

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ ĐIỆN TỬ**

**----🙣🕮🙡----**



**TIỂU LUẬN**

***Đề tài:*** **Thiết kế máy nước nóng lạnh có màn hình hiển thị bằng PIC16F887**

**Giảng viên hướng dẫn : Trương Năng Toàn**

**Nhóm thực hiện : Nhóm 2**

**Lớp : DHDTMT19**

***TP.HCM, ngày 1 tháng 12 năm 2025***

**BÁO CÁO TIỂU LUẬN**

**MÔN VI XỬ LÍ**

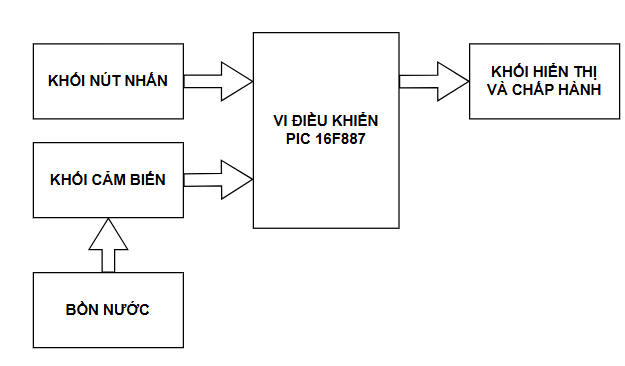
**Tên bài tiểu luận:** Thiết kế máy nước nóng lạnh có màn hình hiển thị bằng PIC16F887

Phân công nhiệm vụ/công việc:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ và Tên** | **Nhiệm vụ/Công việc** |
| 1 | 23673811 | Nguyễn Mạnh Thắng | Viết code, Mạch vật lí |
| 2 | 23688361 | Cao Huỳnh Phi | Báo cáo, Mô phỏng, Mạch vật lí |
| 3 | 23688841 | Huỳnh Hửu Thiên | Báo cáo, Mô Phỏng, Mạch vật lí |
| 4 | 23688591 | Cao Quốc Trọng | Viết code, Mạch vật lí |

**PHẦN I: Thiết kế sơ đồ khối, sơ đồ nguyên lí của mạch trong Proteus.**

**1.1.Sơ đồ khối:**

****

**Giải thích chức năng, nhiệm vụ từng khối.**

**1.Vi điều khiển (PIC16F887).**

**- Chức năng:** Là trung tâm xử lý và điều khiển toàn bộ hệ thống

**- Nhiệm vụ:**

* Nhận dữ liệu từ cảm biến và các nút nhấn.
* So sánh, xử lý và điều khiển các khối khác.
* Xuất dữ liệu ra LCD để hiển thị thông tin.
* Gửi tín hiệu điều khiển đến các phần tử chấp hành (làm nóng, làm lạnh, bơm nước…)

**2. Khối nút nhấn**

**- Chức năng:** Cung cấp giao diện tương tác giữa người dung và hệ thống

**- Nhiệm vụ:**

* MODE: chọn chế độ (Nóng/Lạnh).
* UP/DOWN: tăng hoặc giảm nhiệt độị.
* OUT: rót nước.

**3. Khối cảm biến**

**- Chức năng:** đo nhiệt độ nước và mức nước trong bồn.

**- Nhiệm vụ:**

* LM350 đo nhiệt độ nước nóng và lạnh.
* Cảm biến mức nước (SENSOR) phát hiện các mức nước
* Gửi tín hiệu về PIC16F887 để xử lý .

**4. Khối hiển thị và chấp hành**

**- Chức năng:** Hiển thị thông tin hoạt động và thực hiện điều khiển phần cứng.

**- Nhiệm vụ:**

* Hiển thị nhiệt độ, mức nước, chế độ hoạt động và cảnh báo lỗi trên LCD.
* Điều khiển thiết bị làm nóng (SSR\_HOT), làm lạnh (SSR\_COLD), bơm nước, hoặc đèn báo trạng thái theo lệnh từ vi điều khiển.

