-Fetch

1.- Simular > Iniciar simulació.

2.- Entrar al mòdul Fetch seleccionant-lo i fent click a: 

3.- Seleccionar el mòdul de memòria i carregar el fitxer de memòria (echo.mem): 

**Per simular:**

La simulació la farem tota amb el botó següent enlloc d’executar-ho amb el Play:

 A partir d’ara li diré “Step”

1. Clickem varies vegades el botó Step
2. Posem el commutador de reset a Off
3. Clickem varies vegades al botó Step
4. Posem el commutador de reset a ON
5. Mentres clickem a Step nem veien com avança el Program Counter i el indicador d’instruccions.

Com evoluciona:

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| ... | ... |
|  |  |
|  |  |

**Codi Carregat en memòria:**

00/ 8c090064 # <input:1> lw $t1, 100($zero) # carrega la constant "1"

01/ 8c0a0065 # <input:2> lw $t2, 101($zero) # carrega 256 (0x100)

02/ 8c100066 # <input:3> lw $s0, 102($zero) # carrega @missatge

03/ 8e080000 # <input:7> lw $t0, 0($s0) #llegim caracter de l'string

04/ 11000003 # <input:8> beq $t0, $zero, fi\_no\_null # sortim del bucle

05/ ac08ffff # <input:9> sw $t0, 65535($zero) # escrivim caracter al TTY

06/ 02098020 # <input:10> add $s0, $s0, $t1 # increment index string

07/ 08000003 # <input:11> j mentre\_no\_null # fi\_bucle

08/ 8c04ffff # <input:17> lw $a0, 65535($zero) # llegim tecla

09/ 008a4024 # <input:18> and $t0, $a0,$t2 # mirem si s'ha pitjat (bit 9 a '1')

0a/ 1100fffd # <input:19> beq $t0, $zero, mentre\_sempre # no hi tecla continuem

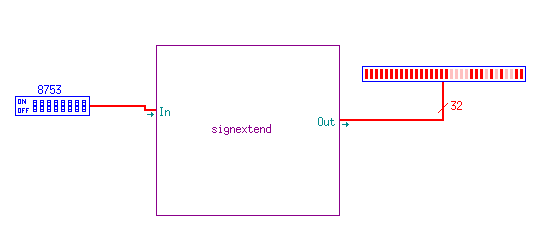
0b/ ac04ffff # <input:20> sw $a0, 65535($zero) # print de tecla

0c/ 08000008 # <input:21> j mentre\_sempre # jump per

0d/ 1000ffff # <input:25> beq $zero, $zero, fi # bucle infinit

Sign Extend

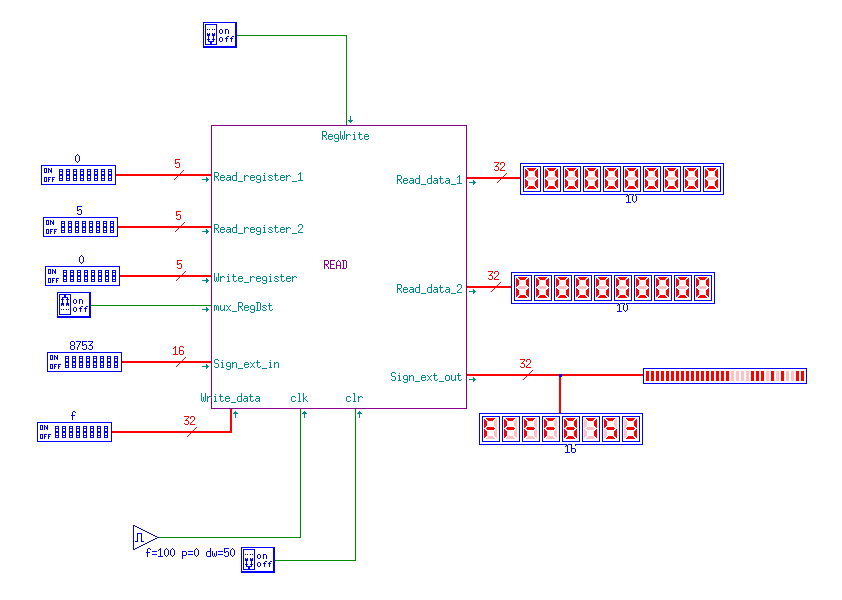
S’ha provat per al valor 0x8753 que passat a binari és 1000 0111 0101 0011:

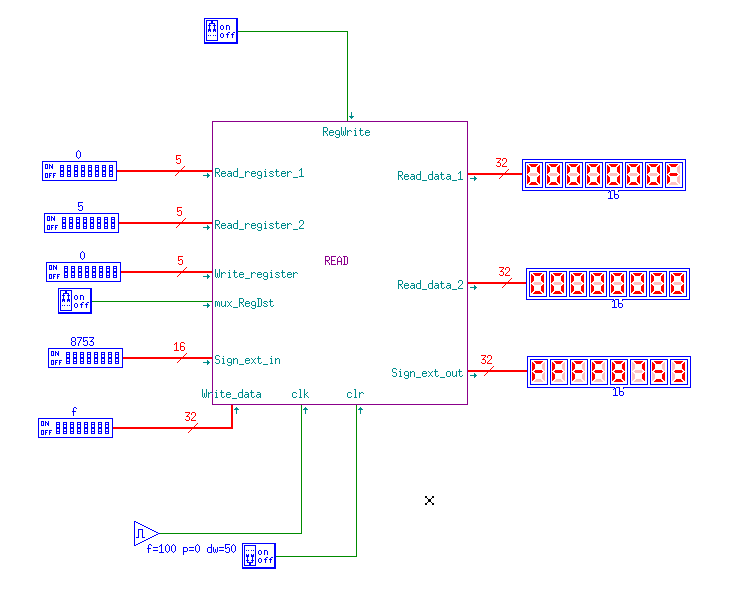


READ

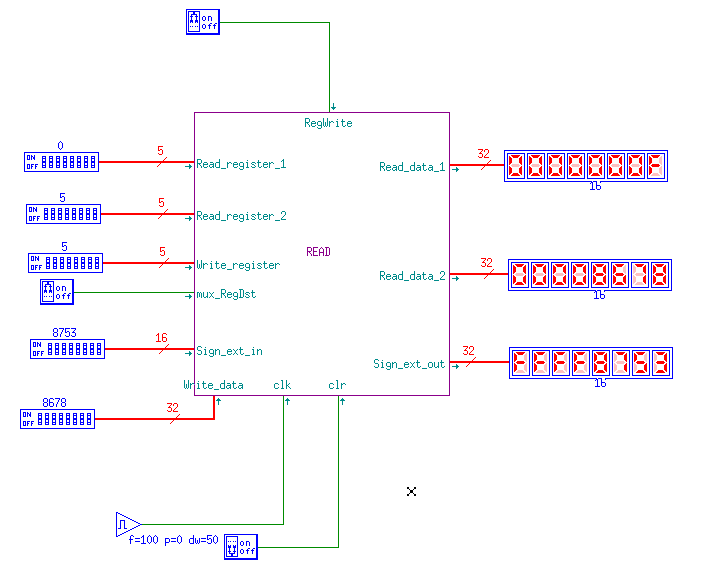
Fem step by step una escriptura/lectura sobre el registre 0 i 5. Abans d'escriure i després de fer un reset veurem que els dos registres estàn a 0. Acte seguit activem la senyal «RegWrite»

1.- Fem un reset i comprovem que tot el que llegim és 0.

