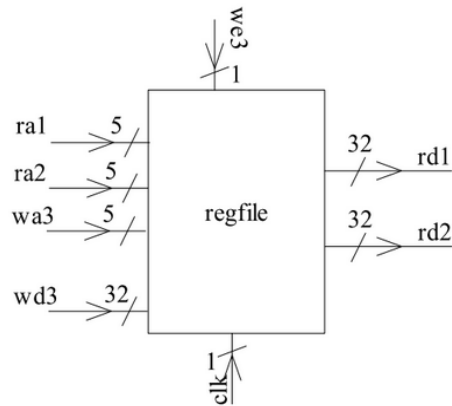


## Práctico N° 1 (Continuación)

Para todos los ejercicios se pide que respeten los nombres de las entidades, entradas y salidas. Representación de datos siempre en *std\_logic* o *std\_logic\_vector*. Internamente, se les da la libertad para que utilicen el diseño que consideren más adecuado siempre y cuando la salida sea la esperada.

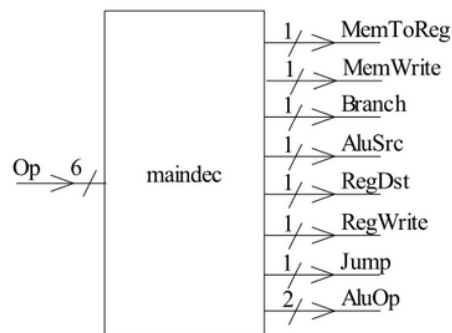
Ejercicio 10) Diseñe una memoria de doble puerto de salida y un puerto de escritura según el diagrama.



Nota: el contenido de la memoria en la dirección 0 podría no estar definido, en caso de que la dirección (*ra1* o *ra2*) sea 0, debe retornarse 0.

Implemente en VHDL, diseñe un *testbench* adecuado, simule con GHDL y GTKWave.

Ejercicio 11) Diseñe un decodificador según la tabla y el diagrama.



Op	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemToReg	Jump	AluOp
000000	1	1	0	0	0	0	0	10
100011	1	0	1	0	0	1	0	00
101011	0	0	1	0	1	0	0	00
000100	0	0	0	1	0	0	0	01
001000	1	0	1	0	0	0	0	00
000010	0	0	0	0	0	0	1	00
Others	-	-	-	-	-	-	-	--

Implemente en VHDL, diseñe un *testbench* adecuado, simule con GHDL y GTKWave.