**Practica 6**

**Objectius**

Els objectius d’aquesta practica han sigut, primer veure i entendre el funcionament de les memòries del microprocessador, després veure el funcionament de la pila i com funciona al cridar a subrutines, i per últim entendre els ports d’entrada i sortida com son els interruptors, teclat, el panell de LEDs i els displays de 7 i 15 segment.

**Explicació de la practica**

La practica consistia en respondre a preguntes sobre el funcionament de la pila y la memòria del microprocessador, i realitzar dos programes, un amb interruptors i panell de LEDs, i l’altre amb el teclat com entrada i el display de 7 segments i la pantalla de text com a sortida.

**Informe:**

Preguntes :

**1. L’adreçament de la instrucció LXI és:**

a) directe

b) indirecte

**c) immediat**

d) implícit

**2. Quina instrucció guarda el PC a la Pila?**

a) PUSH PC

b) POP PC

**c) CALL**

d) MOV M, PC

**3. Quin espai ocupa en memòria la subrutina ‘suma’?**

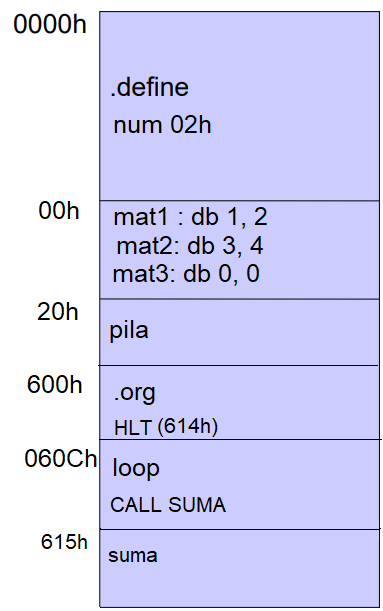
La subrutina suma ocupa 8 bytes en memòria.

**4. Quants cicles triga en executar-se la subrutina ‘suma’?**

La subrutina suma triga 65 cicles en executar-se.

**Part I**

**TASCA 1:**

****

Les instruccions que modifiquen les dades de la memòria son:

STAX D, la cridem per guardar el resultat de la suma sobre la tercera matriu.

CALL SUMA que guardarà el PC a la memòria.

PUSH PSW que guardarà l’ACC i el registre d’estats a memòria.

**TASCA 2:**

La pila comença en la posició 20h en memòria

****

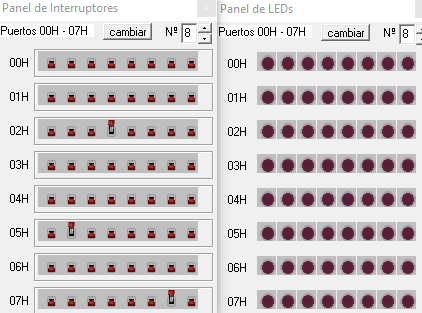
|  |  |  |
| --- | --- | --- |
| **Instrucció** | **Descripció** | **Canvi en la pila** |
| PUSH PSW | Afegeix PSW a la pila. | Resta en 2 bytes. |
| CALL SUMA | Crida a la subrutina suma | Guarda el contingut del PC dins de la pila |
| POP PSW | Restableix els flags condicionals utilitzant en contingut de la localització de memòria especificada per el stack pointer | Augmenta en 2 bytes |
| RET | El programa continua després de la crida a la subrutina. | Treu fora 2 bytes de dades |

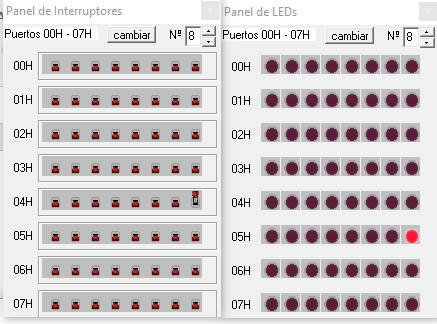
**PART II**

**TASCA 3:**

**Què fa la subrutina ‘ports’? Per això, introduïu dades amb els interruptors o amb el teclat; observeu en un port de sortida el resultat de la subrutina.**

La subrutina ports encén o apaga l’últim LED del panell de LEDS 5h, això ho fa amb la instrucció, es modificant les interruptors de la fila 4h del panell d’interruptors i d’aquesta manera es passa un numero en binari en el codi que es guarda en l’acumulador i després fa una AND amb el numero 1 en binari amb 8 bits (00000001), això fa que es que nomes s’encengui l’últim LED de 5h si l’últim interruptor de 4h esta activat.

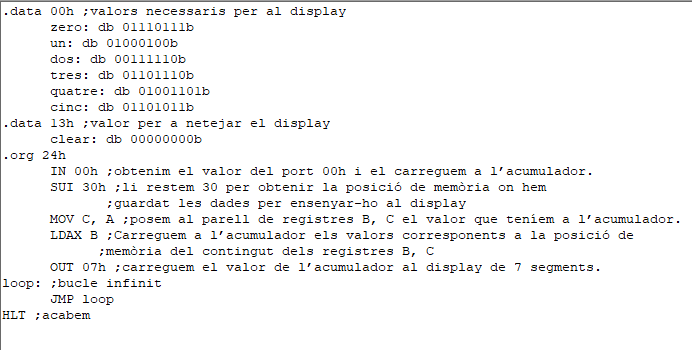
****

****

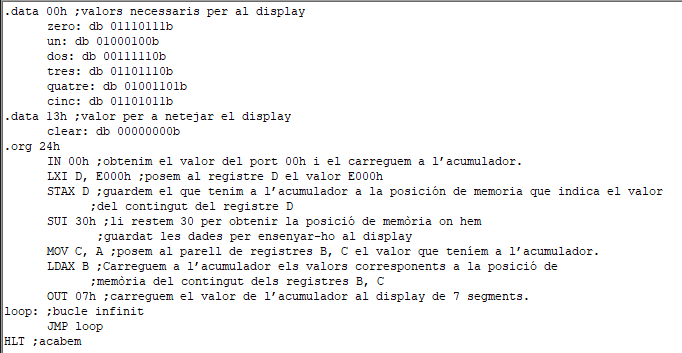
**PART III**

**TASCA 4:**

Programa sense mostrar el valor introduït per teclat a la pantalla de text:

****

Programa mostrant el valor introduït per teclat a la pantalla de text:

****

En aquests programes he guardat en memòria els números en binari i la lletra ‘c’ que s’havien de mostrar per pantalla i en el display de 7 segments per poder accedir a ells quan aquestes tecles es pressionessin al teclat. Desprès de que s’introduís el numero per teclat i que es mostres, he posat un bucle infinit que s’executi esperant una nova interacció amb el teclat per saltar a la interrupció TRAP i així repetir el procés fins que es pari el programa.

**Conclusió**

En aquesta practica he aprés a utilitzar diverses eines que son més visuals com el display i els interruptors i no fer un programa que només modifiqui registres. També he aprés el funcionament de la pila i la memòria quan es criden subrutines.