

# Arquitectura de computadores

## Pràctica 5

24 de maig de 2017

Entrada i sortida de dades

Practica optativa

Antoni Escobet

# PRÀCTICA 5 – MIPS: Entrada i sortida de dades

## 1. Objectius

En aquesta practica es pretén:

- Profunditzar en els conceptes relacionats amb les entrades i sortides d'un sistema microprocessador MIPS.
- Ampliar el camí de dades (conjunt unitat central de processos (CPU), memòries i dispositius d'entrada i sortida) estudiada a la classe de teoria.
- Amplia els coneixements sobre el llenguatge VHDL i l'edició amb QUARTUS II.
- Veure el funcionament de la transferència de dades per interrupció.

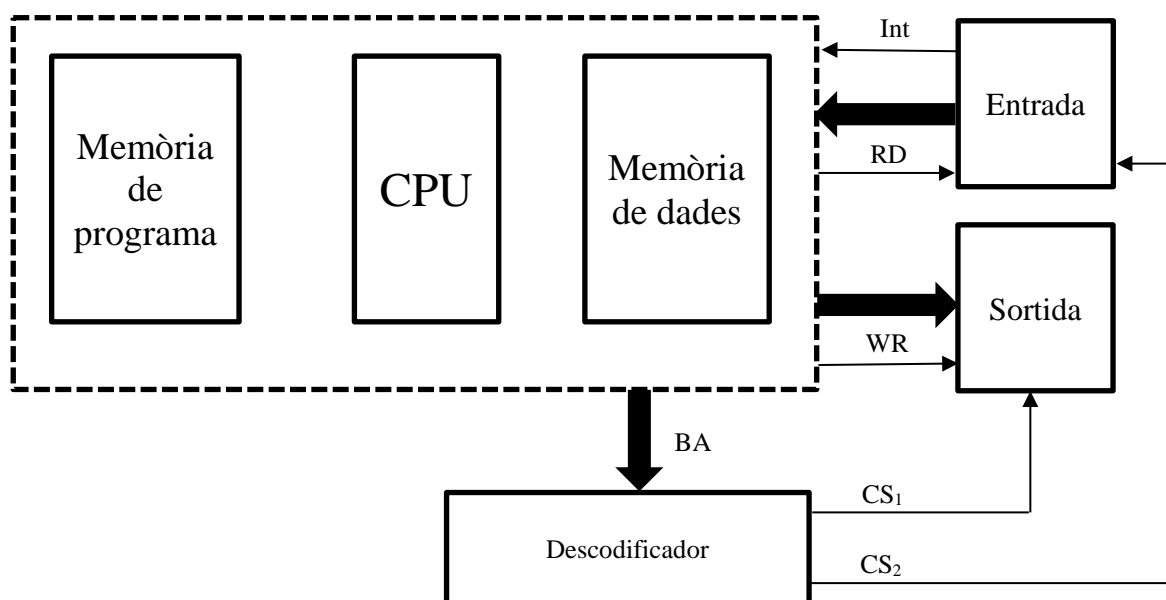
## 2. Material

L'únic material necessari per la realització d'aquesta pràctica és el paquet de programari d'ALTERA, el Quartus instal·lat als ordinadors del laboratori, i que us podeu descarregar gratuïtament de la pàgina web d'Altera (<http://www.altera.com>).

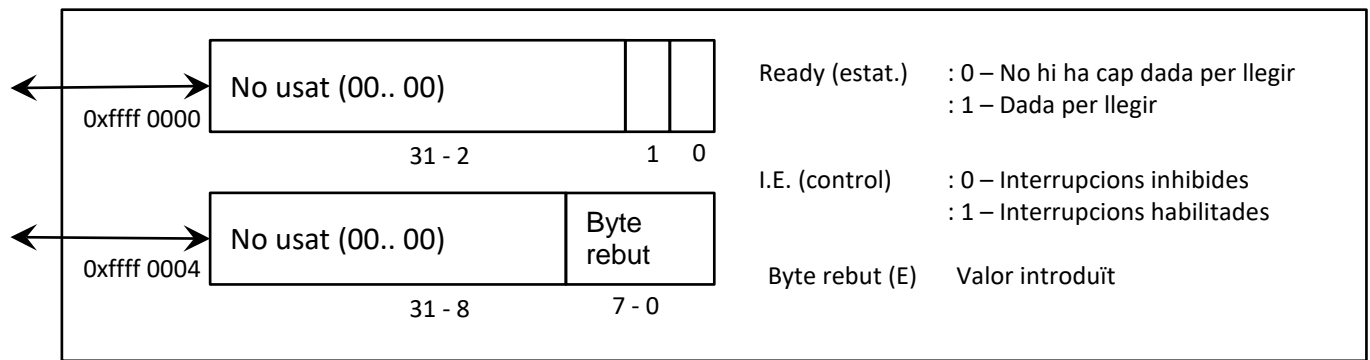
La simulació del vostre disseny s'ha de fer amb el simulador que proporciona el mateix paquet de programari d'Altera (ModelSim-Altera)

