

66:20 Organización de Computadoras

Trabajo Práctico 2: MIPS Datapath

Fecha de presentación 03/11/2011

1. Objetivos

Familiarizarse con el camino de datos de la arquitectura MIPS y la implementación del *pipeline* de cinco etapas.

2. Alcance

Este trabajo práctico es de elaboración grupal, evaluación individual, y de carácter obligatorio para todos alumnos del curso.

3. Recursos

Para la realización de este trabajo práctico, se requiere la herramienta de simulación WinDLX [1], la cual permite simular una arquitectura MIPS 32 con *pipeline* de cinco etapas, pudiendo ver los valores de cada registro dentro del archivo de registros, la ejecución de instrucciones dentro de cada etapa del *pipeline*, etc.

Esta herramienta es de fácil instalación, y corre en ambientes Linux-x86 utilizando WINE [4].

En [2] y [3] se encuentran ejemplos de programas para la ejecución en el simulador WinDLX.

4. Objetivo

Desarrollar y analizar el comportamiento de segmentos de código en lenguaje Assembly de MIPS que cumplan con las condiciones especificadas en cada ítem de este enunciado. Cada ítem, se debe desarrollar y analizar con la opción de *forwarding* sin habilitar y luego con esta opción activada.

1. Realizar las siguientes operaciones:

```
A=4
B=5
C=7
D=3
X=B+C
Y=D-C
Z=(((A+C)-B)*C)-X)/Y
A=X*Y*3
```

Donde las variables A,B,C,D están en memoria. La variable Z es de tipo double. Determinar el CPI para el conjunto de las operaciones, cantidad y tipo de Stalls.

2. Realizar un reordenamiento del código assembly implementado en el ítem anterior que disminuya el CPI.
3. Implementar el siguiente segmento de código en Assembly MIPS:

```
int a = 0;
int i = 0;

for(i = 0; i < 10; ++i){
    if( i % 2 ){
        a += i;
    }
    if( a > 11){
        break;
    }
}
```

Determinar el CPI para el conjunto de las operaciones, cantidad y tipo de Stalls. Verificar la posibilidad de utilizar *branch delay slot* para realizar una mejora en el tiempo de ejecución.

4. Realizar un reordenamiento del código assembly implementado en el ítem anterior que disminuya el CPI.
5. Implementar un segmento de código que con la adición de instrucciones de tipo *nop* mejore su desempeño. Si no fuera posible, explicar los motivos.

5. Informe

El informe deberá incluir

- Este enunciado.
- Descripción de los segmentos de código realizados que cumplan con los objetivos de los ítems de este trabajo práctico.
- Entregar los segmentos de código implementados en archivos separados con el nombre de cada ítem para poder ser probados en el simulador WinDLX.

6. Referencias

Referencias

- [1] WINDLX, <http://cs.uns.edu.ar/~jechaiz/arquitectura/windlx/windlx.html>
- [2] EJEMPLOS 1, <http://www.ecs.umass.edu/ece/koren/ece568/wdlx1.html>
- [3] EJEMPLOS 2, <http://www.ecs.umass.edu/ece/koren/ece568/wdlx2.html>
- [4] WINE, <http://www.winehq.org/>