



Área Académica Ingeniera en Computadores

Arquitectura de Computadores I

Taller 4

Profesor:

Luis Chavarría Zamora

Estudiante:

Jose Ignacio Granados Marín

Grupo 1

IS-2022

## ❖ Investigación

Según un artículo publicado por Tech Target (2005), un pipe, en las ciencias de computación, es una técnica para pasar información de un proceso de determinado programa a otro. En otras palabras, un pipe pasa un parámetro como la salida de un proceso a otro proceso que lo acepta como entrada y el sistema almacena temporalmente la información hasta que el proceso lo lee.

## ❖ Instalación de herramientas

Luego de haber configurado el CPU con el ISA de x86, un BINARY de gem5.opt y Número de CPU's + 1 de 5, a través del comando scons, se ejecutó un Hello World de prueba para verificar la correcta instalación de la herramienta, dicha prueba se muestra a continuación:

```
hello
gem5 Simulator System. http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 21.2.1.1
gem5 compiled May 16 2022 21:32:22
gem5 started May 16 2022 21:43:28
gem5 executing on nachogranados, pid 29828
command line: build/X86/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello

Global frequency set at 100000000000 ticks per second
build/X86/mem/mem_interface.cc:791: warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
**** REAL SIMULATION ****
build/X86/sim/simulate.cc:194: info: Entering event queue @ 0. Starting simulation...
Hello world!
Exiting @ tick 5943000 because exiting with last active thread context
```

Figura 1. Ejecución de la prueba de Hello World con GEM5.

## ❖ Uso con benchmark

Se ejecutó a cabalidad el benchmark 401.bzip2, dando como resultado los siguientes tiempos de simulación:

```
gem5 Simulator System. http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 21.2.1.1
gem5 compiled May 16 2022 21:32:22
gem5 started May 16 2022 22:03:00
gem5 executing on nachogranados, pid 30680
command line: /home/nachogranados/Documents/GEM5/gem5/build/X86/gem5.opt -d m5out/ /home/nachogranados/Documents/GEM5/gem5/configs/example/se.py -c ./src/benchmark -o ./data/input.program -I 100000000 --cpu-type=TimingSimpleCPU --caches --l2cache --l1d_size=128kB --l1i_size=128kB --l2_size=1MB --l1d_assoc=2 --l1i_assoc=2 --l2_assoc=1 --cacheline_size=64

Global frequency set at 100000000000 ticks per second
build/X86/mem/mem_interface.cc:791: warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
0: system.remote_gdb: listening for remote gdb on port 7000
**** REAL SIMULATION ****
build/X86/sim/simulate.cc:194: info: Entering event queue @ 0. Starting simulation...
build/X86/sim/mem_state.cc:443: info: Increasing stack size by one page.
build/X86/sim/syscall_emul.cc:74: warn: ignoring syscall mprotect(...)
spec_init
Loading Input Data
Input data 131072 bytes in length
Compressing Input Data, level 5
build/X86/sim/mem_state.cc:443: info: Increasing stack size by one page.
Exiting @ tick 202595883500 because a thread reached the max instruction count

real    3m52.227s
user    3m52.046s
sys     0m0.096s
```

Figura 2. Ejecución del benchmark 401.bzip2.

Algunas de las estadísticas más relevantes de la simulación en cuestión son las siguientes:

- system.cpu.dcache.overallHits::total => 55548892
- system.cpu.dcache.overallMisses::total => 421592
- system.cpu.dcache.overallMissRate::total => 0.007532
- system.cpu.icache.overallHits::total => 138473695
- system.cpu.icache.overallMisses::total => 1081
- system.cpu.icache.overallMissRate::total => 0.000008
- system.l2.overallHits::total => 379949
- system.l2.overallMisses::total => 42724
- system.l2.demandMissRate::total => 0.101081

#### ❖ Ejercicio

Cada una de las pruebas y resultados fueron ejecutados y extraídos del benchmark 456.hmmer.

