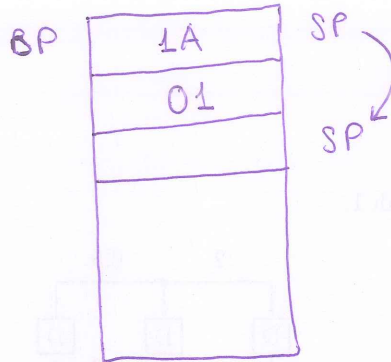


## Giữa kỳ Vi xử lý 20141

Câu 1:

MOV BP, SP; BP = SP = đỉnh stack

PUSH WORD PTR 11Ah; Cất xâu 011A vào Stack  
; SP tự động giảm 2



MOV AL, [BP-1]; AL = nội dung trong ô có địa chỉ BP-1  
; AL = 01

Câu 2:

CMP AL, 39h; So sánh AL với 39h

JA LABEL1; Nếu lớn hơn thì nhảy đến nhãn LABEL1  
; JA = Jump if Above

ADD AL, 37h; AL = AL + 37h  
; nếu sai thì thực hiện lệnh tiếp theo

JMP LABEL2 LABEL1:; JMP là nhảy không điều kiện

ADD AL, 30h; AL = AL + 30h

LABEL2:

Lệnh trên thuộc cấu trúc if-else

~~Câu 3:~~ if (AL > 39h)  
{  
    AL = AL + 30h; // LABEL1  
}  
else  
{  
    AL = AL + 37h;  
}  
// do LABEL2 ...

Long Đặng

Câu 3: Chọn lệnh ADD

LEA: Nạp địa chỉ hiệu dụng vào thanh ghi

MOV: Gán

XCHG: Hoán đổi nội dung hai thanh ghi

ADD: Đích = Nguồn + Đích

Câu 4: Chọn ROM - BIOS

Câu 5: Giá trị của độ dịch được xác định từ trường R/M

MOD = 00: không có dịch chuyển

MOD = 01: dịch chuyển 8 bit

MOD = 10: dịch chuyển 16 bit

MOD = 11: R/M là thanh ghi

Trường R/M phụ thuộc đích

Trường REG phụ thuộc nguồn

Mã lệnh phụ thuộc lệnh

Câu 6: Chọn DF

DF = 1 khi CPU làm việc với chuỗi ký tự theo thứ tự từ phải sang trái.

Câu 7: Chọn Mã lệnh

Câu 8: CMP AL, '9'; so sánh AL với 57

; đang có AL = 3Ch = 60

JA NEXT; nếu đúng AL > 57 thì nhảy đến NEXT

Câu 9: Cặp SS: BP

Địa chỉ vật lý được truy cập đến là:

$$SS * 10h + BP + 3 = 00020h + 0100h + 3 = 0123h$$

Câu 10:  $74 = 2^6 + 2^3 + 2^1 = 01001010$

Câu 11: Chọn SS: SP

Câu 12:  $SS * 10h + SP = 10200h + 0100h = 10300h$

Câu 13: MOV DX, 7000H; nếu địa chỉ của cổng là 16 bit

thì phải đưa vào DX trước  
IN AL, DX; rồi sau đó mới dùng AL hoặc AX đọc

Câu 14: XOR AL, AL; Vì  $1 \text{ xor } 1 = 0$  và  $0 \text{ xor } 0 = 0$

; lệnh này để cài đặt AL = 00h

Long Đặng

CMP AL, AL; So sánh AL với AL

TEST AL, AL; kiểm tra  $AL == 0$

; Thực hiện phép AND giữa AL với AL

; Nếu kết quả bằng 0 thì  $ZF = 1$

; Nếu kết quả khác 0 thì  $ZF = 0$

; Như vậy  $ZF = 1 \Leftrightarrow AL = 0000\ 0000$

AND AL, 0;  $AL = AL \text{ AND } 0 = 0$

Câu 15: MOV AX, 0F978h;  $AX = 0F978h$

SHL AH, 1; Dịch trái AH 1 bit

; Ban đầu  $AH = F9$ ; dịch trái 1 bit  $\rightarrow AH = F2$

; bit 1 ngoài cùng đẩy ra CF

;  $F9 = 1111\ 1001$

; Dịch 1 bit:  $1111\ 0010 = F2$

CF = 1

ADC AL, 02h;  $AL = AL + 02h + CF =$

;  $AL = 78h + 02h + 1$

$$\begin{array}{r} 78 \\ + 02 \\ 1 \\ \hline 7B \end{array}$$

*Long Đặng*

Vậy  $AX = F27Bh$

Câu 16: XOR AX, BX;  $AX \neq BX$  thì  $AX \neq 0$ ;  $ZF = 0$

;  $AX = BX$  thì  $AX = 0$ ;  $ZF = 1$

JNZ Lenh; Nhảy nếu  $ZF$  không bằng 1

Câu 17: Từ CPU đến bộ nhớ và thiết bị ngoại vi

Câu 18: MOV CL, 35;  $CL = 35 = 2^5 + 2^1 + 2^0 = 00100011$

Câu 19: MOV AL, [BX];  $AL =$  nội dung tại địa chỉ DS: BX  
; chế độ địa chỉ gián tiếp qua thanh ghi

