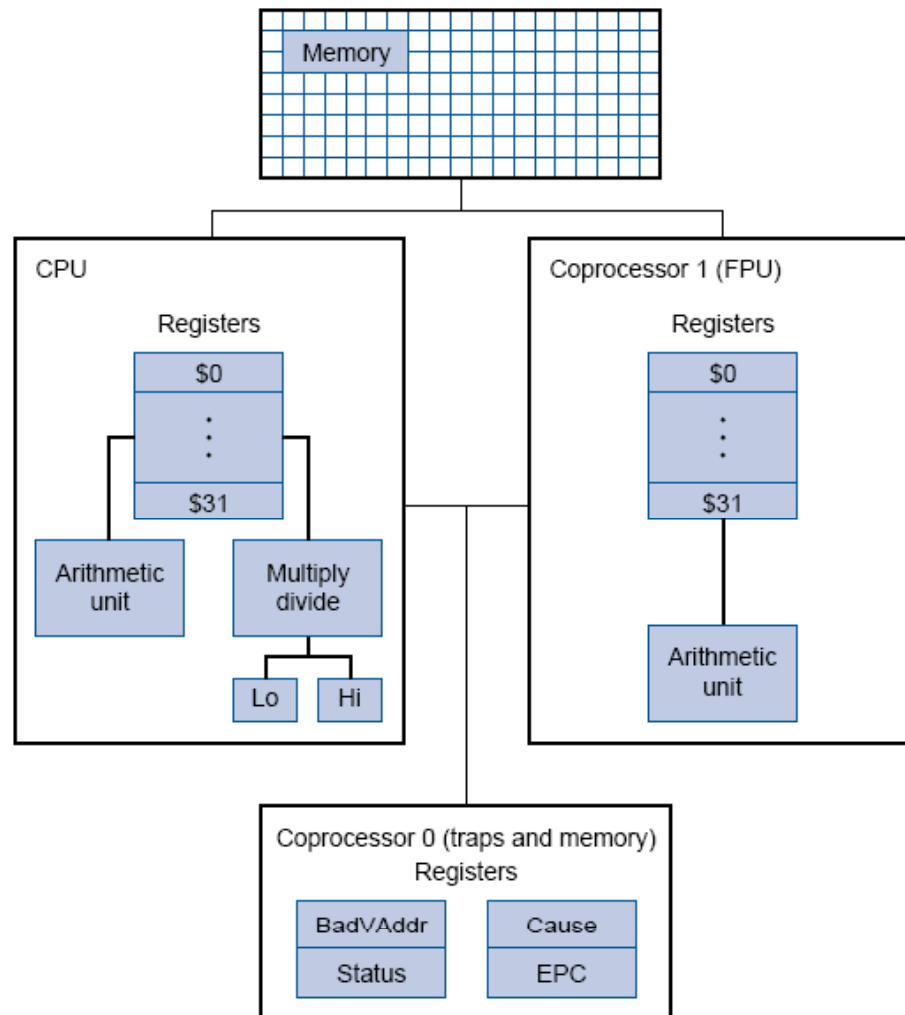


Exceptions and Interrupts

Exceptions and Interrupts

- Exceptions:
 - Para responder a errores de ejecución
- Interrupts:
 - Para atender I/O

Exceptions



Coprocessor 0:

Almacena los datos necesarios para tratar las excepciones y interrupciones.

Exceptions

Registros implementados por SPIM:

Register name	Register number	Usage
BadVAddr	8	memory address at which an offending memory reference occurred
Count	9	timer
Compare	11	value compared against timer that causes interrupt when they match
Status	12	interrupt mask and enable bits
Cause	13	exception type and pending interrupt bits
EPC	14	address of instruction that caused exception
Config	16	configuration of machine

Exceptions

Acceso a los registros del coprocesador 0

Move from coprocessor 0:

`mfc0 rt, rd`

$Rt \leftarrow Rd$

Move register `rd` in a coprocessor to CPU register `rt`.

