

COMPUTER Architecture
CSE511

Project 02 Final-Evaluation
4x4_Multi-Core_System_Using_MESI_&_VI_Protocol

Contributors:

KULDEEP SINGH(2021328)
DAEVAANG KHAIRWAL(2020369)

Evaluator:

Naorem Akshaykumar

To Build The Gem5 With MESI Protocol:

Necessary dependency :

```
sudo apt install build-essential git m4 scons zlib1g zlib1g-dev  
libprotobuf-dev protobuf-compiler libprotoc-dev libgoogle-perftools-dev  
python-dev python
```

Cloning the gem5:

```
git clone https://github.com/gem5/gem5  
cd gem 5
```

Building the build with MESI protocol:

```
scons defconfig build/X86_MESI build_opts/X86  
scons setconfig build/X86_MESI RUBY_PROTOCOL_MESI_TWO_LEVEL=y  
SLICC_HTML=y  
scons build/X86_MESI/gem5.opt
```

Parsec installation:

Create a virtual environment :

```
virtualenv -p python3 venv  
source venv/bin/activate
```

Installing dependencies:

```
pip install gem5art-artifact gem5art-run gem5art-tasks
```

Make a dir disk-image:

```
cd disk-image/parsec-benchmark  
git clone https://github.com/cirosantilli/parsec-benchmark.git
```

Building m5 :

```
cd gem5/utils/m5/  
scons build/x86/gem.opt
```

create file parsec-install.sh:

```
# install build-essential (gcc and g++ included) and gfortran  
  
#Compile PARSEC
```

```

cd /home/gem5/
su gem5
echo "12345" | sudo -S apt update

# Allowing services to restart while updating some
# libraries.
sudo apt install -y debconf-utils
sudo debconf-get-selections | grep restart-without-asking > libs.txt
sed -i 's/false/true/g' libs.txt
while read line; do echo $line | sudo debconf-set-selections; done < libs.txt
sudo rm libs.txt
##

# Installing packages needed to build PARSEC
sudo apt install -y build-essential
sudo apt install -y m4
sudo apt install -y git
sudo apt install -y python
sudo apt install -y python-dev
sudo apt install -y gettext
sudo apt install -y libx11-dev
sudo apt install -y libxext-dev
sudo apt install -y xorg-dev
sudo apt install -y unzip
sudo apt install -y texinfo
sudo apt install -y freeglut3-dev
sudo apt install -y cmake
##

# Building PARSEC

echo "12345" | sudo -S chown gem5 -R parsec-benchmark/
echo "12345" | sudo -S chgrp gem5 -R parsec-benchmark/
cd parsec-benchmark
./install.sh
./get-inputs
cd ..
echo "12345" | sudo -S chown gem5 -R parsec-benchmark/
echo "12345" | sudo -S chgrp gem5 -R parsec-benchmark/
##

```

create post-installation.sh:

```

#!/bin/bash
echo 'Post Installation Started'

```

```
mv /home/gem5/serial-getty@.service /lib/systemd/system/
```

```
mv /home/gem5/m5 /sbin
```

```
ln -s /sbin/m5 /sbin/gem5
```

copy and run outside (host) script after booting

```
cat /home/gem5/runscript.sh >> /root/.bashrc
```

```
echo 'Post Installation Done'
```

create runscript.sh:

```
#!/bin/sh
m5 readfile > script.sh
if [ -s script.sh ]; then
    # if the file is not empty, execute it
    chmod +x script.sh
    ./script.sh
    m5 exit
fi
# otherwise, drop to the terminal
```

Create parsec.json:

```
{
  "builders":
  [
    {
      "type": "qemu",
      "format": "raw",
      "accelerator": "kvm",
      "boot_command":
      [
        "{{ user `boot_command_prefix` }}",
        "debian-installer={{ user `locale` }} auto locale={{ user `locale` }}"
        kbd-chooser/method=us ",
        "file=/floppy/{{ user `preseed` }} ",
        "fb=false debconf/frontend=noninteractive ",
        "hostname={{ user `hostname` }} ",
        "/install/vmlinuz noapic ",
        "initrd=/install/initrd.gz ",
        "keyboard-configuration/modelcode=SKIP keyboard-configuration/layout=USA ",

