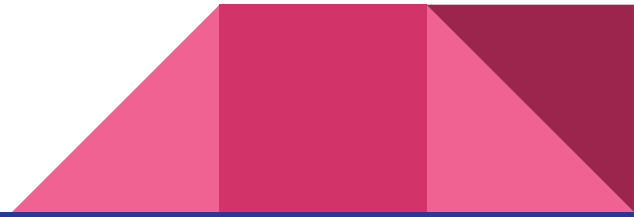
Gem5 Gear-Up Session

Outline

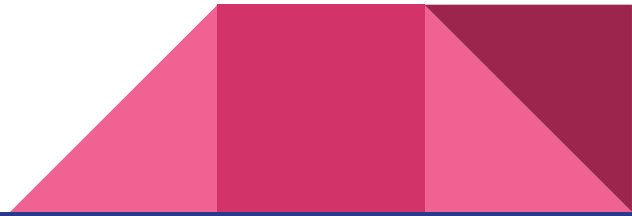
- Overview of gem5 (what is it? why are we using it?)
- Understanding the front-end
- Understanding the back-end
- Debugging in gem5
- *PS: if you are working outside the docker container and haven't compiled gem5 yet, please pull the repo and do that now!*

A Note Before We Start…



The Sales Pitch

“The gem5 simulator provides a flexible, modular simulation system that is capable of **evaluating a broad range of systems** and is widely available to all researchers. This infrastructure provides flexibility by offering **a diverse set of CPU models**, system execution modes, and **memory system models**. A commitment to **modularity** and clean interfaces allows researchers to *focus on a particular aspect of code without understanding the entire code base*.”



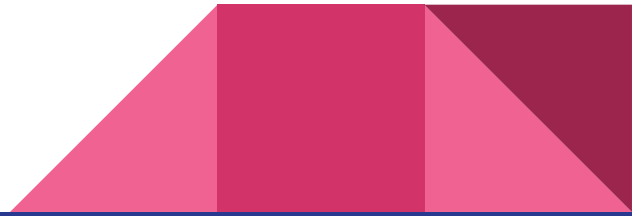
The Good

- Run real programs!
- Custom architecture configurations!
- Different CPU properties!
- Different memory hierarchies!
- Custom devices!
- Easily run different ISAs!

The Bad

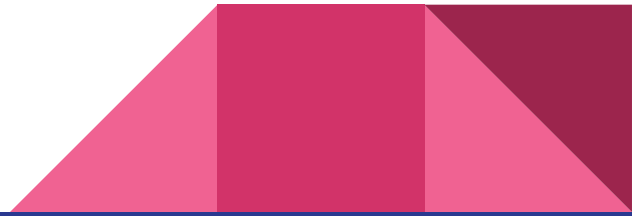
- 10887 unique files; 1.2 million lines of code(!)
- Debugging can be tricky…

- Please don’t try to understand everything



Workflow

- Compile gem5 with *scons* (you can skip through the warnings)
- Whenever you update the back-end, you need to recompile
- If you update the front-end, you don’t need to recompile



Workflow (expected output)

```
sam@moosilauke:~/gem5-assignments-stencil$ ./build/RISCV/gem5.debug configs/gem5-gearup/example1.py
gem5 Simulator System.  https://www.gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 23.1.0.0
gem5 compiled Feb 22 2024 14:50:21
gem5 started Feb 22 2024 16:39:04
gem5 executing on moosilauke, pid 64268
command line: ./build/RISCV/gem5.debug configs/gem5-gearup/example1.py

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/arch/riscv/isa.cc:275: info: RVV enabled, VLEN = 256 bits, ELEN = 64 bits
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
src/mem/coherent_xbar.cc:140: warn: CoherentXBar system.membus has no snooping ports attached!
Beginning simulation!
src/sim/simulate.cc:199: info: Entering event queue @ 0.  Starting simulation...
src/sim/syscall_emul.hh:1014: warn: readlink() called on '/proc/self/exe' may yield unexpected results in various settings.
      Returning '/home/sam/gem5-assignments-stencil/tests/test-progs/hello/bin/riscv/linux/hello'
src/sim/mem_state.cc:448: info: Increasing stack size by one page.
Hello world!
Exiting @ tick 479821000 because exiting with last active thread context
sam@moosilauke:~/gem5-assignments-stencil$
```

Let’s jump into the code!

The slides will have relevant screenshots to refer back to.

