PRÁCTICO U4.1: DESPACHO MÚLTIPLE ESTÁTICO

ARQUITECTURA II

Ejercicio 1 En este ejercicio se compara el desempeño de procesadores de despacho simple y de despacho dual, tomando en cuenta las transformaciones en el código que se pueden hacer para optimizar la ejecución en despacho dual. Los problemas de este ejercicio se refieren al siguiente loop (escrito en lenguaje C):

```
for ( i =0; i != j ; i +=2)
    b[ i ]=a[ i ]-a[ i +1];
```

Al escribir el código MIPS, se debe asumir que las variables son guardadas en los registros especificados más abajo. Los otros registros, excepto aquellos indicados como **Libres**, se usan para otras variables y no se los puede usar.

i	j	a	b	**Libres**
R5	R6	R1	R2	R10, R11, R12

- *Parte* 1 Traducir este código C a instrucciones MIPS. La traducción debe ser directa, sin ordenar las instrucciones para lograr mejor desempeño.
- *Parte* 2 Si la salida del loop ocurre después de ejecutar solamente dos iteraciones, dibujar el diagrama de ciclos para el código MIPS de la *Parte* 1, considerando que ejecuta sobre un procesador dual[1] como el mostrado en la Figura 1. Se asume que el procesador tiene una predicción de salto perfecta y que puede recuperar cualquier par de instrucciones (no solamente instrucciones consecutivas) en el mismo ciclo.
- *Parte* 3 Ordenar el código de la *Parte* 1 para conseguir un mejor desempeño sobre un procesador de despacho dual estático como el de la Figura 1.
- *Parte* 4 Repita la *Parte* 2 pero usando el código MIPS de la *Parte* 3.
- *Parte* 5 ¿Cuál es la ganancia de pasar de un procesador de despacho simple a un procesador de despacho dual como el de la Figura 1? Usar el código obtenido en la *Parte* 1 sobre una arquitectura de despacho simple y luego sobre una de despacho dual. Asuma que se ejecutan 1.000.000 de iteraciones del loop. Al igual que se hizo en la *Parte* 2, asuma que se cuenta con una predicción de saltos perfecta y que el procesador de despacho dual puede recuperar dos instrucciones cualesquiera en el mismo ciclo.
- *Parte* 6 Repita la *Parte* 5 pero esta vez considere la siguiente restricción en el procesador dual: en cada ciclo se puede ejecutar una instrucción de cualquier clase y la otra debe ser una instrucción que no acceda a memoria.

[1] Como los paquetes pueden contener cualquier tipo de instrucción, pueden haber riesgos estructurales entre las instrucciones del paquete.

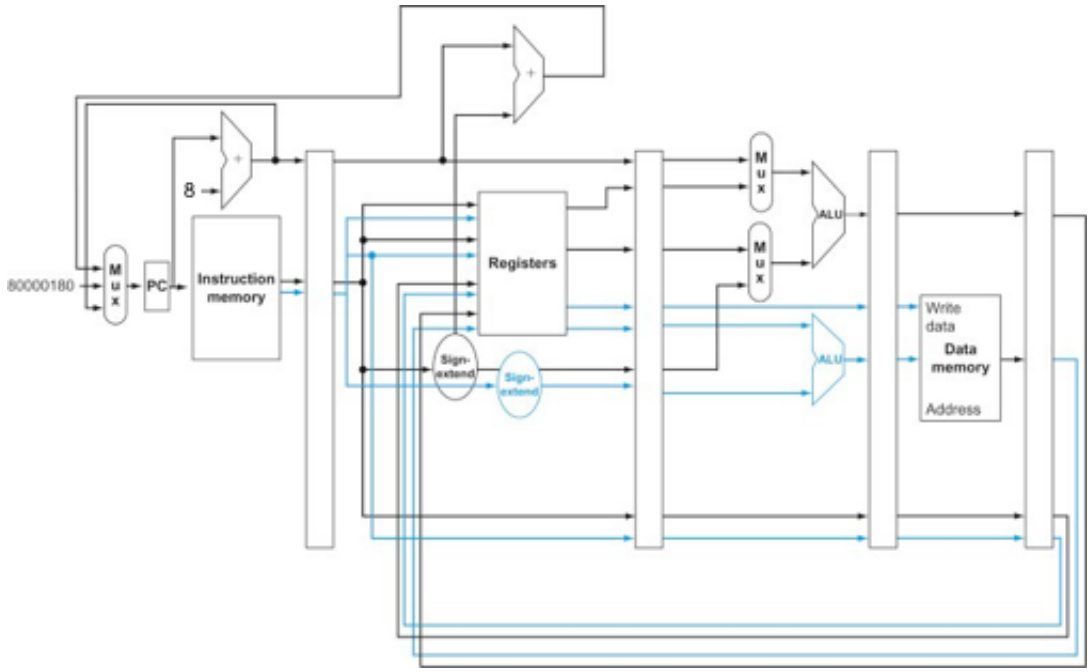

FIGURA 1. **Un camino de datos de emisión dual estática.**

Ejercicio 2 Analizar el siguiente código y resolver suponiendo una arquitectura MIPS:

```
                $1 = 2*n;  $3 = 0;
LOOP:
                LD      $2,  0xA000($3)
                SW      $2,  0xF000($3)
                ADDI    $3,  $3,    4
                ADDI    $1,  $1,   -1
                BNE     $1,  $0,   LOOP
                NOP
```

...$3 es usado cuando termina el loop

- *Parte* 1 Desenrolle el loop utilizando renombramiento (notar que $1 inicialmente es par).
- *Parte* 2 Para el código obtenido en el punto anterior, reordenar las instrucciones para evitar atascos y realizar la planificación estática de los paquetes para una arquitectura con despacho dual. En este caso los paquetes contienen un par de instrucciones ALU/branch-load/store como se vió en teoría.
- *Parte* 3 ¿Cuál es el IPC alcanzado en la *Parte* 2?