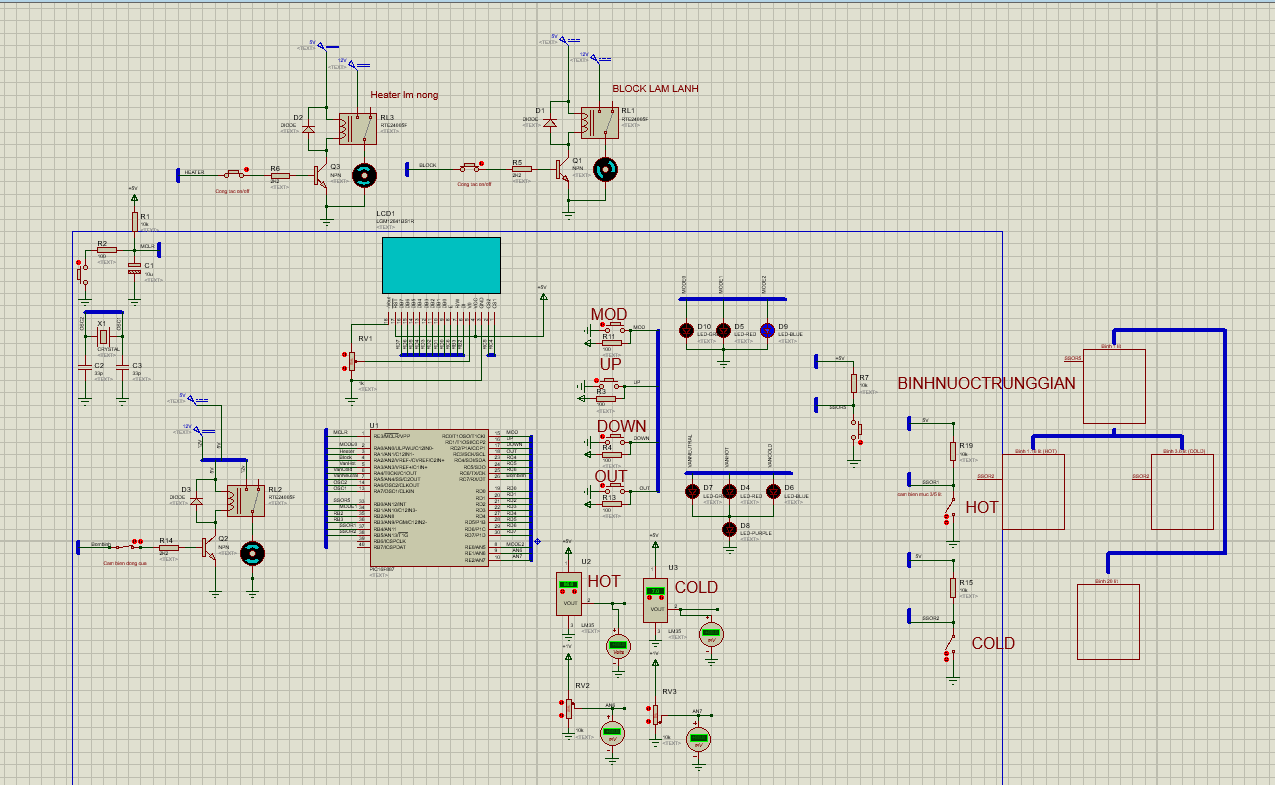
**5. Khối bồn nước (đối tượng điều khiển)**

**- Chức năng:** Là nơi diễn ra quá trình thực tế — chứa nước, nhận nước từ bơm và chịu tác động của hệ thống gia nhiệt / làm lạnh.

**- Nhiệm vụ:**

* Tiếp nhận nước từ bơm.
* Nhiệt độ của nước thay đổi khi SSR\_HOT / SSR\_COLD hoạt động.
* Cung cấp giá trị nhiệt độ và mức nước thực tế cho cảm biến để gửi về vi điều khiển.

**1.2.Sơ đồ nguyên lí của mạch trong Proteus**

****

**- Input(Ngõ vào):**

* Nút nhấn: RC0-1-2-3 - Mode up down out – gửi tín hiệu vi điều khiển:
  + MODE: Chỉnh các chế độ Neutral – Hot - Cold
  + UP/DOWN: Tăng giảm nhiệt độ
  + OUT: Gửi tín hiệu mở van xả nước
* RB0-4-5: Các mức sensor đóng vai trò phao đo lượng nước
* RE1-2: Nhận giá trị nhiệt độ đo được từ bình bằng LM35 (Thay thế bằng biến trở 10k Ohm nguồn 3.3V chia áp còn 1V)

**- Output(Ngõ ra):**

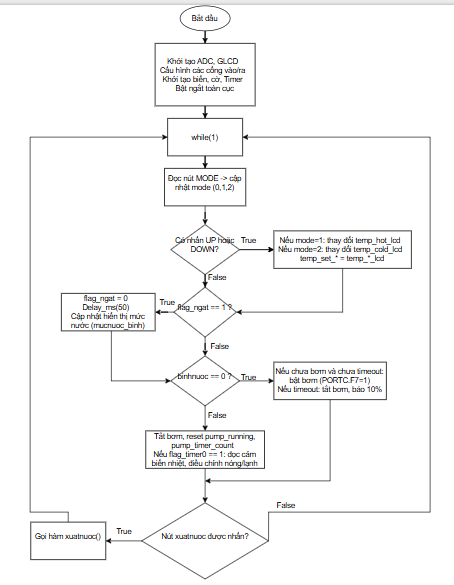
Glcd 128x64 (hiển thị thông tin lượng nước, nhiệt độ):

* DB0-7: RD0-7
* RC6: E
* RB2: D/I
* RB3: RW
* RC4-5: Cs1 Cs2
* RC7: gửi tín hiệu bật motor bơm nước từ bình 20l lên bình trung gian
* RE0, RB1,RA0: lần lượt hiển thị 3 trạng thái mode 0 1 2 - neutral hot cold
* RA1: gửi tín hiệu bật heater làm nóng bình hot
* RA2: gửi tín hiệu bật block làm lạnh bình cold
* RA3-4-5: lần lượt gửi tín hiệu mở van xả nước bình neutral – hot - cold

