

Đề thi kỹ thuật Vi xử lý
ĐT 1,2,3,4,5,6,7,8 – K48
Thời gian : 90 phút

Câu 1 (3 điểm)

Trình bày các khái niệm sau :

Quản lý bộ nhớ theo chế độ thực

Xử lý pipelining

Little endian và big endian

Chế độ địa chỉ gián tiếp qua thanh ghi

Vào ra theo kiểu hỏi vòng

DSP

Câu 2 (2 điểm)

Một hệ vi xử lý bao gồm vxl 8086 ghép nối với 48 KB EPROM sử dụng EPROM 2764 (8K*8)

và 128 KB SRAM sử dụng các IC SRAM giống hệt nhau với 8 bit dữ liệu . Giải mã địa chỉ cho EPROM được thực hiện bởi 74LS139 và giải mã địa chỉ cho SRAM được thực hiện bởi 74LS138

Cho giá trị hiện tại của các thanh ghi : CS = 0600H , IP = 2000H , DS = 1000H , BX = 8000H

Giả thiết rằng chương trình và dữ liệu được lưu trong RAM và liên tiếp theo sẽ được thực hiện là lệnh MOV AL,[BX+1] , lệnh này được lưu trữ tại các byte nhớ đầu tiên của IC thứ 2 thuộc bank thấp .Khi thực hiện lệnh MOV này , byte nhớ đầu tiên của IC thứ 4 thuộc bank cao sẽ được truy cập

Hãy vẽ sơ đồ ghép nối 8086 với bộ nhớ EPROM và SRAM trong hệ vxl được miêu tả ở trên

Câu 3 . (2 điểm)

a.Hãy viết chương trình assembly cho 8086 thực hiện

1.Nhập số N ($N < 9$, nguyên dương) từ bàn phím

2.Tính $S = 1^2 + .. + N^2$

3. Kiểm tra tính chẵn lẻ của S

b.Hãy tối ưu chương trình vừa viết (nếu có thể) theo kích thước của chương trình trong bộ nhớ

Đề II

Câu 1:

Trình bày các khái niệm:

- ghép nối bằng ngắt
- Vi điều khiển
- DMA
- Hyperthearding và dual core
- Làm tươi Dram

Câu 2:

Một hệ VXL bao gồm VXL 8086 ghép nối với 24KB EPROM sử dụng 2732 (4K*8) và 64KB SRAM sử dụng các IC SRAM giống hệt nhau với 8 bit dữ liệu. Giải mã địa chỉ cho EPROM được thực hiện bởi 74LS138 và giải mã địa chỉ cho SRAM được thực hiện bởi 74LS139.

Cho giá trị hiện tại của các thanh ghi: CS = 0300H, IP = 1000H, DS = 0BFFH, BX = 000EH

Giả thiết rằng chương trình và dữ liệu được lưu trữ trong RAM và lệnh ADD AL, [BX + 3], lệnh này được lưu trữ tại các byte nhớ đầu tiên của IC thứ 2 thuộc bank thấp. Khi thực hiện lệnh ADD này, byte nhớ đầu tiên của IC thứ 4 thuộc bank cao sẽ được truy cập.

Hãy vẽ sơ đồ ghép nối 8086 với bộ nhớ EPROM và SRAM trong hệ VXL được miêu tả ở trên

Câu 3:

a). Hãy viết chương trình assembly cho 8086 thực hiện

1. Nhập vào 1 số nguyên N (≤ 9)
2. Nhập N số nguyên (< 10) và tính tổng S của các số đó
3. In S ra màn hình

b). Hãy tối ưu chương trình vừa viết (nếu có thể) theo kích thước của chương trình trong bộ nhớ

Hướng dẫn giải đề vi xử lý

Phần I : Ghép nối 8086 với bộ nhớ

Bài tập : (Đề thi vi xử lý lần I – năm học 2005-2006 – K48 khoa ĐTVT)

Một hệ VXL bao gồm VXL 8086 ghép nối với 24KB EPROM sử dụng 2732 (4K*8) và 64KB SRAM sử dụng các IC SRAM giống hệt nhau với 8 bit dữ liệu. Giải mã địa chỉ cho EPROM được thực hiện bởi 74LS138 và giải mã địa chỉ cho SRAM được thực hiện bởi 74LS139.

Cho giá trị hiện tại của các thanh ghi: CS = 0300H, IP = 1000H,

DS = 0BFFH, BX = 000EH

Giả thiết rằng chương trình và dữ liệu được lưu trữ trong RAM và lệnh ADD AL, [BX + 3] , lệnh này được lưu trữ tại các byte nhớ đầu tiên của IC thứ 2 thuộc bank thấp. Khi thực hiện lệnh ADD này, byte nhớ đầu tiên của IC thứ 4 thuộc bank cao sẽ được truy cập.

Hãy vẽ sơ đồ ghép nối 8086 với bộ nhớ EPROM và SRAM trong hệ VXL được miêu tả ở trên.

Giải :

* ROM :

$$4K \times 8\text{bit} \xrightarrow[\text{đọc}]{6 \text{ IC}} 24K \times 8\text{bit}$$

- Do ROM phải chứa địa chỉ khởi động của CPU là : FFFF0 H nên ta chọn địa chỉ kết thúc của ROM là FFFFF H.

- Mà 24KB = 0000 0110 0000 0000 0000 = 06000 H

⇒ Địa chỉ đầu = Địa chỉ cuối – Dung lượng + 1

$$= \text{FFFFH} - 06000\text{H} + 1$$

$$= \text{FA000H}$$

- Ta có : $4K = 2^2 \times 2^{10} = 2^{12}$

⇒ Địa chỉ thay đổi : $A_1 - A_{12}$

Địa chỉ giải mã : $A_{13} - A_{15}$

Địa chỉ cố định : $A_{16} - A_{19}$

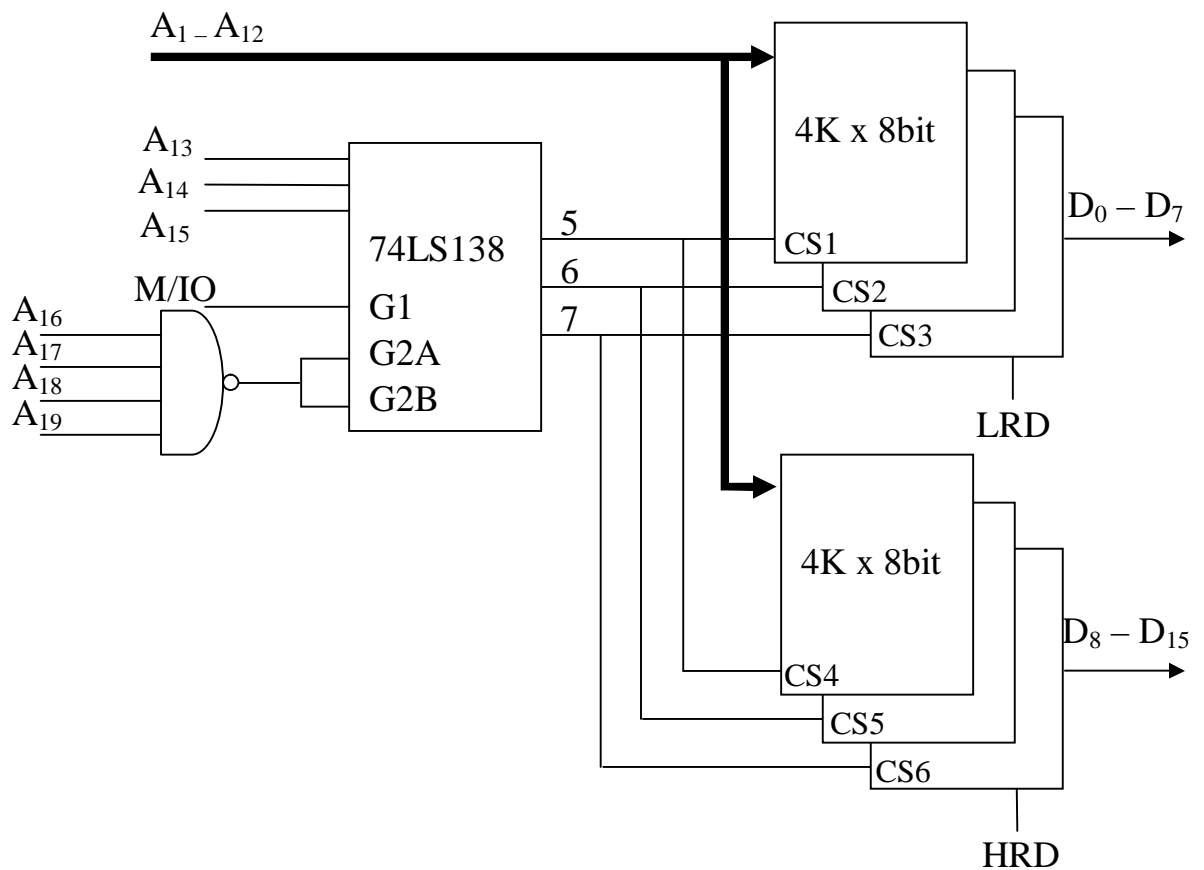
$A_{19}A_{18}A_{17}A_{16}$	$A_{15}A_{14}A_{13}$	$A_{12}A_{11} \dots \dots \dots A_1$	A_0
1 1 1 1	1 0 1	0 0 0	0
1 1 1 1	1 0 1	1 1 1	0
1 1 1 1	1 1 0	0 0 0	0
1 1 1 1	1 1 0	1 1 1	0
1 1 1 1	1 1 1	0 0 0	0
1 1 1 1	1 1 1	1 1 1	0
1 1 1 1	1 0 1	0 0 0	1
1 1 1 1	1 0 1	1 1 1	1

1	1	1	1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1	1	1

Địa chỉ của các IC :

- IC 1 : FA000 H – FBFFE H
- IC 2 : FC000 H – FDFFE H
- IC 3 : FE000 H – FFFFE H
- IC 4 : FA001 H – FBFFF H
- IC 5 : FC001 H – FDFFF H
- IC 6 : FE001 H – FFFFF H

⇒ Hình vẽ :



* RAM :

- Địa chỉ của IC 4 thuộc bank cao = $DS * 16 H + [BX + 3]$
 $= 0BFF0 H + 0011 H = 0C001 H$

⇒ Địa chỉ cuối của IC 4 là : 0FFFF H

- Do bit $A_{16}, A_{17}, A_{18}, A_{19} = 0$ (không thay đổi) nên không thể dùng bit này để chọn IC LS139

--> mỗi bank chỉ có tối đa 4 IC. Tổng cộng cần dùng 8 IC

⇒ $64K : 8 \text{ IC} = 8K$ --> loại SRAM là : $8K \times 8 \text{ bit}$.

$$8K \times 8\text{bit} \xrightarrow[8 \text{ IC}]{\text{đọc}} 64K \times 8\text{bit}$$

- Ta có : $8K = 2^3 \times 2^{10} = 2^{13}$

⇒ Bit dữ liệu : $A_1 - A_{13}$

Bit điều khiển : $A_{14} - A_{15}$

Từ địa chỉ của IC 4 thuộc bank cao ta tính ngược lại sẽ có địa chỉ bắt đầu của IC 1 thuộc bank cao là : 00001 H.

