1. Implementa en RTL la instrucción R 3 ← M [dir + A 0 ]. Su código de operación es 07. "dir"

es una dirección de memoria, su valor se especifica en la siguiente palabra del código de

instrucción (la primer palabra tiene el código de operación; la siguiente, el valor de dir). A 0

es un registro de direcciones, su valor se suma a "dir" para obtener el dato que se almacena en

R 3 .

2. Para la memoria cache de la siguiente figura, responde:

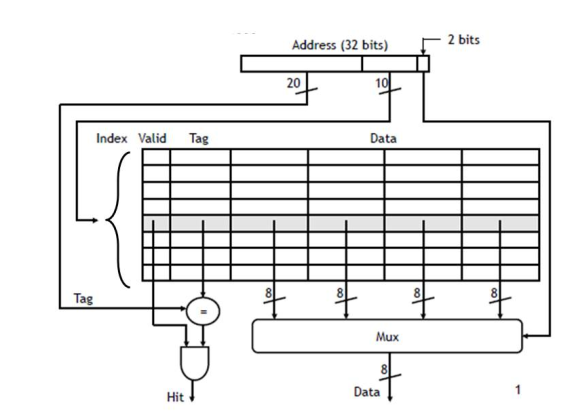
• ¿Cuántos sets (renglones) tiene?

• ¿Cuántos "ways" (bancos) tiene?

• ¿De qué tamaño es un bloque?

• ¿De qué tamaño es la memoria (incluye bits de Tag y valid bit)

• Si la escritura es write through, ¿tiene sentido que haya valid bit?



3. Calcule los ciclos por instrucción (CPI) de un procesador que tiene un CPI promedio para

operaciones en ALU de 1.1, para saltos de 3.0 y un tasa de aciertos (hit) en cache de 60%.

Un hit en cache toma un ciclo, mientras que un fallo (miss) tiene una penalización de 120

ciclos (no hay cache multinivel). Suponga que el 22% de las instrucciones son cargas, 12%

almacenamientos (guarda), 20% saltos, y las demás, operaciones en la ALU.

4. Considere el siguiente segmento de código, que se ejecuta en un procesador con pipeline de

cinco etapas:

ADD

LD

R3, R2, R1

R4, 4(R3)

(a) Muestre un diagrama de tiempos de la ejecución del pipeline si no se tienen atajos (blo-

que el pipeline en caso de conflictos de datos).

(b) Muestre un diagrama de tiempos de la ejecución del pipeline con atajos.

Identifique dos maneras en las que loop unrolling puede incrementar el desempeño de un

programa, y al menos una en la que puede decrementarlo.

6. Un procesador con despacho dinámico tiene tres unidades funcionales (FU): una LD/STORE

que toma dos ciclos, una ADD/SUB, de un ciclo de ejecución, y una MUL/DIV con 2 y 4

ciclos de ejecución, respectivamente. Tiene un solo arreglo de registros y una estación de

reservación con un espacio por FU. El procesador tiene una política de emisión y completado

de instrucciones fuera de orden (OOI, OOC).

Comenzando con la secuencia de instrucciones siguiente en el buffer de instrucciones, y con

los espacios libres en las estaciones de reservación, identifique el ciclo en el que la instrucción

será emitida y en el que escribirá su resultado.

LD

LD

MUL

SUB

DIV

ADD

R6, 34(R12)

R2, 45(R13)

R0, R2, R4

R8, R2, R6

R10, R0, R6

R6, R8, R2

7. conteste cierto o falso

En una arquitectura VLIW, CP I < 1

En una arquitectura Superescalar, CP I > 1

En una arquitectura Vectorial, CP I < 1

En una arquitectura Superpipeline, CP I ≈ 1

Las extensiones MMX son una forma de SIMD

La arquitectura DAXPY es la base de las computadoras Cray

En general, entre más etapas tenga un procesador pipeline, mejor será su desempeño

Algunos conflictos de control se resuelven agregando unidades funcionales

Los conflictos de datos son RAR, RAW, WAR, WAW

En procesadores digitales de señales (DSP), VLIW es más popular que una Arq. Vectorial