```

```

        "keyboard-configuration/variant=USA console-setup/ask_detect=false ",
        "passwd/user-fullname={{ user `ssh_fullname` }} ",
        "passwd/user-password={{ user `ssh_password` }} ",
        "passwd/user-password-again={{ user `ssh_password` }} ",
        "passwd/username={{ user `ssh_username` }} ",
        "-- <enter>"
    ],
    "cpus": "{{ user `vm_cpus` }}",
    "disk_size": "{{ user `image_size` }}",
    "floppy_files":
    [
        "shared/{{ user `preseed` }}"
    ],
    "headless": "{{ user `headless` }}",
    "http_directory": "shared/",
    "iso_checksum": "{{ user `iso_checksum` }}",
    "iso_checksum_type": "{{ user `iso_checksum_type` }}",
    "iso_urls": [ "{{ user `iso_url` }}" ],
    "memory": "{{ user `vm_memory` }}",
    "output_directory": "parsec/{{ user `image_name` }}-image",
    "qemuargs":
    [
        [ "-cpu", "host" ],
        [ "-display", "none" ]
    ],
    "qemu_binary": "/usr/bin/qemu-system-x86_64",
    "shutdown_command": "echo '{{ user `ssh_password` }}'|sudo -S shutdown -P now",
    "ssh_password": "{{ user `ssh_password` }}",
    "ssh_username": "{{ user `ssh_username` }}",
    "ssh_wait_timeout": "60m",
    "vm_name": "{{ user `image_name` }}"
    }
],
"provisioners":
[
    {
        "type": "file",
        "source": "../gem5/util/m5/build/x86/out/m5",
        "destination": "/home/gem5/"
    },
    {
        "type": "file",
        "source": "shared/serial-getty@.service",
        "destination": "/home/gem5/"
    },
    {

```

[illegible]

```
}
```

With this you are ready to create a disk image for x86 parsec

Now,

Install packer:

```
cd disk-image/  
wget https://releases.hashicorp.com/packer/1.4.3/packer_1.4.3_linux_amd64.zip  
unzip packer_1.4.3_linux_amd64.zip
```

Now build the disk image with the following commands :

```
PACKER_LOG=1 ./packer validate parsec/parsec.json  
PACKER_LOG=1 ./packer build parsec/parsec.json
```

NOTE:The image will be created in the parsec-image folder with the name parsec.

cmd code for

```
./build/X86_MESI/gem5.opt configs/example/gem5_library/x86-parsec-benchmarks.py  
--benchmark swaptions --size simsmall > /home/daev/Desktop/output.txt
```

```
./build/X86_MESI/gem5.opt configs/example/gem5_library/x86-parsec-benchmarks.py  
--benchmark facesim--size simsmall > /home/daev/Desktop/output.txt
```

CODE FOR RUNNING PARSEC BENCHMARK

```
import argparse  
import time  
  
import m5  
from m5.objects import Root  
  
from gem5.coherence_protocol import CoherenceProtocol  
from gem5.components.boards.x86_board import X86Board  
from gem5.components.memory import DualChannelDDR4_2400  
from gem5.components.processors.cpu_types import CPUTypes
```

```

from gem5.components.processors.simple_processor import SimpleProcessor
from gem5.isas import ISA
from gem5.resources.resource import obtain_resource
from gem5.simulate.exit_event import ExitEvent
from gem5.simulate.simulator import Simulator
from gem5.resources.resource import DiskImageResource, KernelResource
from gem5.utils.requires import requires

#IMPORTING ALL THE NECESSARY FILES

requires(
    isa_required=ISA.X86,
    coherence_protocol_required=CoherenceProtocol.MESI_TWO_LEVEL,
    kvm_required=False,
)

#CHECKING FOR THE REQUIRED BUILD

benchmark_choices = [
    "swaptions",
]

#DIFFERNT BENCHMARKS
size_choices = ["simsmall", "simmedium", "simlarge"]

parser = argparse.ArgumentParser(
    description="An example configuration script to run the parsec
benchmarks."
)

# ADD ARGUMENTS TO THE CMD LINE CODE
parser.add_argument(
    "--benchmark",
    type=str,
    required=True,
    help="Input the benchmark program to execute.",
    choices=benchmark_choices,
)

parser.add_argument(
    "--size",
    type=str,
    required=True,
    help="Simulation size the benchmark program.",

