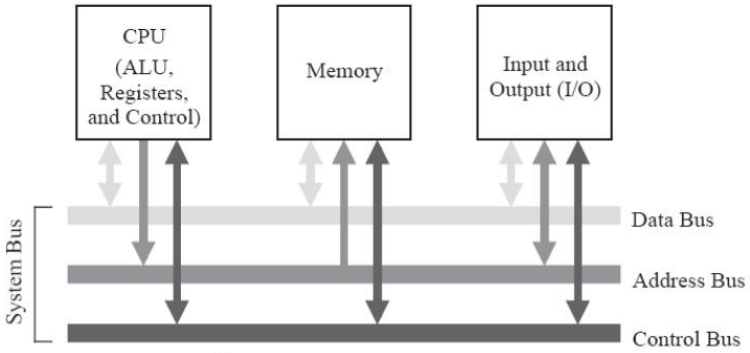
De_thi_EE3480_De1

Câu 1 (2 điểm):



Hình 1: Kiến trúc máy tính theo Von Neumann.

1a) Hãy mô tả ngắn gọn vai trò của từng thành phần trong hệ thống. (1.5 điểm)

1b) Hãy mô tả hoạt động của một lệnh ĐỌC thông tin từ một ô của bộ nhớ vào thanh ghi (0.5 điểm)

a.

1. CPU: bộ vi xử lý trung tâm, bao gồm:

ALU: bộ tính toán số học, nhiệm vụ thực thi các lệnh bằng việc xử lý các tính toán số học, logic.

Thanh ghi (Register): lưu trữ các dữ liệu để xử lý.

Control Unit (CU): đưa ra các tín hiệu điều khiển.

2. Memory: gồm RAM và ROM, có vai trò lưu giữ thông tin.

3. Input/Output: cổng vào ra cho các thiết bị ngoại vi.

b. Hoạt động của một lệnh Đọc thông tin từ một ô nhớ vào thanh ghi:

- Nội dung của PC (program counter) được đặt lên bus địa chỉ.

- Tín hiệu điều khiển READ được xác lập.

- Instruction Opcode của lệnh "ĐỌC" được đọc từ RAM và đưa lên bus dữ liệu.

- Opcode được chốt vào thanh ghi lệnh bên trong CPU.

- PC được tăng để chuẩn bị tìm nạp lệnh kế tiếp từ bộ nhớ.

- Khối giải mã lệnh giải mã Opcode của lệnh ĐỌC và tạo tín hiệu điều khiển. ALU nhận tín hiệu và thực thi lệnh. Quá trình này bao gồm việc đưa địa chỉ ô nhớ cần đọc lên bus địa chỉ, đọc nội dung ô nhớ từ RAM và đưa lên bus dữ liệu, nhận dữ liệu đọc được và chuyển vào thanh ghi, thông báo thực hiện lệnh tiếp theo.

Câu 2 (2 điểm):

Các câu hỏi sau liên quan đến tập lệnh của vi điều khiển MCS-51:

2a) Cho lệnh hợp ngữ (assembly) (1.5 điểm)

MOV A, #20

Câu lệnh này có mã lệnh (instruction code) chiếm mấy bytes trong bộ nhớ? Thực hiện mất bao nhiêu chu kỳ máy? Dịch câu lệnh trên ra mã máy dưới dạng số hexa.

2b) Cho nội dung của bộ nhớ chương trình (0.5 điểm)

Địa chỉ(dạng ***hexa***)	Nội dung ô nhớ (byte) dạng ***hexa***
⋮	
1A0h	74
1A1h	A0
1A2h	36
1A3h	37
1A4h	38
⋮	

- Nếu thanh ghi PC = **1A0h**, hãy dịch ngược mã lệnh từ địa chỉ PC cho đến 1A4h ra các lệnh dạng lệnh hợp ngữ
- Nếu thanh ghi PC = **1A1h**, hãy dịch ngược mã lệnh từ địa chỉ PC cho đến 1A4h ra các lệnh dạng lệnh hợp ngữ

a. MOV A,#20

Lệnh này chiếm 2 bytes trong bộ nhớ.

