Introduction to gem5

gem5 is a versatile, open-source tool widely used in the field of computer architecture research. It allows for the simulation and modeling of a wide range of computer systems, from basic CPUs to complex multi-core architectures with intricate memory hierarchies and network-on-chip systems. One of the key advantages of gem5 is its support for multiple instruction set architectures (ISAs) like x86, ARM, and RISC-V, making it highly adaptable for different research needs.

A significant feature of gem5 is its ability to run both full-system simulations, which replicate the entire computing environment including the operating system, and system-call emulation, which focuses on user-level programs while utilizing the host system's kernel. This makes gem5 particularly useful for testing new architectural ideas, optimizing system performance, and analyzing energy efficiency and security aspects. Researchers can explore and refine new hardware designs using gem5 before committing to the costly process of physical implementation.

References:

Binkert, N., Beckmann, B., Black, G., Reinhardt, S. K., Saidi, A., Basu, A., ... & Wood, D. A. (2011). The gem5 simulator. ACM SIGARCH Computer Architecture News, 39(2), 1-7. doi:10.1145/2024716.2024718.

"The gem5 Simulator System." gem5.org. Available at: https://www.gem5.org/

Environment Setup

Requirements:

Building gem5 required several software dependencies to ensure the build process runs smoothly. Below is a list of the key dependencies (**Obtained from**: **https://www.gem5.org/documentation/general_docs/building#dependencies**)

git : gem5 uses git for version control.

gcc: gcc is used to compiled gem5. Version >=10 must be used. We support up to gcc Version 13.

Clang: Clang can also be used. At present, we support Clang 7 to Clang 16 (inclusive).

SCons : gem5 uses SCons as its build environment. SCons 3.0 or greater must be used.

Python 3.6+ : gem5 relies on Python development libraries. gem5 can be compiled and run in environments using Python 3.6+.

protobuf 2.1+ (Optional): The protobuf library is used for trace generation and playback.

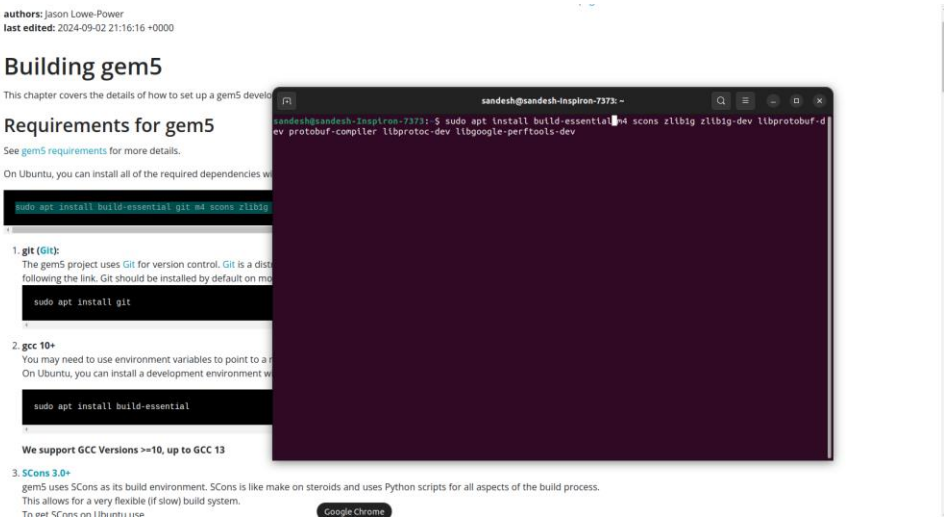
Boost (Optional): The Boost library is a set of general purpose C++ libraries. It is a necessary dependency if you wish to use the SystemC implementation.