Vậy địa chỉ đầu của IC 1 thuộc bank thấp là : 00000 H

⇒ Dải địa chỉ làm việc của SRAM là : 00000 H – 0FFFF H

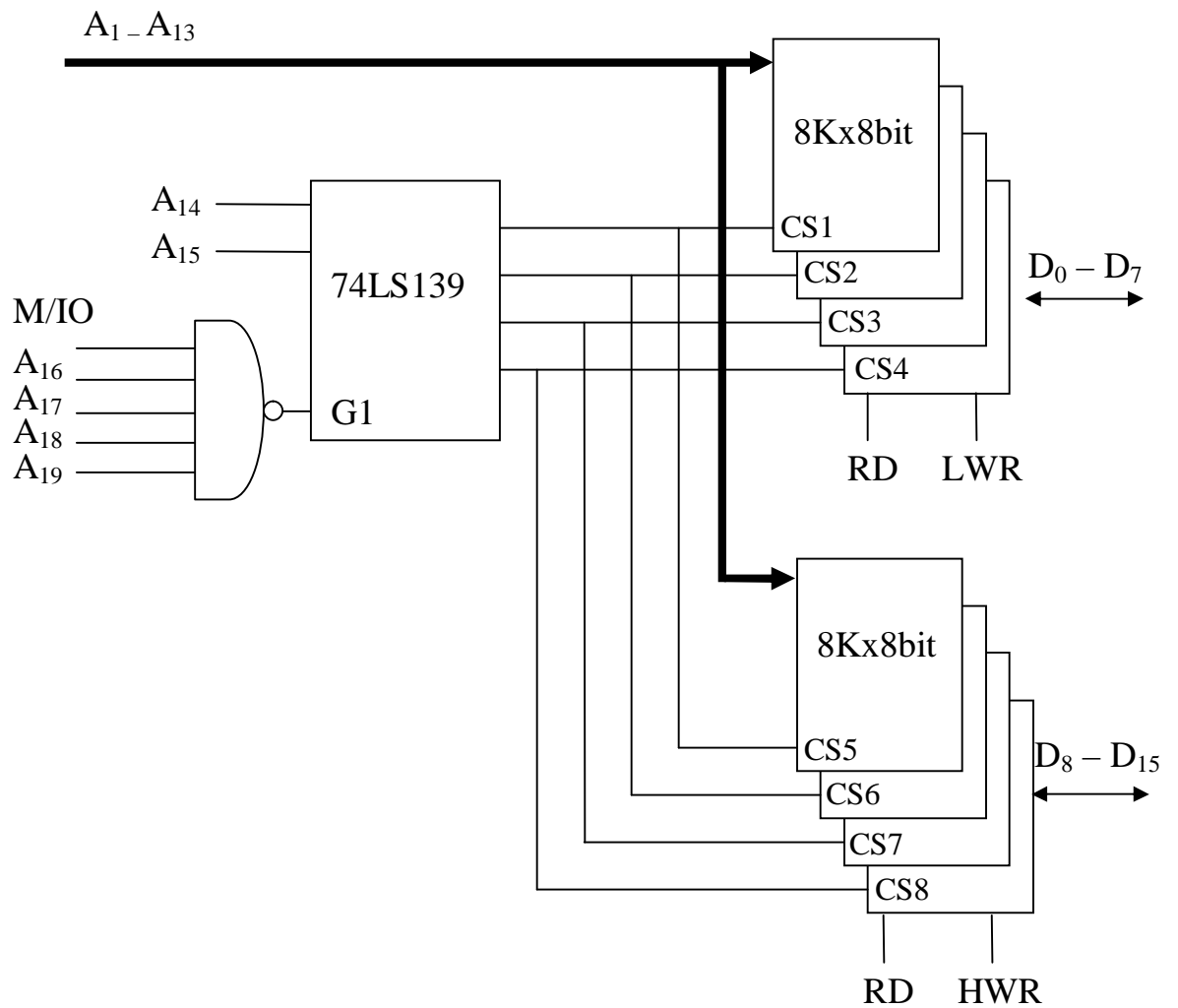
$A_{19}A_{18}A_{17}A_{16}$	$A_{15}A_{14}$	$A_{13}A_{12}A_{11} \dots A_1$	A_0
0 0 0 0	0 0	0 0 0 0	0
0 0 0 0	0 0	1 1 1 1	0
0 0 0 0	0 1	0 0 0 0	0
0 0 0 0	0 1	1 1 1 1	0
0 0 0 0	1 0	0 0 0 0	0
0 0 0 0	1 0	1 1 1 1	0
0 0 0 0	1 1	0 0 0 0	0
0 0 0 0	1 1	1 1 1 1	0
0 0 0 0	0 0	0 0 0 0	1
0 0 0 0	0 0	1 1 1 1	1
0 0 0 0	0 1	0 0 0 0	1
0 0 0 0	0 1	1 1 1 1	1
0 0 0 0	1 0	0 0 0 0	1
0 0 0 0	1 0	1 1 1 1	1
0 0 0 0	1 1	0 0 0 0	1
0 0 0 0	1 1	1 1 1 1	1

⇒ Địa chỉ của các IC :

- IC 1 : 00000 H – 03FFE H
- IC 2 : 04000 H – 07FFE H
- IC 3 : 08000 H – 0BFFE H
- IC 4 : 0C000 H – 0FFFE H
- IC 5 : 00001 H – 03FFF H

- IC 6 : 04001 H – 07FFF H
- IC 7 : 08001 H – 0BFFF H
- IC 8 : 0C001 H – 0FFFF H

è Hình :



Phần II : Lập trình ASM

Bài tập :

a. Hãy viết chương trình assembly cho 8086 thực hiện

1. Nhập số N ($N < 9$, nguyên dương) từ bàn phím

2. Tính $S = 1^2 + .. + N^2$

3. Kiểm tra tính chẵn lẻ của S

b. Hãy tối ưu chương trình vừa viết (nếu có thể) theo kích thước của chương trình trong bộ nhớ

Lời giải :

Cách giải :

Đầu tiên ta nhập 1 số N (< 9) vào từ bàn phím ! N được lưu trong thanh ghi AL

Thực hiện vòng lặp : để tính tổng S

S được lưu trong thanh ghi BX. Kết quả S ở dạng số Hexa (hệ 16)

Tiếp đó ta chuyển lại BX vào AX

Dùng thuật toán chuyển từ Hexa sang thập phân (1)

Các lệnh cơ bản thực hiện trong bài toán :

Sub Đích, Nguồn ; đích = đích – nguồn

ADD đích, nguồn ; đích = đích + nguồn

CMP đích, nguồn ; so sánh đích và nguồn

JNE ; jump if not equal

JB ; jump if below nhảy nếu nhỏ hơn

Mul thừa_số ; $AX = al * thừa_số$

Div số_bị_chia ; $AX = (DX AX) : số_bị_chia$, kết quả lưu trong ax số dư lưu trong dx

Hướng dẫn giải :

1. Nhập số N ($N < 9$, nguyên dương) từ bàn phím

Ta viết chương trình dịch ra file exe

;Allright reserve DTBK groupthank you for viewing this text

```

.model small
.stack 100h
.data
msg1 db 'nhap so N : $'
msg2 db 13,10,'Tong S la : $'
msg3 db 13,10,'S la le $'
msg4 db 13,10,'S la chan $'
N     db ?

```

;Đoạn này 13,10 là số thứ tự của CR(carrier return: về đầu dòng) LF (line feed : xuống dòng) trong bảng ASCII

```

.code
main proc
mov ax,@data
mov ds,ax
mov ah,9 ; ngắt 9 của 21h
lea dx,msg1; chỉ con trỏ tới chuỗi msg1
int 21h ; lúc này máy sẽ in ra chuỗi msg1

```

```

mov ah,1
int 21h
sub al,30h ; Đổi al sang số !! (2)

```

```

xor bx,bx ; bx =0
lap:
xor ah,ah
push ax ; cất ax từ đỉnh ngăn xếp
mul al
add bx,ax ; bx = bx + ax
pop ax ; lấy lại ax từ đỉnh ngăn xếp
dec al
cmp al,0 ; so sánh al và 0
jne lap ; nếu al ko bằng 0 sẽ nhảy về lap:

```

```

mov ah,9
lea dx,msg2
int 21h ; In chuỗi msg2 ra màn hình
; như vậy ta đã thực hiện xong việc tính tổng S , S được lưu trong thanh ghi BX
; ở dạng hexa . Bây giờ ta sẽ tiến hành chuyển S sang hệ thập phân và in ra màn

```



```
xor ax,ax ; ax = 0
mov ax,bx ; chuyển giá trị S vào ax

push ax ; cat ax vào đỉnh ngăn xếp
mov bx,100 ; dung 100 làm số bị chia
mov cx,1 ;
```

```
mov dx,0
begin_print:
```

```
cmp bx,0
jz end_print ; Nếu bx = 0 sẽ nhảy đến end_print:
```

```
cmp cx,0
je calc ; nếu cx=0 sẽ nhảy đến calc
```

```
cmp ax,bx ; so sánh ax và bx
jb giulai ; nếu ax < bx sẽ nhảy đến giữ lại
```

```
; nhãn này sẽ in ra màn hình từng số 1 của kết quả
calc:
```

```
mov cx,0
mov dx,0
div bx
push dx
add al,30h
xor dx,dx
mov dl,al
mov ah,2
int 21h
pop dx
mov ax,dx
```

```
; nhãn này sẽ thực hiện nhiệm vụ giảm bớt bx đi 10 lần vì bx>ax nên ko chia đc
; lúc đầu bx được gán là 100 ( lưu trong thanh ghi bx như sau BH:00h,BL:64
;Sở dĩ ta chỉ gán bx là 100 vì S lưu trong AX chỉ lưu ở AL vì biết trước N=9 thì
;Smax cũng chỉ tầm khoảng 300 -> ko quá 8 bit
;nếu S lớn hơn thì ta gán BX là 1000,10000 ...
```

```
giulai:
push ax
mov dx,0
```

```
mov ax,bx
div cs:muoi ; chia bx cho 10
mov bx,ax ; gan ket qua tro lai bx
pop ax ; lay lai ax tu dinh ngan xep
jmp begin_print
```

muoi dw 10 ;định nghĩa mười -> bắt buộc phải để ở đây , ko đc đưa lên đầu

*;nhân này sẽ kết thúc chương trình khi bx giảm về 0 sau khi thực hiện nhiều
;vòng giulai:*

end_print:

pop ax

pop cx

pop bx

pop dx

mov ah,4ch

int 21h ; thoát về DOS

main endp

END MAIN

Phần kiểm tra tính chẵn lẻ dùng lệnh `div` , số dư sẽ lưu trong `dx`, các bạn thử làm xem

Đề 2 cũng tương tự đề 1 , bài tập bất tính $S = 1 + 2 + \dots + N$

Hy vọng rằng các bạn có thể làm được bài này

Chúc các bạn học và thi tốt

DTBK Group
scorpion & thanhbobo

“If you want to thank us , just go to : <http://dientubachkhoa.com> “

(1) : Thuật toán này tham khảo trong ví dụ Caculator.asm của soft Emulator8086

(2) : Al khi đó là số thự của ASCII , từ 0 -> 9 lần lượt là 30H -> 39H nên ta trừ đi 30H sẽ ra số 0 -> 9

ĐỀ THI KỸ THUẬT VI XỬ LÝ

DT 7, 8, 9, 10, 11, 12- K47

Ngày thi: 21/12/2005

Thời gian làm bài: 90 phút

(Không được sử dụng tài liệu)

Đề 1:

Câu 1 (2 điểm):

Vi xử lý có thể được phân loại như thế nào? Trình bày đặc điểm chung và khuynh hướng phát triển của từng loại.