Front-End

- Gem5 uses a Python front end to declare which components you want to use in the simulation
- Primarily in *configs/…* directory
- Components are connected via *ports* by declaring obj1.port_name = obj2.port_name (ordering doesn't matter)

```
"""
This is the RISCV equivalent to `simple.py` (which is designed to run usin
X86 ISA). More detailed documentation can be found in `simple.py`.
"""

import m5
from m5.objects import *

system = System()

system.clk_domain = SrcClockDomain()
system.clk_domain.clock = "1GHz"
system.clk_domain.voltage_domain = VoltageDomain()

system.mem_mode = "timing"
system.mem_ranges = [AddrRange("512MB")]
system.cpu = RiscvTimingSimpleCPU()

system.membus = SystemXBar()

system.cpu.icache_port = system.membus.cpu_side_ports
system.cpu.dcache_port = system.membus.cpu_side_ports

system.cpu.createInterruptController()

system.mem_ctrl = MemCtrl()
system.mem_ctrl.dram = DDR3_1600_8x8()
system.mem_ctrl.dram.range = system.mem_ranges[0]
system.mem_ctrl.port = system.membus.mem_side_ports

system.system_port = system.membus.cpu_side_ports

thispath = os.path.dirname(os.path.realpath(__file__))
binary = os.path.join(
    thispath,
    "../../../",
    "tests/test-progs/hello/bin/riscv/linux/hello",
)

system.workload = SEWorkload.init_compatible(binary)

process = Process()
process.cmd = [binary]
system.cpu.workload = process
system.cpu.createThreads()

root = Root(full_system=False, system=system)
m5.instantiate()

print(f"Beginning simulation!")
exit_event = m5.simulate()
print(f"Exiting @ tick {m5.curTick()} because {exit_event.getCause()}")
```

Front-End

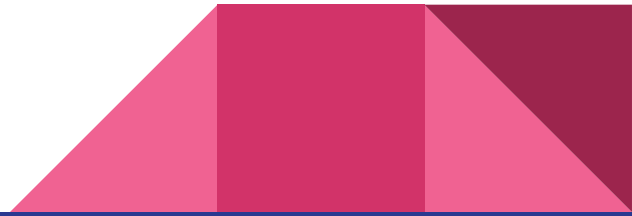
- CPU Models
  - <ISA>TimingSimpleCPU: single-cycle processor
  - <ISA>MinorCPU: 4-stage pipelined processor (we will explore this processor in depth in HW5)
  - <ISA>O3CPU: pipelined, out-of-order (O3) processor
- Memory Models
  - Atomic: useful for checkpoints (will only be relevant for final project, maybe)
  - Functional: used in the simulator boot to preset values relevant for the simulation (i.e., putting the instructions from the binary in memory)
  - Timing: the accurate memory mode for counting stats during the simulation
- Simulation Modes
  - Syscall Emulation (SE): system calls are emulated in the simulator (doesn't model the OS)
  - Full System (FS): disk image (file system, system binaries, etc.), OS are passed as input to the simulations (i.e., customizable)

Front-End (a note about binaries)

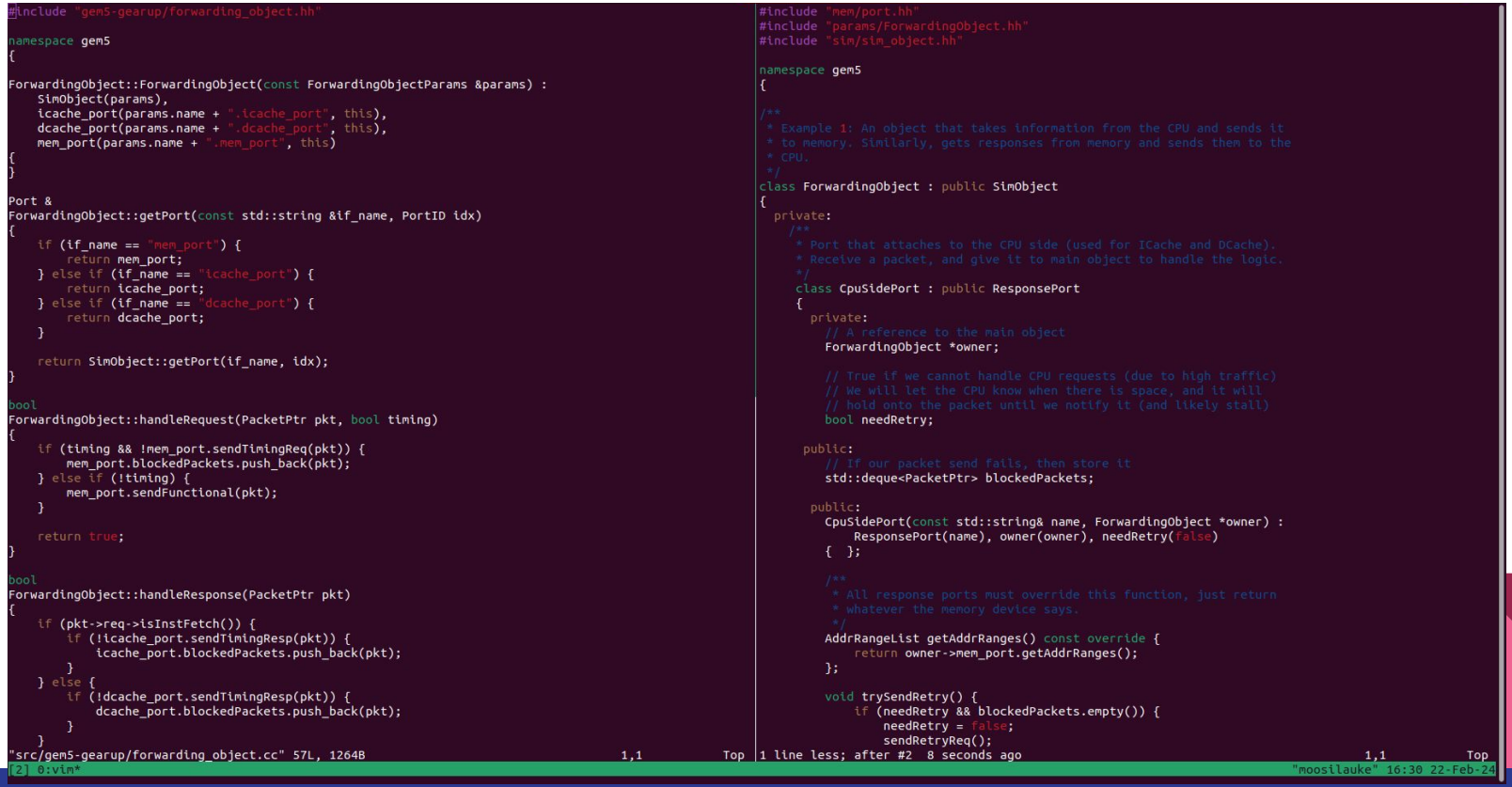
- If you want to install cross compile for RISC-V (optional): https://github.com/riscv-collab/riscv-gnu-toolchain
- All programs that you will need will be included in the stencil repo

