Student Name: Karthik Dhondi
Student ID: 152202007

CS5119: Advanced Computer Architecture
LAB 1

INDIAN INSTITUTE
OF TECHNOLOGY
PALAKKAD

gem5 ”Hello World” Performance Log

1 Introduction

1. **Question** Familiarize with gem5, a popular computer architecture simulator both in industry and research. Follow the instructions in this link to download: gem5

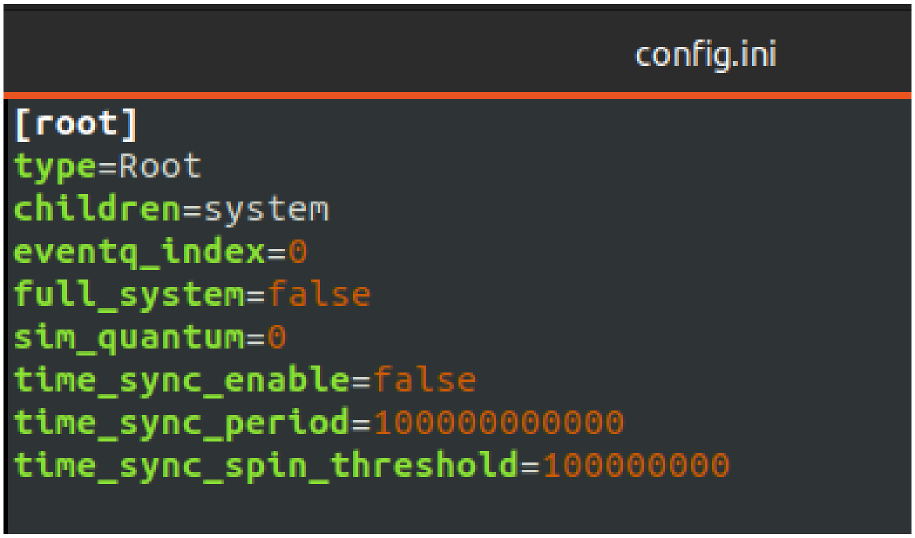
2. **Question** Run the first program available in the link itself to print ”Hello World” and generate the performance log.

3. **Question** Submit the performance log which can be found in ’/gem5/m5out’ folder as ’stats.txt’ and your detailed interpretation of the log (your insights of the important parameters).

Solution:

To start with, to show the performance we need to understand two files.
These two files can be found in the directory ’/gem5/m5out’.

1. **config.ini**
   Config.in file is the definitive version of what was simulated.
   This file displays all parameters associated with each simulated SimObject, whether they were configured in the scripts or default values were utilized.