```



```

        choices=size_choices,
    )
args = parser.parse_args()

from gem5.components.cachehierarchies.ruby.mesi_two_level_cache_hierarchy
import (
    MESITwoLevelCacheHierarchy,
)
# GETTING THE MESITWOLEVEL PROTOCOL

cache_hierarchy = MESITwoLevelCacheHierarchy(
    l1d_size="32kB",
    l1d_assoc=4,
    l1i_size="32kB",
    l1i_assoc=4,
    l2_size="256kB",
    l2_assoc=4,
    num_l2_banks=2,
)
# SETTING UP THE CACHES

memory = DualChannelDDR4_2400(size="4GB")
#SETTING UP THE MEMORY
processor = SimpleProcessor(isa=ISA.X86, cpu_type=CPUTypes.O3,
num_cores=4)
# SETTING UP THE PROCESSORS
board = X86Board(
    clk_freq="3GHz",
    processor=processor,
    memory=memory,
    cache_hierarchy=cache_hierarchy,
)
# ADD A X86 BOARD

command = (
    f"cd /home/gem5/parsec-benchmark;"
    + "source env.sh;"
    + f"parsecgmt -a run -p {args.benchmark} -c gcc-hooks -i {args.size}"
    -n 2;"
    + "sleep 5;"

```

```

        + "m5 exit;"
    )

local_kernel_path = "/home/daev/Desktop/ca
project/parsec/linux-stable/vmlinux-4.19.83"
local_disk_image_path = "/home/daev/Desktop/ca
project/parsec/disk-image/parsec/parsec-image/parsec"
# PATH TO DISK IMAGE AND THE KERNEL IMAGE

board.set_kernel_disk_workload(
    kernel=KernelResource(local_path=local_kernel_path),
    disk_image=DiskImageResource(local_path=local_disk_image_path),
    readfile_contents=command,
)
# SETTING UP THE BOARD WITH THE KERNEL AND DISK IMAGE

def handle_workbegin():
    print("Done booting Linux")
    print("Resetting stats at the start of ROI!")
    m5.stats.reset()
    processor.switch()
    yield False

def handle_workend():
    print("Dump stats at the end of the ROI!")
    m5.stats.dump()
    yield True

simulator = Simulator(
    board=board,
    on_exit_event={
        ExitEvent.WORKBEGIN: handle_workbegin(),
        ExitEvent.WORKEND: handle_workend(),
    },
)

```

```

globalStart = time.time()
# STARTING SIMULATION
print("Running the simulation")
print("Using O3 CPU")

m5.stats.reset()
simulator.run()

print("All simulation events were successful.")

print("Done with the simulation")
print()
print("Performance statistics:")

print("Simulated time in ROI: " + (str(simulator.get_roi_ticks()[0])))
print(
    "Ran a total of", simulator.get_current_tick() / 1e12, "simulated"
    "seconds"
)
print(
    "Total wallclock time: %.2fs, %.2f min"
    % (time.time() - globalStart, (time.time() - globalStart) / 60)
)
#END OF SIMULATION

```

CODE TILL MID EVALUATION

1. Defining the MESITwoLevelCache Class

This class sets up a two-level MESI (Modified-Exclusive-Shared-Invalid) cache system with L1 and L2 caches and coherence controllers.

```
class MESITwoLevelCache(RubySystem):

    def __init__(self):

        if buildEnv['PROTOCOL'] != 'MESI_Two_Level':

            fatal("This system assumes MESI_Two_Level!")

        super(MESITwoLevelCache, self).__init__()

        self._numL2Caches = 4
```

