

实验名称（利用 GEM5 评测系统性能）

智能 1602 任小禹 201608010627

实验目标

利用 GEM5 仿真 x86-64 系统，测量（多线程）FFT 程序运行时间，与实际 x86-64 系统的测试结果进行比较。

实验要求

- 采用 C/C++ 编写程序，选择合适的运行时间测量方法
- 根据自己的机器配置选择合适的输入数据大小 n ，保证足够长度的运行时间
- 回答思考题，答案加入到实验报告叙述中合适位置

思考题

1. GEM5 是什么？怎么使用？
2. 利用仿真器评测系统性能的优点是什么？缺点是什么？
3. GEM5 仿真系统测得的程序性能与实际系统测得的程序性能差别大不大？可能的原因是什么？

实验内容

GEM5 的安装和使用

GEM5 的官方文档在[这里](#)，包括了安装和使用说明。

GEM5 网站也提供了 x86-64 系统 Linux 镜像下载，可以在 GEM5 的[下载](#)页面找到。

利用 GEM5 测试 FFT 程序在 x86-64 系统上的性能

在安装好 GEM5，并掌握如何使用 GEM5 运行自己的测试程序后，可以在 GEM5 上运行 FFT 程序，测试其性能。建议首先使用实验一中的单线程 FFT 程序，然后再尝试实验二中的多线程 FFT 程序，记录测试数据。