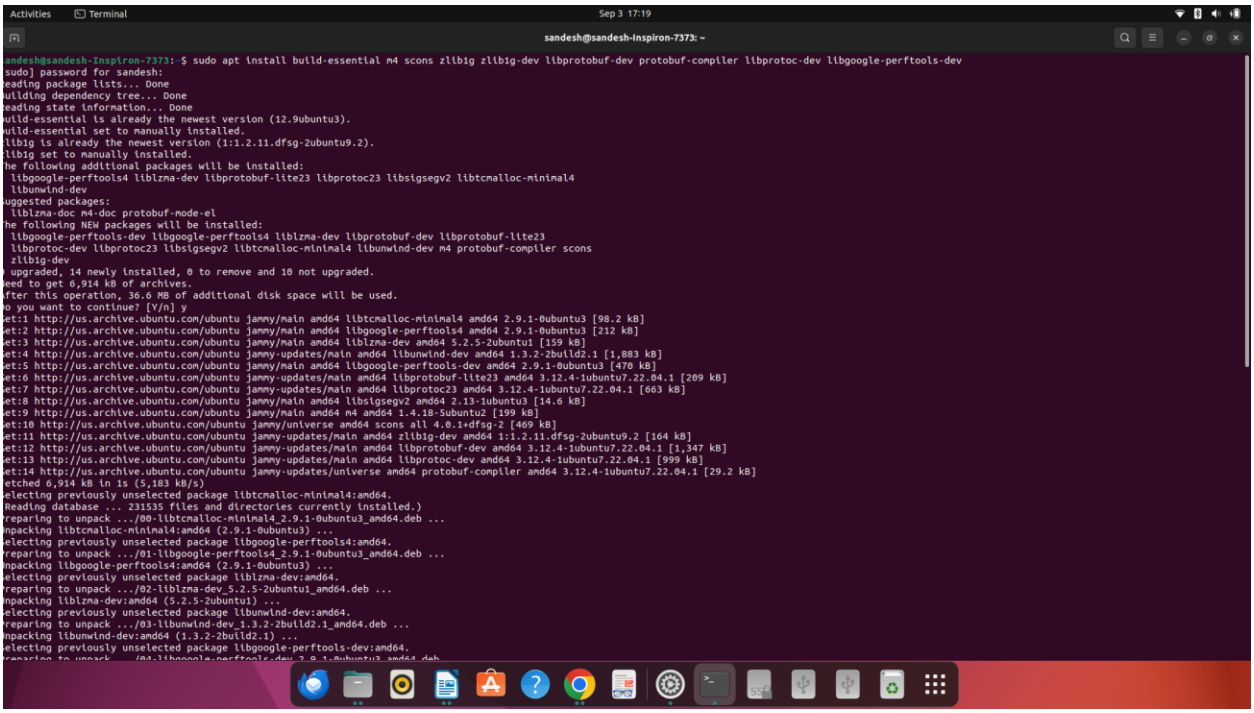
The command I ran to install all dependencies at once:

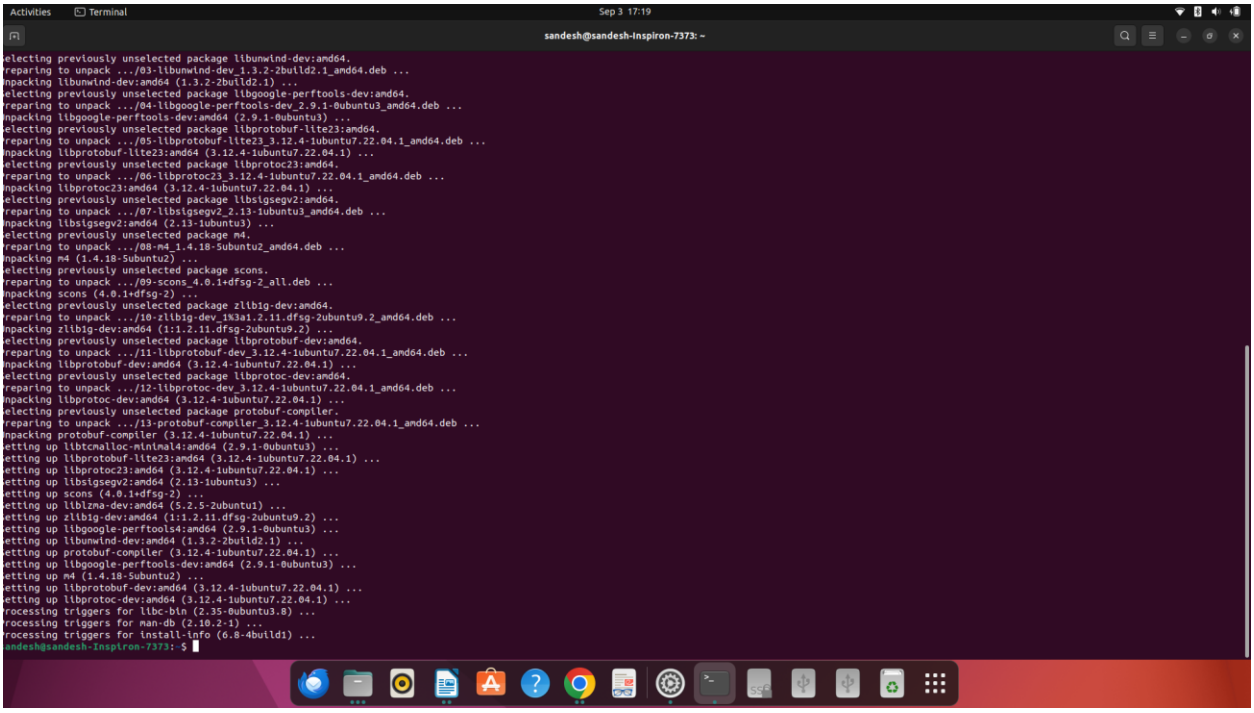
(I did not include git and python as I had that already installed in my machine)