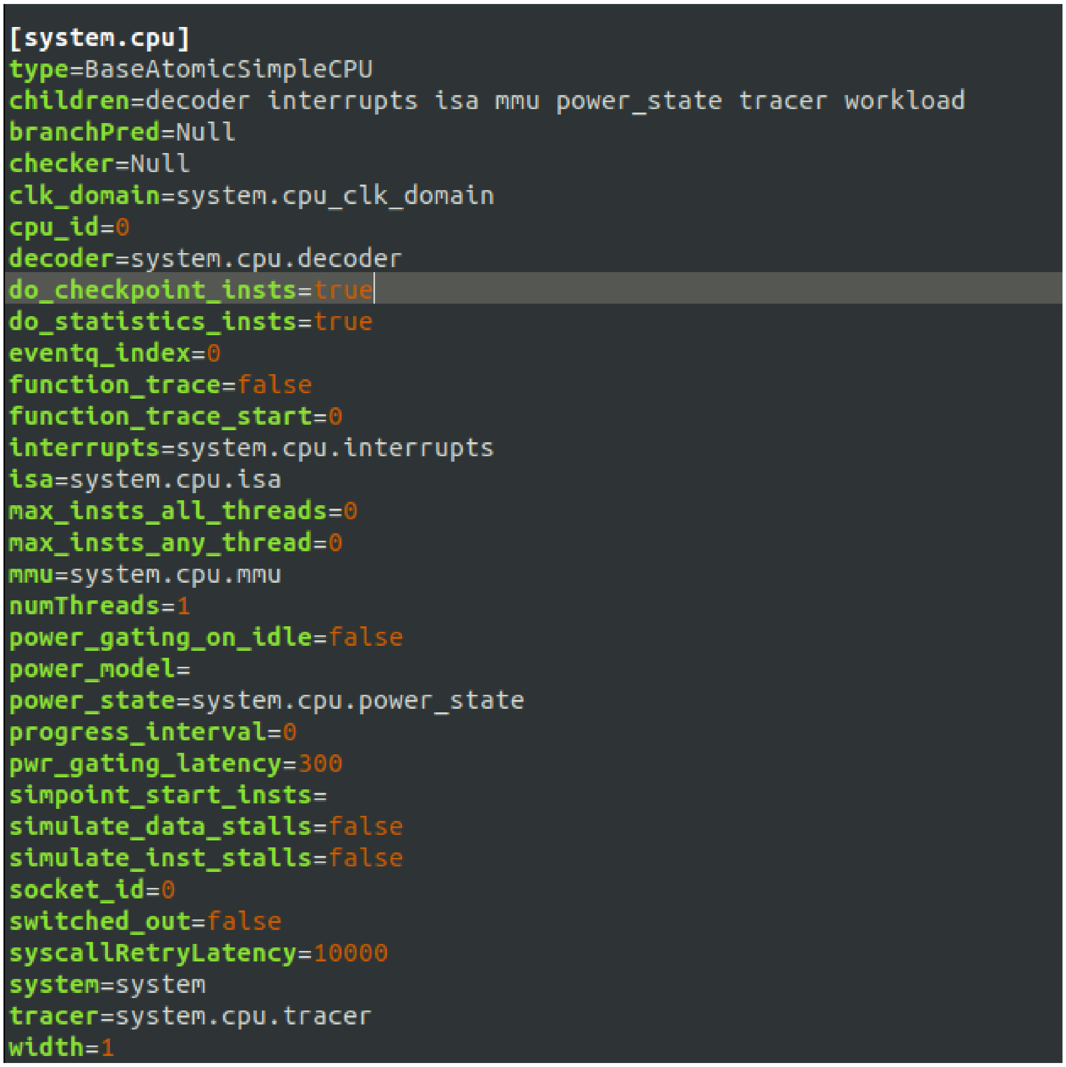


config.ini

```
[root]
type=Root
children=system
eventq_index=0
full_system=false
sim_quantum=0
time_sync_enable=false
time_sync_period=100000000000
time_sync_spin_threshold=100000000
```