测试

测试平台

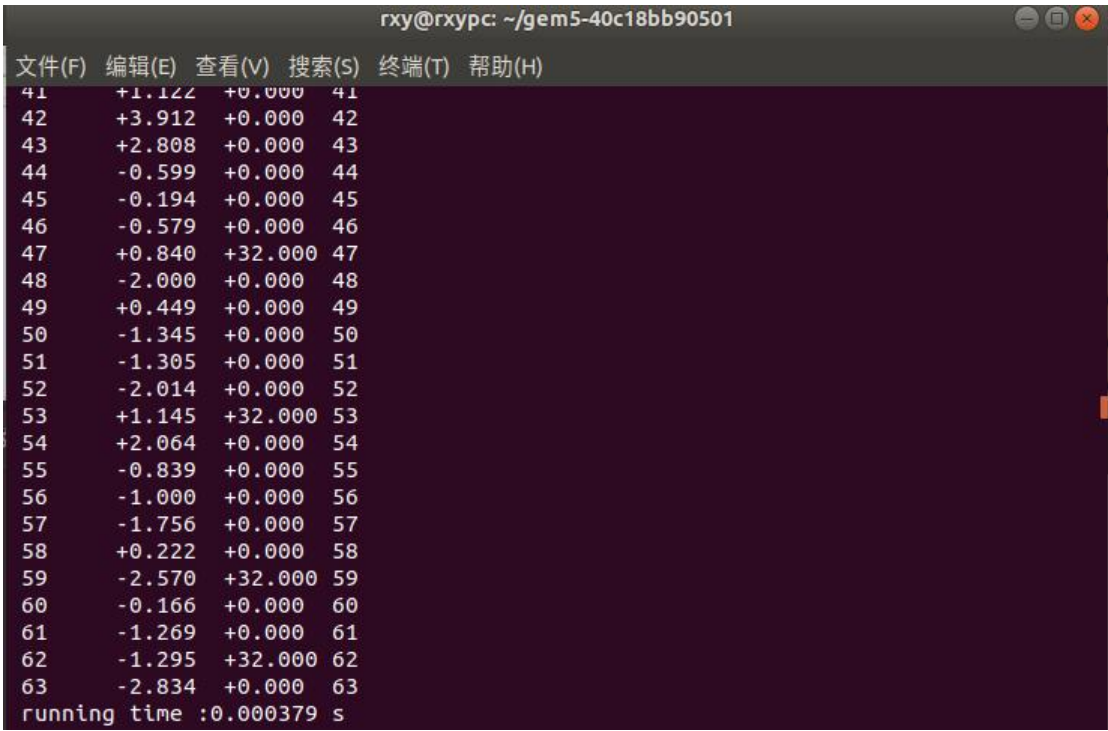
在如下机器上进行了测试：

部件	配置	备注
CPU	core i7-6500U	
内存	DDR4 8GB	
操作系统	Ubuntu 18.04 LTS	

测试记录

测试单线程 FFT 程序：

1) 在实际系统上的测试：



The screenshot shows a terminal window titled 'rxy@rxypc: ~/gem5-40c18bb90501'. The window contains a menu bar with '文件(F)', '编辑(E)', '查看(V)', '搜索(S)', '终端(T)', and '帮助(H)'. Below the menu bar, there is a list of 63 rows of data. Each row contains four columns of numbers: a line number (41 to 63), a value (e.g., +1.122, +3.912, +2.808, -0.599, -0.194, -0.579, +0.840, -2.000, +0.449, -1.345, -1.305, -2.014, +1.145, +2.064, -0.839, -1.000, -1.756, +0.222, -2.570, -0.166, -1.269, -1.295, -2.834), a value (+0.000, +0.000, +0.000, +0.000, +0.000, +0.000, +32.000, +0.000, +0.000, +0.000, +0.000, +0.000, +32.000, +0.000, +0.000, +0.000, +0.000, +0.000, +32.000, +0.000, +0.000, +32.000, +0.000), and a line number (41 to 63). At the bottom of the window, it says 'running time :0.000379 s'.

```
rxy@rxypc: ~/gem5-40c18bb90501
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
41      +1.122  +0.000  41
42      +3.912  +0.000  42
43      +2.808  +0.000  43
44      -0.599  +0.000  44
45      -0.194  +0.000  45
46      -0.579  +0.000  46
47      +0.840  +32.000  47
48      -2.000  +0.000  48
49      +0.449  +0.000  49
50      -1.345  +0.000  50
51      -1.305  +0.000  51
52      -2.014  +0.000  52
53      +1.145  +32.000  53
54      +2.064  +0.000  54
55      -0.839  +0.000  55
56      -1.000  +0.000  56
57      -1.756  +0.000  57
58      +0.222  +0.000  58
59      -2.570  +32.000  59
60      -0.166  +0.000  60
61      -1.269  +0.000  61
62      -1.295  +32.000  62
63      -2.834  +0.000  63
running time :0.000379 s
```

2) 在 GEM5 上进行系统调用模拟（SE 模式）测试：

```
rx@rxypc: ~/gem5-40c18bb90501
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
44      -0.599  +0.000  44
45      -0.194  +0.000  45
46      -0.579  +0.000  46
47      +0.840  +32.000  47
48      -2.000  +0.000  48
49      +0.449  +0.000  49
50      -1.345  +0.000  50
51      -1.305  +0.000  51
52      -2.014  +0.000  52
53      +1.145  +32.000  53
54      +2.064  +0.000  54
55      -0.839  +0.000  55
56      -1.000  +0.000  56
57      -1.756  +0.000  57
58      +0.222  +0.000  58
59      -2.570  +32.000  59
60      -0.166  +0.000  60
61      -1.269  +0.000  61
62      -1.295  +32.000  62
63      -2.834  +0.000  63
running time :0.000728 s
Exiting @ tick 23830554000 because exiting with last active thread context
```

3) 在 GEM5 上进行全系统模拟 (FS 模式) 测试:

进入模拟的系统:

```
rx@rxypc: ~/gem5-40c18bb90501
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
rx@rxypc:~/gem5-40c18bb90501$ sudo mount -o,loop,offset=32256 fs-image/disks/linux-x86.img /mnt
[sudo] rx 的密码:
rx@rxypc:~/gem5-40c18bb90501$ ls /mnt
bin  dev  fft      home  lib32  lost+found  opt  root  sys  usr
boot etc  fft.cpp  lib   lib64  mnt         proc sbin  var
rx@rxypc:~/gem5-40c18bb90501$ sudo cp mountfile/fft.cpp /mnt
rx@rxypc:~/gem5-40c18bb90501$ ls /mnt
bin  dev  fft      home  lib32  lost+found  opt  root  sys  usr
boot etc  fft.cpp  lib   lib64  mnt         proc sbin  var
rx@rxypc:~/gem5-40c18bb90501$ sudo umount /mnt
rx@rxypc:~/gem5-40c18bb90501$ sudo build/X86/gem5.opt configs/example/fs.py
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Dec 17 2018 01:06:06
gem5 started Dec 17 2018 13:08:26
gem5 executing on rxypc, pid 20400
command line: build/X86/gem5.opt configs/example/fs.py

Global frequency set at 100000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
info: kernel located at: /home/rx/gem5-40c18bb90501/fs-image/binaries/x86_64-vmlinux-2.6.22.9
Listening for com_1 connection on port 3456
0: rtc: Real-time clock set to Sun Jan 1 00:00:00 2012
0: system.remote_gdb: listening for remote gdb on port 7000
warn: Reading current count from inactive timer.
**** REAL SIMULATION ****
info: Entering event queue @ 0. Starting simulation...
warn: Don't know what interrupt to clear for console.
warn: x86 cpuid: unknown family 0x8086
92917135500: system.pc.com_1.device: attach terminal 0
warn: Tried to clear PCI interrupt 14
warn: Unknown mouse command 0xe1.
warn: instruction 'wbinvd' unimplemented
warn: instruction 'prefetch_nta' unimplemented
warn: instruction 'prefetch_nta' unimplemented
```

