

2018 计算机系统设计实验报告



湖南大学
HUNAN UNIVERSITY

实验名称：_____ 利用 GEM5 评测系统性能 _____

姓 名：_____ 孟祥炜 _____

学 号：_____ 201607020301 _____

专业班级：_____ 智能 1602 _____

一、实验目标

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

二、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

三、实验内容

首先看一下官方对 gem5 平台的定义，如下

gem5 是一个模块化离散事件驱动的计算机系统模拟器平台。这意味着：

gem5 的组件可以轻松重新排列，参数化，扩展或替换，以满足您的需求。它模拟了一系列离散事件的时间流逝。

其预期用途是以各种方式模拟一个或多个计算机系统。

它不仅仅是一个模拟器;它是一个模拟器平台，可以让您使用尽可能多的预制组件来构建自己的模拟系统。

gem5 主要用 C++ 和 python 编写，大多数组件都是在 BSD 风格的许可下提供的。它可以在完整系统模式（FS 模式）下模拟具有设备和操作系统的完整系统，或者在系统调用模式（SE 模式）下由模拟器直接提供系统服务的用户空间程序。在 CPU 模型上执行 Alpha，ARM，MIPS，Power，SPARC 和 64 位 x86 二进制文件有不同程度的支持，包括两个简单的单个 CPI 模型，一个乱序模型和一个按顺序流水线模型。内存系统可以灵活地由缓存和交叉开关构建。最近，Ruby 模拟器已经与 gem5 集成，以提供更灵活的内存系统建模。

这里没有提到许多组件和功能，但仅从这个部分列表中可以明显看出 gem5 是一个复杂且功能强大的仿真平台。即使今天所有 gem5 都可以做到，积极的开发仍然通过个人和一些公司的支持，并且添加了新功能并且定期改进现有功能。

根据以上对于 gem5 的介绍说明，可以看到，gem5 是一种类似于虚拟机平台。Gem5 平台使得我们可以在一台计算机上，模拟操作系统、应用程序等各类计算机程序在各种处理器架构下的运行情况。有了 gem5，我们可以模拟程序在各种不同架构的处理器下的执行情况而不需要真正的各种架构的处理器实体，能够大幅节约开发成本。

由于 gem5 问世时间很长，而且少有更新，因此现在再 gem5 上运行程序需要一些以前的依赖项，同时由于与现在的操作系统没有很好地兼容性，导致运行 gem5 时需要的时间较长，因此根据老师的说明，改为测量单线程 FFT 的运行时间。

Gem5 的安装过程

① 安装各类库文件及编译器：

```
sudo apt-get install mercurial scons swig gcc m4 python python-dev  
libgoogle-perftools-dev g++ libprotobuf-dev  
sudo apt-get install build-essential
```