Back End

- Mostly C++ in the back end (everything that you will do in this class will be in C++)
- After updating C++, recompile by calling *scons build/RISCV/gem5.debug*
- This step takes forever, prints lots of output, and it’s super memory intensive… don’t worry all of this is normal (413MB binary)



Back End



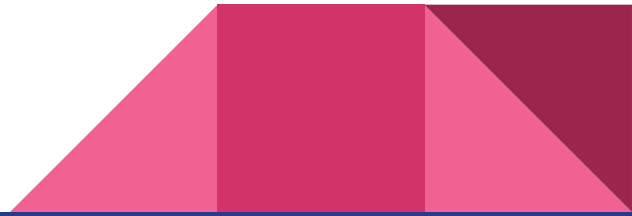
```
#include "gem5-gearup/forwarding_object.hh"

namespace gem5
{

ForwardingObject::ForwardingObject(const ForwardingObjectParams &params) :
    SimObject(params),
    icache_port(params.name + ".icache_port", this),
    dcache_port(params.name + ".dcache_port", this),
    mem_port(params.name + ".mem_port", this)
{
}

Port &
ForwardingObject::getPort(const std::string &if_name, PortID idx)
{
    if (if_name == "mem_port") {
        return mem_port;
    } else if (if_name == "icache_port") {
        return icache_port;
    } else if (if_name == "dcache_port") {
        return dcache_port;
    }

    return SimObject::getPort(if_name, idx);
}

bool
ForwardingObject::handleRequest(PacketPtr pkt, bool timing)
{
    if (timing && !mem_port.sendTimingReq(pkt)) {
        mem_port.blockedPackets.push_back(pkt);
    } else if (!timing) {
        mem_port.sendFunctional(pkt);
    }

    return true;
}

bool
ForwardingObject::handleResponse(PacketPtr pkt)
{
    if (pkt->req->isInstFetch()) {
        if (!icache_port.sendTimingResp(pkt)) {
            icache_port.blockedPackets.push_back(pkt);
        }
    } else {
        if (!dcache_port.sendTimingResp(pkt)) {
            dcache_port.blockedPackets.push_back(pkt);
        }
    }
"src/gem5-gearup/forwarding_object.cc" 57L, 1264B                    1,1           Top
```

```
#include "mem/port.hh"
#include "params/ForwardingObject.hh"
#include "sim/sim_object.hh"

namespace gem5
{

/**
 * Example 1: An object that takes information from the CPU and sends it
 * to memory. Similarly, gets responses from memory and sends them to the
 * CPU.
 */
class ForwardingObject : public SimObject
{
  private:
    /**
     * Port that attaches to the CPU side (used for ICache and DCache).
     * Receive a packet, and give it to main object to handle the logic.
     */
    class CpuSidePort : public ResponsePort
    {
      private:
        // A reference to the main object
        ForwardingObject *owner;

        // True if we cannot handle CPU requests (due to high traffic)
        // We will let the CPU know when there is space, and it will
        // hold onto the packet until we notify it (and likely stall)
        bool needRetry;

      public:
        // If our packet send fails, then store it
        std::deque<PacketPtr> blockedPackets;

       public:
        CpuSidePort(const std::string& name, ForwardingObject *owner) :
            ResponsePort(name), owner(owner), needRetry(false)
        {  };

        /**
         * All response ports must override this function, just return
         * whatever the memory device says.
         */
        AddrRangeList getAddrRanges() const override {
            return owner->mem_port.getAddrRanges();
        };

        void trySendRetry() {
            if (needRetry && blockedPackets.empty()) {
                needRetry = false;
                sendRetryReq();
1 line less; after #2  8 seconds ago                                 1,1           Top
```

```
[2] 0:vim*                                                    "moosilauke" 16:30 22-Feb-24
```