- **Purpose:** Initializes the MESI cache system.
 - **Key Point:** Checks if the protocol is MESI_Two_Level; if not, it raises an error.
 - **Attributes:** Sets _numL2Caches to 4, meaning there are 4 L2 caches.
-

2. Setting up the Cache System and Controllers

This setup method creates the network and cache controllers, associating them with CPUs and memory.

```
def setup(self, system, cpus, mem_ctrls, dma_ports, iobus):

    self.network = MyNetwork(self)

    self.number_of_virtual_networks = 5

    self.controllers = \

        [L1Cache(system, self, cpu, self._numL2Caches) for cpu in cpus] + \

        [L2Cache(system, self, self._numL2Caches) for _ in range(self._numL2Caches)] + \

        [DirController(self, system.mem_ranges, mem_ctrls)] + \

        [DMAController(self) for _ in range(len(dma_ports))]
```

- **Purpose:** Sets up the network, virtual networks, and controllers.

- **Components:**
 - **L1Cache:** Created for each CPU.
 - **L2Cache:** A list of L2 caches (4 in total).
 - **DirController:** A directory controller, responsible for maintaining coherence across all L2 caches.
 - **DMAController:** Direct Memory Access controllers, linked to the dma_ports.
-

3. L1Cache Class

Defines the L1 cache controller with separate instruction and data caches.

```
class L1Cache(L1Cache_Controller):  
  
    def __init__(self, system, ruby_system, cpu, num_l2Caches):  
  
        super(L1Cache, self).__init__()  
  
        self.L1Icache = RubyCache(size='32kB', assoc='4', start_index_bit=block_size_bits,  
is_ocache=True)  
  
        self.L1Dcache = RubyCache(size='32kB', assoc='4', start_index_bit=block_size_bits,  
is_ocache=False)
```

- **Purpose:** Defines separate instruction (L1Icache) and data (L1Dcache) caches.
 - **Attributes:**
 - **size:** 32 KB for both instruction and data caches.
 - **assoc:** 4-way associativity.
-

4. L2Cache Class

Defines the L2 cache controller with a unified cache for instructions and data.

```
class L2Cache(L2Cache_Controller):  
  
    def __init__(self, system, ruby_system, num_l2Caches):  
  
        super(L2Cache, self).__init__()
```

```
self.L2cache = RubyCache(size='256kB', assoc=4,  
start_index_bit=self.getBlockSizeBits(system, num_L2Caches))
```

- **Purpose:** Sets up the L2 cache as a unified cache with a larger capacity.
-

5. Directory Controller (DirController)

This controller maintains coherence across L2 caches.

```
class DirController(Directory_Controller):  
  
    def __init__(self, ruby_system, ranges, mem_ctrls):  
  
        if len(mem_ctrls) > 1:  
  
            panic("This cache system can only be connected to one mem ctrl")  
  
        super(DirController, self).__init__()  
  
        self.directory = RubyDirectoryMemory()  
  
        self.memory_out_port = mem_ctrls[0].port
```

- Directory controller for coherence.

6. MyNetwork Class

This class represents a simple network interconnecting the controllers.

```
class MyNetwork(SimpleNetwork):  
  
    def connectControllers(self, controllers):  
  
        self.routers = [Switch(router_id=i) for i in range(len(controllers))]  
  
        self.ext_links = [SimpleExtLink(link_id=i, ext_node=c, int_node=self.routers[i]) for i, c in  
enumerate(controllers)]
```

- Connects all cache controllers via routers and links.
-

7. O3Core Class

Defining an out-of-order CPU core with branch prediction and reorder buffer (ROB) parameters.

```
class O3Core(RubySystem):  
  
    def __init__(self, cpu_id):  
  
        super(O3Core, self).__init__()  
  
        self.cpu = RiscvO3CPU(cpu_id=cpu_id)  
  
        self.cpu.branchPred = TournamentBP()  
  
        self.cpu.branchPred.localPredictorSize = 2048
```

