

## TRƯỜNG CDKT CAO THẮNG

Độc lập – Tự do – Hạnh phúc

## KHOA ĐIỆN – ĐIỆN TỬ

## ĐỀ THI LẠI CUỐI KỲ

MÔN: VI ĐIỀU KHIỂN

LỚP: CD TĐ23A,B,C

Ngày thi: ..../...../2025

Thời gian: 90 phút (Không kể thời gian chép/phát đề thi)

-----*(Sinh viên sử dụng tài liệu 1 tờ giấy chép tay A4, không trao đổi tài liệu)*-----

Một bộ điều khiển hẹn giờ bật/tắt bơm nước tưới cây, sử dụng vi điều khiển PIC16F877A có nguyên lý hoạt động như sau:

+ Bộ điều khiển hiển thị giờ - phút hiện lên 4 LED 7 đoạn. Tần số thạch anh bộ điều khiển sử dụng là 20[MHz]. Chương trình sử dụng Timer1 chế độ định thời, chức năng ngắt khi tràn Timer1, bộ chia prescale tự chọn. Mặc định lúc cấp nguồn hiện 00-00.

+ Có một tín hiệu kích Rơ-le điều khiển bơm nước theo các khoảng thời gian cố định như sau:

- Buổi sáng: Từ 05-30 tới 06-00 xuất tín hiệu bật bơm chạy.
- Buổi chiều: Từ 16-30 tới 17-00 xuất tín hiệu bật bơm chạy.
- Ngoài các khoảng thời gian này thì tắt bơm.

a) Thiết kế sơ đồ nguyên lý của bộ điều khiển thực hiện được các chức năng nêu trên:

- Có đầy đủ các thành phần cơ bản để vi điều khiển PIC16F877A hoạt động được.(0.5đ)
- Hiển thị thông số giờ - phút bằng phương pháp quét led 7 đoạn.(1đ)
- Thiết kế mạch kích Rơ-le. Và sơ đồ đấu nối với bơm 1 pha. Biết Rơ-le sử dụng là Rơ-le 5VDC - dòng 10A, bơm 1 pha 220VAC, có công suất 0.5Hp (373-W).(1đ)

b) Tính toán số xung cần đếm, giá trị cài lại Timer1, số lần ngắt khi tràn Timer1, để thời gian định thời giây chính xác?(1.5đ)

c) Vẽ lưu đồ giải thuật thực hiện các yêu cầu của bộ điều khiển. Có sử dụng hàm để chương trình tinh gọn và dễ quản lý.(3.5đ)

d) Viết chương trình **hoàn thiện** theo các yêu cầu nêu trên. (2.5đ)

TP. HCM, Ngày .... Tháng ..... Năm 2024

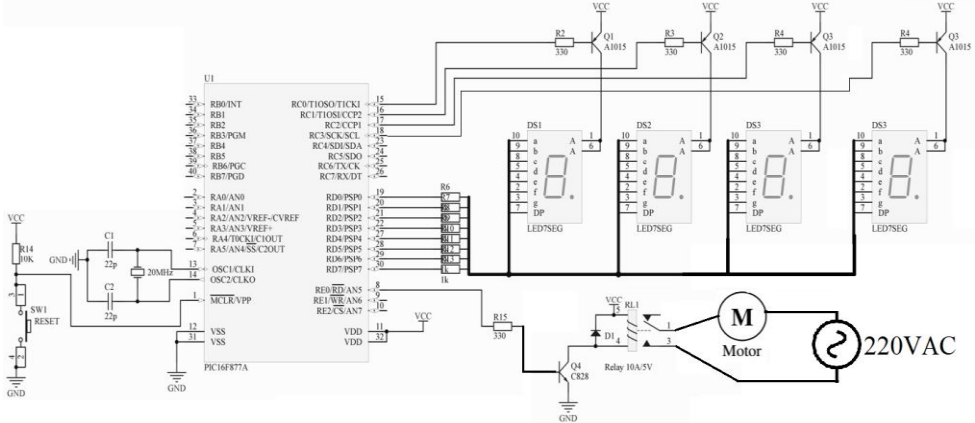
TBM TỰ ĐỘNG HÓA

GV ra đề:

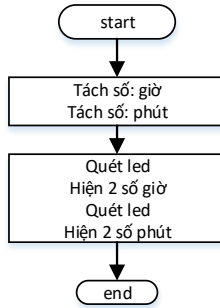
TS. Đặng Đắc Chi

ThS. Cù Minh Phước

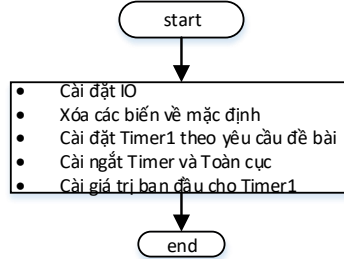
**ĐÁP ÁN ĐỀ THI CUỐI KỲ**  
**MÔN THI: VI XỬ LÝ**  
**LỚP: CĐ TĐ23A,B,C**  
**Thời gian: 90 phút**