Câu 2 (3 điểm)

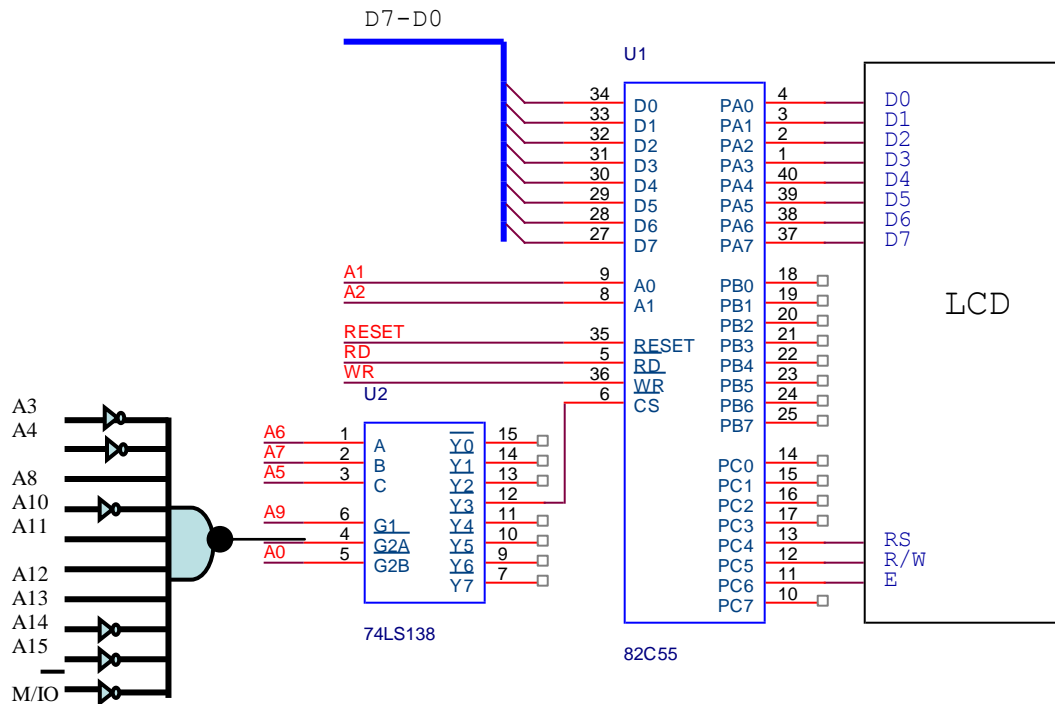
Ghép nối 8086 với bộ nhớ và thiết bị ngoại vi sau, sử dụng bộ giải mã 74LS138:

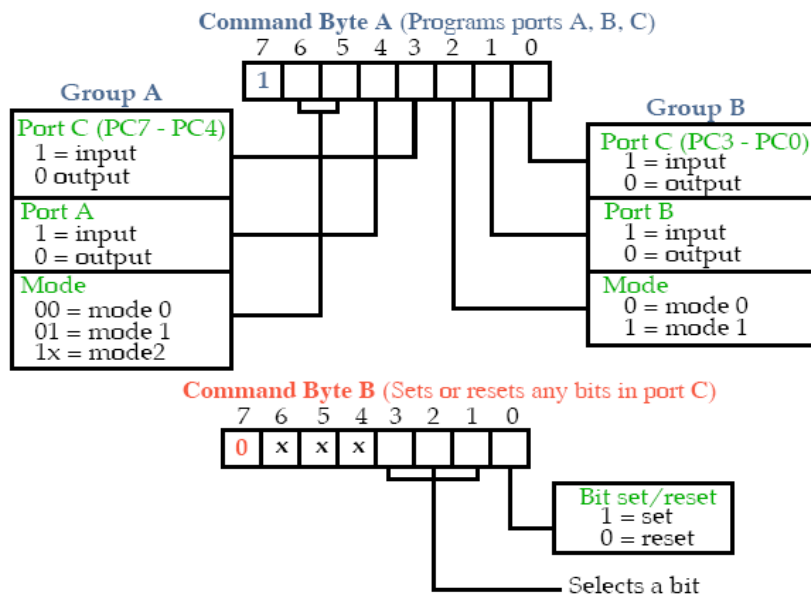
- 192 KB bộ nhớ SRAM sử dụng SRAM 62256 (32K*8), bắt đầu từ địa chỉ 30000H.
- Bộ nhớ EPROM có dải địa chỉ 80000H-9FFFFH sử dụng EPROM 2732 (4K * 8)
- 8 cổng vào ra 8 bit tại các địa chỉ B1H, B3H, B5H, B7H, B9H, BBH, BDH, BFH

Câu 3 (3 điểm)

Cho sơ đồ mạch ghép nối 8086 với 82C55 để điều khiển module LCD như hình vẽ.

- a) Xác định địa chỉ của các cổng và thanh ghi lệnh của 82C55
- b) Viết chương trình hợp ngữ để:
- lập trình cho 82C55 làm việc ở chế độ 0 để điều khiển LCD (xem mô tả ở sau)
 - khởi tạo LCD bằng thủ tục LCD_Init và ghi một ký tự lên LCD bằng thủ tục LCD_Write
 - Hiển thị dòng chữ “Hello World” sử dụng thủ tục LCD_Write. Biết rằng LCD sử dụng bảng ASCII để mã hoá font.





Mô tả các chân của LCD:

-D0-D7 là 8 bit dữ liệu

-RS: tín hiệu chọn thanh ghi (Register Select), RS=0: thanh ghi lệnh, RS=1 thanh ghi dữ liệu

-R/W: chọn chế độ đọc LCD hoặc ghi ra LCD: R/W=0: ghi ra LCD, R/W=1: đọc LCD

-E (Enable): dữ liệu hoặc lệnh sẽ được ghi ra LCD khi E chuyển từ mức cao xuống mức thấp

Các bước khởi tạo LCD:

1. R/W=0 ; chọn chế độ ghi
2. RS=0;
3. E=1;
4. trễ 1ms
5. Đưa dữ liệu 38H ra LCD
6. trễ 1ms
7. E=0;
8. E=1;
9. trễ 1ms
10. Đưa dữ liệu 0CH ra LCD
11. trễ 1ms
12. E=0;
13. E=1;
14. trễ 1ms
15. Đưa dữ liệu 06H ra LCD
16. trễ 1ms
17. E=0;
18. E=1;
19. trễ 1ms
20. Đưa dữ liệu 1CH ra LCD
21. trễ 1ms
22. E=0;

Các bước viết một ký tự ra LCD:

1. RS=1;
2. trễ 1ms
3. E=1;
4. trễ 1ms
5. Đưa ký tự cần ghi ra LCD
6. trễ 1ms
7. E=0;
8. RS=0;
9. trễ 1 ms

ĐỀ THI KỸ THUẬT VI XỬ LÝ

ĐT 1-12- K49

Ngày thi: 15/06/2007

Thời gian làm bài: 90 phút

(Không được sử dụng tài liệu)

Đề 1:

Câu 1 (2 điểm): Trả lời 4 trong 6 câu sau:

1. So sánh SRAM và DRAM
2. Trình bày và cho ví dụ về chế độ địa chỉ tức thì
3. Anh/chị hiểu thế nào về các thông tin: **Pentium D925 - 3.0 GHz - 4MB - 64 bit - Dual Core - bus 800 - SK 775** trong một báo giá vi xử lý
4. Trình bày về DMA
5. Trình bày các tiêu chí khi lựa chọn vi xử lý cho việc thiết kế một máy ảnh kỹ thuật số
6. Trình bày ưu điểm của công nghệ vi xử lý đa lõi

Câu 2 (2 điểm)

Cho IC RAM 32K x 4 (một ngăn nhớ 4 bit dữ liệu) và IC ROM 16K x 8 (một ngăn nhớ 8 bit dữ liệu)

Hãy dùng các IC trên ghép nối với VXL 8086 để tạo thành hệ thống nhớ 192KB RAM và 160KB ROM. Biết rằng địa chỉ bắt đầu của các IC RAM là 90000h, địa chỉ cuối cùng của các IC ROM là FFFFFh

Câu 3 (3 điểm)

Viết chương trình hợp ngữ thông báo cho người sử dụng gõ vào một ký tự và in ra mã ASCII của ký tự dưới dạng hex ở dòng tiếp theo. Lặp lại cho đến khi người sử dụng gõ ENTER.

Ví dụ: Hãy gõ vào một ký tự: Z

Mã ASCII của Z dưới dạng hex là: 5A

ĐỀ THI KỸ THUẬT VI XỬ LÝ

ĐT 1-12- K49

Ngày thi: 15/06/2007

Thời gian làm bài: 90 phút

(Không được sử dụng tài liệu)

Đề 2:

Câu 1 (2 điểm): Trả lời 4 trong 6 câu sau:

1. So sánh vào ra bằng ngắt (Interrupt) và vào ra theo kiểu hỏi vòng (polling)
2. Trình bày và cho ví dụ về chế độ địa chỉ gián tiếp qua thanh ghi
3. Anh/chị hiểu thế nào về các thông tin: **Core 2 Duo E4300 - 1.8 GHz - 2MB - 64 bit - bus 800 - SK 775** trong một báo giá vi xử lý
4. Trình bày về kỹ thuật xử lý pipeline
5. Trình bày ưu điểm của công nghệ vi xử lý đa lõi
6. Trình bày nguyên lý quản lý bộ nhớ ở chế độ thực (real mode)

Câu 2 (2 điểm)

Cho IC RAM 32K x 8 (một ngăn nhớ 8 bit dữ liệu) và IC ROM 16K x 4 (một ngăn nhớ 4 bit dữ liệu)

Hãy dùng các IC trên ghép nối với VXL 8086 để tạo thành hệ thống nhớ 192KB RAM và 160KB ROM. Biết rằng địa chỉ bắt đầu của các IC RAM là 90000h, địa chỉ cuối cùng của các IC ROM là FFFFFh

Câu 3 (3 điểm)

Viết chương trình hợp ngữ thông báo cho người sử dụng gõ vào một ký tự và in ra màn hình số bit 1 trong mã ASCII dưới dạng nhị phân của ký tự đó ở dòng tiếp theo. Lặp lại cho đến khi người sử dụng gõ ENTER.

