

## PRÁCTICA 1: RENDIMIENTO

### Bibliografía de Referencia:

- [1] *"Measuring Computer Performance: A practitioners guide"* de David J. Lilja.
- [2] *"SPEC CPU2006 BENCHMARK SUITE: The Newest Iteration of a Widely Used Measure of Processor Performance"* de Harlan McGhan.
- [3] Application Report *"MSP430 Competitive Benchmarking"* de Texas Instruments.

1) Se desea mejorar el rendimiento de un computador introduciendo un coprocesador matemático que realice las operaciones en punto flotante en la mitad de tiempo respecto del disponible. Calcular la mejora en la velocidad del sistema para la ejecución de un determinado programa si el 60% del tiempo de ejecución del mismo se compone de operaciones aritméticas en punto flotante. Si el programa tarda 12 segundos en ejecutarse sin la mejora, ¿cuánto tardará con la mejora?

2) Una operación muy común en el procesamiento de gráficos es la raíz cuadrada. Las diferentes implementaciones de esta operación en punto flotante (FP) afectan considerablemente la performance de los procesadores. Suponga que la raíz cuadrada en punto flotante (FPSQR) es responsable del 25% del tiempo de ejecución de un benchmark de gráficos. El resto de las operaciones de punto (FPO) flotante representan el 20% del total. Se presentan dos propuestas de mejora para una cierta arquitectura. La primera propuesta consiste en mejorar el hardware de FPSQR en un factor de 5. La segunda alternativa es mejorar los tiempos de ejecución de todas las instrucciones de FP (FPSQR + FPO) por un factor 1,7.

- a) Seleccione la alternativa más provechosa, suponiendo que ambas modificaciones representan el mismo esfuerzo económico.
- b) Repita si se utiliza otro benchmark diferente en el que las FPSQR y las FPO representan el 25% y el 40% del tiempo de ejecución respectivamente.

3) Supongamos que se está considerando mejorar un procesador añadiéndole un modo vectorial<sup>1</sup>. Cuando se ejecuta un cálculo en modo vectorial el proceso es 20 veces más rápido que en el modo normal de ejecución (escalar). Llamemos *porcentaje de vectorización* al porcentaje de tiempo que puede emplearse el modo vectorial.

- a) Dibujar un gráfico donde se muestre la *mejora neta* (cuántas veces más rápido es el procesador modificado respecto del original) que puede obtenerse como una función del porcentaje de vectorización.
- b) ¿Qué porcentaje de vectorización es necesario para conseguir una mejora de 2 en los tiempos de ejecución?
- c) ¿Qué porcentaje de vectorización es necesario para conseguir la mitad de la mejora máxima alcanzable utilizando el modo vectorial?
- d) Suponga que se ha medido el porcentaje de vectorización de un grupo de programas, obteniéndose que es, en promedio, del 70%. El grupo de diseño de hardware asegura que puede duplicar la velocidad de la parte vectorizada con una inversión significativa de ingeniería adicional. Se desea saber si el equipo de compilación puede incrementar la utilización del modo vectorial como otra aproximación para incrementar el rendimiento. ¿Qué incremento en el porcentaje de vectorización (relativo a la utilización actual) se necesitaría para obtener la misma ganancia de rendimiento? En su opinión, ¿qué inversión sería más recomendable?

4) Un sistema de procesador tiene una jerarquía de memoria compuesta por una cache de nivel 1 (un ciclo de tiempo de acceso), caché de nivel 2 (10 ciclos de tiempo de acceso), y memoria principal (100 ciclos de tiempo de acceso). Las tasas de acierto son del 95% para la cache de

<sup>1</sup> Juego de instrucciones adicionales que permite operar grupos de registros (vectores) en forma simultánea.

nivel 1 y 97% para la de nivel 2. Las instrucciones de acceso a memoria son el 30% del total de instrucciones, y todas las demás instrucciones toman un único ciclo de procesador.

- a) ¿Cuál es el CPI promedio de la máquina?
- b) Se propone reducir el tamaño de la cache de nivel 2, lo que reduciría su tasa de aciertos hasta un 92% pero permitiría incrementar la frecuencia en un 10%. ¿Qué mejora obtiene la máquina modificada respecto de la original?
- c) Alternativamente se propone reducir el tamaño de la cache de nivel 1, lo que reduciría su tasa de aciertos hasta un 90% pero permitiría incrementar la frecuencia en un 12%. ¿Qué mejora obtiene la máquina modificada respecto de la original?
- d) Explique con sus palabras por qué un caso da una mejora y el otro una desmejora.