<p>a) Thiết kế sơ đồ nguyên lý của bộ điều khiển thực hiện được các chức năng nêu trên:</p> <ul style="list-style-type: none"> <li>+ Vẽ đầy đủ 3 thành phần cơ bản để vi điều khiển hoạt động: Nguồn, nút reset, bộ dao động thạch anh.(0.5đ)</li> <li>+ Vẽ đầy đủ và đúng nguyên lý ngoại vi như đề bài yêu cầu: 4 LED 7 đoạn A chung hoặc K chung, phân cực kích 4 Transistor, có đầu dây vào các chân I/O đầy đủ.(1đ)</li> <li>+ Thiết kế mạch kích Rơ-le. Và sơ đồ đấu nối với bơm 1 pha. Biết Rơ-le sử dụng là Rơ-le 5VDC - dòng 10A, bơm 1 pha 220VAC, có công suất 0.5Hp (373-W).(1đ)</li> </ul>	1đ
	
<p>b) Tính toán số xung cần đếm, giá trị cài lại Timer1, số lần ngắt khi tràn Timer1, để thời gian định thời giây chính xác?(1.5đ)</p> <p><b>Ví dụ: Tính toán giá trị cài Timer1.</b></p> <ul style="list-style-type: none"> <li>+ Tần số xung vào timer: <math>f_{xung} = \frac{f_{osc}}{4 * pres} = \frac{20000000}{4 * 8} = 625000 \text{ Hz}</math></li> <li>+ Thời gian đếm 1 xung: <math>T_{xung} = \frac{1}{f_{xung}} = \frac{1}{625000} = 1.6 * 10^{-6} \text{ s}</math></li> <li>+ Số xung cần đếm để được 1s: <math>\frac{T_{định\ thời}}{T_{xung}} = \frac{1 \text{ s}}{1.6 * 10^{-6} \text{ s}} = 625000 \text{ Xung (0.5đ)}</math></li> <li>+ Số lần ngắt tràn: <math>\frac{\text{Số xung cần đếm}}{\text{Số xung đếm 1 lần}} = \frac{625000}{5000} = 125 \text{ lần (0.5đ)}</math></li> <li>+ Giá trị cài ban đầu: <math>65536 - 5000 = 60536 \text{ (0.5đ)}</math></li> </ul>	2đ
<p>c) Vẽ lưu đồ giải thuật thực hiện các yêu cầu của bộ điều khiển. Có sử dụng hàm để chương trình tinh gọn và dễ quản lý.(3.5đ)</p>	

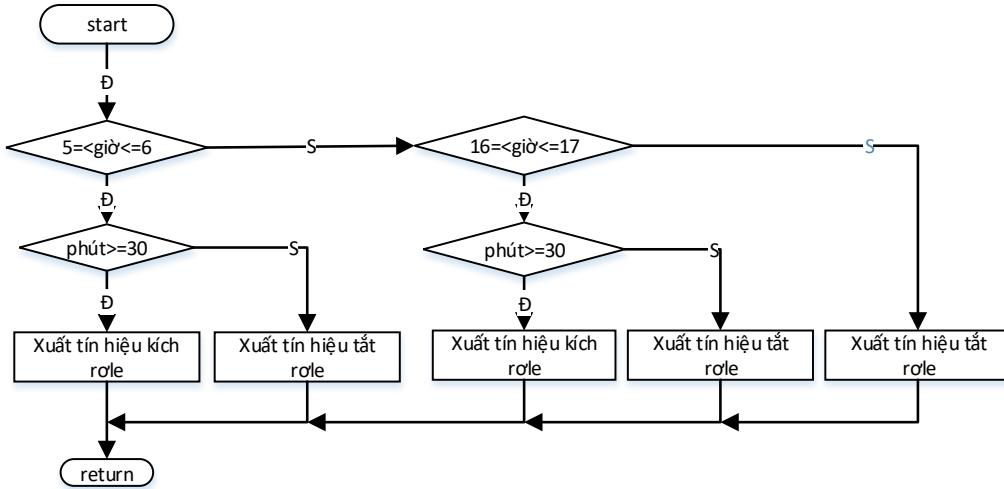
+ Hàm con hiển thị quét led hiển thị giờ - phút: tachso\_ql() (0,5đ)



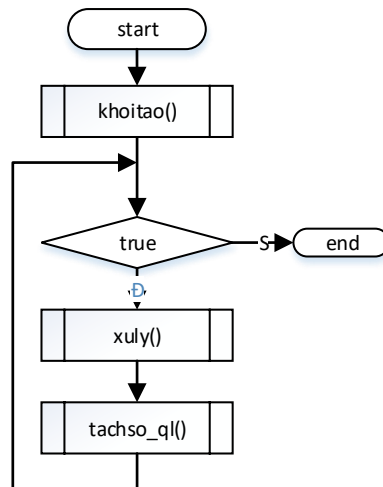
+ Hàm con khởi tạo: khoitao() (0,5đ)