Ví dụ: Hãy gõ vào một ký tự: A

Số bit 1 trong mã ASCII dưới dạng nhị phân của A là: 2

ĐỀ THI KỸ THUẬT VI XỬ LÝ

ĐT 1-12- K49

Ngày thi: 15/06/2007

Thời gian làm bài: 90 phút

(Không được sử dụng tài liệu)

Đề 3:

Câu 1 (2 điểm): Trả lời 4 trong 6 câu sau:

1. So sánh hợp ngữ và ngôn ngữ lập trình bậc cao
2. Trình bày và cho ví dụ về chế độ địa chỉ trực tiếp
3. Anh/chị hiểu thế nào về các thông tin: **Pentium 4 641 - 3.2 GHz - 2MB - 64 bit - bus 800MHz - HT - SK 775** trong một báo giá vi xử lý
4. So sánh CISC và RISC
5. Trình bày lý do và nguyên lý của việc làm tươi DRAM
6. Trình bày ưu điểm của công nghệ vi xử lý đa lõi

Câu 2 (2 điểm)

Cho IC RAM 16K x 8 (một ngăn nhớ 8 bit dữ liệu) và IC ROM 8K x 4 (một ngăn nhớ 4 bit dữ liệu)

Hãy dùng các IC trên ghép nối với VXL 8086 để tạo thành hệ thống nhớ 192KB RAM và 48KB ROM. Biết rằng địa chỉ bắt đầu của các IC RAM là 40000h, địa chỉ cuối cùng của các IC ROM là FFFFFh

Câu 3 (3 điểm)

Viết chương trình hợp ngữ thông báo cho người sử dụng gõ vào một ký tự và in ra mã ASCII của ký tự dưới dạng hex ở dòng tiếp theo. Lặp lại cho đến khi người sử dụng gõ ENTER.

Ví dụ: Hãy gõ vào một ký tự: Z

Mã ASCII của Z dưới dạng hex là: 5A

ĐỀ THI KỸ THUẬT VI XỬ LÝ

ĐT 1-12- K49

Ngày thi: 15/06/2007

Thời gian làm bài: 90 phút

(Không được sử dụng tài liệu)

Đề 4:

Câu 1 (2 điểm): Trả lời 4 trong 6 câu sau:

1. So sánh ngắt cứng và ngắt mềm
2. Trình bày và cho ví dụ về chế độ địa chỉ tương đối cơ sở
3. Anh/chị hiểu thế nào về các thông tin: **Core 2 Duo E6320 - 1.86 GHz - 4MB - 64 bit - bus 1066 - SK 775** trong một báo giá vi xử lý
4. Số nguyên có dấu được biểu diễn thế nào trong máy tính? Cho ví dụ minh họa.
5. Trình bày đặc điểm chính của 4 thế hệ máy tính điện tử
6. Trình bày ưu điểm của công nghệ vi xử lý đa lõi

Câu 2 (2 điểm)

Cho IC RAM 16K x 4 (một ngăn nhớ 4 bit dữ liệu) và IC ROM 8K x 8 (một ngăn nhớ 8 bit dữ liệu)

Hãy dùng các IC trên ghép nối với VXL 8086 để tạo thành hệ thống nhớ 192KB RAM và 48KB ROM. Biết rằng địa chỉ bắt đầu của các IC RAM là 40000h, địa chỉ cuối cùng của các IC ROM là FFFFFh

Câu 3 (3 điểm)

Viết chương trình hợp ngữ thông báo cho người sử dụng gõ vào một ký tự và in ra màn hình số bit 1 trong mã ASCII dưới dạng nhị phân của ký tự đó ở dòng tiếp theo. Lặp lại cho đến khi người sử dụng gõ ENTER.

Ví dụ: Hãy gõ vào một ký tự: A

Số bit 1 trong mã ASCII dưới dạng nhị phân của A là: 2

Giải bài tập Vi xử lý – BachDH – TTM K51 v1.0

Phần 1: Kỹ thuật vi xử lý

- 1.** Bộ vi xử lý 8088 được khởi tạo đoạn ngăn xếp tại SS = 4200H. Giả thiết tại một thời điểm BP=00FCH, BX=1234H, AX= 2006H, CX= 5566H, SP=0100H đang trỏ vào đỉnh ngăn xếp. Hãy tính địa chỉ và nội dung các byte nhớ trong ngăn xếp sau các lệnh sau:

```
PUSH  AX
PUSH  BX
POP   DX
PUSH  CX
MOV   DL,[BP]
POP   SI
```

Giải:

Coi độ lớn Stack là 1 byte và lưu trữ theo Little endian.

+ Ban đầu:

	Địa chỉ	Nội dung
SP →	0100H	

SS →	4200H	

+ PUSH AX

SP →	00FEH	06H
	00FFH	20H
	0100H	

SS →	4200H	

+ PUSH BX

SP →	00FCH	34H
	00FDH	12H
	00FEH	06H

SS →	00FFH	20H
	0100H	

	4200H	

+ POP DX

SP →		
	00FEH	06H
	00FFH	20H
	0100H	
SS →
	4200H	

+ PUSH CX

SP →	00FCH	66H
	00FDH	55H
	00FEH	06H
	00FFH	20H
	0100H	
SS →
	4200H	

+ MOV DL,[BP] – ngăn xếp không thay đổi

SP →	00FCH	66H
	00FDH	55H
	00FEH	06H
	00FFH	20H
	0100H	
SS →
	4200H	

+ POP SI

SP →		
	00FEH	06H
	00FFH	20H

SS →	0100H	

	4200H	

2. Bộ vi xử lý 8088 được khởi tạo đoạn dữ liệu tại DS = 4200H. Giả thiết tại một thời điểm BX=ABF8H, SI=4E5CH, DI= 13C2H. Hãy tính địa chỉ toán hạng nguồn của các lệnh sau :

- a) MOV AL,[BX]+5
- b) ADD AL,[3A5CH]
- c) MOV CL,[DI+7]
- d) MOV BX,[SI+BX+0FH]

Giải:

- a) Chế độ địa chỉ tương đối cơ sở
 $4200H:ABFDH = 4CBFDH$
- b) Chế độ địa chỉ trực tiếp
 $4200H:3A5CH = 45A5CH$
- c) Chế độ địa chỉ tương đối chỉ số
 $4200H:13C9H = 433C9H$
- d) Chế độ địa chỉ tương đối chỉ số cơ sở
 $4200H:FA63H = 51A63H$

3. Cho nội dung các thanh ghi trong của 8088 như sau.

AX= 94B3H ; BX=5AE4H ; CX= A4B7H ; DX= EA8DH. Hãy cho biết kết quả các phép toán sau và nội dung các cờ CF,OF,ZF sau mỗi phép toán

a) ADD AX,BX

b) SUB BX,CX

c) ADD CL,DL

d) OR BH,AL

Giải:

a) Kết quả:

ADD	94B3H	1001 0100 1011 0011
	5AE4H	0101 1010 1110 0100
AX =	EF97H	1110 1111 1001 0111

Cờ: CF = 0, OF = 0, ZF = 0

b) Kết quả:

	A4B7H	1010 0100 1011 0111
	bù 1	0101 1011 0100 1000
	bù 2	0101 1011 0100 1001
ADD	5AE4H	0101 1010 1110 0100
BX =	B62DH	1011 0110 0010 1101

Cờ: CF = 0, OF = 1, ZF = 0

c) Kết quả:

ADD	B7H	1011 0111
	8DH	1000 1101
CL =	44H	0100 0100

Cờ: CF = 1, OF = 1, ZF = 0

d) Kết quả:

OR	5AH	0101 1010
	B3H	1011 0011
BH =	FBH	1111 1011

Cờ: CF = 0, OF = 0, ZF = 0

4. a) Hãy sử dụng các mạch giải mã 1/4, các mạch logic, các vi mạch EPROM 512B thiết kế bộ nhớ 2kB đặt địa chỉ cuối cùng là FFFFFH .

b) Liệt kê địa chỉ của từng vi mạch EPROM.

Giải:

Dung lượng EPROM = 512B = 2^9 B = 200H \rightarrow 9 chân địa chỉ (A0 – A8)

Dung lượng bộ nhớ = 2kB = 2^{11} B = 4.2⁹B = 800H \rightarrow phải dùng 4 vi mạch EPROM 512B

Địa chỉ đầu của bộ nhớ = Địa chỉ cuối của bộ nhớ - (Dung lượng bộ nhớ - 1)
= FFFFFH – (800H – 1) = FF800H

Mỗi vi mạch EPROM có dung lượng 200H nên địa chỉ của từng vi mạch EPROM là:

IC 1: Địa chỉ đầu = FF800H

Địa chỉ cuối = FF9FFH

IC2: Địa chỉ đầu = FFA00H

Địa chỉ cuối = FFBFFH

IC3: Địa chỉ đầu = FFC00H

Địa chỉ cuối = FFDFFH

IC4: Địa chỉ đầu = FFE00H

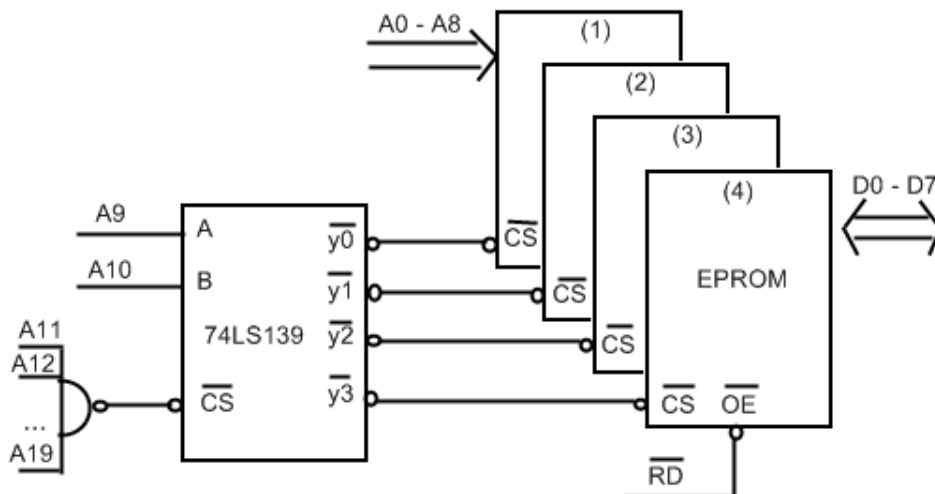
Địa chỉ cuối = FFFFFH

Sơ đồ địa chỉ:

	A ₁₉	A ₁₈	A ₁₇	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	IO/ \overline{M}
IC 1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	
IC 2	1	1	1	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	
	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	
IC 3	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	
	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	
IC 4	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
	địa chỉ mô đun nhớ									địa chỉ IC	nối // add bus										

Chọn mạch giải mã 1/4 (74LS139)

Sơ đồ ghép nối:



5. Thiết kế bộ nhớ dung lượng 24KB từ các vi mạch ROM 8KB và bộ giải mã 74138 (1/8) ghép nối với bộ vi xử lý 8088 (Chế độ MIN) với địa chỉ đầu từ AA000H.

