

IIINSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE  
OCCIDENTE

DIPLOMADO VERIFICACIÓN PRE-SILICIO

ARQUITECTURA DE COMPUTADORAS

**Tarea 3: Register File**

Profesor: Dr. José Luis Pizano Escalante

Alumno: Cesar Enrique Alvarado Plascencia



## Register File

El register file es un módulo funcional donde se almacena físicamente los registros internos del procesador, en la micro arquitectura MIPS se manejan registros de 32 bits.

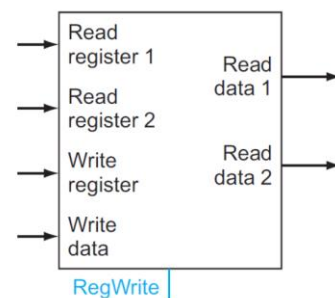
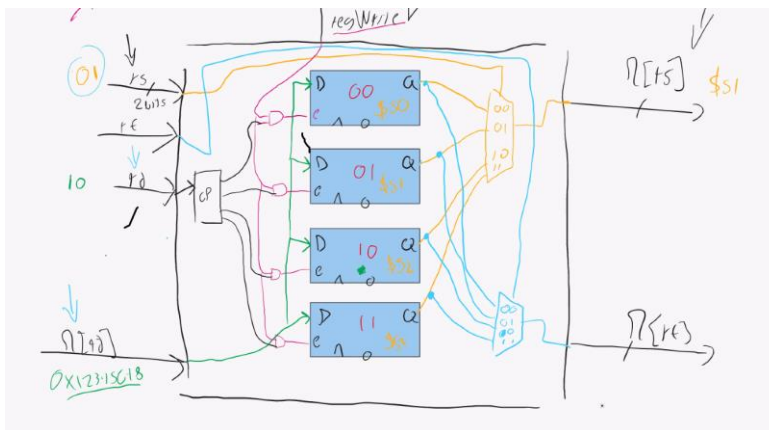
En la siguiente imagen se puede observar el White box del register file,

Señales de entrada

- Clk: señal que sirve para coordinar las acciones de varios circuitos.
- Reset: señal asíncrona que limpia los registros y los pone con valor de 0
- RegWrite: Uno de los dos habilitadores de los registros, se establece en 1 para que sea el módulo CP el que decida en que registro se va a escribir.
- WriteData: señal de 32bits que será la información que queremos escribir en el registro o en los registros deseados. (esta señal llega a todos los registros a la vez)
- WriteRegister: señal de 5 bits, con 32 posibles combinaciones, emplea un módulo CP que es un codificador de prioridad para elegir por medio de salida en one-hot (a excepción del registro 0) que registro vamos a habilitar para la escritura.
- Read\_Register: Dos señales de 5 bits encargadas de habilitar la lectura de datos con los multiplexores.