5) Considere dos procesadores con diferentes estrategias para el manejo de saltos condicionales:

Procesador A: Utiliza dos instrucciones para realizar el salto. La primera es una comparación que prepara los *flags* de condición. Estos son examinados por la segunda instrucción que es el salto condicional.

Procesador B: En una única instrucción realiza comparación y salto.

En ambos procesadores las instrucciones de salto condicional emplean dos ciclos de reloj y las demás instrucciones solamente un ciclo. Según estadística recopilada en un grupo de programas, el 20% de todas las instrucciones del **Procesador A** son saltos condicionales (recordar que habrá también 20% de instrucciones de comparación asociadas a estos saltos). La frecuencia de reloj del **Procesador A** es un 25% más elevada que la del **Procesador B**.

- a) ¿Cuál de los dos procesadores es más rápido? Justificar.
- b) Considere que se logra aumentar la velocidad del reloj del **Procesador B**, de modo que la diferencia con respecto al reloj del **Procesador A** se reduzca a un 10%. Nuevamente, se desea saber cuál de los dos procesadores es más rápido.

6) En una primera implementación de una determinada arquitectura, cuya CPU trabaja a 400Mhz, un programa de aplicación se ejecuta en 10 segundos. Se desea diseñar una nueva implementación para la misma arquitectura, tal que el mismo programa se ejecute en 6 segundos. Se optará por aumentar la frecuencia de reloj, lo cual se sabe de antemano que implicará un aumento del CPI a 1,2 veces el que se tenía. ¿Cuál deberá ser la nueva frecuencia de funcionamiento para lograr la mejora deseada?

7) Considere un procesador del tipo *carga/almacenamiento*, en el cual las únicas instrucciones con acceso a memoria son las de transferencia entre memoria y registros. Todas las demás instrucciones operan entre registros. Suponga que se ejecutan programas con las siguientes características:

Operaciones	Frecuencia	CPI
Aritmético / Lógicas	43%	1
Carga de registros	21%	2
Almacenamientos	12%	2
Saltos	24%	2

y que el 25% de las operaciones de la ALU utiliza un operando que debe ser cargado previamente y que no se vuelve a utilizar. Se propone agregar instrucciones aritmético/lógicas que tengan un operando fuente en memoria y que requieran 2 ciclos de reloj. Este cambio determina también que el CPI de los saltos aumente a 3 sin modificar el ciclo de reloj.

- a) Se desea saber si el cambio propuesto mejoraría el rendimiento de la CPU. Suponiendo que la duración del período del reloj es de 10ns, calcule los MIPS en cada caso.
- b) Considere el impacto que tendría utilizar un compilador optimizado que consigue reducir a la mitad la cantidad de instrucciones de la ALU, sin poder reducir cargas, ni almacenamientos ni saltos. Sacar conclusiones.

8) Del estudio de la frecuencia de utilización de instrucciones de los lenguajes de alto nivel surge que las llamadas a los procedimientos son una de las operaciones más costosas. Suponga que ha diseñado una modificación para un procesador que reduce las operaciones de carga y almacenamiento normalmente asociadas con las llamadas y retornos de procedimientos, a expensas de velocidad de reloj. Para verificar los resultados se realizan algunos experimentos con y sin esta optimización. Los experimentos utilizan el mismo compilador/optimizador en ambas versiones del computador.

Los experimentos realizados revelan lo siguiente:

- La versión no optimizada tiene una frecuencia un 5% mayor que la optimizada.
- El 30% de las instrucciones de la versión no optimizada son operaciones de carga o almacenamiento.
- La versión optimizada ejecuta 1/3 menos de operaciones de carga y almacenamiento que la versión no optimizada. Para las demás instrucciones el recuento se mantiene.
- Todas las instrucciones, incluyendo las de carga y almacenamiento, emplean un ciclo de reloj.

¿Qué versión del procesador es más rápida? Justificar cuantitativamente.

9) Del recuento de instrucciones de un determinado programa en condiciones normales de operación resulta que el 40% de las instrucciones son operaciones de punto flotante, de las cuales el 75% son sumas y el 25% multiplicaciones. Se dispone de un procesador que, con una frecuencia de operación de 800 MHz, emplea 20 ciclos para realizar una multiplicación en punto flotante, 10 ciclos para una suma y 5 ciclos para el resto de las instrucciones. Se propone una nueva implementación en la cual todas las operaciones de punto flotante emplean 10 ciclos y el resto 5 ciclos, pero se sabe que para que la nueva implementación sea viable deberá disminuirse la frecuencia de operación. Indique el límite inferior por debajo del cual no debería bajar la frecuencia para que la modificación propuesta no deje de ser una mejora.