Figure 1: Simobject and its parameters

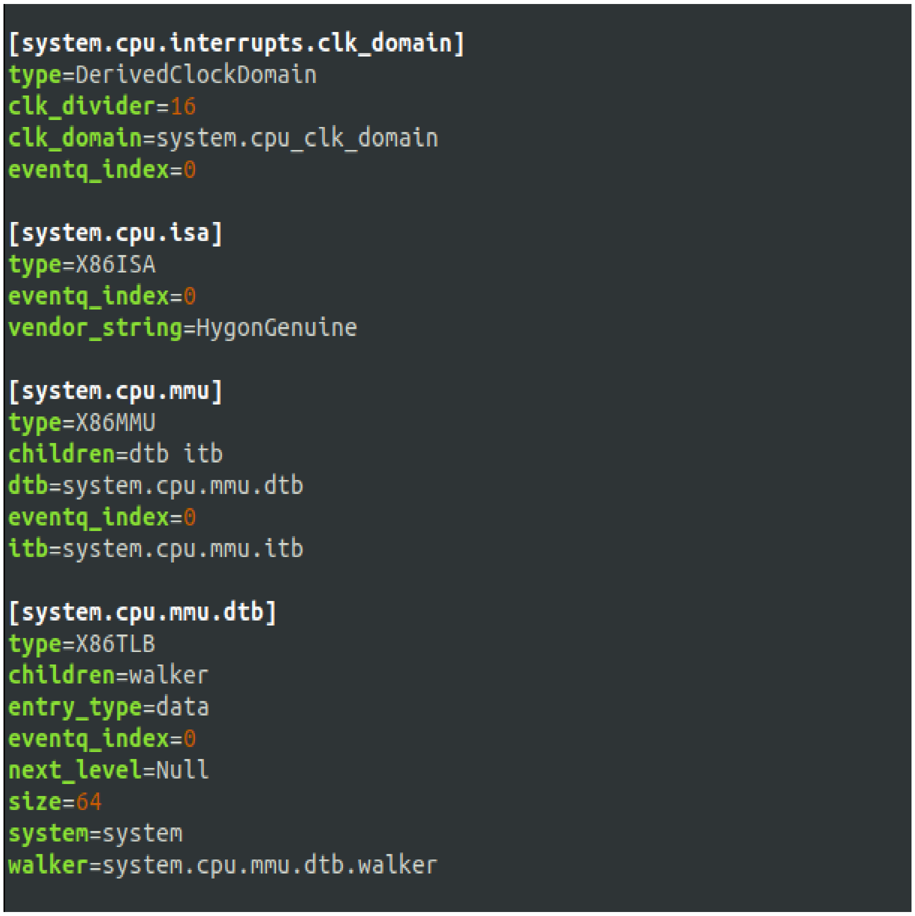
In Figure 1. If we see, **[root]** is a SimObject. And type, children, full_system,..... etc, are parameters of the respective SimObject.



```
[system.cpu]
type=BaseAtomicSimpleCPU
children=decoder interrupts isa mmu power_state tracer workload
branchPred=Null
checker=Null
clk_domain=system.cpu_clk_domain
cpu_id=0
decoder=system.cpu.decoder
do_checkpoint_insts=true
do_statistics_insts=true
eventq_index=0
function_trace=false
function_trace_start=0
interrupts=system.cpu.interrupts
isa=system.cpu.isa
max_insts_all_threads=0
max_insts_any_thread=0
mmu=system.cpu.mmu
numThreads=1
power_gating_on_idle=false
power_model=
power_state=system.cpu.power_state
progress_interval=0
pwr_gating_latency=300
simpoint_start_insts=
simulate_data_stalls=false
simulate_inst_stalls=false
socket_id=0
switched_out=false
syscallRetryLatency=10000
system=system
tracer=system.cpu.tracer
width=1
```

Figure 2: Simobject and its parameters

In Figure 2. As we see, **[system.cpu]** is a SimObject. And type=BaseAtomicSimpleCPU, children, branchPred,.....etc, are the parameters of the SimObjects. By observing these necessary parameters after simulations, we can get to know, whether they were configured in the scripts or default values were utilized.