Giải:

Dung lượng ROM = 8KB = 2^{13} B = 2000H \rightarrow 13 chân địa chỉ (A0 – A12)

Dung lượng bộ nhớ = 24KB = 6000H \rightarrow cần dùng 3 vi mạch ROM để thiết kế

Địa chỉ đầu bộ nhớ = AA000H

Địa chỉ cuối bộ nhớ = Địa chỉ đầu + (Dung lượng – 1)
 $= \text{AA000H} + 6000\text{H} - 1 = \text{AFFFFH}$

Mỗi vi mạch ROM có dung lượng 2000H nên địa chỉ đầu và cuối của mỗi vi mạch là:

IC1: Địa chỉ đầu = AA000H
 Địa chỉ cuối = ABFFFH

IC2: Địa chỉ đầu = AC000H
 Địa chỉ cuối = ADFFFH

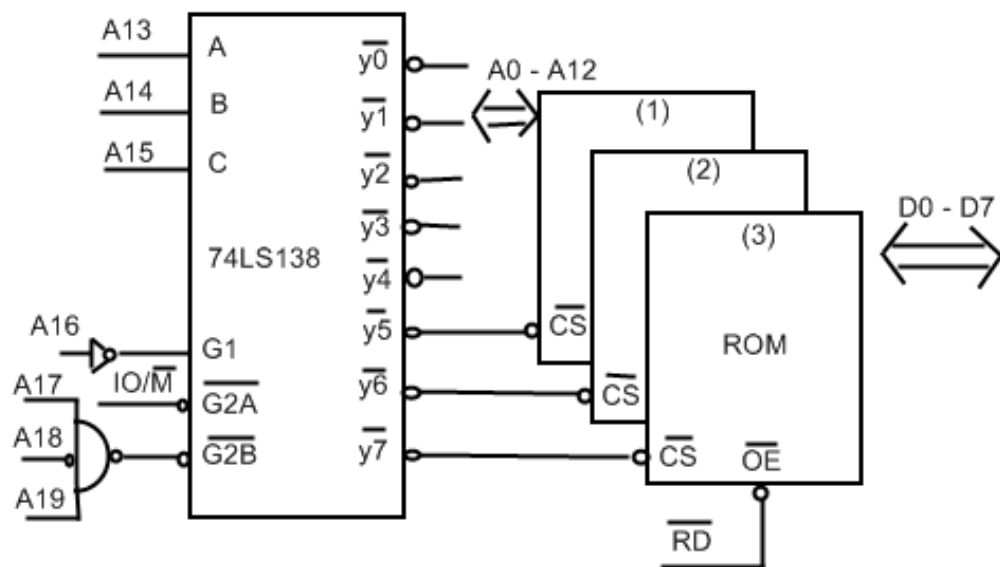
IC3: Địa chỉ đầu = AE000H
 Địa chỉ cuối = AFFFFH

Sơ đồ địa chỉ:

	A ₁₉	A ₁₈	A ₁₇	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	IO/ $\overline{\text{M}}$
IC 1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
IC 2	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	1	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	
IC 3	1	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
	địa chỉ mô đun nhớ				địa chỉ IC			nối // add bus													

Chọn mạch giải mã 1/8 (74LS138)

Sơ đồ nối:



6. Thiết kế mạch giải mã địa chỉ cho 8 cổng ra có địa chỉ 3A8H – 3AFH dùng các mạch giải mã 74139 (1/4).

Giải:

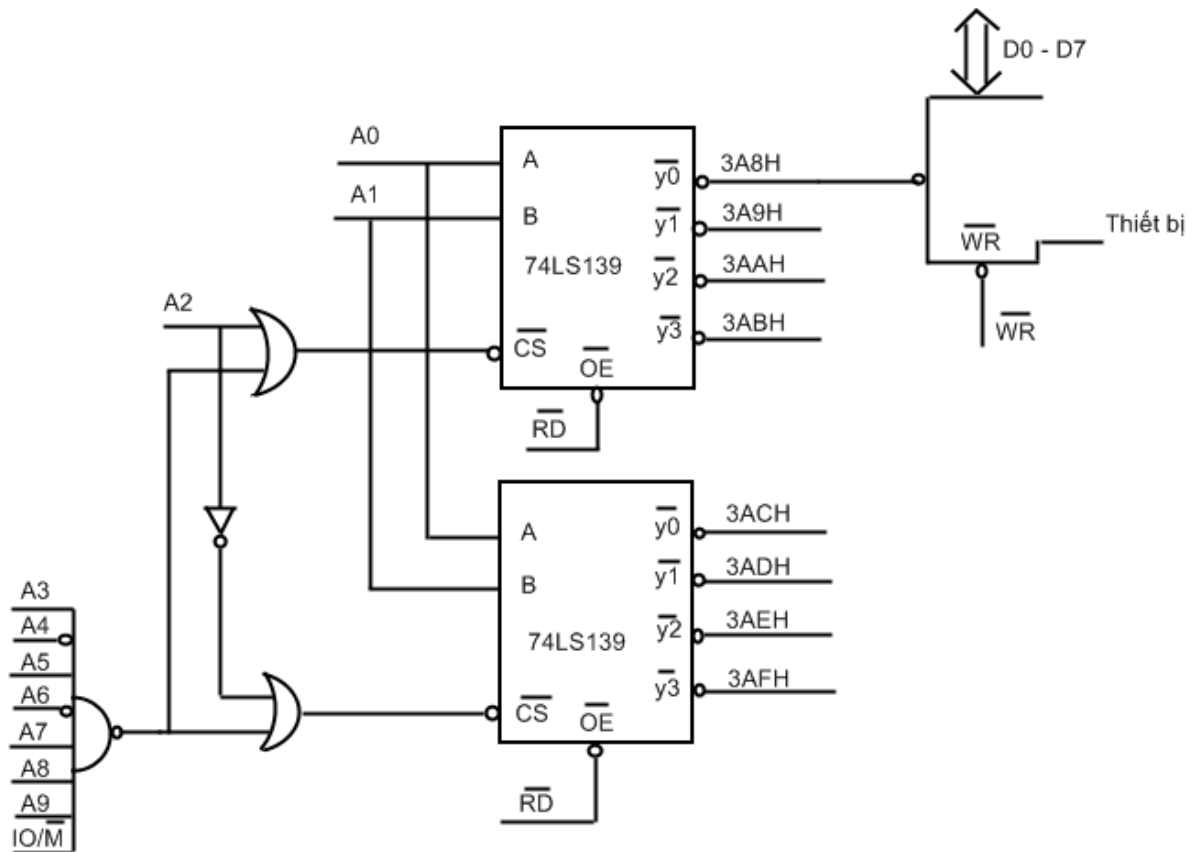
Mỗi mạch giải mã 1/4 có 4 cổng ra \rightarrow cần dùng 2 mạch giải mã 1/4 để thiết kế.

Sơ đồ địa chỉ:

	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	IO/ \overline{M}
3A8H	1	1	1	0	1	0	1	0	0	0	1
3A9H	1	1	1	0	1	0	1	0	0	1	
3AAH	1	1	1	0	1	0	1	0	1	0	
3ABH	1	1	1	0	1	0	1	0	1	1	
3ACH	1	1	1	0	1	0	1	1	0	0	
3ADH	1	1	1	0	1	0	1	1	0	1	
3AEH	1	1	1	0	1	0	1	1	1	0	
3AFH	1	1	1	0	1	0	1	1	1	1	
								Địa chỉ cổng			

Chọn mạch giải mã 1/4 (74LS139)

Sơ đồ nối:



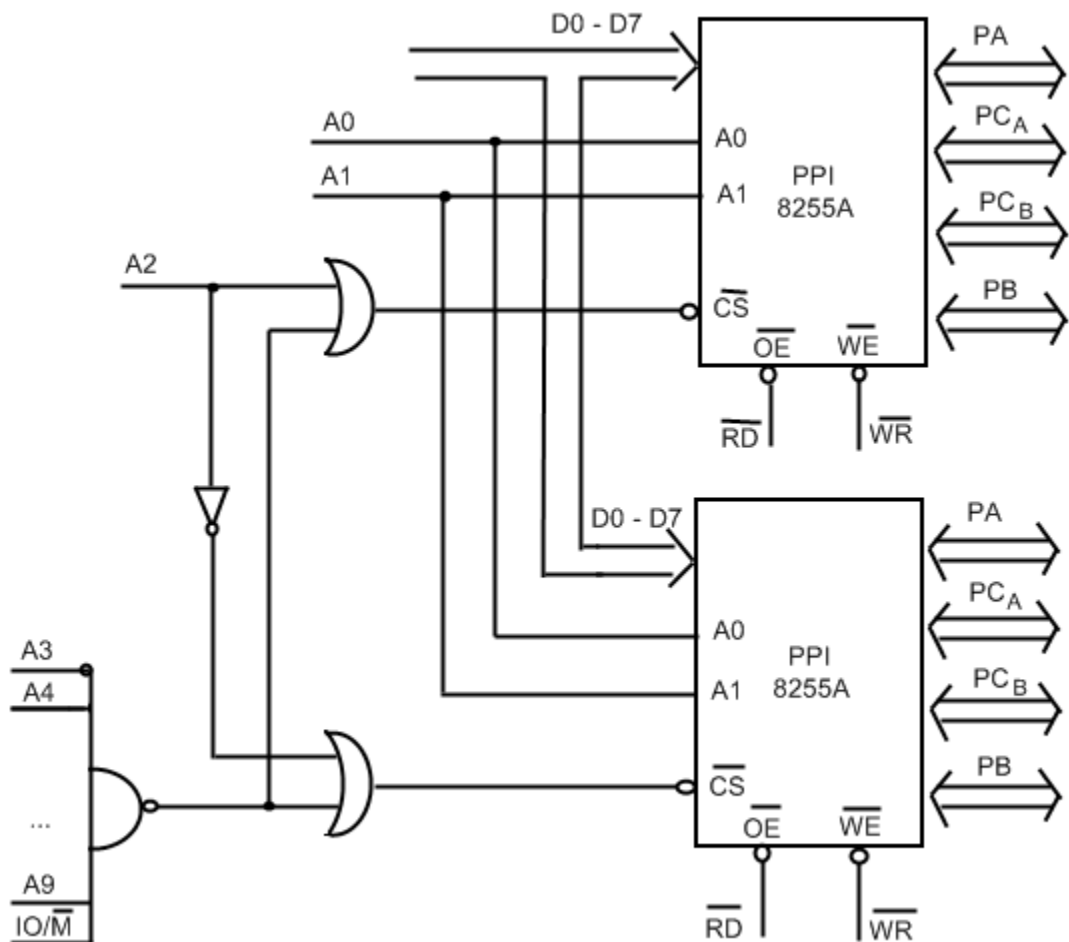
7. Hệ vi xử lý 8088 có 2 vi mạch cổng PPI 8255A (Mỗi vi mạch có 4 địa chỉ). Hãy thiết kế mạch giải mã địa chỉ biết địa chỉ cơ sở của vi mạch 1 là 2B0H còn vi mạch 2 là 2B4H.

