Trabajo Práctico Número 5

Arquitectura de Computadoras

Esp. Ing. Esteban Volentini (evolentini@herrera.unt.edu.ar)
Ing. Lucas Castillo (lecastillodelacroix@herrera.unt.edu.ar)

http://www.microprocesadores.unt.edu.ar/arqcom/

¿De qué se trataba este TP?

- Predicción de saltos estática y dinámica.
- Caminos de datos superescalares.
- Emisión múltiple de instrucciones.
- Ejecución de instrucciones fuera de orden.
- Manejo de interrupciones y excepciones.
- Optimización de código: desenrollado lazos.
- Renombramiento de registros.

Planteo del problema

- Ejercicio del Primer Parcial en 2019.
- Se desea comparar variantes
- Una suma de vectores C = A + B
- Cada vector tiene 100 elementos
- Cada elemento es de 32 bits
- Todos se ubican en direcciones contiguas
- Primer vector a partir de la dirección 0

Planteo del problema

- El código para la suma es el siguiente:

```
        li   a0,0          # Puntero al primer elemento
        li   a1,400        # Limite de un vector
loop:   lw   t0,0(a0)      # Carga del primer vector
        lw   t1,400(a0)    # Carga del segundo vector
        add  t0,t0,t1      # Suma los dos elementos
        sw   t0,800(a0)    # Almacena en el resultado
        addi a0,a0,4       # Incrementa el puntero
        bne  a1,a0,loop    # Compara con el limite
```

Planteo de la solución

a) ¿Cuánto demora en ejecutarse el código en la mejor implementación del procesador segmentado de cinco etapas visto en clase? ¿Cuántas instrucciones promedio por ciclo (IPC) se ejecutan? *Sugerencia: desprecie los tiempos de llenado y vaciado del pipeline.*

- Determinar la duración de cada instrucción
- Analizar los riegos y el adelantamiento

La solución en detalle

- Solo un riesgo no se puede resolver por adelantamiento
- Un load seguido de un instrucción que utiliza el mismo registro
- El lazo tiene esa secuencia de instrucciones
- Suponemos que hay predicción del salto y que no falla nunca
- Las seis instrucciones demoran 7T

La solución en números

- 100 elementos implican 100 iteraciones
- 100 iteraciones ejecutan 600 instrucciones
- 100 iteraciones demoran 700T
- $IPC = \frac{Instrucciones}{Tiempo} = \frac{600}{700} = 0{,}86$

Planteo de la solución

b) Desenrolle el código una vez, indique cuánto es la duración en este caso, cuánto es el IPC, y cuánto es la aceleración con respecto al tiempo calculado en el apartado (a).

- Se escribe dos veces el cuerpo del lazo
- Se escribe una vez el control del lazo
- Se corrigen las constantes
- Se renombran los registros
- Se reordenan las instrucciones

Desenrollado del cuerpo

```
loop: lw   t0,0(a0)     # Carga del primer vector
      lw   t1,400(a0)   # Carga del segundo vector
      add  t0,t0,t1     # Suma los dos elementos
      sw   t0,800(a0)   # Almacena en el resultado

      lw   t0,4(a0)     # Carga del primer vector
      lw   t1,404(a0)   # Carga del segundo vector
      add  t0,t0,t1     # Suma los dos elementos
      sw   t0,804(a0)   # Almacena en el resultado

      addi a0,a0,8      # Incrementa el puntero
      bne  a1,a0,loop   # Compara con el limite
```

Reordenando el cuerpo

```
loop: lw   t0,0(a0)     # Carga del primer vector
      lw   t2,4(a0)     # Carga del primer vector
      lw   t1,400(a0)   # Carga del segundo vector
      lw   t3,404(a0)   # Carga del segundo vector

      add  t0,t0,t1     # Suma los dos elementos
      add  t2,t2,t3     # Suma los dos elementos
      sw   t0,800(a0)   # Almacena en el resultado
      sw   t2,804(a0)   # Almacena en el resultado

      addi a0,a0,8      # Incrementa el puntero
      bne  a1,a0,loop   # Compara con el limite
```

La solución en números

- Ya no hay riesgos insalvables
- 100 elementos implican 50 iteraciones
- 50 iteraciones ejecutan 500 instrucciones
- 50 iteraciones demoran 500T
- $IPC = \frac{Instrucciones}{Tiempo} = \frac{500}{500} = 1$
- $A = \frac{Performance_b}{Performance_a} = \frac{Tiempo_a}{Tiempo_b} = \frac{700}{500} = 1{,}4$

Planteo de la solución

c) Desenrolle el código tres veces, indique cuánto es la duración en este caso, cuánto es el IPC, y cuánto es la aceleración con respecto al tiempo del apartado (a).

