

## Informe de la pràctica 2

### Introducció

En esta 2ª práctica de Introducció als Ordinadors encontraremos los objetivos, los ejercicios planteados y conclusiones:

Familiarizarnos con el funcionamiento de los registros que hay en la CPU a través del simulador SIMR.

Familiarizarnos con las instrucciones del simulador Ripes.

Aquesta pràctica és novament una sessió guiada pel professor.

El professor recordarà el conjunt d'instruccions, el funcionament del simulador i l'adreçament a memòria.

ADD R0,R0,R1=> Suma el contingut de R0 + el contingut de R0 i es desa en R1

#### **Com seria aquesta instrucció amb el simulador Ripes?**

add a0,a0,a1

Els registres de propòsit específic són:

- El registre Acumulador, que en el nostre simulador el tenim en una de les entrades de la ALU
- El registre d'estat, que el tenim connectat a la ALU per detectar valors negatius, zero,i OV
- El registre d'Instruccions, on es guarda la instrucció a executar
- El registre Program Counter, on tenim l'adreça de la següent instrucció a executar.
- Un registre específic per guardar les adreces en cas de salt, al costat del PC

#### **Exercici 1 (Guiat)**

Carrega el següent programa en el simulador RIPES.

```
.data
Dada1: .word 9
Dada2: .word 3
Dada3: .word 4
Dada4: .word 1
Reserva: .word 0
```

```
.text                #directiva d'inici de programa
main:
    sub a0,a0,a0      #posem A0=0. A0 actuarà com a comptador
    lw a1, Dada1      #carreguem el contingut de Dada1 en A1
    lw a2, Dada2
    lw a3, Dada3
    lw a4, Dada4
loop:
    add a5,a1,a2
    add a6,a4,a2
    addi a3,a3,-1
    bge a3,zero,loop
    sw a5, a0(Reserva)
    addi a0,a0,1
    sw a6, a0(Reserva)
end:    nop
#final de programa
```

Feu el següent exercici a casa utilitzant el simulador SIMR. Compareu-lo amb el fet a classe. Indiqueu diferències i similituds.

Las diferencias residen en:

La forma de cargar las palabras: en clase usamos las instrucciones la y lw (ex: la a7, Dada1 y lw a1, 0(a7)) mientras que aquí usamos directamente lw.

La forma de usar un contador para terminar el bucle: en SIMR usamos el registro R7 y en Ripes controlamos el loop con el registro a3.

Las similitudes son las instrucciones del bucle (sumas de registros y resta del contenido del registro R3/a3) junto con las condición de salto para finalizar el loop.

Dades: .DW 9,4,3,1

.begin inici ;directiva d'inici de programa

```
inci:
    SUB R7, R7, R7      ;posem R7 = 0. R7 actua com a comptador
    LOAD Dades(R0), R1  ;R1 <=9
    ADDI R7, #1, R7      ;R7 <=1
    LOAD Dades(R7), R2   ;R2 <=4
    ADDI R7, #1, R7      ; R7 <=2
    LOAD Dades(R7), R3   ;R3 <=3
    ADDI R7, #1, R7      ; R7<=3
    LOAD Dades(R7), R4   ;R4 <=1
```

```
loop:
    ADD R1, R2, R5
```

```

        ADD R3, R4, R6
        SUBI R3,#1, R3
        BG loop
    .end

```

Quines operacions fan aquests codis?

El código de SIMR primero cargamos todas las variables en los registros R1,R2,R3 y R4 mientras vamos incrementando en una unidad el registro R7. Dentro del bucle sumamos el contenido de R1 y R2 ( $9+4=13$ ,  $R5 = 000D$ ) y R3 y R4 ( $R6=0004$ ,  $R6=0003$ ,  $R6=0002$  según vamos reduciendo el contenido de R3) para luego decrementar en una unidad R3. Continuará mientras R3 sea mayor que 0.