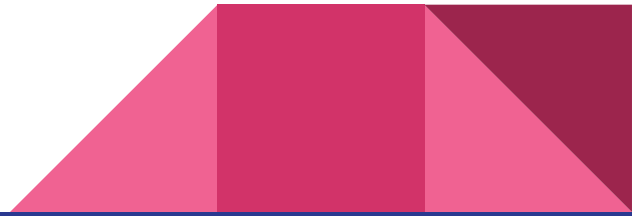
Back End

- Information sent between simulated devices as “packets”
  - We will try to make sure all of the relevant packet functions are in the assignment handout for each assignment
- Packets are communicated via ports, which act as the starting/stopping point for the logic of each device



Back End

- Gem5 is an event-based simulator to model timing
- There is an “EventQueue” object where all routines are enqueued with a lambda function and a cycle time to execute the lambda function
    - The EventQueue pops each routine and uses the scheduled execution time to “advance time”



Back End (clean compilation)

```
Checking Python version... (cached) 3.10.12
Checking for accept(0,0,0) in C++ library None... (cached) yes
Checking for zlibVersion() in C++ library z... (cached) yes
Checking for C library tcmalloc_minimal... (cached) no
Checking for C library tcmalloc... (cached) no
Warning: You can get a 12% performance improvement by installing tcmalloc (libgoogle-perftools-dev package on Ubuntu or RedHat).
Building in /home/sam/gem5-assignments-stencil/build/RISCV
"build_tools/kconfig_base.py" "/home/sam/gem5-assignments-stencil/build/RISCV/gem5.build/Kconfig" "/home/sam/gem5-assignments-stencil/src/Kconfig"
Checking for C header file linux/if_tun.h... (cached) yes
Checking for backtrace_symbols_fd((void *)1, 0, 0) in C library None... (cached) yes
Checking size of struct kvm_xsave ... (cached) yes
Checking for shm_open("/test", 0, 0) in C library None... (cached) yes
Warning: While checking protoc version: [Errno 2] No such file or directory: 'protoc'
Warning: Protocol buffer compiler (protoc) not found.
         Please install protobuf-compiler for tracing support.
Checking for C header file capstone/capstone.h... (cached) no
Warning: Header file <capstone/capstone.h> not found.
         This host has no capstone library installed.
Checking for C header file linux/kvm.h... (cached) yes
Checking for timer_create(CLOCK_MONOTONIC, NULL, NULL) in C library None... (cached) yes
Checking for member exclude_host in struct perf_event_attr...(cached) yes
Checking for C header file fenv.h... (cached) yes
Checking for C header file png.h... (cached) no
Warning: Header file <png.h> not found.
         This host has no libpng library.
         Disabling support for PNG framebuffers.
Checking for clock_nanosleep(0,0,NULL,NULL) in C library None... (cached) yes
Checking for C header file valgrind/valgrind.h... (cached) no
Checking for pkg-config package hdf5-serial... (cached) no
Checking for pkg-config package hdf5... (cached) no
Checking for H5Fcreate("", 0, 0, 0) in C library hdf5... (cached) no
Warning: Couldn't find HDF5 C++ libraries. Disabling HDF5 support.
Checking whether __i386__ is declared... (cached) no
Checking whether __x86_64__ is declared... (cached) yes
Checking for compiler -Wno-self-assign-overloaded support... (cached) yes
Checking for linker -Wno-free-nonheap-object support... (cached) yes
scons: done reading SConscript files.
scons: Building targets ...
 [VER TAGS]  -> RISCV/sim/tags.cc
scons: `build/RISCV/gem5.debug' is up to date.
scons: done building targets.
*** Summary of Warnings ***
Warning: You can get a 12% performance improvement by installing tcmalloc (libgoogle-perftools-dev package on Ubuntu or RedHat).
Warning: While checking protoc version: [Errno 2] No such file or directory: 'protoc'
Warning: Protocol buffer compiler (protoc) not found.
         Please install protobuf-compiler for tracing support.
Warning: Header file <capstone/capstone.h> not found.
         This host has no capstone library installed.
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.
sam@moosilauke:~/gem5-assignments-stencil$
```