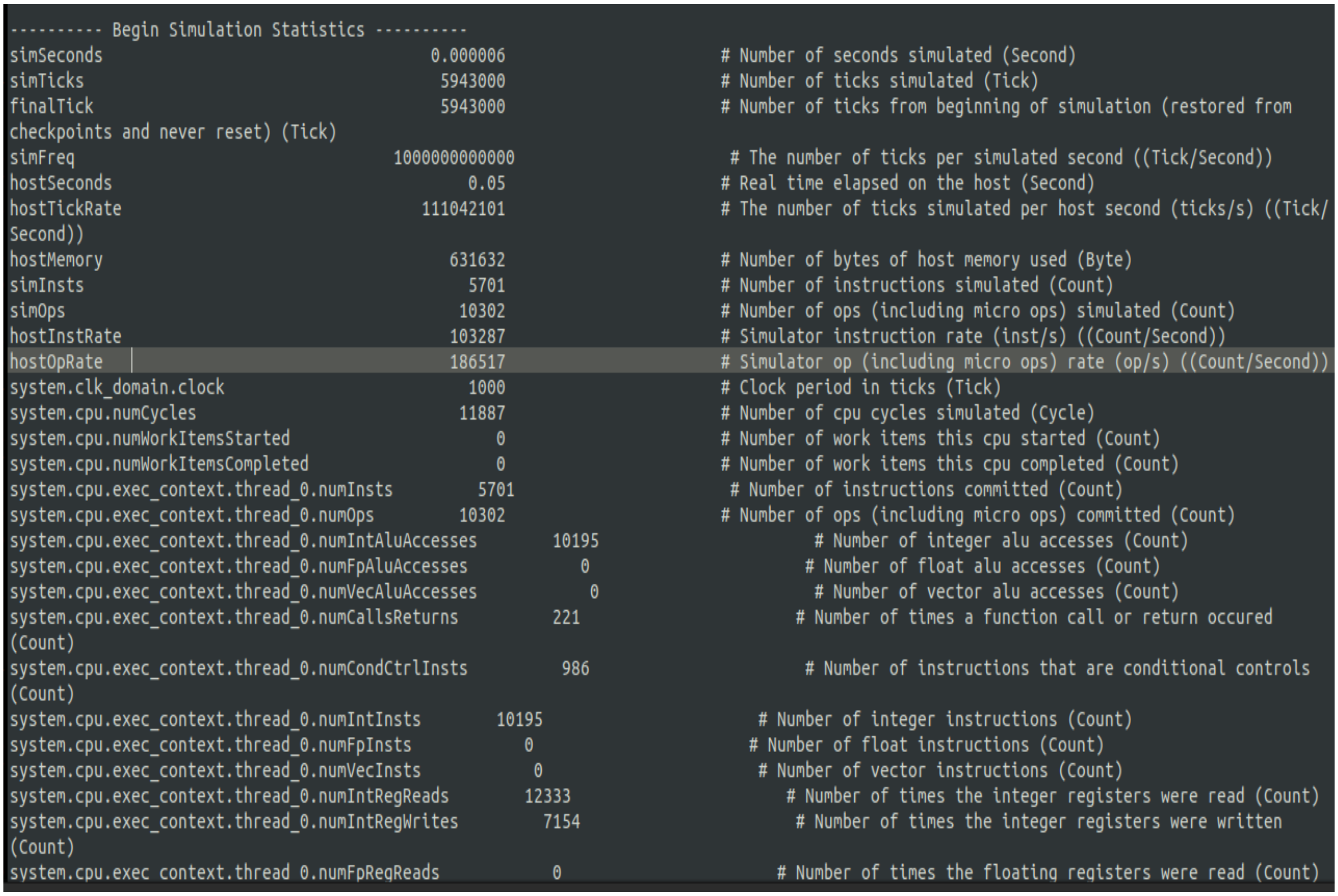


```
[system.cpu.interrupts.clk_domain]
type=DerivedClockDomain
clk_divider=16
clk_domain=system.cpu_clk_domain
eventq_index=0

[system.cpu.isa]
type=X86ISA
eventq_index=0
vendor_string=HygonGenuine

[system.cpu.mmu]
type=X86MMU
children=dtb itb
dtb=system.cpu.mmu.dtb
eventq_index=0
itb=system.cpu.mmu.itb

[system.cpu.mmu.dtb]
type=X86TLB
children=walker
entry_type=data
eventq_index=0
next_level=Null
size=64
system=system
walker=system.cpu.mmu.dtb.walker
```

Figure 3: Simobject and its parameters

In Figure 3. We can see a few more Simobjects. Not only these we have a few more SimObjects in config.ini file, these are just for examples to show.