Thực hiện lệnh mất 1 chu kỳ máy.

Lệnh dưới dạng Hexa: **74 14**

b.

Nếu PC = 1A0H	**Nếu PC = 1A1H**

MOV A,#0A0H
ADDC A,@R0
ADDC A,@R1
ADDC A,R0
	ORL C,36H
	ADDC A,@R1
	ADDC A,R0

Câu 3 (2 điểm):

3a) Trình bầy nguyên lý vào ra bằng ngắt. Vào ra bằng ngắt có ưu/nhược điểm gì so với vào ra bằng chương trình. (1.5 điểm)

3b) Để MCS-8051 sử dụng ngắt ngoài INT0 và INT1. Ngắt INT1 có mức ưu tiên cao hơn thì cần đặt các thanh ghi (SFR) nào, giá trị cài đặt cho các thanh ghi đó (dạng Hex hoặc Binary). Giải thích tại sao đặt như vậy. (0.5 điểm)

a. Nguyên lý vào ra bằng ngắt

Trong hệ vi xử lý, mỗi khi có một thiết bị cần được phục vụ thì thiết bị sẽ báo cho bộ vi điều khiển bằng cách gửi một tín hiệu ngắt. Khi nhận được tín hiệu này, CPU sẽ kiểm tra xem ngắt có được phép hay không, nếu được phép, việc thực thi chương trình chính tạm dừng và CPU thực hiện việc rẽ nhánh đến trình phục vụ ngắt ISR. CPU thực thi ISR để thực hiện một công việc và kết thúc việc thực thi này khi gặp lệnh RETI (trở về từ ngắt), chương trình chính được tiếp tục tại nơi bị tạm dừng.

Ưu nhược điểm so với vào ra bằng chương trình

Ưu điểm của ngắt:

+ Giúp bộ Vi điều khiển có thể phục vụ nhiều thiết bị.

+ Có thể gán mức ưu tiên phục vụ cho các thiết bị.

+ Cho phép che hoặc bỏ qua yêu cầu phục vụ của thiết bị.

+ Không làm mất thời gian do vi xử lý chỉ cần thực hiện ISR khi gặp tín hiệu ngắt, không cần dò hỏi thiết bị hay tín hiệu bằng vòng lặp liên tục.

Nhược điểm của ngắt:

Quá trình lưu trạng thái và khôi phục trạng thái dễ gây ra lỗi, đặc biệt là nếu trong chương trình phục vụ ngắt có tác động đến ngăn xếp. Chương trình có sử dụng ngắt có hoạt động phức tạp dễ gây khó khăn, nhầm lẫn cho người lập trình.

b. Thanh ghi cho phép ngắt IE

Là thanh ghi cho phép ngắt. Để ngắt có thể làm việc, phải thiết lập bit **EA** (Enable All) của thanh ghi này lên 1. Các bit còn lại là cho phép ngắt nối tiếp (ES), ngắt Timer 1, Timer 0 (ET) và ngắt ngoài (EX)

EA	--	ET2	ES	ET1	EX1	ET0	EX0

Giá trị cần đặt cho IE: 1000 0101

Cho 2 bit ngắt ngoài EX1 và EX0 lên cao.

Thanh ghi ưu tiên ngắt IP

Điều khiển ưu tiên ngắt.