Back End (compilation error)

```
"build_tools/kconfig_base.py" "/home/sam/gem5-assignments-stencil/build/RISCV/gem5.build/Kconfig" "/home/sam/gem5-assignments-stencil/src/Kconfig"
Checking for C header file linux/if_tun.h... (cached) yes
Checking for backtrace_symbols_fd((void *)1, 0, 0) in C library None... (cached) yes
Checking size of struct kvm_xsave ... (cached) yes
Checking for shm_open("/test", 0, 0) in C library None... (cached) yes
Warning: While checking protoc version: [Errno 2] No such file or directory: 'protoc'
Warning: Protocol buffer compiler (protoc) not found.
         Please install protobuf-compiler for tracing support.
Checking for C header file capstone/capstone.h... (cached) no
Warning: Header file <capstone/capstone.h> not found.
         This host has no capstone library installed.
Checking for C header file linux/kvm.h... (cached) yes
Checking for timer_create(CLOCK_MONOTONIC, NULL, NULL) in C library None... (cached) yes
Checking for member exclude_host in struct perf_event_attr...(cached) yes
Checking for C header file fenv.h... (cached) yes
Checking for C header file png.h... (cached) no
Warning: Header file <png.h> not found.
         This host has no libpng library.
         Disabling support for PNG framebuffers.
Checking for clock_nanosleep(0,0,NULL,NULL) in C library None... (cached) yes
Checking for C header file valgrind/valgrind.h... (cached) no
Checking for pkg-config package hdf5-serial... (cached) no
Checking for pkg-config package hdf5... (cached) no
Checking for H5Fcreate("", 0, 0, 0) in C library hdf5... (cached) no
Warning: Couldn't find HDF5 C++ libraries. Disabling HDF5 support.
Checking whether __i386__ is declared... (cached) no
Checking whether __x86_64__ is declared... (cached) yes
Checking for compiler -Wno-self-assign-overloaded support... (cached) yes
Checking for linker -Wno-free-nonheap-object support... (cached) yes
scons: done reading SConscript files.
scons: Building targets ...
 [      CXX] src/gem5-gearup/buggy_object1.cc -> RISCV/gem5-gearup/buggy_object1.do
src/gem5-gearup/buggy_object1.cc: In member function ‘bool gem5::BuggyObject1::handleRequest(gem5::PacketPtr, bool)’:
src/gem5-gearup/buggy_object1.cc:43:16: error: expected ‘;’ before ‘}’ token
   43 |     return true
      |                ^
      |                ;
   44 | }
      | ~
scons: *** [build/RISCV/gem5-gearup/buggy_object1.do] Error 1
scons: building terminated because of errors.
*** Summary of Warnings ***
Warning: You can get a 12% performance improvement by installing tcmalloc (libgoogle-perftools-dev package on Ubuntu or RedHat).
Warning: While checking protoc version: [Errno 2] No such file or directory: 'protoc'
Warning: Protocol buffer compiler (protoc) not found.
         Please install protobuf-compiler for tracing support.
Warning: Header file <capstone/capstone.h> not found.
         This host has no capstone library installed.
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.
sam@moosilauke:~/gem5-assignments-stencil$
```

Debugging (incorrect outputs)

```
sam@moosilauke:~/gem5-assignments-stencil$ ./build/RISCV/gem5.debug configs/gem5-gearup/example2.py
gem5 Simulator System.  https://www.gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 23.1.0.0
gem5 compiled Feb 22 2024 14:50:21
gem5 started Feb 22 2024 16:40:55
gem5 executing on moosilauke, pid 64370
command line: ./build/RISCV/gem5.debug configs/gem5-gearup/example2.py

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/arch/riscv/isa.cc:275: info: RVV enabled, VLEN = 256 bits, ELEN = 64 bits
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
src/mem/coherent_xbar.cc:140: warn: CoherentXBar system.membus has no snooping ports attached!
Beginning simulation!
src/sim/simulate.cc:199: info: Entering event queue @ 0.  Starting simulation...
```

