AI+X: Report 1

Hanxi Lin

September 4, 2025

Task 1

Run the test command, we have the following output in `network_stats.txt`:

```
packets_injected = 5                        (Unspecified)
packets_received = 2                        (Unspecified)
average_packet_queueing_latency = 1000                    (Unspecified)
average_packet_network_latency = 3000                    (Unspecified)
average_packet_latency = 4000                        (Unspecified)
average_hops = 1.500000                          (Unspecified)
```

After changing the global frequency to `2GHz`, , we have the following output in `network_stats.txt`:

```
packets_injected = 3165                          (Unspecified)
packets_received = 3162                          (Unspecified)
average_packet_queueing_latency = 2                          (Unspecified)
average_packet_network_latency = 13.557559                          (Unspecified)
average_packet_latency = 15.557559                          (Unspecified)
average_hops = 5.269450                          (Unspecified)
```

Task 2

By modifying the bash command to

```
#! /bin/bash

./build/NULL/gem5.opt \
configs/example/garnet_synth_traffic.py \
--network=garnet --num-cpus=64 --num-dirs=64 \
--topology=Mesh_XY --mesh-rows=8 \
--inj-vnet=0 --synthetic=uniform_random \
--sim-cycles=10000 --injectionrate=0.01

echo > network_stats.txt
grep "packets_injected::total" m5out/stats.txt | sed 's/system.ruby.network.packets_injected
    ::total\s*/packets_injected = /' >> network_stats.txt
grep "packets_received::total" m5out/stats.txt | sed 's/system.ruby.network.packets_received
    ::total\s*/packets_received = /' >> network_stats.txt
```

```
grep "average_packet_queueing_latency" m5out/stats.txt | sed 's/system.ruby.network.
    average_packet_queueing_latency\s*/average_packet_queueing_latency = /' >> network_stats
    .txt
grep "average_packet_network_latency" m5out/stats.txt | sed 's/system.ruby.network.
    average_packet_network_latency\s*/average_packet_network_latency = /' >> network_stats.
    txt
grep "average_packet_latency" m5out/stats.txt | sed 's/system.ruby.network.
    average_packet_latency\s*/average_packet_latency = /' >> network_stats.txt
grep "average_hops" m5out/stats.txt | sed 's/system.ruby.network.average_hops\s*/
    average_hops = /' >> network_stats.txt
```

We added the `reception_rate` metric to `network_stats.txt`. And by adding `.unit()` in `GarnetNetwork::regStats()`, we completed the units of statistics. In addition, there is an error in `GarnetSyntheticTraffic.cc`: the phrase `curTick() >= simCycles` should be `curCycle() >= simCycles` The output in `network_stats.txt` is as follows:

```
packets_injected = 6296                          (Count)
packets_received = 6290                          (Count)
reception_rate = .009828                          (Packet/(Node*Cycle))
average_packet_queueing_latency = 2                        (Tick)
average_packet_network_latency = 13.558188                        (Tick)
average_packet_latency = 15.558188                       (Tick)
average_hops = 5.269952                          (Count)
```

Task 3

1.

- `-h, --help`: show help message and exit.
- `-n NUM_CPUS, --num-cpus NUM_CPUS`: Number of CPUs.
- `--sys-voltage SYS_VOLTAGE`: Top-level voltage for blocks running at system power supply
- `--sys-clock SYS_CLOCK`: Top-level clock for blocks running at system speed.
- `--list-mem-types`: List available memory types.
- `--mem-type {CfiMemory,DDR3_1600_8x8,DDR3_2133_8x8,DDR4_2400_16x4,DDR4_2400_4x16,DDR4_2400_8x8,DDR5_4400_4x8,DDR5_6400_4x8,DDR5_8400_4x8,DRAMInterface,GDDR5_4000_2x32,HBM_1000_4H_1x128,HBM_1000_4H_1x64,HBM_2000_4H_1x64,HMC_2500_1x32,LPDDR2_S4_1066_1x32,LPDDR3_1600_1x32,LPDDR5_5500_1x16_8B_BL32,LPDDR5_5500_1x16_BG_BL16,LPDDR5_5500_1x16_BG_BL32,LPDDR5_6400_1x16_8B_BL32,LPDDR5_6400_1x16_BG_BL16,LPDDR5_6400_1x16_BG_BL32,NVMInterface,NVM_2400_1x64,QoSMemSinkInterface,SimpleMemory,WideIO_200_1x128}`

  type of memory to use
- `--mem-channels MEM_CHANNELS`: number of memory channels
- `--mem-ranks MEM_RANKS`: number of memory ranks per channel