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

Screenshot

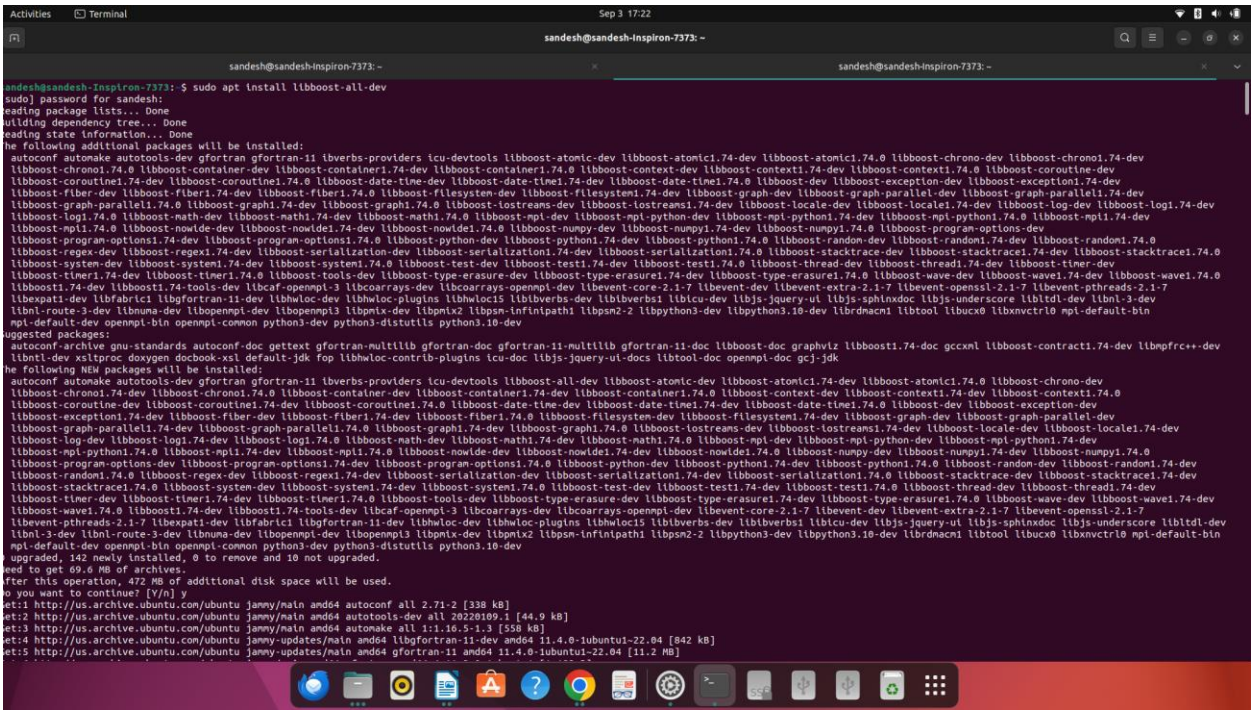






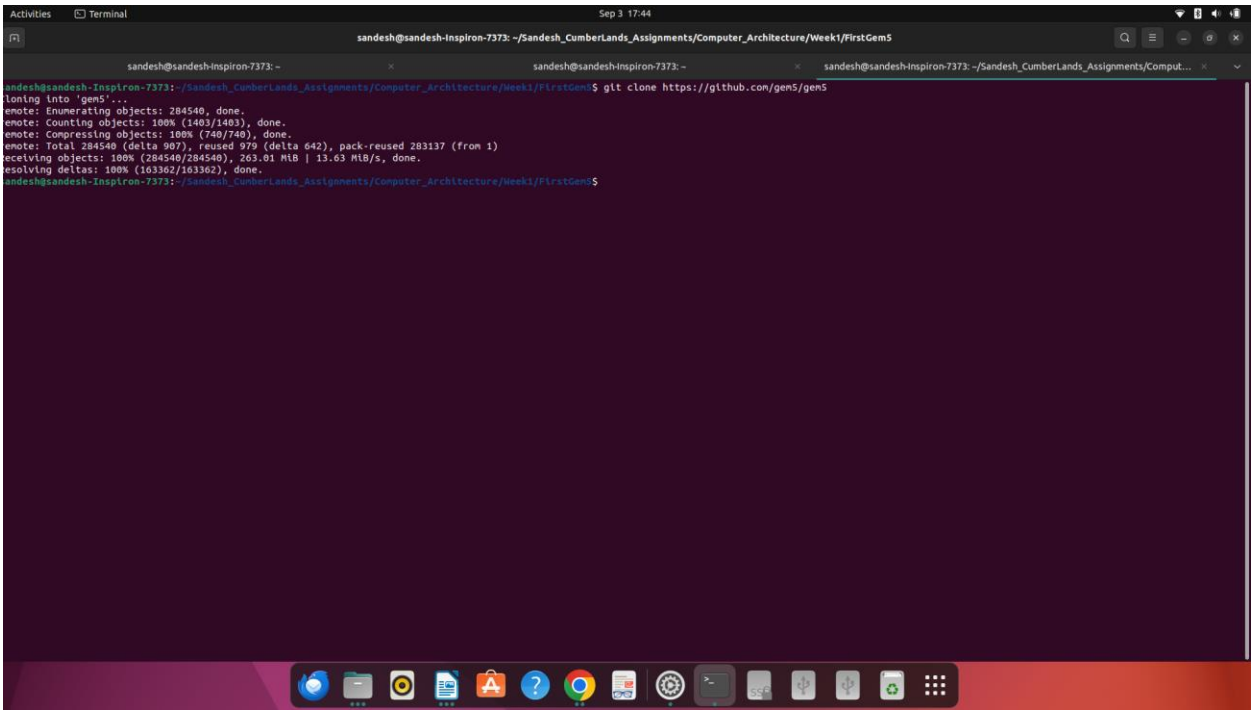
I also installed Boost, which was optional.

Command used for Boost Installation: sudo apt install libboost-all-dev



Cloning the gem5 repository

For cloning the gem5 repository, I used the following command:
git clone https://github.com/gem5/gem5



Building gem5 for a specific CPU model (X86 in our case)

Command used for building:
scons build/X86/gem5.opt -j8 (8 as I have 8 cores in my computer)
(Referenced from:
https://www.gem5.org/documentation/general_docs/building#dependencies)