La configuración por defecto del benchmark 456.hmmer se muestra a continuación:

- --cpu-type=TimingSimpleCPU
- --caches
- --l2cache
- --l1d\_size=128kB
- --l1i\_size=128kB
- --l2\_size=1MB
- --l1d\_assoc=2
- --l1i\_assoc=2
- --l2\_assoc=1
- --cacheline\_size=64

Al ejecutar el benchmark 456.hmmer con la configuración por defecto, se obtuvieron las siguientes estadísticas de simulación:

- system.cpu.dcache.overallHits::total => 2815
- system.cpu.dcache.overallMisses::total => 232
- system.cpu.dcache.overallMissRate::total => 0.076140
- system.cpu.icache.overallHits::total => 9188
- system.cpu.icache.overallMisses::total => 457
- system.cpu.icache.overallMissRate::total => 0.047382
- system.l2.overallHits::total => 2
- system.l2.overallMisses::total => 687
- system.l2.overallMissRate::total => 0.997097

Conocida la configuración anterior, se propusieron 4 cambios de 2 parámetros cada uno (donde los datos que poseen el símbolo “-” representan la configuración por defecto mientras que los que poseen el símbolo “+” corresponde a la nueva modificación), con el objetivo de determinar las posibles mejoras que dichas alteraciones pueden generar. Dichas modificaciones se muestran a continuación:

➤ Cambio 1:

Las modificaciones propuestas para este cambio son las siguientes:

- --l1d\_size=512kB
- --l1i\_size=512kB

Si se comparan los cambios anteriores con la configuración por defecto, se puede notar que tanto el tamaño de la memoria de datos de nivel 1 (l1d\_size) como la memoria de instrucciones de nivel 1 (l1i\_size) se cuadruplicaron, esto con el objetivo de almacenar una mayor cantidad de datos, de manera que se mejore la lectura y escritura de los mismos y ejecutar una amplia cantidad de instrucciones, generando así un proceso más eficiente. Por esta razón, se espera que las estadísticas posean diferencias significativas con respecto a la configuración por defecto.

Ahora bien, las estadísticas obtenidas son las siguientes:

- -system.cpu.icache.overallHits::total => 9188
- +system.cpu.icache.overallHits::total => 9189
- -system.cpu.icache.overallMisses::total => 457
- +system.cpu.icache.overallMisses::total => 456
- -system.cpu.icache.overallMissRate::total => 0.047382
- +system.cpu.icache.overallMissRate::total => 0.047278
- -system.l2.overallHits::total => 2
- +system.l2.overallHits::total => 1
- -system.l2.overallMissRate::total => 0.997097
- +system.l2.overallMissRate::total => 0.998547

Tal y como se puede observar, a pesar de incrementar el tamaño de las memorias de datos e instrucciones en un factor de 4, las estadísticas no reflejan cambios significativos en la cantidad de aciertos, desaciertos y tasa de desaciertos en dichos módulos, ya que las diferencias entre ellos no superan una unidad.

➤ Cambio 2:

Las modificaciones propuestas para este cambio son las siguientes:

- `--l1d_assoc=4`
- `--l1i_assoc=4`

Si se comparan los cambios anteriores con la configuración por defecto, se puede notar que tanto la asociatividad de la memoria de datos de nivel 1 (l1d\_assoc) como de la memoria de instrucciones de nivel 1 (l1i\_assoc) se duplicaron, esto con el objetivo reducir la cantidad de desaciertos al momento realizar la lectura y escritura de datos e instrucciones. Por esta razón, se espera que los desaciertos y la tasa de desaciertos posean diferencias significativas respecto a la configuración por defecto.

Ahora bien, las estadísticas obtenidas son las siguientes:

- `-system.cpu.icache.overallHits::total => 9188`
- `+system.cpu.icache.overallHits::total => 9189`
- `-system.cpu.icache.overallMisses::total => 457`
- `+system.cpu.icache.overallMisses::total => 456`
- `-system.cpu.icache.overallMissRate::total => 0.047382`
- `+system.cpu.icache.overallMissRate::total => 0.047278`
- `-system.l2.overallMissRate::total => 0.997097`
- `+system.l2.overallMissRate::total => 0.998547`

Similar al cambio anterior, a pesar haber incrementado la asociatividad de las memorias de datos e instrucciones en un factor de 2, las estadísticas no reflejan cambios significativos en la cantidad de desaciertos y tasa de desaciertos como se esperaba debido a la modificación de únicamente la memoria de nivel 1.