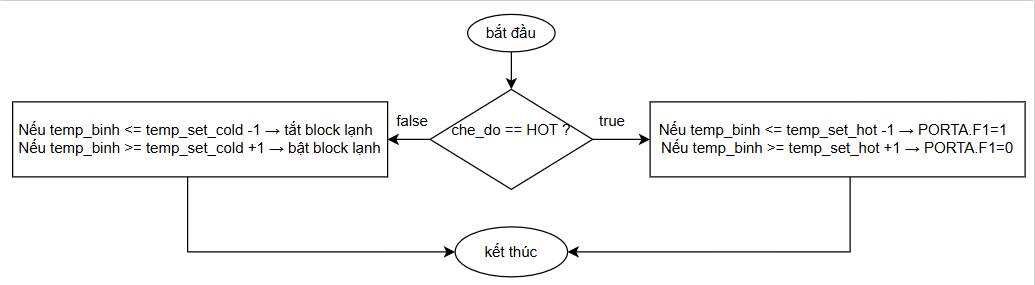
**PHẦN II: Lưu đồ giải thuật cho chương trình, Lâp trình code cho PIC 16F887**

**2.1.Lưu đồ giải thuật cho chương trình**

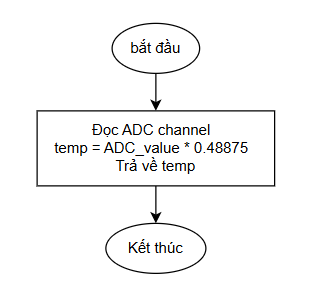
**\*Lưu đồ hàm defmain:**

****

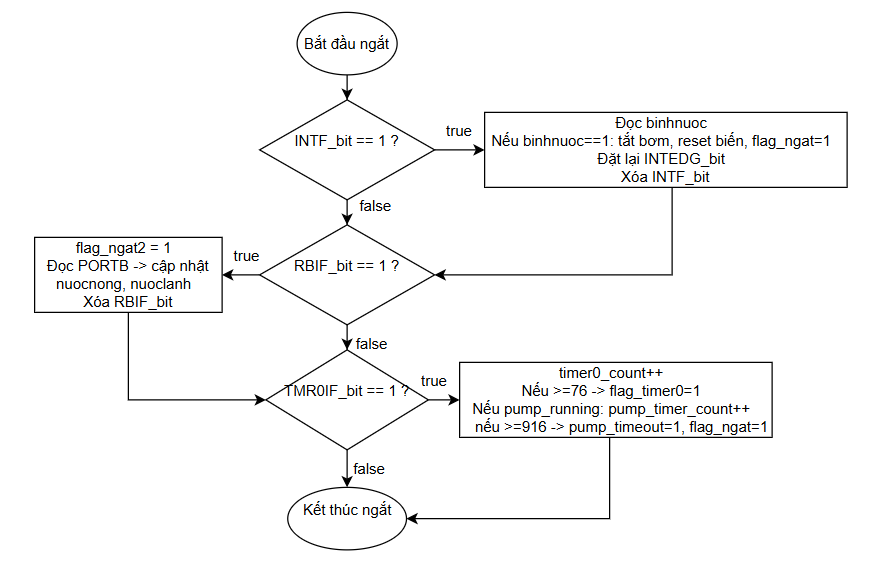
**\*Lưu đồ hàm dieuchinhnhietdo:**

****

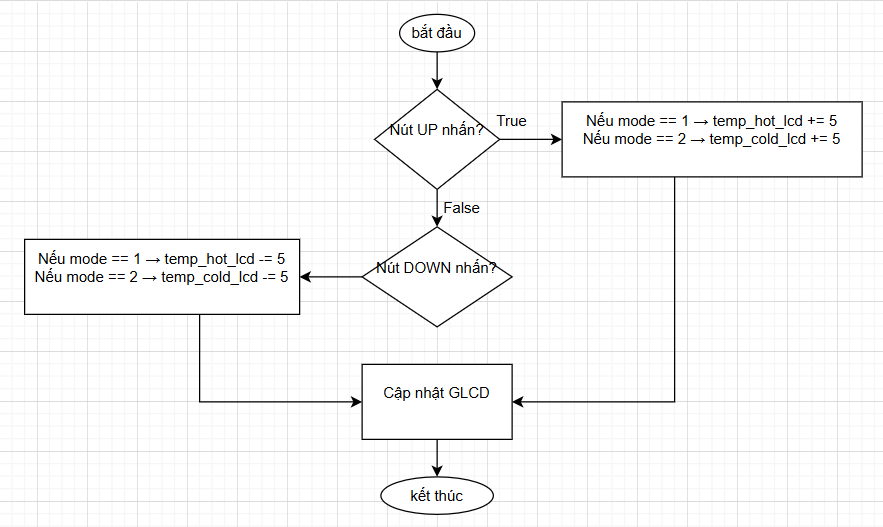
**\*Lưu đồ hàm donhietdobinh:**

****

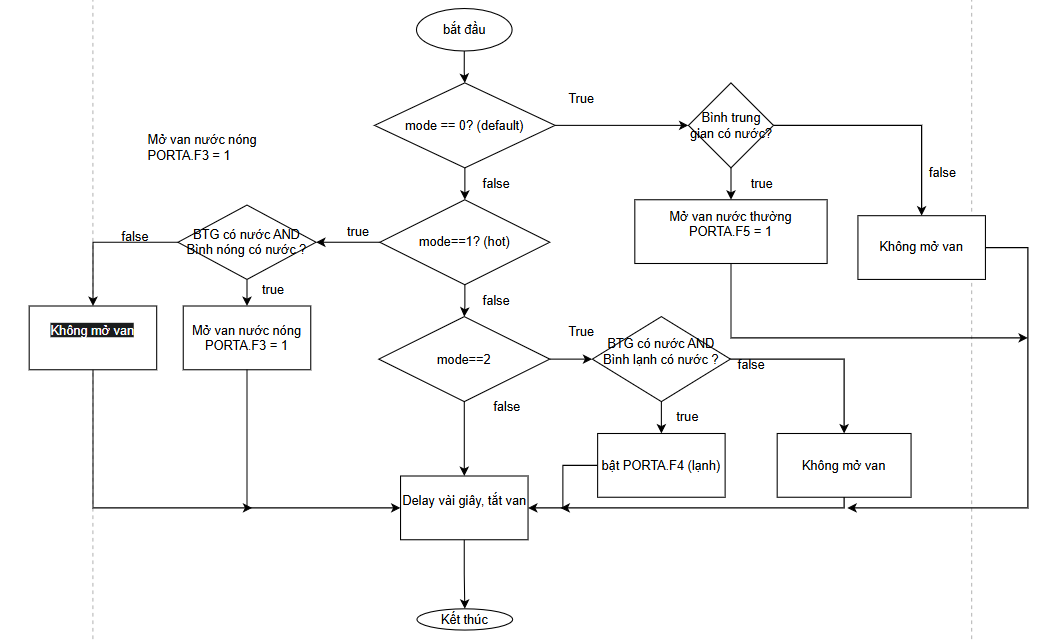
**\*Lưu đồ hàm interrupt:**

****

**\*Lưu đồ hàm up\_down\_button:**

****

**\*Lưu đồ hàm xuatnuoc:**

****

**2.2Lập trình code cho PIC 16F887 ( MikroC )**

/\*

  De tai: Cay nuoc nong lanh hien thi len LCD

  Tham khao: toshiba rwf-w1830uvbv(t) 650w

  Timeout 60s cho motor bom nuoc - cong suat bom nuoc tu binh 20l len binh trung gian 1l la 4-6W

  => mat 60s de bom day

\*/

//====================== GLCD CONNECTIONS ======================//

sbit GLCD\_DataPort at PORTD;

sbit GLCD\_D7 at RD7\_bit;

sbit GLCD\_D6 at RD6\_bit;

