
            break;
        case 402:
            break;
        case 403:
            break;
        case 404:
            break;
        case 405:
            lcd_putc('\f');
            delay_ms(200);
            return 4;
    }
}
```

```
float calcul_voltage()
{
```

```
int16 q;
float p;

port_b_pullups(true);
set_tris_b(0x01);

setup_adc(adc_clock_internal);
set_adc_channel(2);
delay_us(20);
q=read_adc();
p=((5.0*q*4)/1024.0)+220;

return p;
}

float calcul_current()
{
int16 q;
float p;

port_b_pullups(true);
set_tris_b(0x01);

setup_adc(adc_clock_internal);
set_adc_channel(3);
delay_us(20);
q=read_adc();
p=(5.0*q*3)/1024.0;

return p;
}

float calcul_power(float voltage, float current)
{
PW = (voltage*current)/1000;
return PW;
}
```

```
overpower(float PW)
{
    if (PW>=3.0)
    {
        i2c=10;
        envio_i2c();
        lcd_gotoxy(14,2);
        printf(lcd_putc,"OVERPW");
    }
    if ((PW<3.0)&&(PW>2.5))
    {
        i2c=11;
        envio_i2c();
    }
    if (PW<=2.5)
    {
        i2c=12;
        envio_i2c();
    }
}

write_eeprom(long int address, int power_data)
{
    short int status;
    i2c_start();
    i2c_write(0xA0);
    i2c_write(address>>8);
    i2c_write(address);
    i2c_write(power_data);
    i2c_stop();
    i2c_start();
    status=i2c_write(0xA0);

    while (status==1)
    {
        i2c_start();
        status=i2c_write(0xA0);
    }
}
```

```
}
```

```
long int read_eeprom(long int address)
```

```
{
```

```
    int data;
```

```
    i2c_start();
```

```
    i2c_write(0xA0);
```

```
    i2c_write(address>>8);
```

```
    i2c_write(address);
```

```
    i2c_start();
```

```
    i2c_write(0xA1);
```

```
    data=i2c_read(0);
```

```
    i2c_stop();
```

```
    return(data);
```

```
}
```

```
long int consume(byte sec)
```

```
{
```

```
    long int memory;
```

```
    if (sec==5)
```

```
    {
```

```
        memory=0;
```

```
    }
```

```
    if (sec==10)
```

```
    {
```

```
        memory=1;
```

```
    }
```

```
    if (sec==15)
```

```
    {
```

```
        memory=2;
```

```
    }
```

```
    if (sec==20)
```

```
    {
```

```
        memory=3;
```

```
    }
```

```
    return(memory);
```



```
}
```

```
int Daily_consume(int cons1, cons2, cons3, cons4)
```

```
{
```

```
    int suma;
```

```
    suma=(cons1+cons2+cons3+cons4)/4;
```

```
    return(suma);
```

```
}
```

```
//-----  
-----
```

```
int16 menu_savingplan(long item)
```

```
{
```

```
    int16 n_menus=5;
```

```
    lcd_putc('\f');
```

```
    while (1)
```

```
    {
```

```
        saving_plan (sec);
```

```
        if (input(pin_A4)==1)
```

```
        {
```

```
            item ++;           //Si pulsa aumenta la variable
```

```
            delay_ms(200);
```

```
            lcd_putc('\f');
```

```
            if (item>=(n_menus+500))
```

```
            {
```

```
                item=500;
```

```
            }
```

```
        }
```

```
        switch(item)
```

```
        {
```

```
            case 500:
```

```
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Plugs state");
    lcd_gotoxy(1,2);
    printf(lcd_putc,"PL:%c ",PL);
    break;
case 501:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Plugs OFF");
    lcd_gotoxy(1,2);
    printf(lcd_putc,"PL:%c ",PL);
    break;
case 502:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Plugs ON");
    lcd_gotoxy(1,2);
    printf(lcd_putc,"PL:%c ",PL);
    break;
case 501:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Saving Plan");
    lcd_gotoxy(1,2);
    printf(lcd_putc,"State:%c ",save);
    if (input(pin_B3)==0)
    {
        save = "X";
    }
    if (input(pin_B3)==1)
    {
        save = "O";
    }
    break;
case 502:
    lcd_gotoxy(1,1);
    printf(lcd_putc,"Go Back");
    break;
}

if(input(pin_A5)==1)
```

```
{
    delay_ms(200);

    switch(item)
    {
        case 500:
            break;
        /*case 501:
            i2c=13;
            envio_i2c();
            PL='X';
            break;
        case 502:
            i2c=14;
            envio_i2c();
            PL='O';
            break;*/
        case 503:
            break;
        case 504:
            lcd_putc('\f');
            delay_ms(200);
            return 5;
    }
}
}
```

```
saving_plan (sec)
{
    if(input(pin_B2)==1)
    {
        PL="O";
        i2c=13;
        envio_i2c();
    }
    if(input(pin_B2)==0)
    {
```

```
    PL="X";
    i2c=14;
    envio_i2c();
}

if (input(pin_B3)==0)
{
    save = "X";
}
if (input(pin_B3)==1)
{
    save = "O";
}
if ((sec==5)&&(input(pin_B3)==1))
{
    i2c=15;
    envio_i2c();
}
if ((sec==7)&&(input(pin_B3)==1))
{
    i2c=16;
    envio_i2c();
}
if ((sec==10)&&(input(pin_B3)==1))
{
    i2c=17;
    envio_i2c();
}
if ((sec==12)&&(input(pin_B3)==1))
{
    i2c=18;
    envio_i2c();
}
}
```

```
//-----
-----
```

```
clock()
{
    //ds1307_get_date(day,month,yr,dow);
    ds1307_get_time(hrs,min,sec);
    //lcd_gotoxy(1,1);
    //printf(lcd_putc, " %02d/%02d/%02d ", day, month, yr);
    lcd_gotoxy(1,2);
    printf(lcd_putc, " %02d:%02d:%02d ", hrs, min, sec);
    //delay_ms(1000);
}
```

```
//-----  
-----
```

```
void main()
{
    set_tris_a(0xFF);        //(11111111)

    set_tris_b(0xFF);        //(11111111)

    bit_set(trisc,0);
    bit_set(trisc,1);
    bit_set(trisc,2);
    //bit_set(trisc,5);
    //bit_set(trisc,6);
    //bit_set(trisc,7);

    set_tris_e(0xFF);

    lcd_init();

    EEPROM_ADDRESS address;

    //ds1307_init(DS1307_OUT_ON_DISABLED_HIHG | DS1307_OUT_ENABLED |
DS1307_OUT_32_KHZ);
    ds1307_set_date_time(31,10,94,6,0,0,0);

    output_high(pin_B1);
```

```
menu();  
  
}
```