- Configures a single RISC-V out-of-order core.
 - **Branch Predictor:** Uses a tournament predictor for accuracy.**Reorder Buffer (ROB):** Allows speculative execution.
-

8. System Configuration

Defining the full system with CPU, memory, and cache configuration.

```
system = System()  
  
system.clk_domain = SrcClockDomain(clock="1GHz")  
  
system.mem_ranges = [AddrRange("512MB")]  
  
system.cpu = [O3Core(i) for i in range(4)]
```

- Sets up the main system configuration.
- **Components:**

- **Clock:** 1 GHz clock for timing.
 - **Memory Range:** 512MB.
 - **CPUs:** Creates four out-of-order cores.
-

9. Setting up Workloads

Assigning a workload (a simple RISC-V binary) to each CPU.

```
binary = os.path.join(thispath, "../..", "tests/test-progs/hello/bin/RISCV/linux/hello")

process = Process()

process.cmd = [binary]

for cpu in system.cpu:

    cpu.workload = process

    cpu.createThreads()
```

10. Running the Simulation

Initializes the simulation and runs until completion.

```
root = Root(full_system=False, system=system)

m5.instantiate()

exit_event = m5.simulate()

print(f"Exiting @ tick {m5.curTick()} because {exit_event.getCause()}")
```


Output

```
daev@daev-VirtualBox: ~/Desktop/ca project/parsec/gen5$ ./build/X86_MESI/gen5.opt configs/example/gen5_library/x86-parsec-benchmarks.py --benchmark swaption
s --size simsmall > /home/daev/Desktop/output.txt
src/mem/dram_interface.cc:698: warn: DRAM device capacity (16384 Mbytes) does not match the address range assigned (2048 Mbytes)
src/mem/dram_interface.cc:698: warn: DRAM device capacity (16384 Mbytes) does not match the address range assigned (2048 Mbytes)
src/sim/kernel_workload.cc:46: info: kernel located at: /home/daev/Desktop/ca project/parsec/linux-stable/vmlinux-4.19.83
src/base/statistics.hh:279: warn: One of the stats is a legacy stat. Legacy stat is a stat that does not belong to any statistics::Group. Legacy stat is deprecated.
board.pc.com.1.device: Listening for connections on port 3456
src/base/statistics.hh:279: warn: One of the stats is a legacy stat. Legacy stat is a stat that does not belong to any statistics::Group. Legacy stat is deprecated.
src/dev/intel_8254_timer.cc:128: warn: Reading current count from inactive timer.
board.remote_gdb: Listening for connections on port 7000
src/sim/simulate.cc:199: info: Entering event queue @ 0. Starting simulation...
src/mem/ruby/system/Sequencer.cc:688: warn: Replacement policy updates recently became the responsibility of SLICC state machines. Make sure to setMRU() near callbacks in .sm files!
build/X86_MESI/arch/x86/generated/exec-ns.cc.inc:27: warn: instruction 'finit' unimplemented
build/X86_MESI/arch/x86/generated/exec-ns.cc.inc:27: warn: instruction 'sgdt_Ms' unimplemented
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
build/X86_MESI/arch/x86/generated/exec-ns.cc.inc:27: warn: instruction 'wbinvd' unimplemented
src/arch/x86/interrupts.cc:538: hack: Assuming logical destinations are 1 << id.
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
src/cpu/o3/fetch.cc:604: warn: Address 0xffffffff is outside of physical memory, stopping fetch
src/dev/x86/pc.cc:117: warn: Don't know what interrupt to clear for console.

src/dev/x86/i8042.cc:298: warn: Write to unknown i8042 (keyboard controller) command port.
```

```
daev@daev-VirtualBox: ~/Downloads/gen5-resources-develop/src/parsec/disk-image