Giải:

Sơ đồ địa chỉ:

	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	IO/ \overline{M}
2B0H	1	0	1	0	1	1	0	0	0	0	1
2B1H	1	0	1	0	1	1	0	0	0	1	
2B2H	1	0	1	0	1	1	0	0	1	0	
2B3H	1	0	1	0	1	1	0	0	1	1	
2B4H	1	0	1	0	1	1	0	1	0	0	
2B5H	1	0	1	0	1	1	0	1	0	1	
2B6H	1	0	1	0	1	1	0	1	1	0	
2B7H	1	0	1	0	1	1	0	1	1	1	
								Địa chỉ cổng			

Sơ đồ nối:



8. Cho vi mạch cổng 8255A có 4 cổng PA, PC_A, PB, PC_B. Hãy Viết đoạn chương trình đặt cấu hình cho các cổng ở mode 0 như sau: PA, PC_A là cổng vào, còn PB, PC_B là cổng ra.
Giải:

Vì ở mode 0 nên $\rightarrow MA_1MA_0 = 00$
 $MB = 0$

Vì PA, PC_A là cổng vào $\rightarrow A = C_A = 1$

Vì PB, PC_B là cổng ra $\rightarrow B = C_B = 0$

Vậy từ điều khiển là:

IO/ \overline{M}	MA ₁	MA ₀	A	C _A	MB	B	C _B
1	0	0	1	1	0	0	0

\rightarrow Từ điều khiển = 98H

Đoạn chương trình đặt cấu hình cho các cổng:

Định nghĩa các hằng:

CW EQU 98H

Cấu hình cho 8255A:

```
MOV DX,CWR    ;đưa CWR vào DX
MOV AL,CW     ;từ điều khiển chứa trong AL
OUT DX,AL     ;đưa từ điều khiển ra CWR
```

Phần 2: Lập trình hợp ngữ

A. Giải thích chương trình

1. Cho đoạn chương trình hợp ngữ sau, hãy giải thích từng lệnh (theo ngữ cảnh) và cho biết kết quả trên màn hình.

```
MOV BH,41H      ;Gán BH=41H (kí tự 'A')
MOV BL,4         ;Gán BL=4
MOV AH,2         ;Gán AH=2
```

LAP:

```
MOV CX,5         ;Gán CX=5
```

LAI:

```
MOV DL,BH        ;Chuyển nội dung BH vào DL
INT 21H          ;Gọi ngắt 21H, in ra màn hình kí tự trong DL(=BH)
LOOP LAI         ;Lặp lại việc in ra màn hình kí tự trong DL 5 lần (CX=5)
MOV DL,'_'       ;Gán DL= kí tự '_'
INT 21H          ;Gọi ngắt 21H 3 lần, in ra màn hình 3 kí tự '_' liên tiếp
INT 21H
INT 21H
INC BH           ;Tăng BH 1
DEC BL           ;Giảm BL 1
JNZ LAP          ;Nhảy tới nhãn LAP nếu cờ ZF chưa bật
```

Kết quả trên màn hình:

AAAAA_BBBBBB_CCCCC_DDDDD

2. Giải thích từng lệnh và viết kết quả trên màn hình của đoạn chương trình sau

```
MOV BH,48        ;Gán BH=48 (kí tự '0')
MOV BL,4         ;Gán BL=4
```

FO1:

```
MOV CX,5         ;Gán CX=5
MOV AH,2         ;Gán AH=2
```

FO2:

```
MOV DL,BH        ;Chuyển nội dung BH vào DL
INT 21H          ;Gọi ngắt 21H, in ra màn hình kí tự trong DL (=BH)
INC DL           ;Tăng DL 1
LOOP FO2         ;Lặp lại việc in ra màn hình kí tự trong DL 5 lần
MOV DL,0AH       ;Gán DL=0AH (LF)
INT 21H          ;Gọi ngắt 21H, in kí tự điều khiển LF (tạo một dòng mới)
MOV DL,0DH       ;Gán DL=0DH (CR)
INT 21H          ;Gọi ngắt 21H, in kí tự điều khiển CR (về đầu dòng)
INC BH           ;Tăng BH 1
DEC BL           ;Giảm BL 1
JNZ FO1          ;Nhảy tới nhãn FO1 nếu cờ ZF chưa bật
```

Kết quả trên màn hình:

00000

11111

22222

33333

3. Cho đoạn chương trình hợp ngữ sau.

```
.....  
.DATA  
    M1 DB 0Bh,10h,15h,2DUP(?)  
    B1 DB 10  
.CODE  
Thu  proc  
.....  
    MOV  AL,B1      ;AL = B1 = 10 (= 0Ah)  
    LEA  BX,M1      ;BX trở tới M1  
    ADD  [BX+2],AL   ;[BX+2] += AL (= M1[2] = 15h + 0Ah = 1Fh)  
    ADD  AL,[BX]+1   ;AL += [BX+1] (= 0Ah + 10h = 1Ah)  
    MOV  BX,3        ;BX = 3  
    MOV  M1[BX],AL   ;M1[3] = AL = 1Ah  
    SUB  B1,10h      ;B1 -= 0Ah - 10h = FAh  
    MOV  AL,B1       ;AL = B1 = FAh  
    INC  BX          ;BX += 1 = 4  
    MOV  M1[BX],AL   ;M1[4] = AL = FAh  
.....
```

Hãy cho biết giá trị mới của mảng M1 sau các lệnh trên.

M1: 0Bh,10h,1Fh,1Ah,FAh

4. Cho đoạn chương trình hợp ngữ sau.

.....
.DATA

M1 DB 't','h','i','l','a','i',2 DUP('*')

B1 DB 20H

.CODE

.....
MOV AL,B1 ;Gán AL = B1 = 20H

LEA BX,M1 ;BX trở tới M1

MOV CX,6 ;Gán CX = 6

LAP:

SUB [BX],AL ;[BX] -= AL = [BX] – 20H (chữ thường thành chữ hoa)

ADD BX,1 ;BX += 1

LOOP LAP ;Lặp lại 6 lần

.....
MOV AH,2 ;Gán AH = 2

MOV BX,0 ;Gán BX = 0

MOV CX,8 ;Gán CX = 8

LAI:

MOV DL,M1[BX] ;Gán DL = M1[BX]

INT 21H ;In ra màn hình kí tự trong DL

INC BX ;Tăng BX 1

LOOP LAI ;Lặp lại quá trình trên 8 lần (CX = 8)

Hãy cho biết giá trị mới của mảng M1 sau các lệnh trên và kết quả trên màn hình.

M1: 'T','H','I','L','A','I','*','*'

Kết quả trên màn hình:

THILAI**

5. Giải thích từng lệnh và viết kết quả trên màn hình của đoạn chương trình sau

MOV BH,41H ;Gán BH = 41H (kí tự 'A')

MOV BL,4 ;Gán BL = 4

MOV AH,2 ;Gán AH = 2

LAP:

MOV CX,5 ;Gán CX = 5

LAI:

MOV DL,BH ;Chuyển nội dung BH vào DL

INT 21H ;Gọi ngắt 21H, in ra màn hình kí tự trong DL

OR DL,20H ;DL OR 20H (chuyển chữ hoa thành chữ thường)

INT 21H ;Gọi ngắt 21H, in ra màn hình kí tự trong DL

LOOP LAI ;Lặp lại quá trình trên 5 lần (CX = 5)

MOV DL,'_' ;Gán DL = '_'

INT 21H ;Gọi ngắt 21H 3 lần, in ra 3 lần kí tự '_'

INT 21H

INT 21H

INC BH ;Tăng BH 1

DEC BL ;Giảm BL 1

JNZ LAP ;Nhảy tới nhãn LAP nếu cờ ZF bật

.....

Kết quả trên màn hình:

AaAaAaAaAa__BbBbBbBbBb__CcCcCcCcCc__DdDdDdDdDd__

6. Giải thích từng lệnh và viết kết quả trên màn hình của đoạn chương trình sau

MOV BH,0AH	;Gán BH = 0AH (kí tự điều khiển LF)
MOV DL,39H	;Gán DL = 39H (kí tự '9')
ADD DL,BH	;DL += BH = 39H + 0AH = 43H (kí tự 'C')
MOV CX,5	;Gán CX = 5
MOV AH,2	;Gán AH = 2

LAP:

INT 21H	;Gọi ngắt 21H, in ra màn hình kí tự trong DL
LOOP LAP	;Lặp lại quá trình trên 5 lần (CX = 5)
XCHG BH,DL	;Đổi chỗ giá trị trong BH và DL (BH = 43H, DL = 0AH)
INT 21H	;Gọi ngắt 21H, in ra màn hình kí tự trong DL (kí tự LF)
MOV DL,0DH	;Gán DL = 0DH (kí tự điều khiển CR)
INT 21H	;Gọi ngắt 21H, in ra màn hình kí tự trong DL (kí tự CR)
MOV CX,8	;Gán CX = 8

MA:

XOR DL,DL	;DL XOR DL (Xóa DL = 0)
ROL BH,1	;Quay trái BH 1, CF = MSB
RCL DL,1	;Quay trái DL qua cờ CF 1, LSB = CF
ADD DL,30H	;DL += 30H (đổi số thành mã ASCII)
INT 21H	;Gọi ngắt 21H, in ra màn hình kí tự trong DL
LOOP MA	;Lặp lại quá trình trên 8 lần (CX = 8)

Kết quả trên màn hình:

CCCCC

01000011

Bài tập lập trình

1. Viết chương trình hợp ngữ đầy đủ thực hiện công việc sau

- Thông báo để vào một dòng chữ bất kỳ, kết thúc bằng Enter.
- Vào một dòng chữ.
- Đếm số chữ hoa trong dòng.
- Nếu không có chữ hoa hiển thị thông báo “Không có chữ hoa”.
- Ngược lại hiển thị số chữ hoa lên màn hình.(giả thiết số chữ hoa không quá 9).
- Về DOS.

Giải:

.MODEL SMALL

.STACK 100H

.DATA

S1 DB 'Vao mot dong chu bat ki: \$'

S2 DB 'Khong co chu hoa\$'

S3 DB 0AH,0DH,'\$'

.CODE

MAIN PROC

MOV AX,@DATA ;Khởi tạo DS

MOV DS,AX

MOV AH,9 ;In ra thông báo để vào một dòng chữ bất kỳ

LEA DX,S1

INT 21H

MOV AH,1 ;Bắt đầu đọc kí tự vào

XOR CX,CX ;CX chứa số chữ hoa trong dòng

LAP1:

INT 21H ;Đọc một kí tự

CMP AL,0DH ;Xem có phải là Enter không?

