

Informe de la pràctica 6: Memòries i Subrutines

Introducció

En aquesta pràctica es pretén conèixer l'espai de memòria del microprocessador, comprendre el funcionament de la pila i veure un exemple d'utilització de les subrutines. Per fer-ho, fem ús de:

- Ports d'entrada i sortida
- Components com un display de set segments o un teclat
- Interrupcions

Part 1

1. Qüestions

a. *L'adreçament de la instrucció LXI és:*

- i. La opció c), immediat

b. *Quina instrucció guard el PC a la Pila?*

- i. La a), PUSH PC.

2. Pregunta 1: *Quin espai ocupa en memòria la subrutina sumar?*

- a. Ocupa 7 posicions, ja que està formada per PUSH (1), LDAX (1), ADD (1), STAX (1), INX (1), INX (1) i POP(1). En total 7.

3. Pregunta 2: *Quants cicles triga en executar-se la subrutina sumar?*

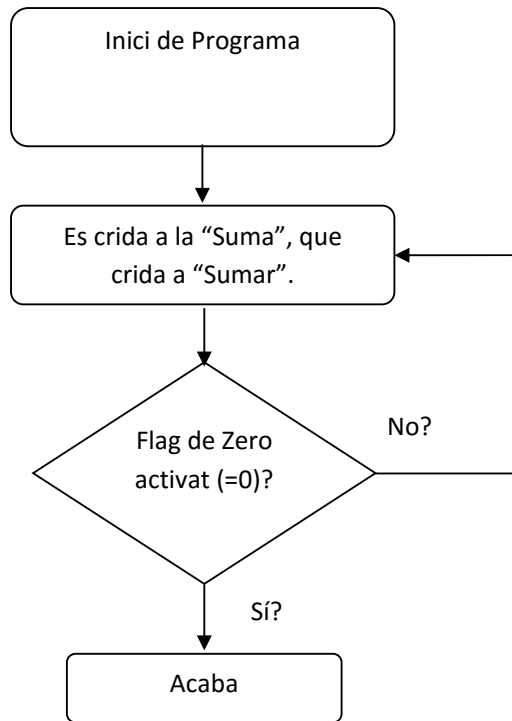
- a. 55 cicles en total, ja que PUSH ocupa 12, LDAX 7, ADD 7, STAX 7, INX 6 (x2) i POP 10 cicles. En total 52 cicles.

4. **TASCA 1:** *Dibuixeu el mapa de memòria de dades: direccions i contingut. Indiqueu les instruccions que modifiquen les dades de la memòria. En cadascuna d'elles, indiqueu quines modificacions es produeixen. Situeu la memòria de programa. Dins d'ella, localitza el subloc que pertany a la subrutina 'sumar'.*

<u>Direcció</u>	<u>Dada</u>	<u>Instrucció que les Modifica</u>
<u>0000</u>	<u>04</u>	<u>INX H</u>
<u>0001</u>	<u>06</u>	<u>INX H</u>
<u>0002</u>	<u>03</u>	
<u>0003</u>	<u>04</u>	
	.	
	.	
	.	
<u>0024</u>	<u>01</u>	<u>2 cops LDAX D</u>
<u>0025</u>	<u>ED</u>	<u>LDAX D</u>
<u>0026</u>	<u>13</u>	<u>PUSH PSW</u>
<u>0027</u>	<u>06</u>	<u>PUSH PSW</u>
<u>0028</u>	<u>09</u>	
<u>0029</u>	<u>08</u>	
	.	
	.	
	.	
<u>600</u>	<u>21</u>	
<u>601</u>	<u>28</u>	
<u>602</u>	<u>00</u>	
<u>603</u>	<u>F9</u>	
<u>604</u>	<u>3E</u>	
<u>605</u>	<u>17</u>	
<u>606</u>	<u>D6</u>	
<u>607</u>	<u>2A</u>	

Totes les direccions que no es mostren es perquè no tenen cap dada emmagatzemada (00). Les instruccions en vermell son les instruccions de Programa, les de color groc són les de la Pila. En verd està el cos del programa, i en blau es marca la subrutina "sumar". Només les instruccions INX H, PUSH PWS i LDAX D modifiquen les dades.