Move to coprocessor 0:

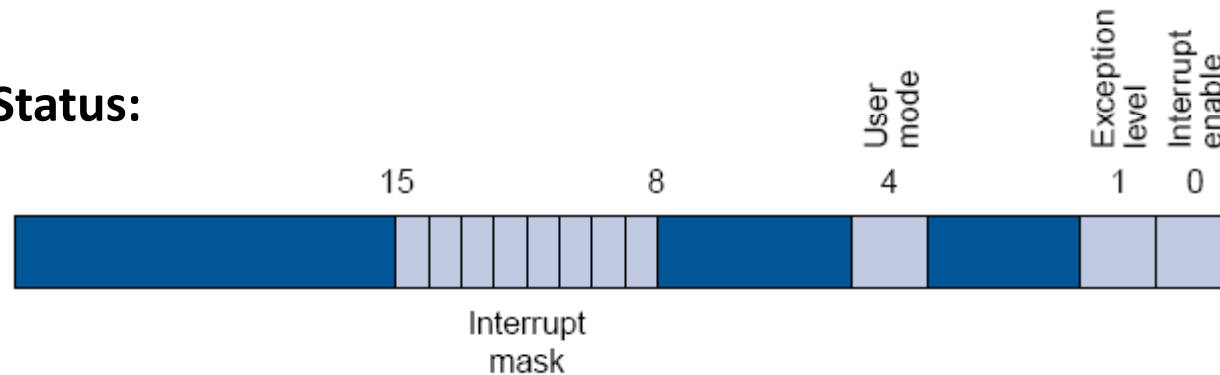
`mtc0 rd, rt`

$Rd \leftarrow Rt$

Move CPU register `rt` to register `rd` in a coprocessor.

Exceptions

Registro de Status:



Interrupt mask:

- 6 niveles hardware y 2 niveles software de interrupciones.
- bit a 1 permite interrupciones.

User mode bit:

- 0 si modo *Kernel* y 1 modo *User* (SPIM siempre 1).

Exception level bit:

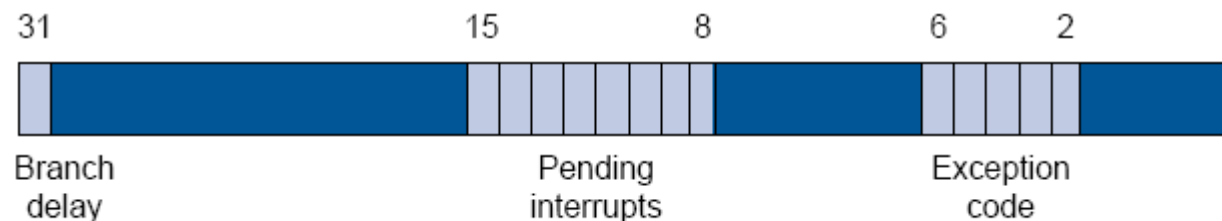
- 0 se permiten, 1 se deshabilitan interrupciones (evitar conflictos).

Interrupt enable:

- 0 NO se permiten interrupciones, 1 se permiten.

Exceptions

Registro de Cause:



Branch delay bit:

- Valor a 1 cuando la ultima excepción ocurrió durante el slot delay de un branch. Para saber la instrucción que fallo debemos mirar EPC+4.

Pending interrupt mask:

- Cuando se genera una interrupción siempre se actualiza registro "Cause" y se activa el bit de interrupción pendiente.

Exception code:

- Causa de la interrupción.

Exceptions

Exception codes:

Number	Name	Cause of exception
0	Int	interrupt (hardware)
4	AdEL	address error exception (load or instruction fetch)
5	AdES	address error exception (store)
6	IBE	bus error on instruction fetch
7	DBE	bus error on data load or store
8	Sys	syscall exception
9	Bp	breakpoint exception
10	RI	reserved instruction exception
11	CpU	coprocessor unimplemented
12	Ov	arithmetic overflow exception
13	Tr	trap
15	FPE	floating point

Exceptions

Ejemplo: exceptions.s

Interrupts: I/O

SPIM disponen de un Terminal con dos funcionalidades independientes.