Debugging Event-Based Program

```
#24 0x00005555570b7df5 in pybind11::detail::argument_loader<gem5::SimObject*>::call_impl<void, pybind11::cpp_function::cpp_function<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::
sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(gem5::SimObject*)#1}&, 0ul, pybind11::detail::void_type>(pybind11::cpp_function:
:cpp_function<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{l
ambda(gem5::SimObject*)#1}&, std::integer_sequence<unsigned long, 0ul>, pybind11::detail::void_type&&) && (this=0x7fffffffd030, f=...) at ext/pybind11/include/pybind11/detail/../cast.h:1443
#25 0x00005555570b709a in pybind11::detail::argument_loader<gem5::SimObject*>::call<void, pybind11::detail::void_type, pybind11::cpp_function::cpp_function<void, gem5::SimObject, , pybind11::name, pybind1
1::is_method, pybind11::sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(gem5::SimObject*)#1}&>(pybind11::cpp_function::cpp_funct
ion<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(gem5
::SimObject*)#1}&) && (this=0x7fffffffd030, f=...) at ext/pybind11/include/pybind11/detail/../cast.h:1417
#26 0x00005555570b5d69 in pybind11::cpp_function::initialize<pybind11::cpp_function::initialize<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)()
, pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(gem5::SimObject*)#1}, void, gem5::SimObject*, pybind11::name, pybind11::is_method, pybind11::sibling>(pybind11::cpp_
function::initialize<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling cons
t&)::{lambda(gem5::SimObject*)#1}&&, void (*)(gem5::SimObject*), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(pybind11::detail::function_call&)#3}::operator()(pybi
nd11::detail::function_call&) const (__closure=0x0, call=...) at ext/pybind11/include/pybind11/pybind11.h:248
#27 0x00005555570b5e73 in pybind11::cpp_function::initialize<pybind11::cpp_function::initialize<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)()
, pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(gem5::SimObject*)#1}, void, gem5::SimObject*, pybind11::name, pybind11::is_method, pybind11::sibling>(pybind11::cpp_
function::initialize<void, gem5::SimObject, , pybind11::name, pybind11::is_method, pybind11::sibling>(void (gem5::SimObject::*)(), pybind11::name const&, pybind11::is_method const&, pybind11::sibling cons
t&)::{lambda(gem5::SimObject*)#1}&&, void (*)(gem5::SimObject*), pybind11::name const&, pybind11::is_method const&, pybind11::sibling const&)::{lambda(pybind11::detail::function_call&)#3}::_FUN(pybind11::
detail::function_call&) () at ext/pybind11/include/pybind11/pybind11.h:223
--Type <RET> for more, q to quit, c to continue without paging--
#28 0x0000555556ad9667 in pybind11::cpp_function::dispatcher (self=0x7ffff5b49ec0, args_in=0x7ffff5326350, kwargs_in=0x0) at ext/pybind11/include/pybind11/pybind11.h:939
#29 0x00007ffff7928023 in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#30 0x00007ffff78dfadc in _PyObject_MakeTpCall () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#31 0x00007ffff78e241a in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#32 0x00007ffff787b9c8 in _PyEval_EvalFrameDefault () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#33 0x00007ffff79c23af in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#34 0x00007ffff787b9c8 in _PyEval_EvalFrameDefault () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#35 0x00007ffff79c23af in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#36 0x00007ffff79bd3de in PyEval_EvalCode () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#37 0x00007ffff79bd96d in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#38 0x00007ffff79287b3 in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#39 0x00007ffff787863e in _PyEval_EvalFrameDefault () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#40 0x00007ffff79c23af in ?? () from /lib/x86_64-linux-gnu/libpython3.10.so.1.0
#41 0x0000555556ae2dd8 in pybind11::detail::simple_collector<(pybind11::return_value_policy)1>::call (this=0x7fffffffdd30, ptr=0x7ffff5360ee0) at ext/pybind11/include/pybind11/detail/../cast.h:1465
#42 0x00005555557040cc7 in pybind11::detail::object_api<pybind11::detail::accessor<pybind11::detail::accessor_policies::str_attr> >::operator()<(pybind11::return_value_policy)1> (this=0x7fffffffdde0)
    at ext/pybind11/include/pybind11/detail/../cast.h:1635
#43 0x00005555557161b11 in main (argc=2, argv=0x7fffffffe0b8) at src/sim/main.cc:87
(gdb)
```