sbit GLCD\_D5 at RD5\_bit;

sbit GLCD\_D4 at RD4\_bit;

sbit GLCD\_D3 at RD3\_bit;

sbit GLCD\_D2 at RD2\_bit;

sbit GLCD\_D1 at RD1\_bit;

sbit GLCD\_D0 at RD0\_bit;

sbit GLCD\_D7\_Direction at TRISD7\_bit;

sbit GLCD\_D6\_Direction at TRISD6\_bit;

sbit GLCD\_D5\_Direction at TRISD5\_bit;

sbit GLCD\_D4\_Direction at TRISD4\_bit;

sbit GLCD\_D3\_Direction at TRISD3\_bit;

sbit GLCD\_D2\_Direction at TRISD2\_bit;

sbit GLCD\_D1\_Direction at TRISD1\_bit;

sbit GLCD\_D0\_Direction at TRISD0\_bit;

sbit GLCD\_CS1 at RC5\_bit;

sbit GLCD\_CS2 at RC4\_bit;

sbit GLCD\_RS  at RB2\_bit;

sbit GLCD\_RW  at RB3\_bit;

sbit GLCD\_EN  at RC6\_bit;

sbit GLCD\_RST at RB5\_bit;

sbit GLCD\_CS1\_Direction at TRISC5\_bit;

sbit GLCD\_CS2\_Direction at TRISC4\_bit;

sbit GLCD\_RS\_Direction  at TRISB2\_bit;

sbit GLCD\_RW\_Direction  at TRISB3\_bit;

sbit GLCD\_EN\_Direction  at TRISC6\_bit;

sbit GLCD\_RST\_Direction at TRISB5\_bit;

// End LCD module connections

#define temp\_hot\_default  80;