## 3. Problema proposat

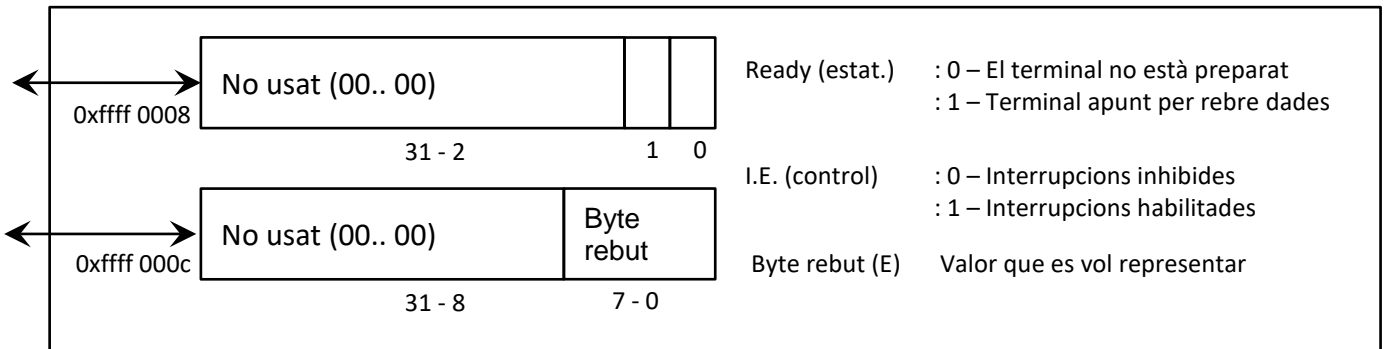
En aquesta pràctica heu d'ampliar disseny del vostre processador, per connectar-lo a l'exterior. Es tracta d'una connexió digital molt simple mitjançant uns ports d'entrada i sortida de 8 bits.



El port d'entrada té el mateix format que l'entrada del teclat del simulador SPIM. Està format per dos registres de 32 bits, el primer és el registre de control i està ubicat a l'adreça 0xFFFF0000, el segon és el registre de dades ubicat a l'adreça 0xFFFF0004.



El port de sortida representa la sortida a la pantalla. També està format per dos registres: Control a l'adreça 0xFFFF0008 i dades a l'adreça 0xFFFF000c.



## 4. Realització pràctica

### 4.1. Exercici 1

Feu les modificacions necessàries al vostre MIPS monocicle per poder connectar aquest dos ports i que sigui capaç de gestionar les interrupcions generades des del port d'entrada. Fixeu la rutina de servei d'interrupció a una adreça vàlida de la vostre memòria de programa.

### 4.2. Exercici 2

Implementeu una nova instrucció per poder retornar de la subrutina de servei d'interrupció.

```
rfe    010000 10000000000000000000 010000
```

Recordeu que quan es genera una interrupció, el coprocessador zero és l'encarregat de gestionar-la. S'ha de guardar l'adreça de retorn i la paraula d'estat, en el nostre cas, els flacs de zero i de carry. Per no complicar el disseny del nostre processador, enlloc d'utilitzar els registres del CP0, podem usar dos dels registres interns de la CPU.

### 4.3. Exercici 3

Feu un programa que en el moment en que es detecta una interrupció, llegeixi el port d'entrada i carregui al port de sortida el valor en BCD de l'entrada binària que s'ha activat.

Si s'ha activat el bit de menys pes de l'entrada, s'ha de mostrar un zero en BCD, si s'ha activat el bit de més pes (bit 7), s'ha de mostrar un 7.

#### **4.4. Exercici 4**

Demostreu el correcte funcionament del nou processador. Mostreu el seu correcte funcionament al simulador.

*Al tractar-se d'una practica optativa s'ha de demostrar el funcionament correcte de la pràctica abans del dia 26 de juny.*

*En tots els casos, heu de presentar el codi VHDL realitzat per a cadascun dels exercicis, així com les simulacions que demostrin el funcionament correcte dels circuits realitzats*