因为静态链接之后的运行执行文件会一直报错: FATAL: kernel too old, 所以采用直接在 GEM5 上编译并运行:

```
rxypc@rxypc: ~/gem5-40c18bb90501/util/term
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
VFS: Mounted root (ext2 filesystem).
Freeing unused kernel memory: 232k freed
INIT: version 2.86 booting
mounting filesystems...
loading script...
Script from M5 readfile is empty, starting bash shell...
(none) / # g++ fft.cpp -o fft
(none) / # serial8250: too much work for irq4
serial8250: too much work for irq4
./fft
bash: ♦♦♦♦./fft: No such file or directory
(none) / # ./fft
n      x[]      X[]      f
0      +0.000  +0.000  0
1      +2.834  +0.000  1
2      +1.295  +32.000  2
3      +1.269  +0.000  3
4      +0.166  +0.000  4
5      +2.570  +32.000  5
6      -0.222  +0.000  6
7      +1.756  +0.000  7
8      +1.000  +0.000  8
9      +0.839  +0.000  9
10     -2.064  +0.000  10
```

```
rxypc@rxypc: ~/gem5-40c18bb90501/util/term
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
42     +3.912  +0.000  42
43     +2.808  +0.000  43
44     -0.599  +0.000  44
45     -0.194  +0.000  45
46     -0.579  +0.000  46
47     +0.840  +32.000  47
48     -2.000  +0.000  48
49     +0.449  +0.000  49
50     -1.345  +0.000  50
51     -1.305  +0.000  51
52     -2.014  +0.000  52
53     +1.145  +32.000  53
54     +2.064  +0.000  54
55     -0.839  +0.000  55
56     -1.000  +0.000  56
57     -1.756  +0.000  57
58     +0.222  +0.000  58
59     -2.570  +32.000  59
60     -0.166  +0.000  60
61     -1.269  +0.000  61
62     -1.295  +32.000  62
63     -2.834  +0.000  63
running time :0.001667 s
```

分析和结论

从测试记录来看，FFT 程序在实际系统上测得的执行时间为 0.000379s，在 GEM5 上的全系统模拟执行时间是 0.001667s，是实际系统上测得的执行时间的 5 倍，系统调用模拟执行时间为 0.00728s，是实际系统上测得的执行时间的 2 倍。