#define temp\_cold\_default  5;

unsigned char binhnuoc, nuocnong, nuoclanh, nhietdobinhnong, nhietdobinhlanh, temp\_hot\_lcd, temp\_cold\_lcd, temp\_set\_hot, temp\_set\_cold;

unsigned short mode = 0;

bit flag\_ngat, flag\_ngat2, flag\_timer0;

bit pump\_running;           // o danh dau motor dang chay

bit pump\_timeout;           // Co bao timeout 10s

unsigned int timer0\_count = 0;      // Bien dem cho do nhiet do (5s)

unsigned int pump\_timer\_count = 0;  // Bien dem cho timeout motor (10s)

// =============================================

//   Ham ve giao dien GLCD

// =============================================

void Draw\_Interface() {

    char temp\_str[5];

    unsigned char water\_level;

    unsigned char bar\_top;

    unsigned char bar\_bottom;

    unsigned char bar\_height;

    unsigned char fill\_start;

    Glcd\_Fill(0x00);

    Glcd\_Set\_Font(Font5x7, 5, 7, 32);

    // Thanh cot ben trai

    Glcd\_Box(5, 10, 20, 55, 1);

    Glcd\_Rectangle(6, 11, 19, 54, 0);

    // TTinh muc nuoc

    if (binhnuoc == 1) {

        water\_level = 90;

    } else {

        water\_level = 10;

    }

    // Ve muc nuoc

    bar\_bottom = 53;

    bar\_top = 12;

    bar\_height = bar\_bottom - bar\_top;

    fill\_start = bar\_bottom - (water\_level \* bar\_height / 100);

    Glcd\_Box(7, 12, 18, 53, 0);

    if(water\_level > 0) {

        Glcd\_Box(7, fill\_start, 18, bar\_bottom, 1);

    }

     // NEU (NEUTRAL)

    Glcd\_Write\_Text("Neu", 25, 2, 1);

    // HOT

    Glcd\_Write\_Text("Hot", 55, 2, 1);

    IntToStr(temp\_hot\_lcd, temp\_str);

    Ltrim(temp\_str);

    Glcd\_Write\_Text(temp\_str, 55, 4, 1);

    Glcd\_Write\_Text("\*C", 70, 4, 1);

    // COLD

    Glcd\_Write\_Text("Cold", 85, 2, 1);

    IntToStr(temp\_cold\_lcd, temp\_str);

    Ltrim(temp\_str);

    Glcd\_Write\_Text(temp\_str, 85, 4, 1);

    Glcd\_Write\_Text("\*C", 100, 4, 1);

    // Vien khung

    Glcd\_Rectangle(0, 0, 127, 63, 1);

}

void interrupt(){

    if (INTF\_bit) {

        Delay\_ms(50);

        binhnuoc = PORTB.F0;

        // Neu binh day (RB0=1), tat motor va reset timeout

        if (binhnuoc == 1) {

            INTEDG\_bit = 0;

            portc.f7 = 0;

            porte.f0 = 0;

            pump\_running = 0;

            pump\_timer\_count = 0;  // Reset bien dem timeout

            pump\_timeout = 0;       // Xoa co timeout

            flag\_ngat = 1;          // Cap nhat hien thi 90%

        } else {

            INTEDG\_bit = 1;

            // RB0=0: KHONG set flag\_ngat, giu nguyen hien thi

        }

        INTF\_bit = 0;

    }

    if (RBIF\_bit) {

         flag\_ngat2 = 1;

         if (portb.f4==0) nuocnong = 0; //muc 3/5

         else nuocnong = 1;

         if (portb.f5 == 0) nuoclanh = 0;

         else nuoclanh = 1;

         RBIF\_bit = 0;

    }

    // Ngat Timer0

    if (TMR0IF\_bit) {

        TMR0IF\_bit = 0;

        TMR0 = 0;

        // dem nhiet do (5s = 76 lan)

        timer0\_count++;

        if (timer0\_count >= 76) {

            timer0\_count = 0;

            flag\_timer0 = 1;

        }

        // dem cho timeout motor (60s = 916 lan)

        if (pump\_running) {

            pump\_timer\_count++;

            if (pump\_timer\_count >= 916) {  // 916 x 65.536ms = 60s

                pump\_timer\_count = 0;

                pump\_timeout = 1;  // dat co timeout

                pump\_running = 0;

                flag\_ngat = 1;     // Bao cap nhat hien thi 10%

            }

        }

    }

}

void mucnuoc\_binh(){

    unsigned char w, f;

    w = (binhnuoc) ? 90 : 10; //w la muc nuoc hien tai, 90% hoac 10%

    f = 53 - (w \* 41 / 100); //f la vi tri bat dau ve muc nuoc tren GLCD

    Glcd\_Box(7, 12, 18, 53, 0); //xoa muc nuoc cu

    if(w > 0) Glcd\_Box(7, f, 18, 53, 1); //ve muc nuoc moi

}

int button\_mode(){

    unsigned short changed = 0;

    if (PORTC.F0 == 0) {  // nut mode dc nhan

       Delay\_ms(50);

       if (PORTC.F0 == 0) {

          mode++;

          changed = 1;  // danh dau co nut dc nhan

          if (mode > 2) mode = 0;

          while(PORTC.F0 == 0);