JE TIEP1 ;Nếu là Enter thì nhảy tới nhãn TIEP1

CMP AL,41H ;Xem có <'A' không?

JL NOTUPPER ;Nếu <'A' thì nhảy tới nhãn NOTUPPER

CMP AL,5AH ;Nếu >='A', xét xem có >'Z' không?

JG NOTUPPER ;Nếu lớn hơn thì nhảy tới nhãn NOTUPPER

INC CX ;Nếu <='Z' thì là chữ hoa, tăng CX 1

NOTUPPER:

JMP LAP1 ;Nếu không phải là chữ hoa thì lặp lại việc đọc kí tự

TIEP1:

MOV AH,9 ;In ra chuỗi CRLF, xuống dòng

LEA DX,S3

INT 21H

CMP CX,0	;Xem CX có bằng 0 không?
JE NOUPPER	;Nếu CX = 0, nhảy tới nhãn NOUPPER
MOV DX,CX	; Nếu CX <> 0, chuyển số chữ hoa trong CX sang DX
ADD DL,30H	;Đổi sang kí tự số
MOV AH,2	
INT 21H	;In ra số chữ hoa
JMP EXIT	;Nhảy tới nhãn EXIT
NOUPPER:	
MOV AH,9	;Nếu không có chữ hoa, in ra thông báo S2
LEA DX,S2	
INT 21H	
EXIT:	
MOV AH,4CH	;Trả lại điều khiển cho hệ thống
INT 21H	
MAIN ENDP	
END MAIN	

2. Cho một mảng số liệu có tên M1 gồm 50 phần tử cỡ WORD. Viết chương trình hợp ngữ đầy đủ đếm và hiển thị số lượng phần tử là số âm lên màn hình.
(Giả thiết có chương trình con hiển thị số hệ 10 tên là IN_DEC, số cần hiển thị để trong AX.)

Giải:

```
.MODEL SMALL
.STACK 100H
.DATA
    M1    DW    -10,-9,2,-3,5,45 DUP(-1)
.CODE
MAIN PROC
    MOV AX,@DATA    ;Khởi tạo DS
    MOV DS,AX

    XOR CX,CX        ;CX chứa số số âm
    XOR BX,BX        ;BX đóng vai trò chỉ số khi chạy trong mảng
    LEA SI,M1        ;SI trỏ vào đầu mảng M1
LAP1:
    CMP BX,50        ;So sánh BX với 50
    JE TIEP1        ;Nếu BX = 50 nhảy tới nhãn TIEP1
    MOV AX,[SI]      ;AX = phần tử trỏ bởi SI
    CMP AX,0         ;So sánh AX với 0
    JGE SO_DUONG     ;Nếu AX >= 0, nhảy tới nhãn SO_DUONG
    INC CX           ;Nếu AX < 0, tăng số số âm 1
SO_DUONG:
    INC BX           ;Tăng BX 1
    ADD SI,2         ;Tăng SI 2 (do mỗi phần tử trong M1 là 1 word)
    JMP LAP1         ;Lặp lại đến khi duyệt hết mảng M1
TIEP1:
    MOV AX,CX        ;In ra số số âm hệ 10
    CALL IN_DEC

    MOV AH,4CH       ;Trả lại điều khiển cho hệ thống
    INT 21H
MAIN ENDP
IN_DEC PROC          ;Thủ tục in ra số hệ 10 chứa trong AX
    PUSH AX          ;Lưu lại các thanh ghi
    PUSH BX
    PUSH CX
    PUSH DX

    CMP AX,0         ;So sánh AX với 0
    JGE SODUONG      ;Nếu AX >= 0, nhảy tới SODUONG
    PUSH AX          ;Nếu AX < 0, lưu lại số trong AX
    MOV AH,2         ;In ra dấu âm '-'
    MOV DL,'-'
    MOV AH,0
    INT 21H
    POP DX
    POP CX
    POP BX
    POP AX
    RET
IN_DEC ENDP
```

INT 21H	
POP AX	;Lấy lại số âm trong AX
NEG AX	;Rồi đổi dấu số âm trong AX thành số dương
SODUONG:	
XOR CX,CX	;CX chứa số chữ số của số cần in
MOV BX,10	;Gán BX = 10
LAY_SO_DU:	
XOR DX,DX	;Chuẩn bị cho phép chia DXAX cho 10
DIV BX	;Chia AX cho 10
PUSH DX	;Số dư cất vào Stack
INC CX	;Tăng số chữ số lên 1
CMP AX,0	;So sánh thương với 0
JNE LAY_SO_DU	;Nếu $\neq 0$ thì lặp lại quá trình chia DXAX cho 10
MOV AH,2	;In ra các chữ số của AX
IN_SO_DU:	
POP DX	;Lấy các số dư khi chia DXAX cho 10
OR DL,30H	;Đổi thành kí tự số trong bảng ASCII
INT 21H	;In ra
LOOP IN_SO_DU	;Lặp lại số lần bằng số chữ số của AX
POP DX	;Khôi phục các thanh ghi
POP CX	
POP BX	
POP AX	
RET	
IN_DEC ENDP	
END MAIN	

3. Cho một mảng số liệu có tên M1 gồm 80 phần tử, mỗi phần tử cỡ 1 BYTE chứa 1 kí tự. Viết chương trình hợp ngữ đầy đủ làm các công việc sau:
- + Vào một dòng kí tự lưu trong mảng.
 - + Đếm và hiển thị những kí tự là chữ số ở dòng tiếp theo.

Giải:

```
.MODEL SMALL
```

```
.STACK 100H
```

```
.DATA
```

```
    M1    DB    80 DUP(0)
    S1     DB    'Vao mot dong ki tu: $'
    S2     DB    'So chu so: $'
    S3     DB    'Cac chu so: $'
    CRLF DB    13,10,'$'
```

```
.CODE
```

```
MAIN PROC
```

```
    MOV AX,@DATA    ;Khởi tạo DS
    MOV DS,AX
```

```
    MOV AH,9        ;In ra thông báo vào một dòng kí tự
    LEA DX,S1
    INT 21H
```

```
    XOR CX,CX        ;CX chứa số chữ số
    LEA SI,M1        ;SI trở vào đầu mảng M1
    MOV AH,1
```

```
LAP1:
```

```
    INT 21H          ;Đọc các kí tự nhập vào
    CMP AL,0DH        ;Xem có phải phím Enter không
    JE TIEP1          ;Nếu đúng thì nhảy đến nhãn TIEP1
    MOV [SI],AL        ;Lưu kí tự đọc được vào ô nhớ trỏ bởi SI
    INC SI             ;Tăng SI 1 (do các phần tử trong mảng kiểu BYTE)
    CMP AL,48          ;So sánh kí tự nhập vào với '0'
    JL LAP1            ;Nếu nhỏ hơn thì đọc kí tự tiếp theo
    CMP AL,57          ;So sánh kí tự nhập vào với '9'
    JG LAP1            ;Nếu lớn hơn thì đọc kí tự tiếp theo
    INC CX             ;Nếu < '9' thì tăng số chữ số lên 1
    JMP LAP1           ;Đọc kí tự tiếp theo
```

```
TIEP1:
```

```
    MOV [SI],'$'      ;Gán cho kí tự cuối cùng của xâu là '$'
    MOV AH,9          ;In ra chuỗi CRLF, xuống dòng
    LEA DX,CRLF
```

INT 21H	
LEA DX,S2	;In ra thông báo số chữ số
INT 21H	
MOV AX,CX	;In ra số chữ số dưới dạng hệ 10
CALL IN_DEC	
MOV AH,9	;In ra chuỗi CRLF, xuống dòng
LEA DX,CRLF	
INT 21H	
LEA DX,S3	; In ra thông báo các chữ số
INT 21H	
MOV AH,2	
LEA SI,M1	;SI trở vào đầu mảng M1
LAP2:	
CMP [SI],'\$'	;So sánh [SI] với kí tự '\$'
JE TIEP2	;Nếu đúng thì nhảy tới nhãn TIEP2
MOV DL,[SI]	;Nếu không thì gán DL bằng [SI]
CMP DL,'0'	;So sánh DL với '0'
JL NOT_NUMBER	;Nếu <'0' thì nhảy tới nhãn NOT_NUMBER
CMP DL,'9'	;Nếu >='0' thì so sánh với '9'
JG NOT_NUMBER	;Nếu >'9' thì nhảy tới nhãn NOT_NUMBER
INT 21H	;Nếu <='9' thì in ra chữ số đó
MOV DL,' '	;In ra dấu ' '
INT 21H	
NOT_NUMBER:	
INC SI	;Nếu không phải là chữ số thì tăng SI 1
JMP LAP2	;Chuyển sang xét kí tự tiếp theo trong mảng M1
TIEP2:	
MOV AH,4CH	;Trả lại điều khiển cho hệ thống
INT 21H	
MAIN ENDP	
IN_DEC PROC	;Thủ tục in ra số hệ 10 chứa trong AX
PUSH AX	;Lưu lại các thanh ghi
PUSH BX	
PUSH CX	
PUSH DX	
CMP AX,0	;So sánh AX với 0

JGE SODUONG	;Nếu AX>=0, nhảy tới SODUONG
PUSH AX	;Nếu AX<0, lưu lại số trong AX
MOV AH,2	;In ra dấu âm '-'
MOV DL,''	
INT 21H	
POP AX	;Lấy lại số âm trong AX
NEG AX	;Rồi đổi dấu số âm trong AX thành số dương
SODUONG:	
XOR CX,CX	;CX chứa số chữ số của số cần in
MOV BX,10	;Gán BX = 10
LAY_SO_DU:	
XOR DX,DX	;Chuẩn bị cho phép chia DXAX cho 10
DIV BX	;Chia AX cho 10
PUSH DX	;Số dư cất vào Stack
INC CX	;Tăng số chữ số lên 1
CMP AX,0	;So sánh thương với 0
JNE LAY_SO_DU	;Nếu <> 0 thì lặp lại quá trình chia DXAX cho 10
MOV AH,2	;In ra các chữ số của AX
IN_SO_DU:	
POP DX	;Lấy các số dư khi chia DXAX cho 10
OR DL,30H	;Đổi thành kí tự số trong bảng ASCII
INT 21H	;In ra
LOOP IN_SO_DU	;Lặp lại số lần bằng số chữ số của AX
POP DX	;Khôi phục các thanh ghi
POP CX	
POP BX	
POP AX	
RET	
IN_DEC ENDP	
END MAIN	

- 4.** Viết chương trình hợp ngữ đầy đủ tính $a=b+c$ với điều kiện:
- + a,b,c là các biến kiểu byte.
 - + Thông báo để vào giá trị b,c bằng một số hệ 10 từ bàn phím.
 - + Tính $a=b+c$ và thông báo kết quả lên màn hình.
- (Nếu $a>9$ hãy trừ đi 10 và in ra số 1, Đổi a ra chữ số và hiển thị.)
- + Về DOS.