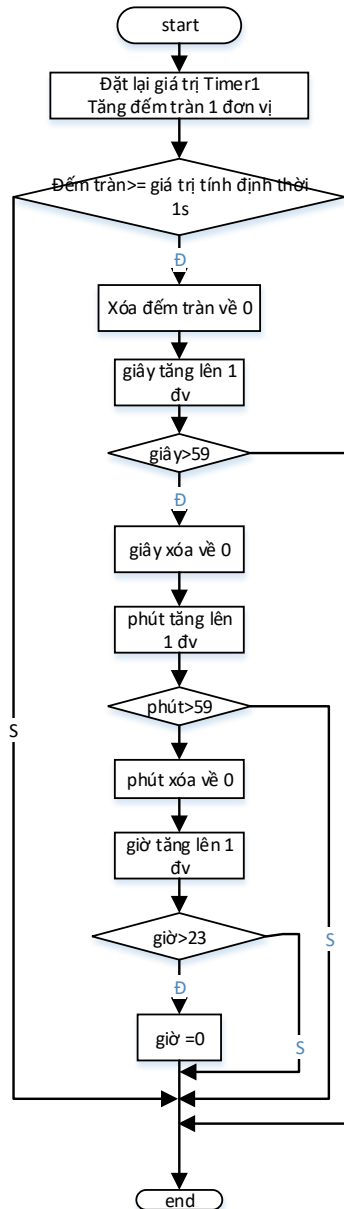
+ Hàm con xử lý giờ bom cây: xuly() (1đ)



+ Chương trình chính: (0,5đ)



+ Chương trình xử lý ngắt tràn timer 1: #INT\_TIMER1 (1đ)



d) Viết chương trình hoàn thiện theo các yêu cầu nêu trên (2.5đ)

+ **Viết hoàn thiện chương trình có đầy đủ các thành phần: 0,25đ**

```
#include <16F877A.h>
```

```
#FUSES NOWDT, HS
```

```
#use delay(clock = 20000000)
```

+ **Khai báo biến tối ưu: 0,25đ**

```
int8 maled[10]={0xc0,0xf9,0xa4,0xb0,0x99,0x92,0x82,0xf8,0x80,0x90};
```

```
int16 demtran=0;
```

```
int8 giay=0, phut=0, gio=0;
```

+ **Tách số quét led đúng theo sơ đồ nguyên lý: (0,5đ) – Sinh viên có thể chia 2 phần**

```
void tachso_ql()
```

```
{
```

```
    int8 Chuc;
```

```
    int8 Donvi;
```

```

Chuc = gio/10;
Donvi = gio%10;
output_low(PIN_C0);
output_d(maled[Chuc]);
delay_ms(10);
output_high(PIN_C0);
output_low(PIN_C1);
output_d(maled[Donvi]);
delay_ms(10);
output_high(PIN_C1);
Chuc = phut/10;
Donvi = phut%10;
output_low(PIN_C2);
output_d(maled[Chuc]);
delay_ms(10);
output_high(PIN_C2);
output_low(PIN_C3);
output_d(maled[Donvi]);
delay_ms(10);
output_high(PIN_C3);
}

```

**+ Xử lý thời gian bật tắt bơm nước: (0,5đ)**

```

void xuly()
{
    if(gio>=5&& gio<=6)
    {
        if(phut>=30) output_high(PIN_E0); // bật bơm
        else output_low(PIN_E0); // tắt bơm
    }
    else if (gio>=16&& gio<=17)
    {
        if(phut>=30) output_high(PIN_E0); // bật bơm
        else output_low(PIN_E0); // tắt bơm
    }
    else output_low(PIN_E0); // tắt bơm
}

```

**+ Khai báo khởi tạo đầy đủ: (0,5đ)**

```

void khoitao()
{
+ Khai báo I/O đúng theo nguyên lý vẽ: (0,25đ)
set_tris_c(0x00);
set_tris_d(0x00);
set_tris_e(0x00);
output_e(0x00);
output_c(0xff);
}

```

+ Cài đặt ngắt Timer1 theo đúng thông số đã tính toán: (0,25đ)

```
gio =0;
phut=0;
setup_timer_1(T1_INTERNAL/T1_DIV_BY_8);
enable_interrupts(INT_TIMER1);
enable_interrupts(GLOBAL);
set_timer1(60536);
}
```

+ Xử lý đếm thời gian khi ngắt tràn timer xảy ra: 0,5đ

```
#INT_TIMER1
void ngatTimer1()
{
    set_timer1(65036);
    demtran++;
    if (demtran >= 125)
    {
        demtran=0;
        giay++;
        if (giay>59)
        {
            giay=0;
            phut++;
            if(phut>59)
            {
                gio++;
                if(gio>23) gio =0;
            }
        }
    }
}
}
void main()
{
    khoitao();
    while (true)
    {
        xuly();
        tachso_ql();
    }
}
```

TBM TỰ ĐỘNG HÓA

GV ra đề:

TS. Đặng Đức Chi

ThS. Cù Minh Phước