2.- Escrivim al registre 0 el valor «f»:

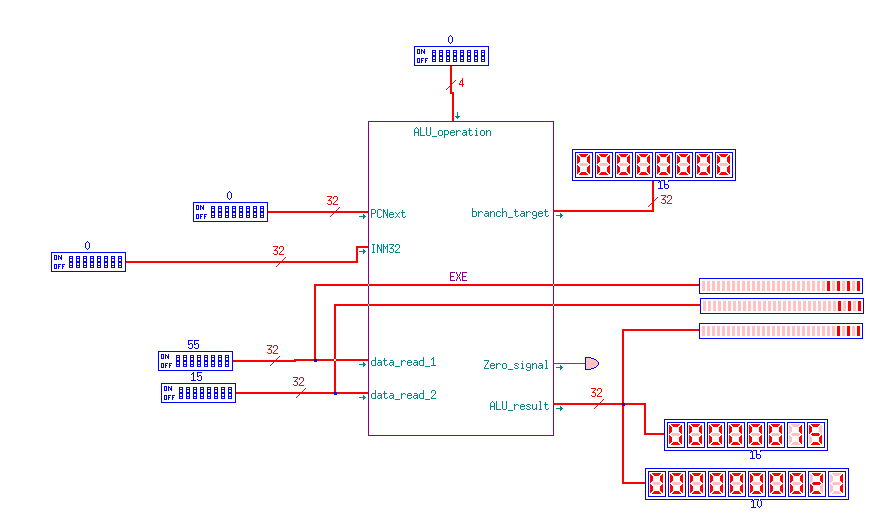


3.- Escrivim al registre 5 el valor 0x8678.