Giải:

.MODEL SMALL

.STACK 100H

.DATA

S1 DB 'Vao cac chu so tu 0 - 9: \$'

S2 DB 'b = \$'

S3 DB 'c = \$'

S4 DB 'Tong cua b va c la: \$'

CRLF DB 13,10,'\$'

.CODE

MAIN PROC

MOV AX,@DATA ;Khởi tạo DS

MOV DS,AX

MOV AH,9 ;In ra thông báo vào các chữ số

LEA DX,S1

INT 21H

LEA DX,CRLF ;Xuống dòng

INT 21H

LEA DX,S2 ;In ra thông báo nhập b =

INT 21H

MOV AH,1 ;Nhập giá trị cho b

INT 21H

SUB AL,30H ;Đổi b thành số và chứa trong BL

MOV BL,AL

MOV AH,9 ;Xuống dòng

LEA DX,CRLF

INT 21H

LEA DX,S3 ;In ra thông báo nhập c =

INT 21H

MOV AH,1 ;Nhập giá trị cho c

INT 21H	
SUB AL,30H	;Đổi c thành số
ADD BL,AL	;Thực hiện b + c và chứa kết quả trong BL
MOV AH,9	;Xuống dòng
LEA DX,CRLF	
INT 21H	
LEA DX,S4	;In ra thông báo tổng của b và c là
INT 21H	
CMP BL,10	;So sánh tổng với 10
JL NHO_HON_10	;Nếu nhỏ hơn thì nhảy tới nhãn NHO_HON_10
MOV AH,2	;Nếu lớn hơn 10 thì
MOV DL,'1'	;In ra số 1
INT 21H	
SUB BL,10	;Và trừ tổng đi 10
NHO_HON_10:	
MOV AH,2	
MOV DL,BL	;Gán DL = BL
ADD DL,30H	;Đổi ra mã ASCII của chữ số tương ứng
INT 21H	;In ra chữ số trong BL
MOV AH,4CH	;Trả lại điều khiển cho hệ thống
INT 21H	
MAIN ENDP	
END MAIN	

5. Viết chương trình hợp ngữ đầy đủ thực hiện công việc sau
- Thông báo để vào một từ bất kỳ.(Từ dài nhất cũng không quá 15 kí tự)
 - Vào một từ và lưu trong mảng có tên Name (Đếm số chữ).
 - Nếu chữ đầu là chữ thường đổi thành chữ hoa.
 - Nếu các chữ tiếp theo là hoa đổi thành thường.
 - Hiện thị từ đã sửa ở dòng tiếp theo.
 - Về DOS.

Ví dụ: Bạn hãy vào một cái tên: BÌNH
 Tên đã sửa: Binh

Giải:

.MODEL SMALL

.STACK 100H

.DATA

S1 DB 'Ban hay vào mot cai ten: \$'

S2 DB 'Ten da sua: \$'

S3 DB 15 DUP(0)

CRLF DB 13,10,'\$'

.CODE

MAIN PROC

MOV AX,@DATA ;Khởi tạo DS

MOV DS,AX

MOV AH,9 ;In ra thông báo vào một cái tên

LEA DX,S1

INT 21H

XOR CX,CX ;CX chứa số chữ cái của xâu nhập vào

LEA SI,S3 ;SI trỏ vào đầu mảng S3

MOV AH,1

LAP1:

INT 21H ;Đọc một kí tự

CMP AL,13 ;Xem có ấn phím Enter không?

JE TIEP1 ;Nếu là phím Enter, nhảy tới nhãn TIEP1

MOV [SI],AL ;Lưu kí tự vừa nhập vào mảng

INC CX ;Tăng số chữ cái lên 1

INC SI ;Tăng SI 1

JMP LAP1 ;Đọc kí tự tiếp theo

TIEP1:

MOV [SI],'\$' ;Gán kí tự cuối cùng của xâu là '\$'

MOV AH,9 ;Xuống dòng

LEA DX,CRLF	
INT 21H	
LEA DX,S2	;In ra thông báo “Tên đã sửa: “
INT 21H	
LEA SI,S3	;SI trở vào đầu mảng S3
MOV AL,[SI]	;Gán AL = [SI]
CMP AL,'a'	;So sánh kí tự đầu với 'a'
JL NOT_LOWER	;Nếu <'a' thì nhảy tới nhãn NOT_LOWER
CMP AL,'z'	;Nếu >='a' thì so sánh với 'z'
JG NOT_LOWER	;Nếu >'z' thì nhảy tới nhãn NOT_LOWER
SUB AL,20H	;Nếu <='z' thì đổi thành chữ hoa
NOT_LOWER:	
MOV AH,2	;In ra kí tự đầu tiên của xâu chứa trong AL
MOV DL,AL	
INT 21H	
DEC CX	
LAP2:	
INC SI	;Tăng SI, trở tới kí tự tiếp theo trong xâu
MOV AL,[SI]	;Gán AL = [SI]
CMP AL,'A'	;So sánh AL với 'A'
JL NOT_UPPER	;Nếu <'A' thì nhảy đến nhãn NOT_UPPER
CMP AL,'Z'	;Nếu >='A' thì so sánh với 'Z'
JG NOT_UPPER	;Nếu >'Z' thì nhảy đến nhãn NOT_UPPER
ADD AL,20H	;Nếu <='Z' thì là chữ hoa, đổi thành chữ thường
NOT_UPPER:	
MOV AH,2	;In ra kí tự trong AL
MOV DL,AL	
INT 21H	
LOOP LAP2	;Lặp lại với các kí tự tiếp theo
MOV AH,4CH	;Trả lại điều khiển cho hệ thống
INT 21H	
MAIN ENDP	
END MAIN	

6. Viết chương trình hợp ngữ đầy đủ thực hiện công việc sau
- Hiện thị thông báo : 'Hãy gõ vào một chuỗi chữ cái hoa, CR để thôi'
 - Nhận chuỗi chữ cái hoa.
 - Xuống dòng về đầu dòng.
 - Hiện thị thông báo : 'Chữ cuối cùng đã đổi ra chữ thường'
 - Hiện thị chữ cuối đã đổi ra chữ thường.
 - Về DOS.

Giải:

.MODEL SMALL

.STACK 100H

.DATA

S1 DB 'Hay go vao mot chuoai chu cai hoa, CR de thoi: \$'

S2 DB 'Chu cuoi cung da doi ra chu thuong: \$'

CRLF DB 13,10,'\$'

.CODE

MAIN PROC

MOV AX,@DATA ;Khởi tạo DS

MOV DS,AX

MOV AH,9 ;In ra thông báo vào một chuỗi chữ hoa

LEA DX,S1

INT 21H

MOV AH,1

LAP1:

INT 21H ;Đọc một kí tự

CMP AL,13 ;Xem có ấn Enter không?

JE TIEP1 ;Nếu ấn Enter thì nhảy đến nhãn TIEP1

MOV BL,AL ;Chứa kí tự vừa đọc trong BL

JMP LAP1 ;Đọc kí tự tiếp theo

TIEP1:

MOV AH,9 ;Xuống dòng

LEA DX,CRLF

INT 21H

LEA DX,S2 ;In ra thông báo chữ cuối cùng đổi ra chữ thường

INT 21H

MOV AH,2

MOV DL,BL ;Chuyển chữ cuối cùng chứa trong BL vào DL

ADD DL,20H ;Đổi ra chữ thường

INT 21H ;In ra chữ cuối cùng đã đổi thành chữ thường

```
        MOV AH,4CH      ;Trả lại điều khiển cho hệ thống
        INT 21H
MAIN ENDP
END MAIN
```

7. Viết chương trình hợp ngữ đầy đủ thực hiện công việc sau
- Hiển thị thông báo : 'Hãy gõ vào một chuỗi chữ cái, hai chữ giống nhau để thôi'
 - Nhận chuỗi chữ cái, lưu vào mảng, nếu là chữ thường đổi ra chữ hoa.
 - Xuống dòng về đầu dòng.
 - Hiển thị thông báo : 'Chữ cuối cùng đã đổi ra chữ hoa'
 - Hiển thị chữ cuối đã đổi ra chữ hoa.
 - Về DOS.

Giải:

.MODEL SMALL

.STACK 100H

.DATA

S1 DB 'Hay go vao mot chuoai chu cai, hai chu giong nhau de thoi: \$'

S2 DB 'Chu cuoi cung da doi ra chu hoa: \$'

M1 DB 100 DUP(0)

CRLF DB 13,10,'\$'

.CODE

MAIN PROC

MOV AX,@DATA ;Khởi tạo DS

MOV DS,AX

MOV AH,9 ;In ra thông báo nhập vào một chuỗi chữ cái

LEA DX,S1

INT 21H

LEA SI,M1 ;SI trỏ vào đầu mảng M1

XOR CX,CX ;CX chứa số kí tự của chuỗi

MOV AH,1

INT 21H ;Nhập kí tự đầu tiên

MOV BL,AL ;Chứa trong BL

MOV [SI],AL ;Lưu kí tự đầu tiên vào mảng

INC SI ;Tăng SI 1

INC CX ;Tăng CX 1

LAP1:

INT 21H ;Đọc kí tự tiếp theo

CMP AL,BL ;So sánh với kí tự trước đây

JE TIEP1 ;Nếu bằng nhau thì nhảy đến nhãn TIEP1

MOV BL,AL ;Nếu khác thì lưu lại kí tự vừa nhập vào BL

MOV [SI],AL ;Lưu kí tự vừa nhập vào mảng

INC SI ;Tăng SI 1

INC CX ;Tăng CX 1

JMP LAP1	;Đọc kí tự tiếp theo
TIEP1:	
MOV [SI],AL	;Lưu kí tự cuối cùng vào mảng
INC CX	;Tăng CX 1
MOV [SI+1], '\$'	;Lưu kí tự cuối cùng của xâu là '\$'
LEA SI,M1	;SI trở vào đầu mảng M1
LAP2:	
MOV BL,[SI]	;Gán BL = [SI]
CMP BL,'a'	;So sánh BL với 'a'
JL NOT_LOWER	;Nếu <'a' thì nhảy đến nhãn NOT_LOWER
CMP BL,'z'	;Nếu >='a' thì so sánh với 'z'
JG NOT_LOWER	;Nếu >'z' thì nhảy đến nhãn NOT_LOWER
SUB BL,20H	;Nếu <='z' thì đổi thành chữ hoa
MOV BH,BL	;Lưu lại chữ thường cuối cùng đổi thành chữ hoa
NOT_LOWER:	
INC SI	;Tăng SI
LOOP LAP2	;Xét phần tử tiếp theo trong mảng, lặp lại CX lần
MOV AH,9	;Xuống dòng
LEA DX,CRLF	
INT 21H	
LEA DX,S2	;In ra thông báo chữ cuối cùng đã đổi thành chữ hoa
INT 21H	
MOV AH,2	
MOV DL,BH	;In ra chữ cuối cùng trong BL
INT 21H	
MOV AH,4CH	;Trả lại điều khiển cho hệ thống
INT 21H	
MAIN ENDP	
END MAIN	

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.LookupTables.com