Receiver:

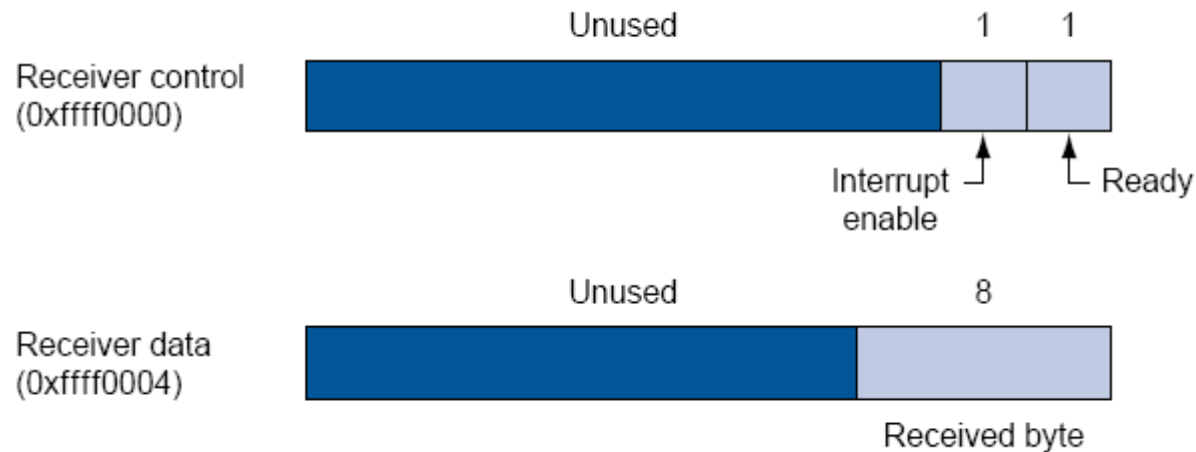
- Lee los caracteres del teclado.

Transmitter:

- Muestra caracteres por pantalla.

Ambos son completamente independiente, si queremos que una tecla pulsada por teclado sea mostrada en pantalla hace falta hacer echo.

Interrupts: I/O



Interrupt enable bit (rw):

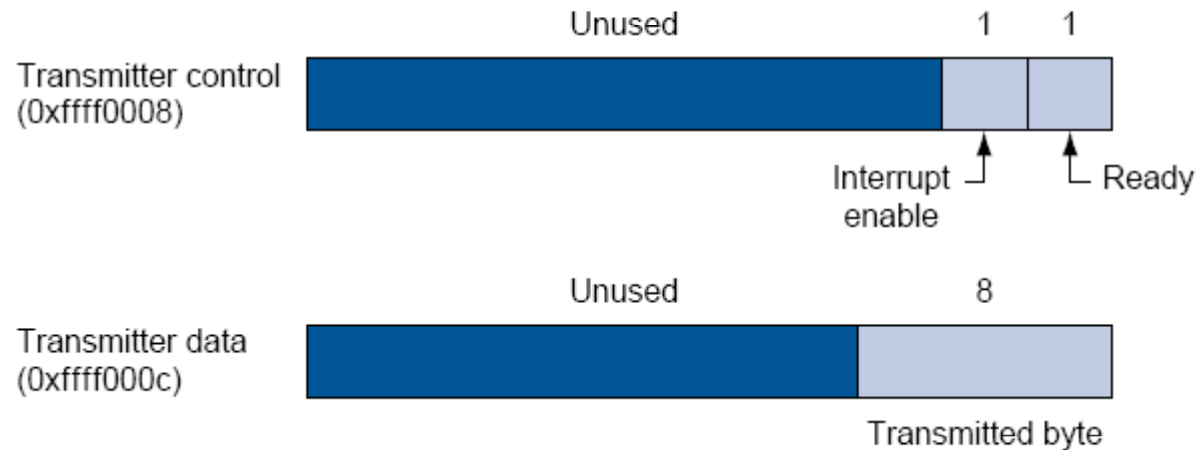
- 1 acepta pulsaciones de teclado.
- 0 no acepta pulsaciones, valor por defecto.

Ready bit (read-only):

- 1 hay un carácter listo para leer.

Nota: Para poder leer hace falta que también estén habilitadas las interrupciones a nivel del registro de Status.

Interrupts: I/O



Interrupt enable bit (rw):

- 1 acepta un nuevo carácter.
- 0 ocupado.

Ready bit (read-only):

- 1 puede aceptar un nuevo carácter para la pantalla.
- 0 esta ocupado escribiendo un carácter por pantalla.

Interrupts: I/O

Ejemplos:

-in-out print.s

-In-out read.s