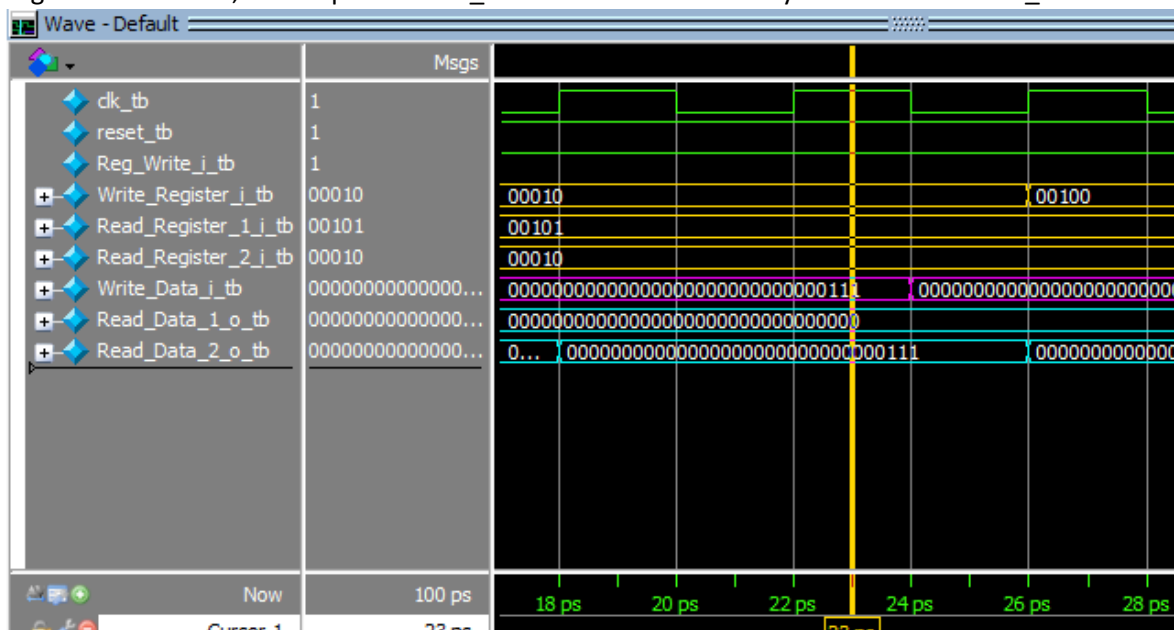
Señales de salida

- Read\_Data: Señal de 32 bits, es la lectura de los registros rs y rt.