➤ Cambio 3:

Las modificaciones propuestas para este cambio son las siguientes:

- `--l2_assoc=4`
- `--cacheline_size=256`

Si se comparan los cambios anteriores con la configuración por defecto, se puede notar que tanto la asociatividad de la memoria nivel 2 (l2\_assoc) como el tamaño de la memoria cache de línea (cacheline\_size) se cuadruplicaron, esto con el objetivo de reducir la cantidad de desaciertos, así como también mejorar la disponibilidad de datos e instrucciones cuando el CPU los requiera.

Ahora bien, las estadísticas obtenidas son las siguientes:

- -system.cpu.dcache.overallHits::total => 2815
- +system.cpu.dcache.overallHits::total => 2951
- -system.cpu.dcache.overallMisses::total => 232
- +system.cpu.dcache.overallMisses::total => 93
- -system.cpu.dcache.overallMissRate::total => 0.076140
- +system.cpu.dcache.overallMissRate::total => 0.030552
- -system.cpu.icache.overallHits::total => 9188
- +system.cpu.icache.overallHits::total => 9442
- -system.cpu.icache.overallMisses::total => 457
- +system.cpu.icache.overallMisses::total => 203
- -system.cpu.icache.overallMissRate::total => 0.047382
- +system.cpu.icache.overallMissRate::total => 0.021047
- -system.l2.overallHits::total => 2
- +system.l2.overallHits::total => 4
- -system.l2.overallMisses::total => 687
- +system.l2.overallMisses::total => 292
- -system.l2.overallMissRate::total => 0.997097
- +system.l2.overallMissRate::total => 0.986486

A diferencia de los cambios anteriores, las modificaciones propuestas reflejan pocos cambios en la cantidad de aciertos, pero diversas mejoras en la suma de desaciertos y tasa de desaciertos en las memorias de datos cache, instrucciones cache y nivel 2. Para este caso en particular, los valores de overallMisses y overallMissRate disminuyen en aproximadamente un factor de 2.5.

#### ➤ Cambio 4:

Las modificaciones propuestas para este cambio son las siguientes:

- --l2\_size=2MB
- --l2\_assoc=2

Si se comparan los cambios anteriores con la configuración por defecto, se puede notar que tanto el tamaño (l2\_size) como la asociatividad de la memoria de la memoria nivel 2 (l2\_assoc) se duplicaron, esto con el objetivo de mejorar la lectura y escritura de datos e instrucciones y reducir la cantidad de desaciertos del CPU.

Ahora bien, las estadísticas obtenidas son las siguientes:

- -system.l2.replacements => 15
- +system.l2.replacements => 0
- -system.l2.tags.avgOccs::total => 0.020603
- +system.l2.tags.avgOccs::total => 0.010455
- -system.l2.tags.dataAccesses => 1387
- +system.l2.tags.dataAccesses => 2087

En contraste con el resto de cambios realizados, al modificar las características de la memoria de nivel 2, las mejoras se observan únicamente en dicho módulo. Tal y como lo demuestran los datos anteriores hubo una reducción significativa en la cantidad de remplazos. Además, el porcentaje promedio de ocupación de caché (avgOccs) se disminuye aproximadamente a la mitad. Sin embargo, hubo un incremento de 700 accesos a memoria y esto se debe principalmente a la ampliación del tamaño de la memoria en cuestión.

## Referencias:

Tech Target. (2005, 21 septiembre). *Pipe*. WhatIs.Com.

<https://www.techtarget.com/whatis/definition/pipe#:~:text=In%20computer%20programming%2C%20especially%20in,is%20one%2Dway%20communication%20only.>