2024/11/24 21:31:49 [INFO] RPC client: Communicator ended with: 0
2024/11/24 21:31:49 [INFO] RPC endpoint: Communicator ended with: 0
2024/11/24 21:31:49 packer-provisioner-shell plugin: [INFO] RPC client: Communicator ended with: 0
2024/11/24 21:31:49 packer-builder-qemu plugin: [DEBUG] Opening new ssh session
2024/11/24 21:31:49 packer-builder-qemu plugin: [DEBUG] starting remote command: rm -f
2024/11/24 21:31:49 packer-builder-qemu plugin: [INFO] RPC endpoint: Communicator ended with: 0
2024/11/24 21:31:49 [INFO] RPC client: Communicator ended with: 0
2024/11/24 21:31:49 [INFO] RPC endpoint: Communicator ended with: 0
2024/11/24 21:31:49 packer-provisioner-shell plugin: [INFO] RPC client: Communicator ended with: 0
2024/11/24 21:31:49 [INFO] (telemetry) ending shell
==> qemu: Gracefully halting virtual machine...
2024/11/24 21:31:49 packer-builder-qemu plugin: Executing shutdown command: echo '12345'|sudo
2024/11/24 21:31:49 packer-builder-qemu plugin: [DEBUG] Opening new ssh session
2024/11/24 21:31:49 packer-builder-qemu plugin: [DEBUG] starting remote command: echo '12345'|
==> qemu: [sudo] password for gen5:
2024/11/24 21:31:51 packer-builder-qemu plugin: [ERROR] Remote command exited without exit status
2024/11/24 21:31:51 packer-builder-qemu plugin: Waiting max 5m0s for shutdown to complete
2024/11/24 21:31:52 packer-builder-qemu plugin: VM shut down.
2024/11/24 21:31:52 packer-builder-qemu plugin: failed to unlock port lockfile: close tcp 127.
2024/11/24 21:31:52 packer-builder-qemu plugin: failed to unlock port lockfile: close tcp 127.
2024/11/24 21:31:52 packer-builder-qemu plugin: Deleting floppy disk: /tmp/packer389833492
2024/11/24 21:31:52 [INFO] (telemetry) ending qemu
Build 'qemu' finished.
==> Builds finished. The artifacts of successful builds are:
==> Builds finished. The artifacts of successful builds are:
2024/11/24 21:31:52 machine readable: qemu.artifact-count [{}string{"1"}
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "builder-id", "transcend.qemu"}
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "id", "VM"}
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "string", "VM files in directory: parsec/parsec-image"}
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "files-count", "1"}
--> qemu: VM files in directory: parsec/parsec-image
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "file", "0", "parsec/parsec-image/parsec"}
2024/11/24 21:31:52 machine readable: qemu.artifact [{}string{"0", "end"}
2024/11/24 21:31:52 [INFO] (telemetry) Finalizing.
2024/11/24 21:31:53 waiting for all plugin processes to complete...
2024/11/24 21:31:53 /home/daev/Downloads/gen5-resources-develop/src/parsec/disk-image/packer: plugin process exited
2024/11/24 21:31:53 /home/daev/Downloads/gen5-resources-develop/src/parsec/disk-image/packer: plugin process exited
2024/11/24 21:31:53 /home/daev/Downloads/gen5-resources-develop/src/parsec/disk-image/packer: plugin process exited
2024/11/24 21:31:53 /home/daev/Downloads/gen5-resources-develop/src/parsec/disk-image/packer: plugin process exited
2024/11/24 21:31:53 /home/daev/Downloads/gen5-resources-develop/src/parsec/disk-image/packer: plugin process exited
daev@daev-VirtualBox: ~/Downloads/gen5-resources-develop/src/parsec/disk-image$ ls
build.sh packer packer-1.6.0-linux-amd64.zip packer_cache parsec shared
daev@daev-VirtualBox: ~/Downloads/gen5-resources-develop/src/parsec/disk-image$
```