- `--mem-size MEM_SIZE`: Specify the physical memory size (single memory)
- `--enable-dram-powerdown`: Enable low-power states in DRAMInterface
- `--mem-channels-intlv MEM_CHANNELS_INTLV`: Memory channels interleave
- `--memchecker`
- `--external-memory-system EXTERNAL_MEMORY_SYSTEM`: use external ports of this `port_type` for caches
- `--tlm-memory TLM_MEMORY`: use external port for SystemC TLM cosimulation
- `--caches`
- `--l2cache`
- `--num-dirs NUM_DIRS`
- `--num-l2caches NUM_L2CACHES`
- `--num-l3caches NUM_L3CACHES`
- `--l1d_size L1D_SIZE`
- `--l1i_size L1I_SIZE`
- `--l2_size L2_SIZE`
- `--l3_size L3_SIZE`
- `--l1d_assoc L1D_ASSOC`
- `--l1i_assoc L1I_ASSOC`
- `--l2_assoc L2_ASSOC`
- `--l3_assoc L3_ASSOC`
- `--cacheline_size CACHELINE_SIZE`
- `--ruby`
- `-m TICKS, --abs-max-tick TICKS`: Run to absolute simulated tick specified including ticks from a restored checkpoint
- `--rel-max-tick TICKS`: Simulate for specified number of ticks relative to the simulation start tick (e.g. if restoring a checkpoint)
- `--maxtime MAXTIME`: Run to the specified absolute simulated time in seconds
- `-P PARAM, --param PARAM`: Set a `SimObject parameter` relative to the root node. An extended Python multi range slicing syntax can be used for arrays.

- `--synthetic {uniform_random,tornado,bit_complement,bit_reverse,bit_rotation,neighbor,shuffle,transpose}`
- `-i I, --injectionrate I`: Injection rate in packets per cycle per node. Takes decimal value between 0 to 1 (eg. 0.225). Number of digits after 0 depends upon `--precision`.
- `--precision PRECISION`: Number of digits of precision after decimal point for injection rate
- `--sim-cycles SIM_CYCLES`: Number of simulation cycles
- `--num-packets-max NUM_PACKETS_MAX`: Stop injecting after –num-packets-max. Set to -1 to disable.
- `--single-sender-id SINGLE_SENDER_ID`: Only inject from this sender. Set to -1 to disable.
- `--single-dest-id SINGLE_DEST_ID`: Only send to this destination. Set to -1 to disable.
- `--inj-vnet {-1,0,1,2}`: Only inject in this vnet (0, 1 or 2). 0 and 1 are 1-flit, 2 is 5-flit. Set to -1 to inject randomly in all vnets.
- `--ruby-clock RUBY_CLOCK`: Clock for blocks running at Ruby system's speed
- `--access-backing-store`: Should ruby maintain a second copy of memory
- `--ports PORTS`: used of transitions per cycle which is a proxy for the number of ports.
- `--numa-high-bit NUMA_HIGH_BIT`: high order address bit to use for numa mapping. 0 = highest bit, not specified = lowest bit
- `--interleaving-bits INTERLEAVING_BITS`: number of bits to specify interleaving in directory, memory controllers and caches. 0 = not specified
- `--xor-low-bit XOR_LOW_BIT`: hashing bit for channel selection. See MemConfig for explanation of the default parameter. If set to 0, `xor_high_bit` is also set to 0.
- `--recycle-latency RECYCLE_LATENCY`: Recycle latency for ruby controller input buffers
- `--topology TOPOLOGY`: check configs/topologies for complete set
- `--mesh-rows MESH_ROWS`: the number of rows in the mesh topology
- `--network {simple,garnet}`: 'simple'—'garnet' (garnet2.0 will be deprecated.)
- `--router-latency ROUTER_LATENCY`: number of pipeline stages in the garnet router. Has to be ¿= 1. Can be over-ridden on a per router basis in the topology file.
- `--link-latency LINK_LATENCY`: latency of each link the simple/garnet networks. Has to be ¿= 1. Can be over-ridden on a per link basis in the topology file.
- `--link-width-bits LINK_WIDTH_BITS`: width in bits for all links inside garnet.
- `--vcs-per-vnet VCS_PER_VNET`: number of virtual channels per virtual network inside garnet network.

- `--routing-algorithm ROUTING_ALGORITHM`: routing algorithm in network. 0: weight-based table 1: XY (for Mesh. see garnet/RoutingUnit.cc) 2: Custom (see garnet/RoutingUnit.cc)
- `--network-fault-model`: enable network fault model: see `src/mem/ruby/network/fault_model/`
- `--garnet-deadlock-threshold GARNET_DEADLOCK_THRESHOLD`: network-level deadlock threshold.
- `--simple-physical-channels`: SimpleNetwork links uses a separate physical channel for each virtual network

The default values are defined in `configs/common/Options.py` and `configs/example/garnet_synth_traffic.py`.

2. The unit of `sim-cycles` is cycles and the unit of `router_latency,link_latency` is ticks. Cycles and ticks are related by the clock period. By default, 1 clock cycle is 1ns, and 1 tick is 1ps. The `setGlobalFrequency()` function sets the tick frequency.

3. Packets per node per cycle.

4. `GarnetNetworkInterface` is defined in `src/mem/ruby/network/garnet/GarnetNetworkInterface.hh` and implemented in `src/mem/ruby/network/garnet/GarnetNetworkInterface.cc`. `Router` is defined in `src/mem/ruby/network/garnet/Router.hh` and implemented in `src/mem/ruby/network/garnet/Router.cc`.

5. The packets are generated and injected into the network in `GarnetSyntheticTraffic`, buffered in `VirtualChannel` and `GarnetNetworkInterface`. In `SwitchAllocator` and `CrossbarSwitch`, the program determines if packets can be sent downstream.