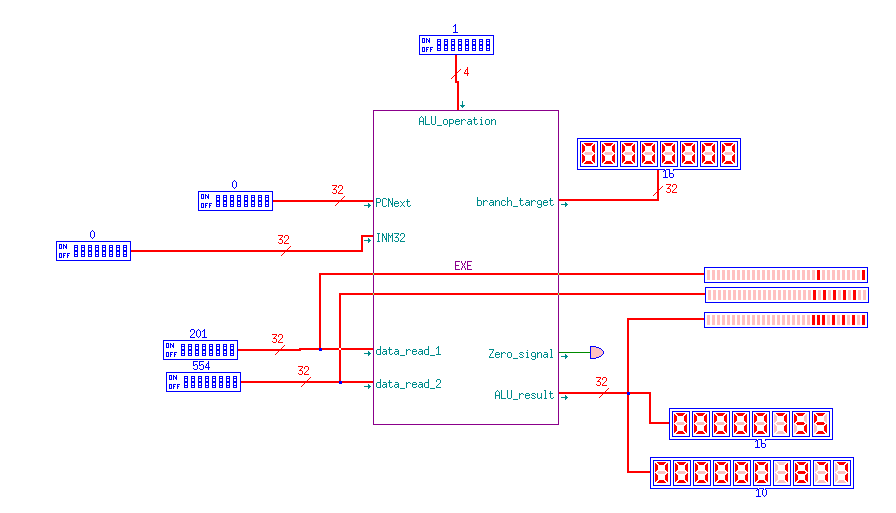


ALU

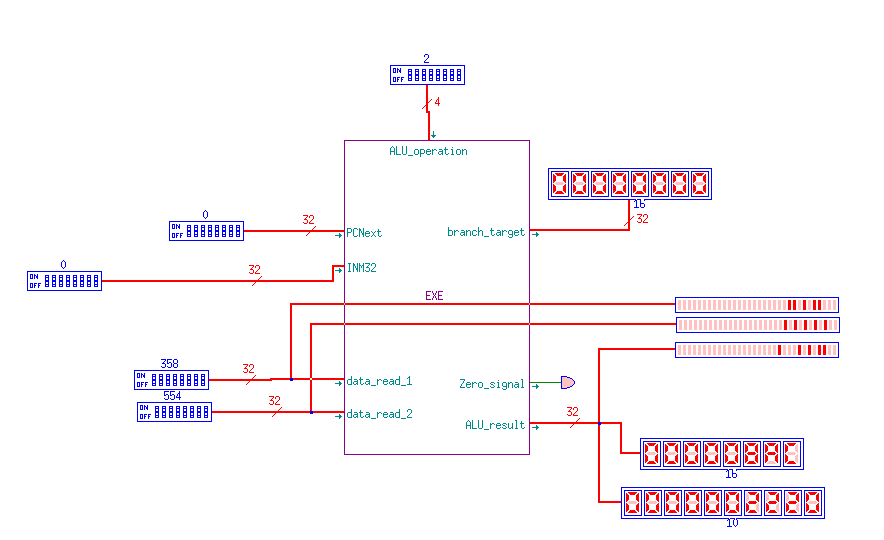
1.-AND



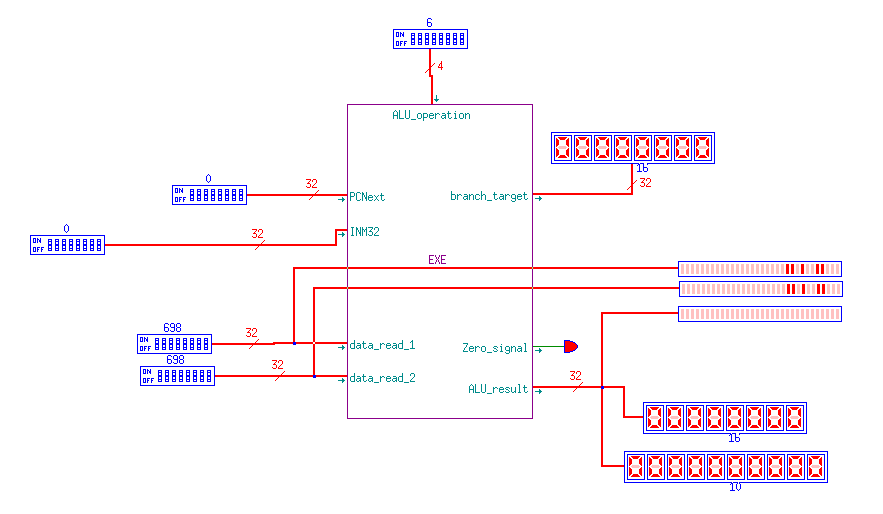
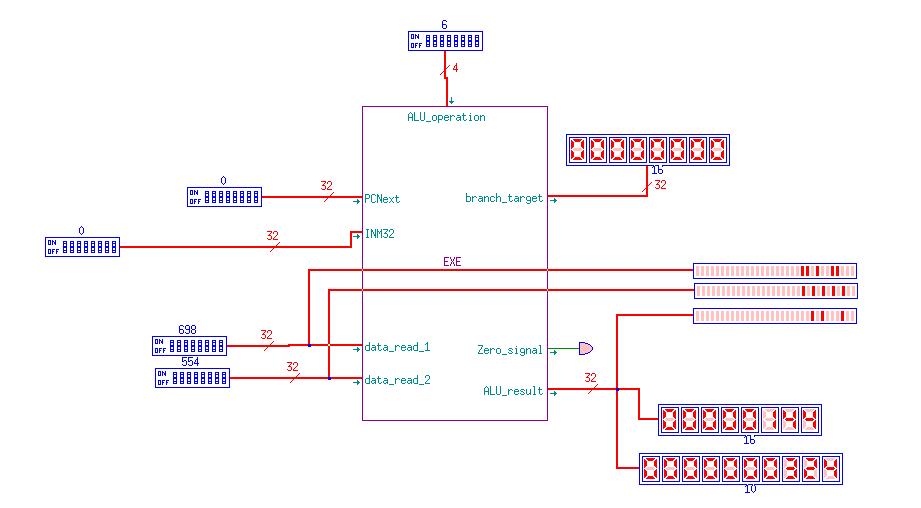
2.- OR



3.-SUMA



4.- Resta amb valor diferent de 0

4.- Resta amb valor igual a 0

APUNTS

Dades que ha donat el profe:

* EL PC ja agrupa de 4 en 4 (adreçament de memòria a 32 enlloc de 8 bits), de manera que al fer ep PC+4 haurem de fer PC+1 en el tkgate
* Degut a això, no fara falta fer un shift register de 2 (no hem d'afegir aquests 2 bits de més)