<u>608</u>	<u>06</u>	
<u>609</u>	<u>02</u>	
<u>60A</u>	<u>11</u>	
.		
.		
.		
<u>60D</u>	<u>21</u>	
<u>60E</u>	<u>02</u>	
<u>60F</u>	<u>00</u>	
<u>610</u>	<u>CD</u>	
<u>611</u>	<u>19</u>	
<u>612</u>	<u>06</u>	
<u>613</u>	<u>05</u>	
<u>614</u>	<u>C2</u>	
<u>615</u>	<u>10</u>	
<u>616</u>	<u>06</u>	
<u>617</u>	<u>00</u>	
<u>618</u>	<u>76</u>	
<u>619</u>	<u>F5</u>	
<u>61A</u>	<u>1A</u>	
<u>61B</u>	<u>86</u>	
<u>61C</u>	<u>12</u>	
<u>61D</u>	<u>23</u>	
<u>61E</u>	<u>13</u>	
<u>61F</u>	<u>F1</u>	
<u>620</u>	<u>C9</u>	

Diagrama del codi 1**TASCA 2**

Amb les instruccions .data N abans de declarar la pila, situarem la pila en la posició N. Com s'ha dit en la tasca 1, la pila comença en el 0028h que és el 40 decimal que està declarat en el programa. La pila la modifica el PUSH PSW, que copia en la pila l'estat dels bits de condició i el registre A, i LDAX D, que carrega a l'acumulador el registre D.

Direcció ↓	Instrucció que les modifica	Dada Abans->Després
0027	PUSH PSW	00->06
0026	PUSH PSW	00->13
0025	LDAX D	00->ED
0024	LDAX D	00->95
0024	2 cops LDAX D	95->01

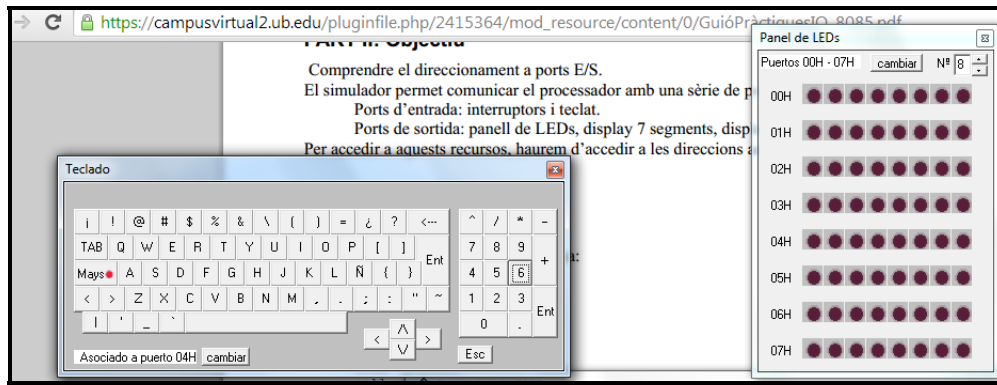
La pila creix cap a dalt, des de el 0029 fins al 0024, és a dir, des de la part inferior a la superior i emmagatzema la informació en la memòria de Dades.

Part 2

Tasca 3: Què fa la subrutina 'puertos' ?

La subrutina “puertos” és una subrutina que agafa una dada qualsevol amb la instrucció in 04h, ja sigui amb un dispositiu d'entrada com un teclat, on 04h és el port d'entrada. A continuació, fa un AND d'aquesta dada que ha entrat amb el número binari 00000001. Aquest resultat, l'envia a través de la instrucció out 05h, on 05 és el port del dispositiu de sortida, com una pantalla de LED's o un set segments.

Al multiplicar qualsevol número parell per 00000001, el resultat sempre serà zero, de manera que no es mostrarà res en el nostre dispositiu de sortida. En canvi, si el número d'entrada és senar, sempre s'enviarà una dada diferent de zero al dispositiu de sortida, que s'encendrà. Ho podem veure aquí:



S'ha introduït el 6, parell, i no es mostra res. Aquí a baix s'ha introduït un 5, i s'il·lumina:

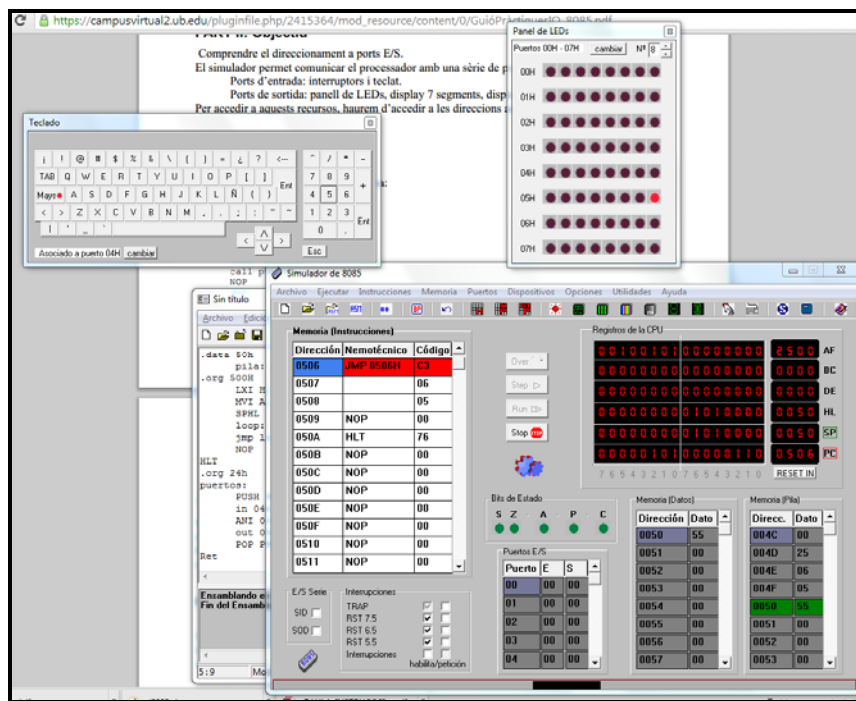
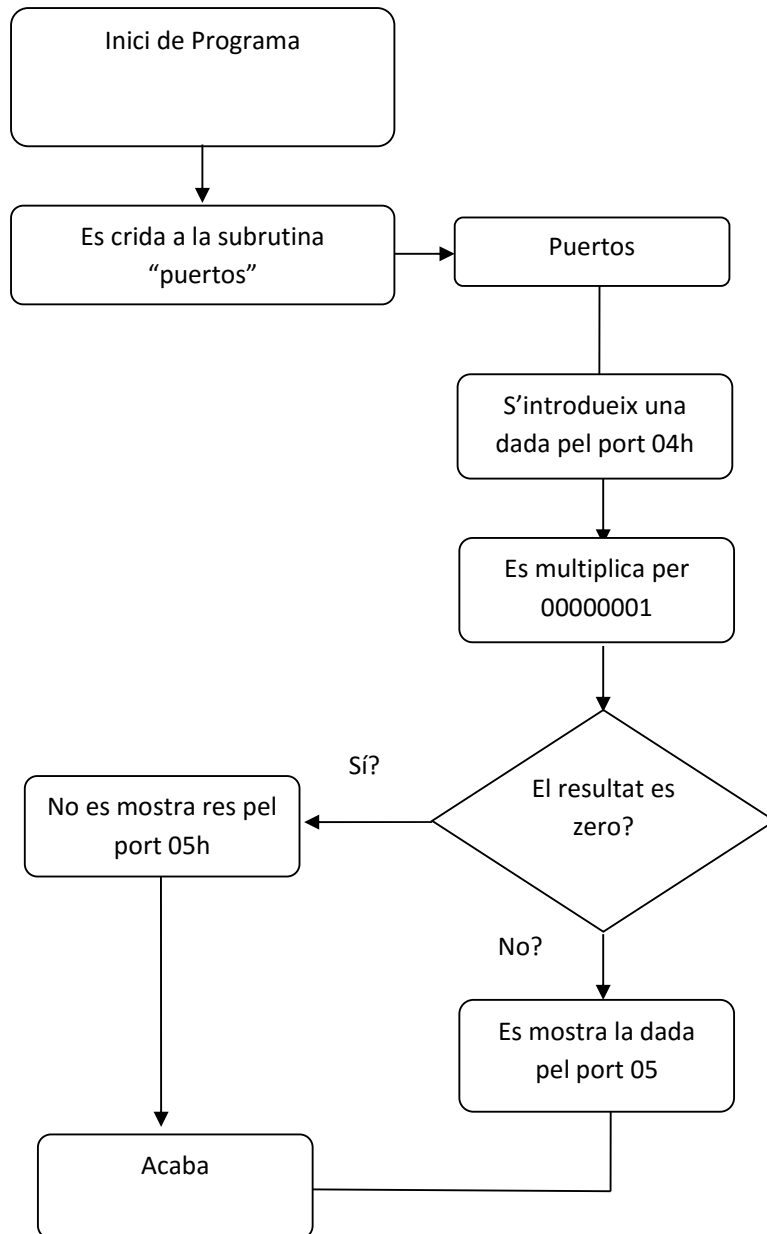