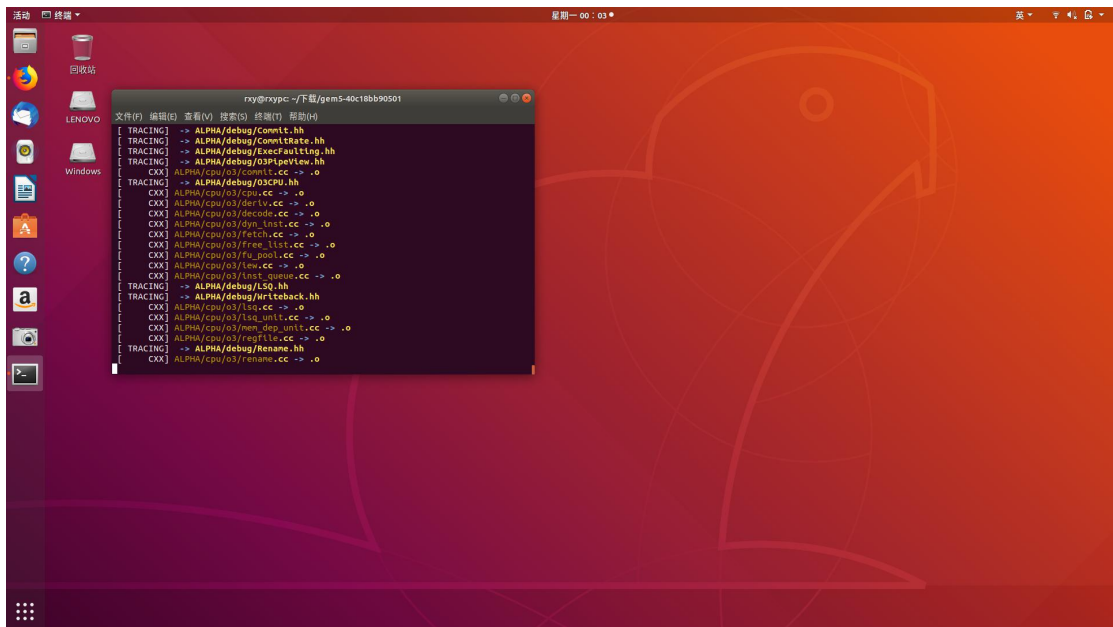
思考题

1. GEM5 是什么？如何使用？

1) gem5 模拟器提供了四个不同的 CPU 模型,两个不同的系统模型以及两个不同的内存系统模型,并且支持多种指令集 (ARM、ALPHA、MIPS、Power、SPARC 和 x86),其中可以再 ARM、ALPHA 和 x86 三种架构上运行 Linux。gem5 的许可证是基于 BSD 的,这就为工业界和学术界的合作搭建了一个好的桥梁。

2) 首先要安装 GEM5 依赖环境和 GEM5,接着配置环境,启动全仿真模拟,挂载镜像并添加文件到镜像中,卸载镜像,连接到模拟的系统,编译并运行程序。

编译架构的过程:




```
rxyc@rxypc: ~/下载/gem5-40c18bb90501
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[ SHCC] libelf/gelf_symsndx.c -> .os
[ SHCC] libelf/gelf_xlate.c -> .os
[ SHCC] libelf/libelf.c -> .os
[ SHCC] libelf/libelf_align.c -> .os
[ SHCC] libelf/libelf_allocate.c -> .os
[ SHCC] libelf/libelf_ar.c -> .os
[ SHCC] libelf/libelf_checksum.c -> .os
[ SHCC] libelf/libelf_data.c -> .os
[ SHCC] libelf/libelf_ehdr.c -> .os
[ SHCC] libelf/libelf_extended.c -> .os
[ SHCC] libelf/libelf_phdr.c -> .os
[ SHCC] libelf/libelf_shdr.c -> .os
[ SHCC] libelf/libelf_xlate.c -> .os
[ M4] libelf/elf_types.m4, libelf_convert.m4 -> libelf_convert.c
[ SHCC] libelf/libelf_convert.c -> .os
[ M4] libelf/elf_types.m4, libelf_fsize.m4 -> libelf_fsize.c
[ SHCC] libelf/libelf_fsize.c -> .os
[ M4] libelf/elf_types.m4, libelf_msize.m4 -> libelf_msize.c
[ SHCC] libelf/libelf_msize.c -> .os
[ AR] -> libelf/libelf.a
[ RANLIB] -> libelf/libelf.a
[ LINK] -> X86/gem5.opt
scons: done building targets.
rxyc@rxypc:~/下载/gem5-40c18bb90501$
```