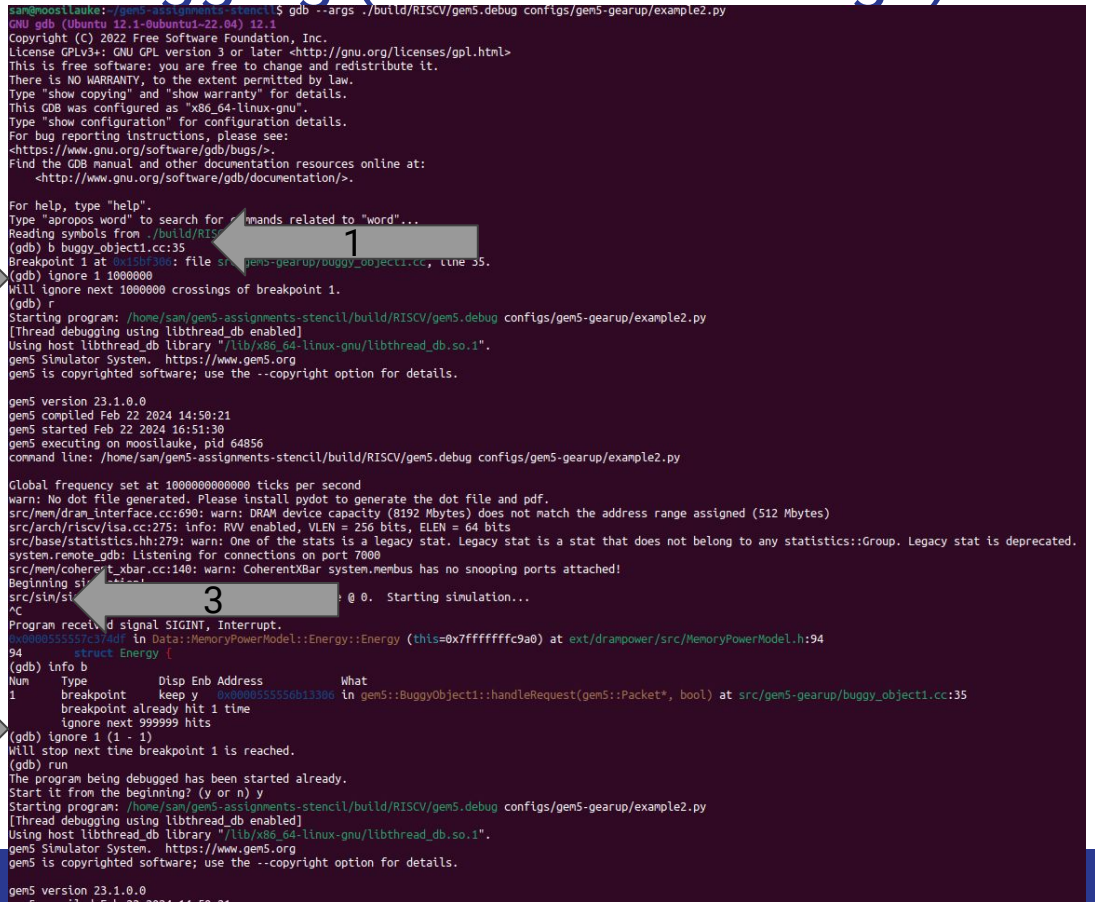
Debugging (Incorrect outputs)

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

gem5 version 23.1.0.0
gem5 compiled Feb 22 2024 14:50:21
gem5 started Feb 22 2024 16:41:12
gem5 executing on moosilauke, pid 64470
command line: ./build/RISCV/gem5.debug configs/gem5-gearup/example3.py

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/arch/riscv/isa.cc:275: info: RVV enabled, VLEN = 256 bits, ELEN = 64 bits
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
src/mem/coherent_xbar.cc:140: warn: CoherentXBar system.membus has no snooping ports attached!
Beginning simulation!
src/sim/simulate.cc:199: info: Entering event queue @ 0.  Starting simulation...
src/sim/syscall_emul.hh:1014: warn: readlink() called on '/proc/self/exe' may yield unexpected results in various settings.
      Returning '/home/sam/gem5-assignments-stencil/tests/test-progs/hello/bin/riscv/linux/hello'
src/sim/mem_state.cc:448: info: Increasing stack size by one page.
gem5 has encountered a segmentation fault!