       }

    }

     return changed;

}

int up\_down\_button(){

    unsigned short changed = 0;

    char temp\_str[7];

    if (portc.f1 == 0) {   // nut up dc nhan

        Delay\_ms(50);

        if (portc.f1 == 0) {

           if (mode == 1 && temp\_hot\_lcd < 95){

                temp\_hot\_lcd += 5;

                changed = 1;  //danh dau co nut dc nhan

                Glcd\_Box(30, 28, 70, 40, 0);   //cap nhat gia tri nhiet do moi len lcd

                IntToStr(temp\_hot\_lcd, temp\_str);

                Ltrim(temp\_str); //xoa khoang trang o dau

                Glcd\_Write\_Text(temp\_str, 55, 4, 1);

                Glcd\_Write\_Text("\*C", 70, 4, 1);

           }

           if (mode == 2 && temp\_cold\_lcd < 9) {

                temp\_cold\_lcd ++;

                changed = 1;

                Glcd\_Box(85, 28, 120, 40, 0);

                IntToStr(temp\_cold\_lcd, temp\_str);

                Ltrim(temp\_str);

                Glcd\_Write\_Text(temp\_str, 85, 4, 1);

                Glcd\_Write\_Text("\*C", 100, 4, 1);

           }

           while(portc.f1 == 0);

        }

    }

    if (portc.f2 == 0) {  //neu nut down dc nhan

        Delay\_ms(50);

        if (portc.f2 == 0) {

           if (mode == 1 && temp\_hot\_lcd > 80) {

                temp\_hot\_lcd -= 5; //giam nhiet do

                changed = 1;

                Glcd\_Box(30, 28, 70, 40, 0);

                IntToStr(temp\_hot\_lcd, temp\_str);

                Ltrim(temp\_str);

                Glcd\_Write\_Text(temp\_str, 55, 4, 1);

                Glcd\_Write\_Text("\*C", 70, 4, 1);

           }

           if (mode == 2 && temp\_cold\_lcd > 5) {

                temp\_cold\_lcd --;

                changed = 1;

                Glcd\_Box(85, 28, 120, 40, 0);

                IntToStr(temp\_cold\_lcd, temp\_str);

                Ltrim(temp\_str);

                Glcd\_Write\_Text(temp\_str, 85, 4, 1);

                Glcd\_Write\_Text("\*C", 100, 4, 1);

           }

           while(portc.f2 == 0);

        }

    }

    return changed;

}

int donhietdobinh(unsigned char *channel*){

    unsigned int adc\_value;

    adc\_value = ADC\_Read(*channel*);

    return ((adc\_value \* 5.0)/ 1023.0) \* 100.0; //chuyen doi sang do C

}

void dieuchinhnhietdo(int *temp\_binh*, unsigned char *che\_do*){

     if (*che\_do* == 1 && nuocnong == 1) {  // HOT

         if (*temp\_binh* <= (temp\_set\_hot - 1)) porta.f1 = 1; // neu nhiet do binh < nhiet do setting -1 thi bat heater

         if (*temp\_binh* >= (temp\_set\_hot + 1)) porta.f1 = 0; // neu nhiet do binh > nhiet do setting +1 thi tat heater

     }

     if  (*che\_do* == 2 && nuoclanh == 1) {  // COLD

         if (*temp\_binh* <= (temp\_set\_cold - 1)) porta.f2 = 0; // neu nhiet do binh < nhiet do setting -1 thi bat cooler/block

         if (*temp\_binh* >= (temp\_set\_cold + 1)) porta.f2 = 1;// neu nhiet do binh > nhiet do setting +1 thi tat cooler/block

     }

}

void xuatnuoc(){

     if (mode == 0 && binhnuoc == 1) { // an 1 lan: xuat nuoc, an them lan nua: khoa nuoc

         porta.f5 = ~porta.f5; // xa nuoc neu r5 = 1 va khoa nuoc neu ra5 = 0

         Delay\_ms(50);

     }

     if (mode == 1 && binhnuoc == 1) {

         porta.f3 = ~porta.f3;

         Delay\_ms(50);

     }

     if (mode == 2 && binhnuoc == 1) {

         porta.f4 = ~porta.f4;

         Delay\_ms(50);

     }

}

void main() {

    ANSEL = 0b11000000;

    ANSELH = 0;

    ADC\_Init();

    C1ON\_bit = 0;

    C2ON\_bit = 0;

    TRISB = 0b00110001;

    TRISA = 0;

    TRISD = 0;

    TRISE1\_bit = 1;

    TRISE2\_bit = 1;

    TRISE0\_bit = 0;

    TRISC = 0b00001111;

    PORTD = 0;

    PORTB = 0;

    PORTE = 0;

    PORTA = 0;

    portc.f7 = 0;

    pump\_running = 0; pump\_timeout = 0;