Diagrama del exercici 3:

Part 3

El meu programa és aquest:

```
.data 08H
    pila: db 77h,44h,3Eh,6Eh,4Dh,6Bh ;números del zero al cinc en codi 7 segments
.org 24h ;en el primer cas o en cas d'interrupció
    in 00h; passes el numero introduit en el port 00h a l'acumulador
    LXI D, pila; carregues la pila en D
    call visor; crides a la subrutina visor

.org 500H
visor: ;rutina que comprova si es vol esborrar
    SUI 30h ;restes 30 del numero introduit
    MOV B,A ;copies l'acumulador a B
    SUI 13h ;Li restes 13h per assegurar-te que no es una C
    JZ borrar; Si es C, saltes a la rutina esborrar
    MOV C,B; copies l'acumulador a C
    INR C; Incrementes C en 1
    call visualitzar; crides a visualitzar
    NOP ;instrucció que no fa res
    HLT ;Acabes

borrar:
    MVI A,00h; Copies 00h a l'acumulador
    out 07h ;Passes aquest zero al port 07 perquè no es mostri res (esborrar)
    NOP ;instrucció que no fa res
    HLT ;Acabes

visualitzar:
    LDAX D ;copies a l'acumulador en contingut de la posició en D
    INX D ;incrementes D
    DCR C ;Decrementes C
    JNZ visualitzar; Si no es zero, segueixes fent el bucle
OUT 07h ;mostres el que hi ha a l'acumulador pel port 07h
Ret
```

De manera que queda mes o menys així:

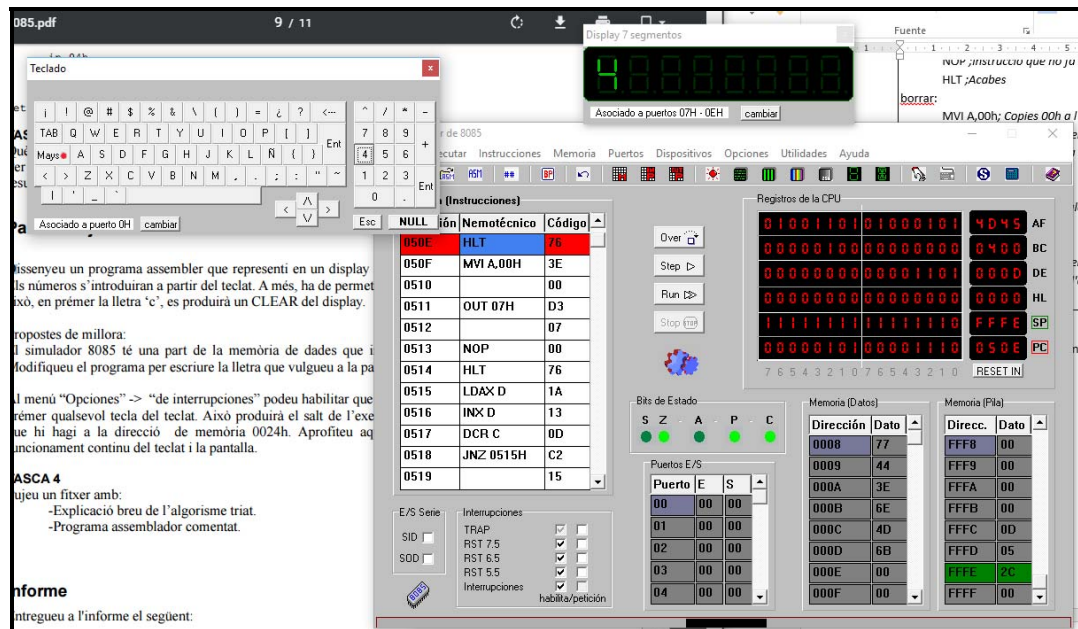
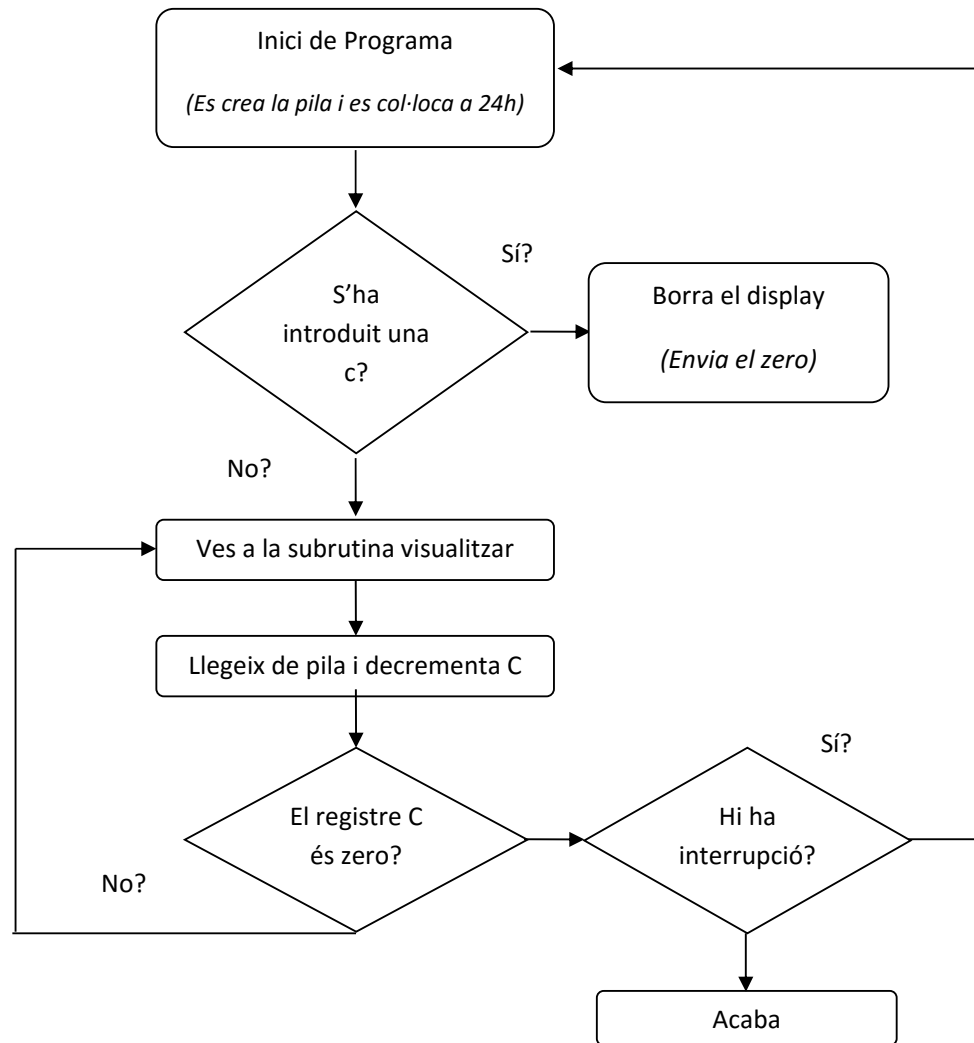


Diagrama exercici 4

En resum, el meu programa comença per la 0024 i allà fa les crides perquè en cas d'interrupció, pugui tornar a funcionar, després segueix comprovant si l'introduït és una C, sinó, passa a mostrar el número que li toca, opera fins trobar-lo i finalment, el mostra.

Conclusions

He aconseguit conèixer com funciona l'espai de memòria del microprocessador, he comprès el funcionament de la pila i he vist un exemple d'utilització de les subrutines. He comprès el funcionament dels ports d'entrada sortida i he entès el funcionament del display i les interrupcions.