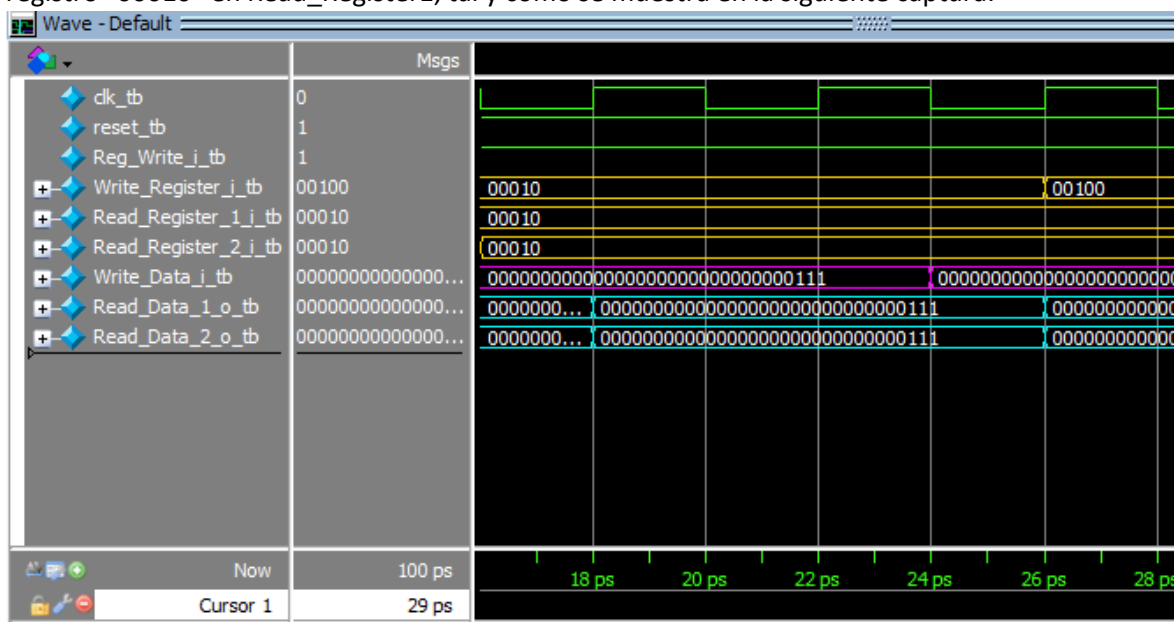


## Funcionamiento

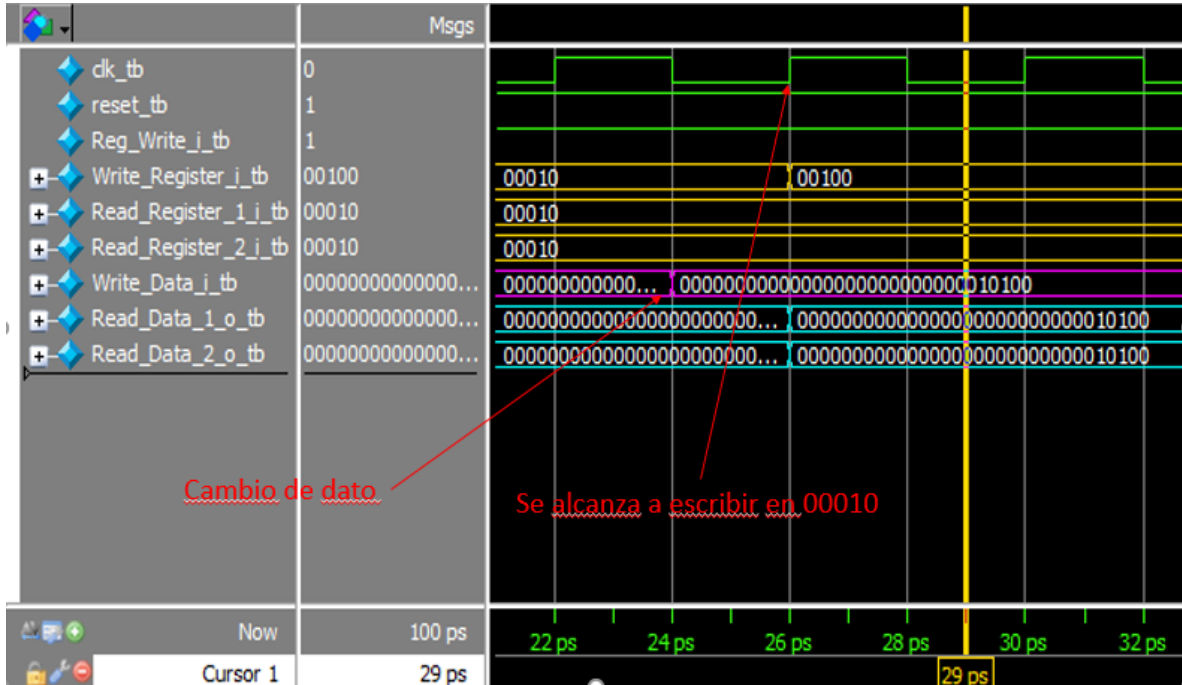
Se puede observar que tanto Write\_Register y Read\_Register2 están referenciando al mismo registro el "00010"; Por lo que la Write Data escribe en "00010" y se leerá en el Read\_Data2.



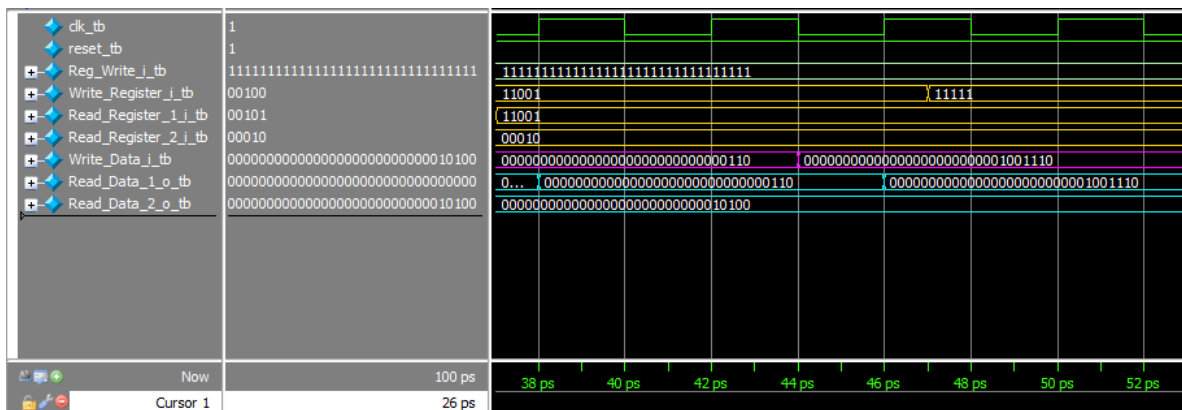
Si quisiésemos que esta misma entrada se refleje en Read\_Data1 tendremos que seleccionar el registro “00010” en Read\_Register1, tal y como se muestra en la siguiente captura.



En la siguiente imagen se observa, que la palabra (Write\_Data) ya cambio, y Read\_Register1 y 2 alcanzaron a escribir el dato a copiar el dato en Read\_Data2



En la siguiente imagen se cambia de dato y se habilita la escritura en el registro "11001", aquí solo Read\_Register1 es el que lee estos cambios. Ya que Read\_Register\_2 no cambio.



## \$ZERO

El registro numero 0 siempre debe valer 0 es por eso que al realizar el PC se queda sin 1. En la siguiente imagen se puede observar que a pesar de que hay un dato en Write\_Data, y que Write\_Register este habilitando el registro 0 para escritura, no se guarda nada en el debido a esta especificación de arquitectura. Este registro es muy útil para realizar instrucciones, sobre todo las de formato I. como, por ejemplo.

And Immediate      `andi`      I       $R[rt] = R[rs] \& \text{ZeroExtImm}$       (3)       $c_{hex}$

Or Immediate      `ori`      I       $R[rt] = R[rs] | \text{ZeroExtImm}$       (3)       $d_{hex}$