10) Se desea analizar el impacto que tendría sobre una determinada arquitectura la inclusión de una nueva técnica de *carga especulativa*. Esta modificación permitiría reducir de 2 a 1 el CPI de la instrucción de carga (LOAD), pero tendría éxito sólo en un 70% de las veces que se ejecute la instrucción. Lamentablemente la frecuencia de reloj de la máquina debe reducirse para que la nueva técnica funcione. Suponiendo que las cargas representan el 15% de las instrucciones y que el resto de las instrucciones se ejecutan en 1 ciclo calcule el límite hasta el cual se justificaría reducir la frecuencia de reloj para implementar la nueva técnica.

11) El documento de la referencia [3] es un reporte publicado por Texas Instruments donde compara un procesador comercializado por dicha firma contra otros procesadores utilizados en el mismo segmento comercial (sistemas embebidos de muy bajo consumo).

La tabla lateral que se puede encontrar en la **página siguiente** es un extracto de los datos presentados por Texas para tres procesadores de interés. En ella se puede ver la performance de estos sistemas evaluada ejecutando tres benchmarks diferentes: Dhrystone, Whetstone, y un tercer benchmark sintético que calcula la salida de un filtro digital FIR (fundamentalmente operaciones aritméticas de suma y producto, e indexado de datos en memoria).

Para minimizar el impacto de la calidad del compilador sobre las mediciones se utilizó una misma familia de compiladores de lenguaje C en todos los casos. Además se recopilaron datos con y sin optimización (*opt* y *unopt* en la tabla, respectivamente) en dos dimensiones de diferentes de la performance que son importantes en el segmento de los sistemas embebidos de bajo consumo: el tamaño del código en bytes y la cantidad de ciclos de ejecución que insume la ejecución de cada programa.

- Las mediciones de código se muestran sin ninguna optimización y con optimización para espacio (minimizar el tamaño del código).
- Las mediciones de ciclos se muestra sin ninguna optimización y con optimización para velocidad (minimizar la cantidad de ciclos de ejecución).

A partir de lo anterior, analice los datos y responda:

- a) Suponiendo que todos los demás factores permanecen iguales, ¿cuál es la mejora de utilizar la optimización de velocidad del compilador? Haga un cuadro con las mejoras para cada combinación de cada procesador y cada benchmark. ¿Todos los benchmarks se benefician por igual de la optimización? ¿Hay alguna anomalía en los datos presentados por Texas?
- b) ¿Cómo definiría la “mejora” en las mediciones de tamaño del código al aplicar la optimización correspondiente? Haga una tabla para cada combinación. ¿Algún benchmark se beneficia particularmente de esta optimización?
- c) Sabiendo que la medición del tiempo de ejecución del Dhrystone presente en la tabla corresponde a la ejecución de 100 iteraciones del bucle central del benchmark, ¿es posible determinar cuántos Dhrystones es el rendimiento de cada procesador? Si sí, calcúelos. Si no, justifique por qué e indique qué datos adicionales harían falta.
- d) Si ahora además sabemos que en todos los casos el reloj del sistema es de 8 Mhz, ¿es posible determinar los MIPS obtenidos ejecutando cada benchmark? ¿Y los DMIPS? Justifique.
- e) El ARM710T es un procesador de diseño muy semejante al ARM7TDMI que figura en la tabla anterior, con la única diferencia de incorporar una cache unificada (instrucciones + datos) de 8kB para mediar entre el procesador y una memoria principal más amplia (y por lo tanto de acceso más lento) que la del ARM7TDMI.

Asumiendo (correctamente) que el código compilado para el ARM710T ocupa el mismo espacio que para el ARM7TDMI, ¿cuántas veces cabe el código del Dhrystone en la cache del procesador? Discuta qué impacto tiene la jerarquía de memoria en la medición de performance del Dhrystone en estas condiciones: ¿se puede esperar que el benchmark arroje resultados muy diferentes para un ARM7TDMI y un ARM710T? Justifique las respuestas.