Câu 20: MOV CX, 2;  $CX = 2$  và  $CL = 2$

MOV AX, 255;  $AX = 255 = 0000\ 0000\ 1111\ 1111$

DICH: SAL AX, CL; Dịch trái AX 2 bit

;  $AX = 0000\ 0011\ 1111\ 1100$



LOOP DICH ; Lặp lại lệnh DICH 2 lần vì đang có CX = 2  
; Ở sau mỗi lần DICH thì CX tự động giảm 1  
; CX giảm 1 nghĩa là CL giảm 1

; Lần 1: CL = 2 ; AX = AX << 2 = 0000 1111 1111 0000

; Lần 2: CL = 1 ; AX = AX << 1 = 0001 1111 1110 0000

; AX = 1FE0h

Câu 21:

; VALUE < 1000 → VALUE < 0000 0011 1110 1000

MOV BX, AX ; BX = AX = VALUE

SHL AX, CL ; Dịch trái AX 2 bit vì đề bài cho CL = 2

; AX tăng gấp 4 lần (AX = 4 VALUE)

ADD AX, BX ; AX = AX + BX = 5 VALUE

Câu 22: 24 bit địa chỉ → có  $2^{24}$  địa chỉ

Mỗi ô nhớ có 1 địa chỉ duy nhất, mỗi ô nhớ dung lượng 1 B

→ Dung lượng quản lý được là:

$2^{24}$  ô nhớ  $\times 1 \text{ B} = 2^{24} \text{ B} = 16 \text{ MB}$

Câu 23:

MSG DB 10 DUP(48)

; khai báo mảng MSG gồm 10 B chứa 10 giá trị 48

A DW 20, 120

; khai báo mảng A gồm 4 B (2 B lưu 20, 2 B lưu 120)

; vậy chiếm tất cả 14 B trong bộ nhớ.

Câu 24:

TEST AX, 1 ; kiểm tra bit 0 của AX

TEST AL, 1 ; kiểm tra bit 0 của AL

DIV BX ; AX = AX / BX và DX = phần dư

CMP DX, 0 ; so sánh DX với 0

chọn cả 3 phương án

Câu 25:

Ví dụ: MOV AH, 2 ; dùng hàm 2

MOV DL, 'A' ; DL = 'A'

INT 21h ; Ngắt để in ký tự trong DL ra

; AL = 'A'

Long Đăng

Câu 26: TEST AL, 4; 4 = 0000 0100 → kiểm tra bit 2  
 AND AL, 0FBh; AL = AL & 0FBh → thay đổi AL  
 OR AL, 4; AL = AL | 4 → thay đổi AL  
 XOR AL, 4; AL = AL XOR 4 → thay đổi AL

Câu 27: chọn PROC

CALL: gọi chương trình con  
 JMP: nhảy không điều kiện  
 INT: ngắt

Câu 28: Cặp thanh ghi DS: BX; BX = 4

DS: 0000 00 5A 47 C1 50 1A B8 90  
 Lệnh 0 1 2 3 4 5 6 7

→ [BX+2] = [4+2] = [6] = B8

MOV AX, [BX+2]; AL = [BX+2] = B8  
 ; AH = [BX+3] = 90  
 ; AX = 90B8

Câu 29: Biết CX = 0100h = 4

LAP: MOV CL, 2; CL = 2 → CX = 0102h

LOOP LAP; Sau 1 lần lặp CX giảm 1 → CL giảm 1

; vì LAP thì CL được gán lại bằng 2 nên không

; bao giờ có CX = 0

; vậy lặp vô cùng

Câu 30: LAP: TEST CX, CX; kiểm tra CX == 0

JZ THOAT; Nếu ZF = 1 (tức CX = 0) thì ~~thoát~~ THOAT

DEC CX; Giảm CX 1 đơn vị

JMP LAP; Nhảy không điều kiện đến LAP

THOAT:  
 cấu trúc while vì while (CX == 0) { CX--; }

Đáp án:

Long Dũng

Câu	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	C	B	D	D	B	C	D	B	C	D	A	A	C	C	B
Câu	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
	A	C	D	B	D	A	A	A	D	B	A	B	D	D	A