- Se escribe cuatro veces el cuerpo del lazo
- Se repite el mismo procedimiento anterior

Desenrollado del lazo

```
loop: lw   t0,0(a0)     # Carga del primer vector
      ...
      lw   t6,12(a0)    # Carga del primer vector
      lw   t1,400(a0)   # Carga del segundo vector
      ...
      lw   s0,412(a0)   # Carga del segundo vector
      add  t0,t0,t1     # Suma los dos elementos
      ...
      add  t6,t6,s0     # Suma los dos elementos
      sw   t0,800(a0)   # Almacena en el resultado
      ...
      sw   t6,812(a0)   # Almacena en el resultado

      addi a0,a0,16     # Incrementa el puntero
      bne  a1,a0,loop   # Compara con el limite
```

La solución en números

- Ya no hay riesgos insalvables
- Se acabaron los registros temporales
- 100 elementos implican 25 iteraciones
- 25 iteraciones ejecutan 450 instrucciones
- 25 iteraciones demoran 450T
- $IPC = \frac{Instrucciones}{Tiempo} = \frac{450}{450} = 1$
- $A = \frac{Performance_b}{Performance_a} = \frac{Tiempo_a}{Tiempo_b} = \frac{700}{450} = 1{,}56$

Planteo de la solución

d) Repita el apartado (a), pero considerando que el código se ejecuta en procesador con emisión de instrucciones estática doble, que es capaz de ejecutar simultáneamente una instrucción de tipo Ld/St junto con otra de tipo R/Branch.

- Analizar el despacho de instrucciones
- Determinar el tiempo de ejecución

Analisis de la ejecución

```
loop: lw   t0,0(a0)     |    nop
      lw   t1,400(a0)   |    nop
      nop               |    add  t0,t0,t1
      sw   t0,800(a0)   |    addi a0,a0,4
      nop               |    bne  a1,a0,loop
```

La solución en números

- Vuelve a aparecer un riesgo insalvable
- 100 elementos implican 100 iteraciones
- 100 iteraciones ejecutan 600 instrucciones
- 100 iteraciones demoran 600T
- $IPC = \frac{Instrucciones}{Tiempo} = \frac{600}{600} = 1$
- $A = \frac{Performance_b}{Performance_a} = \frac{Tiempo_a}{Tiempo_b} = \frac{700}{600} = 1,17$

Planteo de la solución

e) Repita el apartado (c), pero considerando el procesador dual-issue descrito en el apartado anterior..

- Analizar el despacho de instrucciones
- Determinar el tiempo de ejecución

Analisis de la ejecución

```
loop:  lw   t0,0(a0)      |    nop
       lw   t2,4(a0)      |    nop
       lw   t4,8(a0)      |    nop
       lw   t6,12(a0)     |    nop

       lw   t1,400(a0)    |    nop
       lw   t3,404(a0)    |    nop
       lw   t5,408(a0)    |    add  t0,t0,t1
       lw   s0,412(a0)    |    add  t2,t2,t3

       sw   t0,800(a0)    |    add  t4,t4,t5
       sw   t2,804(a0)    |    add  t6,t6,s0
       sw   t4,808(a0)    |    addi a0,a0,16
       sw   t6,796(a0)    |    bne  a1,a0,loop
```

La solución en números

- Se utiliza un reordenamiento agresivo
- Desaparecen los riegos insalvables
- 100 elementos implican 25 iteraciones
- 25 iteraciones ejecutan 450 instrucciones
- 25 iteraciones demoran 300T
- $IPC = \frac{Instrucciones}{Tiempo} = \frac{450}{300} = 1{,}5$
- $A = \frac{Performance_b}{Performance_a} = \frac{Tiempo_a}{Tiempo_b} = \frac{700}{300} = 2{,}33$

Planteo de la solución

f) Uno de sus compañeros sugiere que en un procesador con emisión de instrucciones estática triple, que es capaz de ejecutar simultáneamente una instrucción de tipo Ld/St junto con dos instrucciones de tipo R/Branch se podría mejorar aún más la performance de este código. ¿Qué piensa Ud. al respecto?

Análisis para la solución

- Tiempo de ejecución limitado por Ld/St
- Agregar más capacidad de ejecución en tipo R/Branch no mejora la performance
- Solo disminuiría la eficiencia en el uso de recursos del procesador
- Para mejorar debería ser dos instrucciones Ld/St junto con una de tipo R/Branch

Conclusiones Finales

- Desenrollar los lazos permite mejorar la performance
- Facilita reordenar instrucciones para evitar riegos insalvables
- También disminuye el impacto de las instrucciones de control
- Impacta en el uso de memoria y de registros

Resumen de la primera mitad

- Les mostramos una computadora por dentro.
- Les enseñamos a medir performance.
- Le enseñamos como se programa y criterios para mejorar la performance.
- Conocimos un procesador con CPI = 1 pero con un ciclo T grande
- Conocimos un procesador con CPI = 1 pero con un ciclo T pequeño
- Resolvimos los riegos de datos y de control

Conclusiones

- Próxima evaluación:
  - Miércoles 20 de Mayo a las 16:00
  - TPN6: Multiprocesamiento