El código de Ripes primero inicializa el registro a0 a 0 para usarlo como un contador, luego carga el contenido de dada1,dada2,dada3 y dada4. Después dentro del bucle, suma los registros a1 y a2 ( $9+3=12$ ,  $a5=12$ ) y los registros a4 y a2 ( $1+3=4$ ,  $a6=4$ ), además de decrementar en una unidad el contenido de a3 (de 4 pasamos a 3). El bucle se mantendrá hasta que a3 sea menor que 0, es decir, cuando sea -1 saldrá del loop y se hará un store en la reserva del contenido de a5 y a6 junto con un incremento en una unidad del contenido de a0.

### Exercici 2 (Guiat)

Feu uns programes similars als de l'exercici 1. Inicialitzeu primer el contingut de R7. Carregueu al registre 1 el valor 0000110000001011b. Carregueu al registre 2 el valor 000000000010001b. Feu l'operació  $R7 \leftarrow R7 + R1$  y decrementeu el valor de R2. Feu això en un bucle fins que el valor contingut en R2 sigui 0. Idem per Ripes:

```

.data
dada1: .word 3087
dada2: .word 17
resultat: .word 0

.text
#inicializamos el contenido a a0
    add a0, zero, zero

# cargamos los valores en los registros a0 y a1
    la a7, dada1
    lw a1, 0(a7)
    la a7, dada2
    lw a2, 0(a7)

loop:
    add a0, a0, a1
    addi a2, a2, -1
    bgt a2, zero, loop

```

```
        #salimos del bucle

        la a7, resultat

        lw a2, 0(a7)

        sw a2, 0(a7)

final:

        j final
```

#### Per SIMR:

```
data:    .DW 3083,17

mem:     .DW 0x22

.begin inici

inici:

        SUB R1,R1,R1

        LOAD mem(R0), R4

        LOAD data(R1), R7

        ADDI R1, #1, R1

        LOAD data(R1), R2

        SUB R1, R1, R1

loop:

        ADD R1, R7, R1

        SUBI R2, #1, R2

        BG loop

        STORE R1, 0(R4)

.end
```

### Exercici 3 ( a fer per l'alumne)

Ens demanen calcular un algoritme que ens faci el següent: Donades dues entrades emmagatzemades en les posicions de memòria A i B fer la comparació. Si  $A > B$  calculem la suma. Si  $B > A$  fem la diferència ( $B - A$ ) i si són iguals que multipliqui el seu valor. Carregueu diferents valors en memòria per tal de comprovar el funcionament del programa. Quina instrucció feu servir per fer la multiplicació? Què és més eficient, una instrucció directament que faci aquesta operació o el càlcul iteratiu? Raoneu la resposta.

#### Per Ripes:

```
.data
dada1: .word 5
dada2: .word 3
resultat: .word 0
.text
#inicializamos el contenido de a0
        add a0, zero, zero
        la a7, dada1
        lw a1, 0(a7)
        la a7, dada2
        lw a2, 0(a7)

loop:
```

```
beq a1, a2, multiplicacion
bgt a1, a2, suma
blt a1, a2, resta
```

multiplicacion:

```
mul a3, a2, a1
sw a3, resultat(a0)
```

suma:

```
add a4, a2, a1
sw a4, resultat(a0)
```

resta:

```
sub a5, a2, a1
sw a5, resultat(a0)
```

end: nop

### Per SIMR:

Dades: .dw 5,3

.begin inici

multiplicacio:

loop:

```
ADD R1, R2, R1
SUBI R2, #1, R2
BG loop
STORE R1, 34(R0)
```

BR final

resta:

```
SUB R2, R1, R6
STORE R6, 34(R0)
BR final
```

suma:

```
ADD R1, R2, R7
STORE R7, 34(R0)
BR final
```

inici:

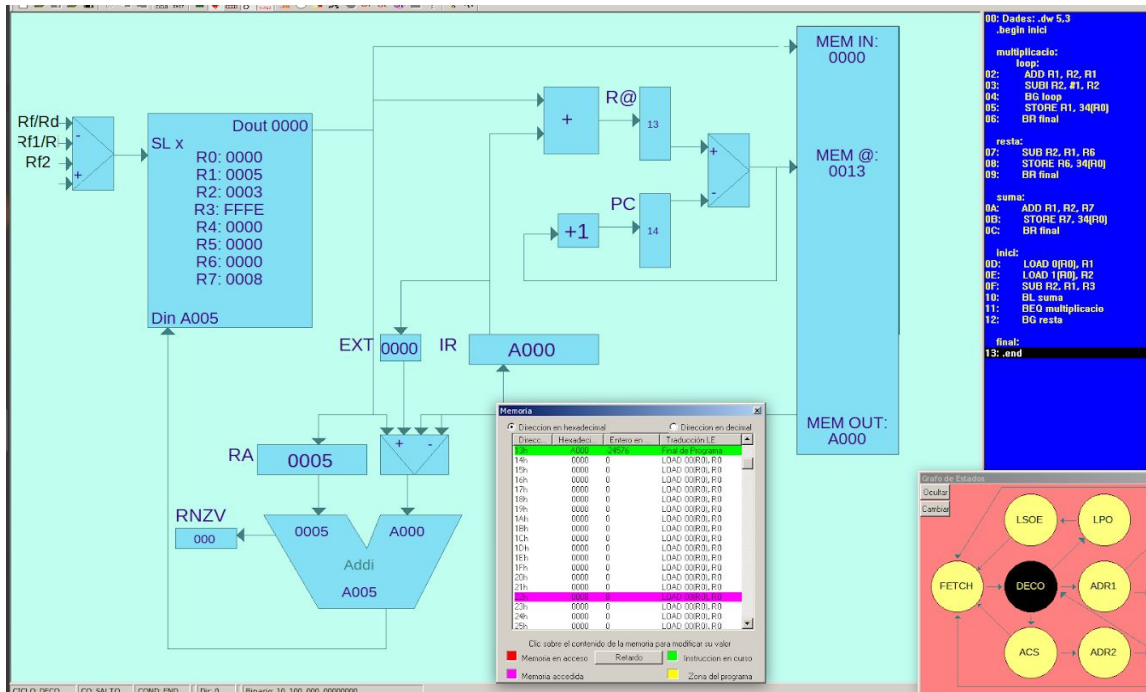
```
LOAD 0(R0), R1
LOAD 1(R0), R2
SUB R2, R1, R3
BL suma
BEQ multiplicacio
BG resta
```

final:

.end

Para multiplicar, en la máquina rudimentaria, SIMR, utilizamos un bucle como el implementado en el ejercicio 2 de la práctica mientras que para Ripes usamos la instrucción

“mul”, que multiplica directamente el contenido de los registros. Por el funcionamiento de ambos códigos, es más eficiente la instrucción “mul” debido a que reducimos el tiempo de ejecución de la operación.



## Informe

Explicaremos el trabajo realizado en esta práctica:

En el ejercicio 1:

- Un cop acabat el programa quant val el contingut de R5? 0009h
- Un cop acabat el programa quant val el contingut de R6? 0008h
- Un cop acabat el programa quant val el contingut de R3? 0000h

En el ejercicio 2:

- Quant val el contingut de A0 quan acaba el programa?  
CCBBh = 52411 (base 10)
- Quantes vegades s'executa el codi?  
17 veges, hasta que el contenido de R2 llega a 0 en el caso de SIMR.
- Al final del programa, just abans del .end, poseu la instrucció sw A5 A0(Reserva). En quina posició de memòria es guarda el resultat? En quina posició de memòria es guarda el contingut del registre A6?

En la posición 00h.

En el ejercicio 3:

i) Expliqueu el funcionament del programa

El programa del ejercicio 3 realiza una comparación de 2 valores, guardados en 2 registros (a1 y a2). Primero guardamos los valores en 2 variables: dada1 y dada2. Después si el contenido es igual (instrucción beq), saltamos fuera del bucle y multiplicamos los valores. Si el valor del primer registro es mayor que el del segundo, sumamos los valores. Por último, si el valor del primer registro es menor que el del segundo, los restamos.

ii) Feu que el resultat es guardi a la posició de memòria 22h

Se guarda en el registro s6 (x22) o en la posición 34(R0).

### **Conclusions**

*Estudiamos más a fondo las instrucciones de salto en SIMR y Ripes y los registros.*

*Entendemos el funcionamiento de la máquina de Ripes.*

*Se han realizado los ejercicios del guión de prácticas de laboratorio.*