    OPTION\_REG = 0b00000111; //dien tro treo

    WPUB = 0b00000001; //cho phep dien tro treo RB0

    Delay\_ms(100);

    Glcd\_Init();

    Delay\_ms(100);

    Glcd\_Fill(0x00);

    Glcd\_Set\_Font(Font5x7, 5, 7, 32); //set font

    Glcd\_Write\_Text("Khoi dong...", 25, 2, 1);

    Glcd\_Write\_Text("Vui long cho", 20, 4, 1);

    Glcd\_Write\_Text("10 giay", 35, 6, 1);

    Delay\_ms(50);

    binhnuoc = PORTB.F0;    //doc trang thai ban dau ssor5 (co nuoc hay ko)

    Delay\_ms(200);

     // doc trang thai ban dau cua binh nong xem co nuoc hay k

     if (PORTB.F4 == 1) nuocnong = 1;

     else nuocnong = 0;

    //tuong tu nhg la binh cold

    if (PORTB.F5 == 1) nuoclanh = 1;

    else nuoclanh = 0;

    if (binhnuoc == 1) {

        INTEDG\_bit = 0;

    } else {

        INTEDG\_bit = 1;

    }

    INTF\_bit = 0;

    INTE\_bit = 1;

    RBIF\_bit = 0;

    RBIE\_bit = 1;

    IOCB = 0b00110000;

    TMR0 = 0;

    TMR0IE\_bit = 1;

    flag\_ngat = 0;

    flag\_ngat2 = 0;

    flag\_timer0 = 0;

    timer0\_count = 0;

    pump\_timer\_count = 0;

    pump\_running = 0;

    pump\_timeout = 0;

    temp\_hot\_lcd = temp\_hot\_default; //luc dau dat lcd = default, setting = default

    temp\_cold\_lcd = temp\_cold\_default;

    temp\_set\_hot = temp\_hot\_default;

    temp\_set\_cold = temp\_cold\_default;

    Delay\_ms(10000);

    Draw\_Interface();

    Delay\_ms(2000);

    GIE\_bit = 1;

    PEIE\_bit = 1;

    porta.f0 = 1; portb.f1 = 0; porte.f0 = 0; //mac dinh o che do NEU

    while (1) {

        if(button\_mode()) {   //xet mode co dc nhan hay k

            if (mode == 0) {porta.f0 = 1; portb.f1 = 0; porte.f0 = 0;}

            if (mode == 1) {porta.f0 = 0; portb.f1 = 1; porte.f0 = 0;}

            if (mode == 2) {porta.f0 = 0; portb.f1 = 0; porte.f0 = 1;} }

        Delay\_ms(50);

        if (up\_down\_button()) { //xet xem co nut up down hay k

            temp\_set\_hot = temp\_hot\_lcd; //dat gia tri nhiet do setting = gt tren lcd

            temp\_set\_cold = temp\_cold\_lcd;

        }

        //xuat nuoc khi nhan nut xuat

        if (portc.f3 == 0) {

            Delay\_ms(50);

            if (portc.f3 == 0) {

                xuatnuoc();

                while (portc.f3 == 0);

            }

        }

        Delay\_ms(50);

        if (flag\_ngat == 1) {

            flag\_ngat = 0;

            Delay\_ms(50);

            // cap nhat hien thi

            if (pump\_timeout && binhnuoc == 0) {

                // Timeout: hien thi 10s

                binhnuoc = 0;

            }

            mucnuoc\_binh(); //ve lai binh nuoc dua tren trang thai hien tai

        }

        Delay\_ms(50);

        // xli motor bom nuoc = timer0

        if (binhnuoc == 0) {  // Binh can (RB0=0)

            if (!pump\_running && !pump\_timeout) {

                // Bat dau bom nuoc

                portc.f7 = 1; //bat cr7 de bat motor ( 7 gio kem 10)

                pump\_running = 1;

                pump\_timer\_count = 0;

            }

            // Kiem tra timeout

            if (pump\_timeout) {

                // Da timeout 60s ma van chua day -> hien thi 10%

                portc.f7 = 0;  // Tat motor

                binhnuoc = 0;  // Dam bao hien thi 10%

                mucnuoc\_binh();

            }

        } else {

            // BBinh day (RB0=1)

            portc.f7 = 0;

            pump\_running = 0;

            pump\_timer\_count = 0;

            pump\_timeout = 0;  // Reset timeout khi day

            if (flag\_ngat2) {

                flag\_ngat2 = 0;

                Delay\_ms(50);

            }

            if (flag\_timer0) {

                flag\_timer0 = 0; //moi 5s doc nhiet do binh va dieu chinh nhiet do 1 lan

                Delay\_ms(50);

                if (nuocnong) {

                    nhietdobinhnong = donhietdobinh(6); //doc nhiet do binh nong

                    dieuchinhnhietdo(nhietdobinhnong, 1); //dieu chinh nhiet do binh nong

                }

                Delay\_ms(200);

                if (nuoclanh) {

                    nhietdobinhlanh = donhietdobinh(7);

                    dieuchinhnhietdo(nhietdobinhlanh, 2);

