

Informe de la pràctica #5

Introducció

En aquesta cinquena pràctica usem per primer cop el simulador i8085, de Intel, i duem a terme un parell de exercicis de prova per tal de assolir els següents conceptes:

- Conèixer els modes d'adreçament del i8085.
- Familiaritzar-se amb les instruccions del simulador.

Exercicis

Aquesta pràctica consta de dos exercicis, el primer demana que donades dues matrius de 5x1, les sumem i les guardem a una tercera matriu buida, el segon demana que, a més, codifiquem la zona de memòria on hem treballat les matrius.

En el primer exercici, primerament copiem la matriu 1 en les posicions de memòria de la matriu 3, i posteriorment sumem el valor de la matriu 2 a la matriu 3, així ens romanen les dues primeres matrius inalterades i tenim el resultat a la matriu 3.

```
.define
    num 4
.data 00h
    mat1: db 1, 2, 3, 4, 5
    mat2: db 6, 7, 8, 9, 0
    mat3: db 0, 0, 0, 0, 0
.org 100h
    LXI H, mat1
    LXI B, mat3
    MVI D, num
bucle1: ;BUCLE PER A GUARDAR MAT1 A MAT3
    MOV A, M
    STAX B ;Guardem al valor 1 de mat3 el Acc
    INX H
    INX B
    DCR D
    JP bucle1
    LXI B, mat3 ;Recarreguem els valors de mat3
    MVI D, num ;Reset del iterador
```

```
bucle2: ;BUCLE PER A SUMAR MAT2 A MAT3
        MOV E, M
        LDAX B
        ADD E
        STAX B ;Guardem al valor 1 de mat3 el Acc
        INX H
        INX B
        DCR D
        JP bucle2

HLT
```

En el segon exercici, ens demanen que codifiquem la zona de memòria on es troben les tres matrius usades, per a això, usem una instrucció XOR entre el valor a codificar i la nostra clau. Usem el mateix codi que al exercici 1, però hi afegim la subrutina per a codificar.

```
        (CALL subrutina)

subrutina: ;Subrutina per a codificar la matriu
        LXI H, mat1 ;Posició inicial de memòria
        MVI B, FFh ;Clau per a codificar, en aquest cas FFh
        MVI C, 00h
        MVI D, 15d ;Numero de posicions a codificar, en aquest cas 15d
xor:
        MOV A, M ;Valor a codificar
        XRA B ;A <== B XOR A
        MOV M, A ;Guardem el codi resultant
        INR C
        INX H
        MOV A, C
        CMP D ;Comparem C (posicio actual) amb D (posicions totals)
        JNZ xor ;Si Z=0 salta

HLT
```

Existeixen també instruccions per a entrar i sortir de les subrutines i no perdre informació. Les instruccions PUSH i POP ens guarden i recuperen la informació dels registres a la pila per a fer un nou ús en la subrutina i la instrucció RET ens guarda a la pila el PC, per a que, al acabar la subrutina, poder recuperar el punt de memòria on havia quedat el programa.

Conclusions

En aquesta practica hem introduït i après a usar el simulador i8085, un simulador que té registres de 8 bits i un adreçament de 16 bits.

- Hem après a adreçar els registres en parelles en el i8085.
- Hem après les instruccions de ensamblador del simulador i8085.
- Hem après a usar el simulador i8085 de Intel i a llegir i usar la informació que proporciona.