--- BEGIN LIBC BACKTRACE ---
./build/RISCV/gem5.debug(_ZN4gem515print_backtraceEv+0x32)[0x55c65ce32b5d]
./build/RISCV/gem5.debug(+0x1c0d47a)[0x55ce5b47a]
/lib/x86_64-linux-gnu/libc.so.6(+0x42520)[0x7f97f0442520]
./build/RISCV/gem5.debug(_ZN4gem512BuggyObject21CpuSidePort13recvTimingReqEPNS_6PacketE+0x3f)[0x55c65c80e537]
./build/RISCV/gem5.debug(_ZN4gem521TimingRequestProtocol7sendReqEPNS_22TimingResponseProtocolEPNS_6PacketE+0x6a)[0x55c65dc2b9ee]
./build/RISCV/gem5.debug(_ZN4gem511RequestPort13sendTimingReqEPNS_6PacketE+0x5f)[0x55c65c8094c1]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU9sendFetchERKSt10shared_ptrINS_9FaultBaseEERKS1_INS_7RequestEEPNS_13ThreadContextE+0x31a)[0x55c65e1fedca]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU16FetchTranslation6finishERKSt10shared_ptrINS_9FaultBaseEERKS2_INS_7RequestEEPNS_13ThreadContextENS_7BaseMMU4ModeE+0x3c)[0x55c65e204796]
./build/RISCV/gem5.debug(_ZN4gem58RiscvISA3TLB15translateTimingERKSt10shared_ptrINS_7RequestEEPNS_13ThreadContextEPNS_7BaseMMU11TranslationENS9_4ModeE+0xc1)[0x55c65cfe9a1f]
./build/RISCV/gem5.debug(_ZN4gem57BaseMMU15translateTimingERKSt10shared_ptrINS_7RequestEEPNS_13ThreadContextEPNS0_11TranslationENS0_4ModeE+0x53)[0x55c65d800d2b]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU5fetchEv+0x422)[0x55c65e1fe986]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU11advanceInstERKSt10shared_ptrINS_9FaultBaseEE+0x637)[0x55c65e1ff677]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU18completeDataAccessEPNS_6PacketE+0xb6b)[0x55c65e20148b]
./build/RISCV/gem5.debug(_ZN4gem515TimingSimpleCPU10DcachePort10DTickEvent7processEv+0x2b)[0x55c65e201d4d]
./build/RISCV/gem5.debug(_ZN4gem510EventQueue10serviceOneEv+0x121)[0x55c65ce48563]
./build/RISCV/gem5.debug(_ZN4gem59doSimLoopEPNS_10EventQueueE+0x240)[0x55c65ce7849d]
./build/RISCV/gem5.debug(_ZN4gem58simulateEm+0x23e)[0x55c65ce77f34]
./build/RISCV/gem5.debug(+0x1b3978a)[0x55c65cd8778a]
./build/RISCV/gem5.debug(+0x1b37742)[0x55c65cd85742]
./build/RISCV/gem5.debug(+0x1b332bf)[0x55c65cd812bf]
./build/RISCV/gem5.debug(+0x1b3332a)[0x55c65cd8132a]
./build/RISCV/gem5.debug(+0x1585667)[0x55c65c7d3667]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(+0x128023)[0x7f97f0d28023]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(_PyObject_Call+0x5c)[0x7f97f0ce1fec]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(_PyEval_EvalFrameDefault+0x4b16)[0x7f97f0c76776]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(+0x1c23af)[0x7f97f0dc23af]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(_PyEval_EvalFrameDefault+0x9d68)[0x7f97f0c7b9c8]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(+0x1c23af)[0x7f97f0dc23af]
/lib/x86_64-linux-gnu/libpython3.10.so.1.0(PyEval_EvalCode+0xbe)[0x7f97f0dbd3de]
```

Debugging (crash)

```
sam@moosilauke:~/gem5-assignments-stencil$ gdb --args ./build/RISCV/gem5.debug configs/gem5-gearup/example3.py
GNU gdb (Ubuntu 12.1-0ubuntu1~22.04) 12.1
Copyright (C) 2022 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./build/RISCV/gem5.debug...
(gdb) run
Starting program: /home/sam/gem5-assignments-stencil/build/RISCV/gem5.debug configs/gem5-gearup/example3.py
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
gem5 Simulator System.  https://www.gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 version 23.1.0.0
gem5 compiled Feb 22 2024 14:50:21
gem5 started Feb 22 2024 16:46:40
gem5 executing on moosilauke, pid 64666
command line: /home/sam/gem5-assignments-stencil/build/RISCV/gem5.debug configs/gem5-gearup/example3.py

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/arch/riscv/isa.cc:275: info: RVV enabled, VLEN = 256 bits, ELEN = 64 bits
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
src/mem/coherent_xbar.cc:140: warn: CoherentXBar system.membus has no snooping ports attached!
Beginning simulation!
src/sim/simulate.cc:199: info: Entering event queue @ 0.  Starting simulation...
src/sim/syscall_emul.hh:1014: warn: readlink() called on '/proc/self/exe' may yield unexpected results in various settings.
      Returning '/home/sam/gem5-assignments-stencil/tests/test-progs/hello/bin/riscv/linux/hello'
src/sim/mem_state.cc:448: info: Increasing stack size by one page.

Program received signal SIGSEGV, Segmentation fault.
0x0000555556b14537 in gem5::BuggyObject2::CpuSidePort::recvTimingReq (this=0x55555a4c2a40, pkt=0x55555a4c5ba0) at src/gem5-gearup/buggy_object2.hh:82
82                      *p = 0xff;
(gdb)
```

Debugging (simulation hangs)



1. Set a breakpoint at the beginning of receiving a packet
2. Ignore the breakpoint *n* times for a big *n*
3. Wait until you think the program should have finished or is hanging, then user interrupt
4. See how many times the breakpoint has been hit (*t*), and ignore it *t - 1* times before running again (so you hit the last iteration through)