Screenshot of my command:

```
gem5
sandesh@sandesh-Inspiron-7373:~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5$ cd gem5/
sandesh@sandesh-Inspiron-7373:~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem5$ ls
build_opts   CODE-OF-CONDUCT.md  CONTRIBUTING.md  ext      KCONFIG.md  MAINTAINERS.yaml          pyproject.toml  RELEASE-NOTES.md  SConstruct  src     TESTING.md  util
build_tools  configs             COPYING          include  LICENSE     optional-requirements.txt  README.md       requirements.txt  site_scons  system  tests
sandesh@sandesh-Inspiron-7373:~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem5sandesh@sandesh-Inspiron-7373:~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem
5sandesh@sandesh-Inspiron-7373:~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem5$ s
 s
cons build/X86/gem5.opt -j8
scons: Reading SConscript files ...

You're missing the pre-commit/commit-msg hooks. These hook help to ensure your
code follows gem5's style rules on git commit and your commit messages follow
our commit message requirements. This script will now install these hooks in
your .git/hooks/ directory.
Press enter to continue, or ctrl-c to abort:
 Cannot find 'pre-commit'. Please ensure all Python requirements are
installed. This can be done via 'pip install -r requirements.txt'.
It is strongly recommended you install the pre-commit hooks before working with gem5. Do you want to continue compilation (y/n)?
y
Mkdir("/home/sandesh/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem5/build/X86/gem5.build")
Checking for linker -Wl,--as-needed support... yes
Checking for compiler -gz support... yes
Checking for linker -gz support... yes
Info: Using Python config: python3-config
Checking for C header file Python.h... yes
Checking Python version... 3.10.12
Checking for accept(0,0,0) in C++ library None... yes
Checking for zlibVersion() in C++ library z... yes
Checking for C library tcmalloc_minimal... yes
```

After taking almost 45 minutes, the build was completed. I could see these logs

```
Activities  Terminal                                                    Sep 3 21:24
sandesh@sandesh-Inspiron-7373: ~/Sandesh_CumberLands_Assignments/Computer_Architecture/Week1/FirstGem5/gem5
sandesh@sandesh-Inspiron-7373: ~   sandesh@sandesh-Inspiron-7373: ~   sandesh@sandesh-Inspiron-7373: ~/Sandesh_CumberLand...   sandesh@sandesh-Inspiron-7373: ~/Sandesh_CumberLand...
[SO Param] m5.objects.QoSMemSinkInterface, QoSMemSinkInterface -> X86/python/_m5/param_QoSMemSinkInterface.cc
[SO Param] m5.objects.QoSMemSinkInterface, QoSMemSinkInterface -> X86/params/QoSMemSinkInterface.hh
[      CXX] X86/mem/qos/QoSPolicy.py.cc -> .o
[      CXX] X86/python/_m5/param_QoSMemSinkInterface.cc -> .o
[      CXX] X86/python/_m5/param_QoSMemSinkCtrl.cc -> .o
[SO Param] m5.objects.QoSPolicy, QoSPolicy -> X86/python/_m5/param_QoSPolicy.cc
[      CXX] X86/python/_m5/param_QoSPolicy.cc -> .o
[SO Param] m5.objects.QoSPolicy, QoSFixedPriorityPolicy -> X86/python/_m5/param_QoSFixedPriorityPolicy.cc
[SO Param] m5.objects.QoSPolicy, QoSFixedPriorityPolicy -> X86/params/QoSFixedPriorityPolicy.hh
[      CXX] X86/python/_m5/param_QoSFixedPriorityPolicy.cc -> .o
[SO Param] m5.objects.QoSPolicy, QoSPropFairPolicy -> X86/python/_m5/param_QoSPropFairPolicy.cc
[SO Param] m5.objects.QoSPolicy, QoSPropFairPolicy -> X86/params/QoSPropFairPolicy.hh
[      CXX] X86/mem/qos/QoSTurnaround.py.cc -> .o
[      CXX] X86/python/_m5/param_QoSPropFairPolicy.cc -> .o
[SO Param] m5.objects.QoSTurnaround, QoSTurnaroundPolicy -> X86/python/_m5/param_QoSTurnaroundPolicy.cc
[      CXX] X86/python/_m5/param_QoSTurnaroundPolicy.cc -> .o
[SO Param] m5.objects.QoSTurnaround, QoSTurnaroundPolicyIdeal -> X86/python/_m5/param_QoSTurnaroundPolicyIdeal.cc
[SO Param] m5.objects.QoSTurnaround, QoSTurnaroundPolicyIdeal -> X86/params/QoSTurnaroundPolicyIdeal.hh
[      CXX] X86/python/_m5/param_QoSTurnaroundPolicyIdeal.cc -> .o
[      CXX] src/mem/qos/policy.cc -> X86/mem/qos/policy.o
[      CXX] src/mem/qos/policy_fixed_prio.cc -> X86/mem/qos/policy_fixed_prio.o
[      CXX] src/mem/qos/policy_pf.cc -> X86/mem/qos/policy_pf.o
[      CXX] src/mem/qos/turnaround_policy_ideal.cc -> X86/mem/qos/turnaround_policy_ideal.o
[      CXX] src/mem/qos/q_policy.cc -> X86/mem/qos/q_policy.o
[      CXX] src/mem/qos/mem_ctrl.cc -> X86/mem/qos/mem_ctrl.o
[      CXX] src/mem/qos/mem_sink.cc -> X86/mem/qos/mem_sink.o
[      CXX] X86/python/m5/defines.py.cc -> .o
[      CXX] X86/python/m5/info.py.cc -> .o
[      CXX] src/base/date.cc -> X86/base/date.o
[     LINK] -> X86/gem5.opt
scons: done building targets.
*** Summary of Warnings ***
Warning: Header file <png.h> not found.
         This host has no libpng library.
         Disabling support for PNG framebuffers.
Warning: Couldn't find HDF5 C++ libraries. Disabling HDF5 support.
Warning: Header file <capstone/capstone.h> not found.
         This host has no capstone library installed.
```