                }

            }

        }

        Delay\_ms(50);

    }

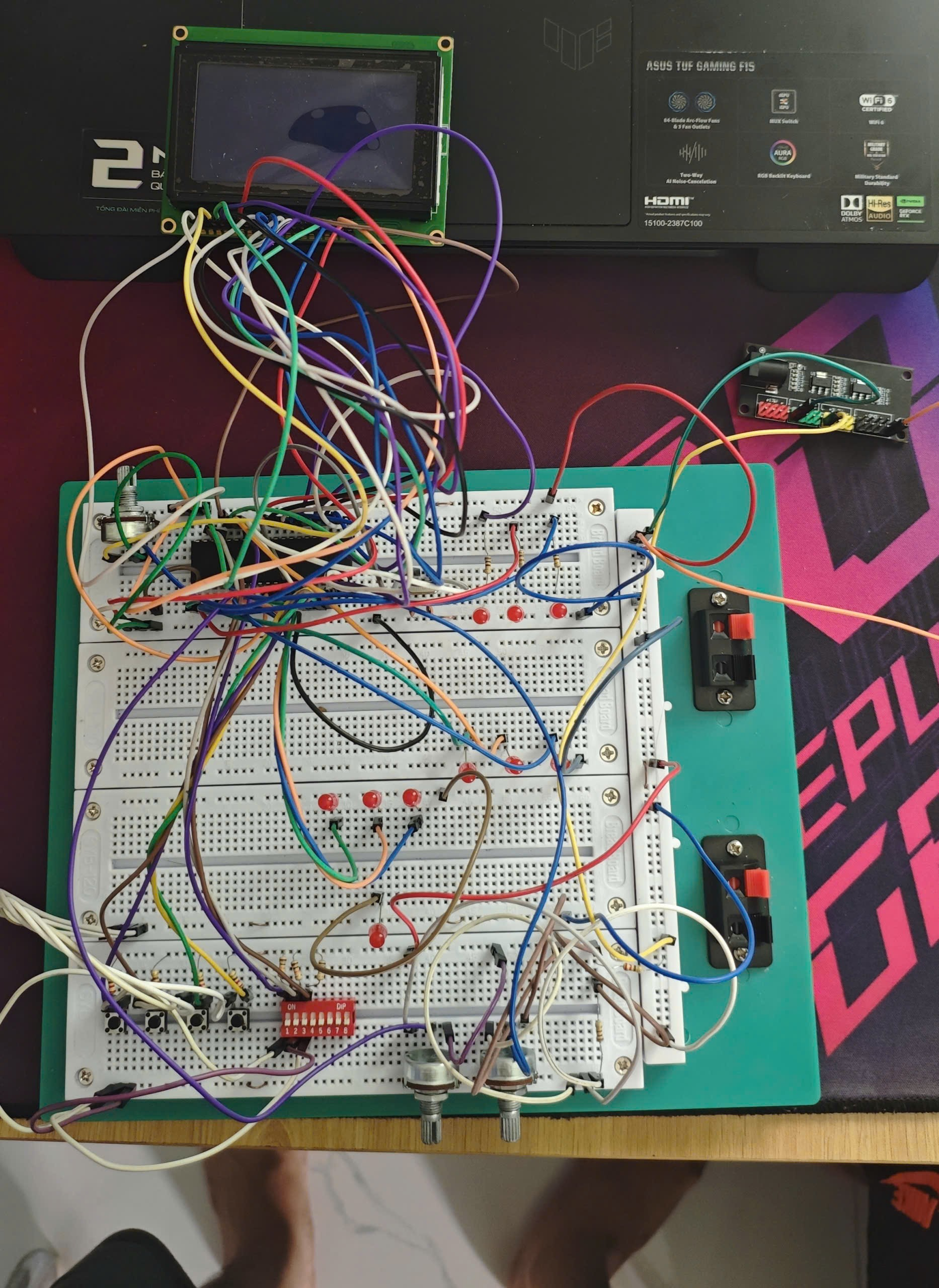
}

**PHẦN III: Thiết kế mạch vật lí và các linh kiện điện tử cần thiết để thiết kế mạch**

**3.1. Các linh kiện cần thiết**

1. Switch 8P DIP x1
2. Nút nhấn 4 chân 6x6x5mm x4
3. Tụ gốm 33pF50V x2
4. Biến trở volume 10k Ohm x3
5. Điện trở 330 Ohm x5
6. Điện trở 10k Ohm x3
7. Điện trở 100 Ohm x4
8. Led đơn x10
9. Thạch anh 4Mhz HC49 x1
10. PIC16F887 x1
11. Hàng rào đực x1
12. Breadboard 240x200x8.5mm x1
13. Dây cắm
14. LCD12864A KS0108 5V x1
15. Nguồn 5V và 3.3V x1

**3.2. Hình ảnh lắp mạch thực tế**

****

**Video:**

<https://youtu.be/aECQ5gtQlio?si=zJlRUO0ByptphZg3>