

4)

a) Ejecutar en simulador con Forwarding habilitado. ¿Qué tarea realiza? ¿Cuál es el resultado y dónde queda indicado?

b) Re-Ejecutar el programa con la opción Configure/Enable Branch Target Buffer habilitada. Explicar la ventaja de usar este método y cómo trabaja.

c) Confeccionar una tabla que compare número de ciclos, CPI, RAWs y Branch Taken Stalls para los dos casos anteriores.

Respuestas:

4)

a) El programa revisa si un numero se encuentra en una tabla, indica con 1 en el registro r10 si el numero se encuentra, o 0 en caso contrario. Para el caso particular en el cual se busca el numero 7 (el cual se encuentra en la tabla) tenemos que r10 finaliza con 1 lo que indica que el 7 efectivamente se encuentra en la misma.

b) Al tener la tecnica de Hardware Branch Target Buffer es una prediccion dinamica que permite a traves de una tabla con informacion sobre si se dio o no el salto en las anteriores veces realizar una prediccion, la misma nos permite en el bucle poder hacer predicciones mas certeras y no tener tantos Branch Taken Stalls. A medida que el bucle tiene mas iteraciones la tabla es mas amplia y permite realizar aun mejores predicciones lo que hace que sea muy buena para bucles.

c)

	Disable Branch Target Buffer	Enable Branch Target Buffer
Cycles	71	67
Cycles Per Instruction (CPI)	1651	1558
RAWs	16	16
Branch Taken Stalls	8	4