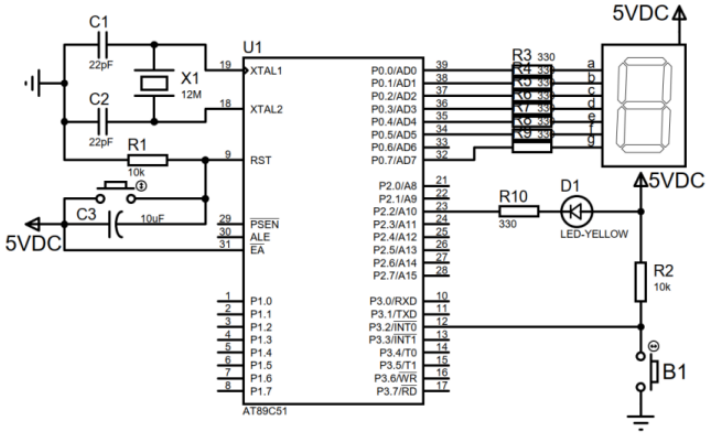
--	--	PT2	PS	PT1	PX1	PT0	PX0

Giá trị cần đặt cho IP: 0000 0100

Đưa bit PX1 lên cao để INT1 ưu tiên cao hơn INT0

Câu 4 (4 điểm):

Cho sơ đồ nguyên lý mạch vi xử lý 8051 như hình *hình 2*



Hình 2

4a) Hãy thành lập các dữ liệu mà khi đưa ra P0 thì LED 7 thanh sẽ hiển thị tương ứng các số: 0,2,4,6,8. (1 điểm)

4b) Hãy viết chương trình điều khiển LED D1 nháy với chu kỳ 120ms. (1.5 điểm)

4c) Hãy viết chương trình để khi ấn phím B1 thì LED 7 thanh sẽ hiện thị lần lượt các số 0,4,8 sau mỗi 60ms và lặp lại cho đến khi nhả phím. (1.5 điểm)

a. Dữ liệu cần đưa ra đầu P0

Số	a (P0.0)	b (P0.1)	c (P0.2)	d (P0.3)	e (P0.4)	f (P0.5)	g (P0.7)
0	0	0	0	0	0	0	1
2	0	0	1	0	0	1	0
4	1	0	0	1	1	0	0
6	0	1	0	0	0	0	0
8	0	0	0	0	0	0	0

b. Chương trình điều khiển nháy LED D1 với chu kỳ 120ms

Để LED D1 nháy với chu kỳ 120ms, cần có Timer tạo trễ 60ms để đưa xung vuông ra đầu P2.2

Tần số XTAL = 12MHz, chu kỳ máy = 12 x chu kỳ thạch anh = 1µs

Số chu kỳ máy để tạo trễ 60ms: 60ms/1µs = 60.000 chu kỳ máy.

Giá trị cần nạp vào 2 thanh ghi của Timer: -60.000 = 15A0H

```
          CLR P2.2
          MOV TMOD,#0000 0001B     ;Timer 0, chế độ 1
LOOP:     MOV TH0,#15H
          MOV TH1,#A0H
```

```
          SETB TR0              ;Khởi động Timer 0
CHECK:    JNB TF0, CHECK        ;Kiểm tra cờ Timer
          CLR TR0
          CLR TF0
          CPL P2.2              ;Đảo bit P2.2 để tạo xung
          SJMP LOOP
```

c. Chương trình để bấm phím B1 thì sẽ xuất hiện 0, 4, 8 mỗi 60ms cho đến khi nhả phím.

Sử dụng ngắt ngoài 0 để thực hiện

```
          ORG 0000H
          LJMP MAIN
;Bảng vector ngắt
          ORG 0003H
          LJMP LEDLOOP
;Chương trình chính
          ORG 30H
MAIN:     MOV IE,#1000 0001B        ;Cho phép ngắt ngoài 0
          MOV TMOD,#0000 0001B      ;Timer 0 chế độ 1
          MOV P0,#00H
;Trình phục vụ ngắt
          ORG 200H
LEDLOOP:  MOV P0,#01H          ;Hiện số 0
          ACALL DELAY
          MOV P0,#98H          ;Hiện số 4
          ACALL DELAY
          MOV P0,#00H          ;Hiện số 8
          ACALL DELAY
          RETI
;Hàm tạo trễ
          ORG 250H
DELAY:
          MOV R3,#240     ;Nạp 240 lần cho vòng lặp ngoài
NEXT:     MOV R2,#255     ;Nạp 250 lần cho vòng lặp trong
AGAIN:    DJNZ R2, AGAIN        ;Lặp lại 250 lần
          DJNZ R3, NEXT         ;Xong 1 vòng lặp trong

          END
```