8. Seleccione la(s) respuesta(s) correcta(s)

Se maximiza el paralelismo en una arquitectura superescalar gracias a:

El hardware El programador El compilador

Se maximiza el paralelismo en una arquitectura VLIW gracias a:

El hardware El programador X El compilador

Se maximiza el paralelismo en una arquitectura EPIC gracias a:

El hardware El programador El compilador

Las arquitecturas Harvard ayudan a resolver los conflictos de:

Datos Control Estructurales

9. Considere el siguiente segmento de código

for (i = 0; i < 100; i ++)

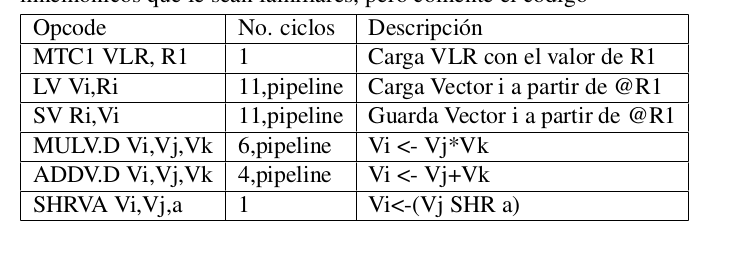
A[i] = ((B[i] \* C[i]) + D[i])/2;

(a) Escriba un programa en ensamblador que lo ejecute en el menor tiempo posible con las

instrucciones VMIPS que se muestran. Suponga que se tienen ocho registros vectoria-

les con 64 elementos cada uno. Para las operaciones escalares que necesite, utilice los

mnemónicos que le sean familiares, pero comente el código.

(b) ¿Cuánto tiempo toma ejecutar el código que desarrolló suponiendo que el procesador

tiene memoria traslapada en 16 bancos, no utiliza chaining y sólo tiene un puerto a mem-

ora (1 LD o STO por ciclo)

(c) Repita la pregunta anterior considerando chaining y un solo puerto a memoria.

10. (a) ¿Qué significan las operaciones "gather/scatter"?

(b) ¿Se puede hacer gather/scatter en una arquitectura vectorial con ISA VMIPS? Explique

muy brevemente

(c) ¿Se puede hacer gather/scatter en una arquitectura SIMD con CUDA? Explique muy

brevemente

11. Para la siguiente instrucción de CUDA, indique cuántos kernels, cuántos bloques y cuántos

hilos se invocan. Explique muy brevemente qué es un kernel, qué es un bloque y qué es un

hilo

cube<<<25, 64>>>(d\_out, d\_in);

12. Una arquitectura GPU permite maximizar:

La latencia El throughput| Ambos: Throughput y latencia Ninguno de los dos

Las líneas que aparecen en la tabla, son segmentos de código de un programa que calcula el

cuadrado de los elementos de un arreglo.

(a) Indique en qué orden deben aparecer las líneas en el código

cudaMalloc((void \*\*)&d\_in,sz);

sz = sizeof(float);

cudaFree(d\_in);

cudaMemcpy(h\_out, d\_out, sz, cudaMemcpyDeviceToHost);

free(h\_out);

float \*h\_in,\*h\_out,\*d\_in,\*d\_out;

CalculaSq<<< 4, 256 >>>(d\_out,d\_in)

h\_in = (float \*)malloc(sz);

(b) ¿Cuántos elementos tiene el arreglo?

(c) Escriba el código de la rutina "CalculaSq", que es la que calcula el cuadrado de los ele-

mentos.

14. ¿Qué estrategia siguen las GPU de NVIDIA si los hilos en un warp divergen en su ejecución?

• Con ayuda del registro de máscara, los hilos se mueven a distintos warps para que no

haya divergencia dentro de un mismo warp

• No pasa nada. Es una arquitectura SIMT en la que los hilos pueden divergir

• Se ejecutan todos los caminos posibles en serie por todos los hilos para que en realidad

no haya divergencia