

Problema entregable 2

Sea un procesador que implementa el algoritmo de Tomasulo sin especulación con las siguientes características:

- Las instrucciones enteras tienen un tratamiento análogo a las instrucciones de punto flotante para lo que:
 - Las unidades enteras tienen asociadas estaciones de reserva, donde las instrucciones aguardan a que se resuelvan sus dependencias LDE.
 - Las instrucciones enteras hacen uso del CDB para escribir sus resultados en el banco de registros enteros.
 - El TAG debe extenderse para considerar las ER enteras (int1, int2, etc.)
- Los recursos de planificación y las latencias de las unidades son las siguientes:

INT: ER=4, lat=1
FPadd: ER=2, lat=2
FPmul: ER=2, lat=5
Load: buf=2, lat=1
Store: buf=2, lat=1

- Se asume que existe una unidad funcional asociada a cada estación de reserva de cada uno de los tipos existentes, así como que las unidades funcionales de suma y multiplicación están segmentadas. Se asume también que el delay slot de los saltos es de una instrucción.
- Las instrucciones dependientes no pueden ejecutarse en el mismo ciclo que se escribe en el CDB el dato que estaban aguardando.

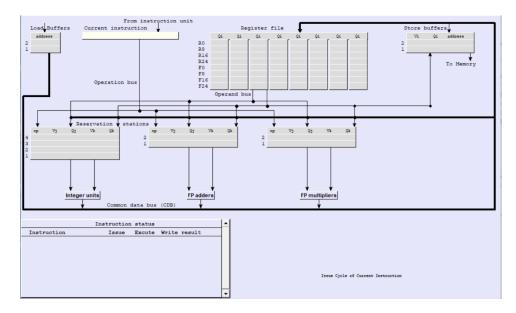


Figura 1: MIPS con Tomasulo

Universidad Complutense de Madrid Facultad de Informática Curso 2021 – 2022



Arquitectura de ComputadoresMarkel Álvarez Martínez

Determinar el estado del procesador tras la ejecución del segundo salto del siguiente código:

ld f2, a
 add r1, r0, xtop
loop: ld f0, 0(r1)
 sub r1, r1, #8
 multd f4, f0, f2
 bnez r1, loop
 sd 8(r1), f4

Datos

Op. int FP add. FP mul. Load Store Lat. 1 2 5 1 1

Ciclo 12			
Instrucción	Issue	Execute	Write
LD FZ,A	1	2	3
ADD RI, RO, KTOP	2.	3	4
LD FO, O(RI) SUB RI, H8 MULTD FY, FO, FZ BNEZ RI, LOOP SD 8(RI), FY	3 4 5 6 7	5. 7-11 8	12 9 14
LD FO, O(RI) SUB RI, RI, #8 HULTD FY, FO, FZ BNEZ RI, LOOP SD 8(RI), FY	8 9 10 11	9 10 11-15 12 4	10 16 13 18

Esta obra está bajo una licencia Creative Commons "Atribución-NoComercial-CompartirIgual 4.0 Internacional".