- 12) Continuando con el ejercicio anterior, considere ahora los siguientes datos extraídos de las hojas de datos de los procesadores anteriores:

Procesador	Consumo en modo ACTIVO	Consumo en modo LOW-POWER
MSP430	0.115 mA/Mhz (@3V)	1.9 uA (@3V)
ATxmega64	1.250 mA/Mhz (@3V)	7.0 uA (@3V)
STR710RZ (ARM7TDMI)	1.100 mA/Mhz (@3V)	10.0 uA (@3V)

Donde el consumo ACTIVO es el que se obtiene en promedio al mantener el procesador en estado plenamente funcional (ejecutando programa), mientras que el modo LOW-POWER es algún modo inactivo (sin ejecución de instrucciones) donde el procesador mantiene solamente los sistemas indispensables en funcionamiento.

En una aplicación de monitoreo ambiental remoto, un sistema embebido debe activarse 100 veces por segundo ejecutando una tarea de procesamiento en cada activación. Dado que el sistema debe funcionar con baterías, el sistema pasa a modo de bajo consumo entre activaciones para conservar energía.

Se desea analizar el impacto de la elección del procesador en la duración de la batería, calculando cada cuántos días deberán

Microcontroller	FIR Filter						Dhrystone						Whetstone					
	Code Size			Cycles			Code Size			Cycles			Code Size			Cycles		
	Unopt	Opt	Unopt	Opt	Unopt	Opt	Unopt	Opt	Unopt	Opt	Unopt	Opt	Unopt	Opt	Unopt	Opt		
MSP430F5438	988	980	111607	107146	1194	780	160672	98039	5776	4726	106451	105651	106451	105651	106451	105651		
ATxmega64A1	1422	1362	250732	242008	2117	1453	269463	185295	7288	4594	114084	173354	114084	173354	114084	173354		
ARM7TDMI (Thumb)	1548	1528	37827	33114	1616	1000	83798	52352	11488	10532	61600	60444	61600	60444	61600	60444		

reemplazarse las baterías para cada una de las tres alternativas posibles (los tres procesadores anteriores) sabiendo qué:

- Todos los sistemas funcionan a partir de un reloj de 8 Mhz.
- Las baterías utilizadas tienen una capacidad de 2200 mAh ("miliamperes Hora").
- Cada activación de la tarea periódica insume 5000 ciclos en el procesador MSP430.
- Por las características del algoritmo a ejecutar se puede suponer que el benchmark del filtro FIR compilado con optimización para velocidad es representativo de la performance relativa en cada uno de los tres sistemas al ejecutar la tarea de procesamiento de la aplicación .
- Además del consumo del procesador hay también un consumo fijo promedio de 1 mA para alimentar los circuitos auxiliares (conversores A/D, comunicación, etc).

---

13) Leer las secciones 2.1 a 2.7 de la referencia [1], y al menos los cuatro primeros párrafos de la sección "Why a New SPEC CPU Benchmark Suite?" (página 2) de [2] y el recuadro "The problem with mips" que se encuentra en la página 12 de esta misma referencia. Luego escriba alguna devolución fundamentada a las siguientes cuestiones:

- a) ¿En qué condiciones tiene sentido utilizar la frecuencia como índice de performance?
- b) ¿Es cierto que utilizar MIPS para medir la performance tiende a favorecer algunas arquitecturas particulares? Si es así, ¿cuáles? ¿por qué?
- c) Según [1] existen dos formas posibles de evaluar los MFLOPS de una máquina que ejecuta instrucciones complejas (como *sin*, *cos*, *tan*, etc.) como secuencias compuestas de instrucciones elementales. ¿Qué método es más "justo" para comparar máquinas diferentes? ¿Qué método probablemente prefiera el vendedor de una máquina que realiza las funciones trigonométricas mediante múltiples instrucciones básicas (sumas, multiplicaciones, etc.)?
- d) Verdadero o falso: Puedo determinar los MIPSs de un procesador a partir de los "dhrystone MIPS" (DMIPS) calculados corriendo el benchmark Dhrystone.
- e) Verdadero o falso: Dhrystone es tan relevante hoy como lo era 30 años atrás.
- f) Verdadero o falso: El benchmark SPEC CPU evalúa exclusivamente la CPU.
- g) El problema con Dhrystone es que es un benchmark sintético.
- h) El tipo de carga propuesto por SPEC para medir la performance es más realista que otros benchmarks.
- i) De la sección 2.6 de [1] se desprende que MFLOPS no sirven para medir performance.