2. **stats.txt**
   The first program is to print ”Hello World” and I have generated the performance log.



```
---------- Begin Simulation Statistics ----------
simSeconds                                  0.000006                       # Number of seconds simulated (Second)
simTicks                                     5943000                       # Number of ticks simulated (Tick)
finalTick                                    5943000                       # Number of ticks from beginning of simulation (restored from
checkpoints and never reset) (Tick)
simFreq                                1000000000000                        # The number of ticks per simulated second ((Tick/Second))
hostSeconds                                     0.05                       # Real time elapsed on the host (Second)
hostTickRate                               111042101                       # The number of ticks simulated per host second (ticks/s) ((Tick/
Second))
hostMemory                                    631632                       # Number of bytes of host memory used (Byte)
simInsts                                        5701                       # Number of instructions simulated (Count)
simOps                                         10302                       # Number of ops (including micro ops) simulated (Count)
hostInstRate                                  103287                       # Simulator instruction rate (inst/s) ((Count/Second))
hostOpRate                                    186517                       # Simulator op (including micro ops) rate (op/s) ((Count/Second))
system.clk_domain.clock                         1000                       # Clock period in ticks (Tick)
system.cpu.numCycles                           11887                       # Number of cpu cycles simulated (Cycle)
system.cpu.numWorkItemsStarted                     0                       # Number of work items this cpu started (Count)
system.cpu.numWorkItemsCompleted                   0                       # Number of work items this cpu completed (Count)
system.cpu.exec_context.thread_0.numInsts         5701                      # Number of instructions committed (Count)
system.cpu.exec_context.thread_0.numOps        10302                       # Number of ops (including micro ops) committed (Count)
system.cpu.exec_context.thread_0.numIntAluAccesses        10195                      # Number of integer alu accesses (Count)
system.cpu.exec_context.thread_0.numFpAluAccesses            0                     # Number of float alu accesses (Count)
system.cpu.exec_context.thread_0.numVecAluAccesses            0                      # Number of vector alu accesses (Count)
system.cpu.exec_context.thread_0.numCallsReturns          221                    # Number of times a function call or return occured
(Count)
system.cpu.exec_context.thread_0.numCondCtrlInsts          986                     # Number of instructions that are conditional controls
(Count)
system.cpu.exec_context.thread_0.numIntInsts        10195                      # Number of integer instructions (Count)
system.cpu.exec_context.thread_0.numFpInsts            0                      # Number of float instructions (Count)
system.cpu.exec_context.thread_0.numVecInsts            0                      # Number of vector instructions (Count)
system.cpu.exec_context.thread_0.numIntRegReads        12333                      # Number of times the integer registers were read (Count)
system.cpu.exec_context.thread_0.numIntRegWrites         7154                      # Number of times the integer registers were written
(Count)
system.cpu.exec_context.thread_0.numFpRegReads            0                      # Number of times the floating registers were read (Count)
```

Figure 4:

To generate the Performance log, we need to run a command, from inside the gem5 directory.

Listing 1: Command to Run Hello World

```
build/X86/gem5.opt configs/example/se.py --cmd=tests/test-progs/hello/bin/x86/linux/hello
```

We can see the Performance log below.

After the Simulation, a stats.txt file will be generated, to know the performance of the system what we have configured, we can check it from the stats.txt file which is in the directory ”/gem5/m5out/stats.txt”

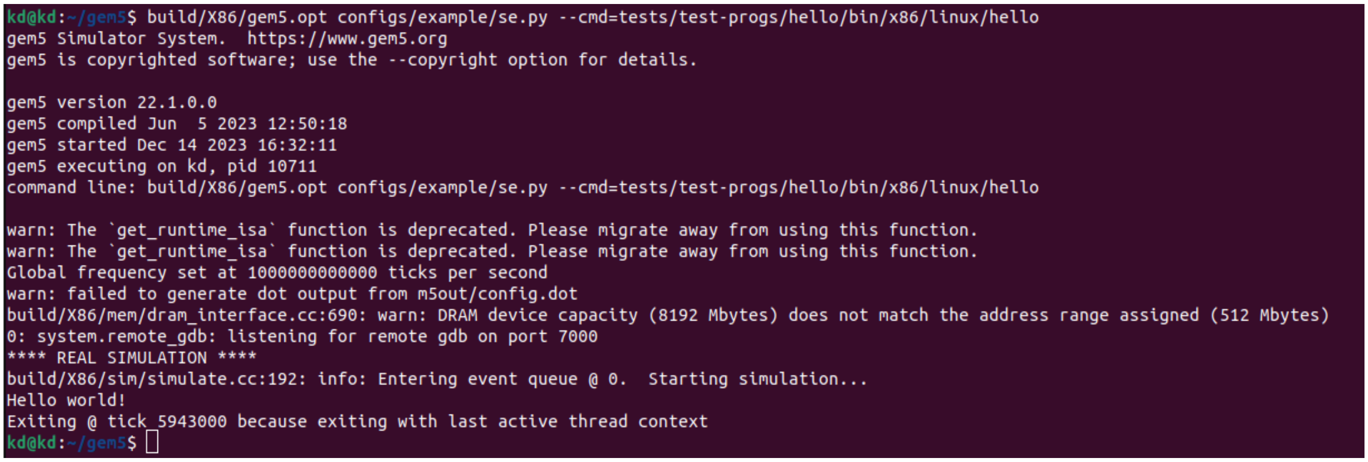
A few important statistics from stats.txt are

1. **sim_seconds** it tell us total simulated time for the simulation.
2. **sim_insts** it tells us number of instructions committed by the CPU
3. **host_inst _rate** it tells us the performance of Gem5.