After the build was completed, I used the following command to run the basic simulation to output “Hello world”. Even if se.py has been deprecated and not suggested for use, I use it for the purpose of this assignment as its starting assignment to get some idea about gem5 and basic simulation.

./build/X86/gem5.opt configs/deprecated/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello

Error occured and Troubleshooting

I did not see any error during building phase but saw some errors after trying to run simulation using some commands. I have listed those commands and the error output below:

1 ./build/X86/gem5.opt --version
-> Usage

=====

gem5.opt [gem5 options] script.py [script options]

gem5.opt: error: no such option: --version

My finding : **It seems that gem5.opt does not support the --version flag directly. Unlike some other software, gem5 doesn't have a built-in command-line option to display its version.**

2. ./build/X86/gem5.opt configs/example/se.py -c "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 24.0.0.1

gem5 compiled Sep 3 2024 19:25:54

gem5 started Sep 3 2024 19:32:43

gem5 executing on sandesh-Inspiron-7373, pid 32372

command line: ./build/X86/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello

fatal: The 'configs/example/se.py' script has been deprecated. It can be found in 'configs/deprecated/example' if required. Its usage should be avoided as it will be removed in future releases of gem5.

My findings: **It looks like gem5 build is working correctly, but the script I tried to run,**

se.py, has been deprecated. The gem5 developers have moved it to a deprecated directory, and they recommend not using it as it may be removed in future versions.

3. ./build/X86/gem5.opt configs/example/starter_se.py -c tests/test-progs/hello/bin/x86/linux/hello

Script configs/example/starter_se.py not found

My findings: I**t seems that the script starter_se.py does not exist in gem5 setup. This could be due to differences in directory structure or script availability in the version of gem5 you are using**

Fix:

Finally I found some files I had been trying to run in depricated folder and thought of using it. Even though it's deprecated, I thought for the purpose of this session I could use it to validate my simulation.

./build/X86/gem5.opt configs/deprecated/example/se.py -c 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 24.0.0.1

gem5 compiled Sep 3 2024 19:25:54

gem5 started Sep 3 2024 19:37:08

gem5 executing on sandesh-Inspiron-7373, pid 32400

command line: ./build/X86/gem5.opt configs/deprecated/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello

warn: The se.py script is deprecated. It will be removed in future releases of gem5.

Global frequency set at 1000000000000 ticks per second

warn: No dot file generated. Please install pydot to generate the dot file and pdf.

src/mem/dram_interface.cc:690: warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)

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.

system.remote_gdb: Listening for connections on port 7000

**** REAL SIMULATION ****

src/sim/simulate.cc:199: info: Entering event queue @ 0. Starting simulation...

Hello world!

Exiting @ tick 5943000 because exiting with last active thread context