挂载镜像、拷贝文件到挂载的镜像中，启动全系统模拟：

```
rxyc@rxypc: ~/gem5-40c18bb90501
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
rxyc@rxypc:~/gem5-40c18bb90501$ sudo mount -o,loop,offset=32256 fs-image/disks/linux-x86.img /mnt
[sudo] rxy 的密码:
rxyc@rxypc:~/gem5-40c18bb90501$ ls /mnt
bin  dev  fft  home  lib32  lost+found  opt  root  sys  usr
boot  etc  fft.cpp  lib  lib64  mnt  proc /sbin  tmp  var
rxyc@rxypc:~/gem5-40c18bb90501$ sudo cp mountfile/fft.cpp /mnt
rxyc@rxypc:~/gem5-40c18bb90501$ ls /mnt
bin  dev  fft  home  lib32  lost+found  opt  root  sys  usr
boot  etc  fft.cpp  lib  lib64  mnt  proc /sbin  tmp  var
rxyc@rxypc:~/gem5-40c18bb90501$ sudo umount /mnt
rxyc@rxypc:~/gem5-40c18bb90501$ sudo build/X86/gem5.opt configs/example/fs.py
gem5 Simulator System. http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Dec 17 2018 01:06:06
gem5 started Dec 17 2018 13:08:26
gem5 executing on rxypc, pid 20400
command line: build/X86/gem5.opt configs/example/fs.py

Global frequency set at 1000000000000 ticks per second
warn: DRAM device capacity (8192 Mbytes) does not match the address range assigned (512 Mbytes)
info: kernel located at: /home/rxy/gem5-40c18bb90501/fs-image/binaries/x86_64-vmlinux-2.6.22.9
Listening for com_1 connection on port 3456
0: rtc: Real-time clock set to Sun Jan 1 00:00:00 2012
0: system.remote_gdb: listening for remote gdb on port 7000
warn: Reading current count from inactive timer.
**** REAL SIMULATION ****
info: Entering event queue @ 0. Starting simulation...
warn: Don't know what interrupt to clear for console.
warn: x86 cpuid: unknown family 0x8086
92917135500: system.pc.com_1.device: attach terminal 0
warn: Tried to clear PCI interrupt 14
warn: Unknown mouse command 0xe1.
warn: instruction 'wbinvd' unimplemented
warn: instruction 'prefetch_nta' unimplemented
warn: instruction 'prefetch_nta' unimplemented
```

连接模拟好的系统：

```
rxxy@rxypc: ~/gem5-40c18bb90501/util/term
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
rxxy@rxypc:~/gem5-40c18bb90501$ cd util/term
rxxy@rxypc:~/gem5-40c18bb90501/util/term$ sudo ./m5term 127.0.0.1 3456
[sudo] rxxy 的密码:
==== m5 slave terminal: Terminal 0 ====
Linux version 2.6.22.9 (blackga@nacho) (gcc version 4.1.2 (Gentoo 4.1.2)) #2 Mon
Oct 8 13:13:00 PDT 2007
Command line: earlyprintk=ttyS0 console=ttyS0 lpj=7999923 root=/dev/hda1
BIOS-provided physical RAM map:
  BIOS-e820: 0000000000000000 - 0000000000009fc00 (usable)
  BIOS-e820: 0000000000009fc00 - 0000000000100000 (reserved)
  BIOS-e820: 0000000000100000 - 00000000020000000 (usable)
  BIOS-e820: 00000000020000000 - 000000000c0000000 (reserved)
  BIOS-e820: 00000000ffff0000 - 0000000100000000 (reserved)
end_pfn_map = 1048576
kernel direct mapping tables up to 100000000 @ 8000-d000
DMI 2.5 present.
Zone PFN ranges:
  DMA      0 ->      4096
  DMA32    4096 ->   1048576
  Normal   1048576 -> 1048576
early_node_map[2] active PFN ranges
  0:      0 ->     159
  0:    256 ->   131072
Intel MultiProcessor Specification v1.4
```

2. 利用仿真器评测系统性能的优点是什么？缺点是什么？

优点：

- 1) 可以完全替代目标系统，完成其的功能。对于 GEM5 的全系统模拟，分析研究目标系统本身，其要跟 CPU 所有内部行为基本一致。
- 2) 具有风险低，成本小、灵活性高的特点。

缺点：

- 1) 仿真器运行速度相比真正的系统差很多。
- 2) 仿真器其内部模拟的 CPU 各个模块也不是完全一样，和真实系统不能保证完全一致。

3. GEM5 仿真系统测得的程序性能与实际系统测得的程序性能差别大不大？可能的原因是什么？

从测试记录来看，FFT 程序在实际系统上测得的执行时间为 0.000379s，在 GEM5 上的全系统模拟执行时间是 0.001667s，是实际系统上测得的执行时间的 5 倍。可能的原因是一条指令的编译可能需要多条指令来解释，所以时间比真实系统长很多。