Simulation Results		
Parameter	**Value**	**Description**
simSeconds	0.000006	# Number of seconds simulated (Second)
simInsts	5701	# Number of instructions simulated (Count)
hostInstRate	103287	# Simulator instruction rate (inst/s) ((Count/Second))

Table 1: Simulation Results

As we see, we have generated Simple code which Prints **Hello world!**. From the results if we see, **5701** instruction are simulated. Just to print the Hello World.



```
kd@kd:~/gem5$ build/X86/gem5.opt configs/example/se.py --cmd=tests/test-progs/hello/bin/x86/linux/hello
gem5 Simulator System.  https://www.gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 22.1.0.0
gem5 compiled Jun  5 2023 12:50:18
gem5 started Dec 14 2023 16:32:11
gem5 executing on kd, pid 10711
command line: build/X86/gem5.opt configs/example/se.py --cmd=tests/test-progs/hello/bin/x86/linux/hello

warn: The `get_runtime_isa` function is deprecated. Please migrate away from using this function.
warn: The `get_runtime_isa` function is deprecated. Please migrate away from using this function.
Global frequency set at 1000000000000 ticks per second
warn: failed to generate dot output from m5out/config.dot
build/X86/mem/dram_interface.cc:690: 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:192: info: Entering event queue @ 0.  Starting simulation...
Hello world!
Exiting @ tick 5943000 because exiting with last active thread context
kd@kd:~/gem5$ 
```

Figure 5: gem5 output

Simulation Results		
Parameter	**Value**	**Description**
system.cpu.numCycles	11887	# Number of cpu cycles simulated (Cycle)
system.cpu.exec_context.thread_0.numMemRefs	2027	# Number of memory refs (Count)
system.cpu.exec_context.thread_0.numLoadInsts	1084	# Number of load instructions (Count)
system.cpu.exec_context.thread_0.numStoreInsts	943	# Number of store instructions (Count)
system.cpu.exec_context.thread_0.notIdleFraction	1.000000	# Percentage of non-idle cycles (Ratio)
system.cpu.exec_context.thread_0.idleFraction	0.000000	# Percentage of idle cycles (Ratio)
system.cpu.exec_context.thread_0.numBranches	1306	# Number of branches fetched (Count)
system.mem_ctrls.dram.rank0.totalEnergy	2282400	# Total energy per rank (pJ) (Joule)

Table 2: Simulation Results

The simulation results provide insights into the performance of the CPU system under consideration. Here are key observations based on the presented statistics:

CPU Cycle Simulation:

The simulation involved a total of 11,887 CPU cycles. This metric serves as a fundamental indicator of the overall computational activity during the simulated period.

2 Memory Access Patterns:

2.1 Memory References:

The CPU executed 2,027 memory references, reflecting the interaction between the processor and the system's memory.

2.2 Load Instructions:

A notable 1,084 load instructions were executed, indicating the frequency at which data was fetched from memory.

2.3 Store Instructions:

The CPU performed 943 store instructions, representing the instances where data was written back to memory.

3 Execution Characteristics:

3.1 Non-idle Cycles:

The CPU exhibited a 100% non-idle fraction, implying continuous utilization without any idle periods.

3.2 Idle Cycles:

Conversely, there were no reported idle cycles, reinforcing the system's constant engagement in computational tasks.

4 Branch Prediction:

The CPU's branch prediction mechanism encountered 1,306 branches during the simulation, showcasing the number of decisions made regarding program flow.

5 Energy Consumption:

The total energy consumption for the DRAM rank 0 was measured at 2,282,400 picojoules (pJ). This metric is crucial for understanding the energy requirements of the memory subsystem.

6 Conclusion:

In summary, the simulation results suggest a highly active and efficient CPU system with continuous utilization, effective memory access, and successful branch predictions. The absence of idle cycles aligns with a system that maximizes computational throughput. The energy consumption metric provides insights into the power requirements of the DRAM subsystem, a critical consideration for overall system efficiency.