注：google 的 protobuf 编译工具能够提高 gem5 的运行速度

② 下载 gem5 源码

hg clone <http://repo.gem5.org/gem5>

下载源码之后，主文件夹中就会有一个 gem5 文件夹，所有 gem5 相关的文件都在其中。

③ 编译 x86 架构

```
scons build/X86/gem5.opt
```

这一步耗时较长，大约在 40~60 分钟左右

这里我们使用 x86 架构编译 gem5，如果需要进行其他架构的处理器模拟，可以根据自己的需要选择。

支持的处理器架构有：Alpha, ARM, MIPS、Power, SPARC, X86

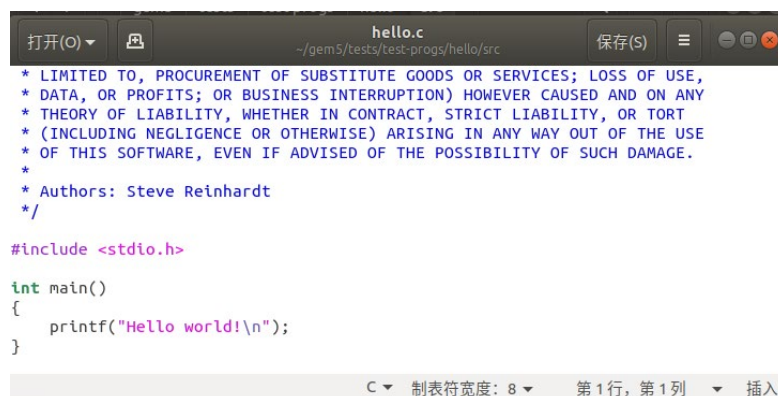
④ 运行 gem5 自带的测试程序

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

在 tests/test-progs/目录下有一些 gem5 自带的测试程序，选取其中一个测试运行，以验证 gem5 是否安装成功

该示例程序运行完之后会输出 hello,world

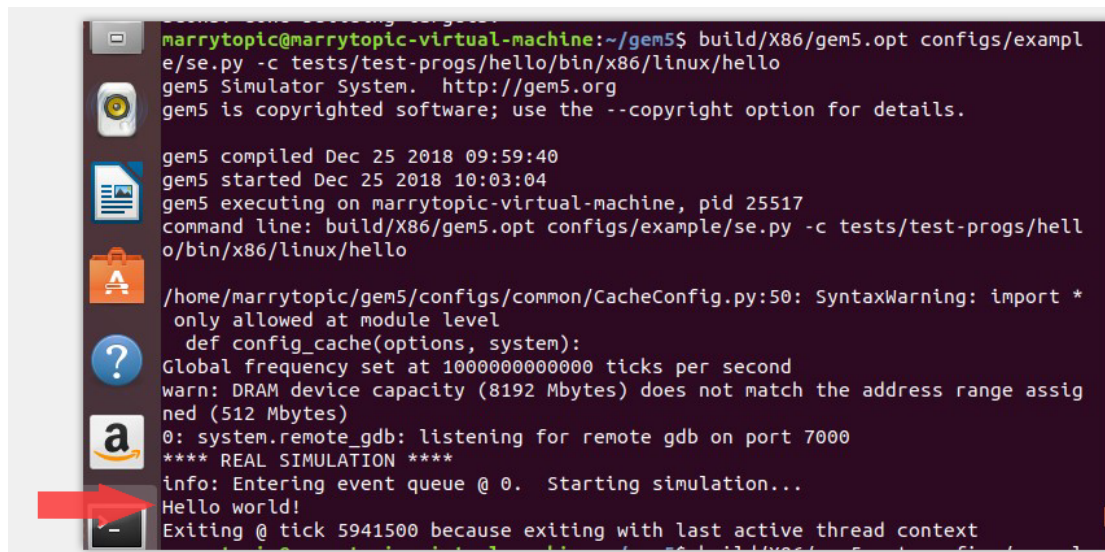
源程序：



```
hello.c  
~/gem5/tests/test-progs/hello/src  
保存(S)  菜单  窗口  退出  
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  
*  
* Authors: Steve Reinhardt  
*/  
  
#include <stdio.h>  
  
int main()  
{  
    printf("Hello world!\\n");  
}
```

C 制表符宽度: 8 第 1 行, 第 1 列 插入

运行结果:



```
marrytopic@marrytopic-virtual-machine:~/gem5$ build/X86/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello
gem5 Simulator System.  http://gem5.org
gem5 is copyrighted software; use the --copyright option for details.

gem5 compiled Dec 25 2018 09:59:40
gem5 started Dec 25 2018 10:03:04
gem5 executing on marrytopic-virtual-machine, pid 25517
command line: build/X86/gem5.opt configs/example/se.py -c tests/test-progs/hello/bin/x86/linux/hello

/home/marrytopic/gem5/configs/common/CacheConfig.py:50: SyntaxWarning: import * only allowed at module level
def config_cache(options, system):
Global frequency set at 1000000000000 ticks per second
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 ****
info: Entering event queue @ 0.  Starting simulation...
Hello world!
Exiting @ tick 5941500 because exiting with last active thread context
```

可知, gem5 已经安装成功

Gem5 支持两种模拟模式:

FS(Full System, 全系统模式), SE(Systemcall simulation, 系统调用模式)

由于这里我们只需要简单的进行程序运行模拟, 因此选择 SE 模式即可。

截取第一次的 FFT 实验代码, 添加测量程序运行时间的代码, 分别在 linux 系统和 gem5 系统上的运行时间进行测量

四、测试平台

CPU: i7-6500U

内存: DDR3 1.5GB

操作系统: ubuntu-18.04.5 LTS

备注: 虚拟机环境运行 (Vmware Workstation Pro 15.0.2)

五、测试记录

Gem5 运行情况

活动 终端 星期二 15:20 英 终端 关闭 退出 刷新

marrytopic@marrytopic-virtual-machine: ~/gem5

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

39	+0.205	+0.000	39
40	+1.000	+0.000	40
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

229291ns
Exiting @ tick 2864912000 because exiting with last active thread context
marrytopic@marrytopic-virtual-machine:~/gem5\$

Linux 系统上的运行情况

marrytopic@marrytopic-virtual-machine: ~/桌面

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

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

59763ns

以上为两种情况下的一次截图
Gem5 上的运行时间在 230000ns 左右
Linux 系统上的运行时间在 60000ns 左右

六、分析和结论

从测量所得的数据可以看出，程序在 gem5 上的运行时间是在 linux 系统上运行时间的 4 倍左右。

可以看到，同样的程序在 gem5 中运行时间比在 linux 系统运行时间慢。

Gem5 作为一个模拟平台，程序在模拟平台上的速度肯定会比在本机系统上的慢一些，主要原因是，程序在本机上运行时，直接调用本机系统的相关链接库等文件即可执行，而在模拟平台上运行时，需要再模拟平台上调用模拟平台的文件，而模拟平台的文件的运行需要再次调用本机的相关文件执行，因此过多的调用导致运行时间较长。同时，gem5 的存储、处理结构、IO 等都与本机上不同，且比本机慢，因此，程序在 gem5 上的运行速度比在本机上的运行速度慢

七、思考题

1. GEM5 是什么？怎么使用？

Gem5 的介绍已经在开头叙述过

下载安装 gem5 后，使用命令使用 gem5 编译运行程序，即可达到与在本机上运行程序一样的效果

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. Gem5 的出现使得研究人员可以在一台机器上实现所有处理器平台的研究，大大节约了成本。
2. 支持多种处理器架构：Alpha, ARM, MIPS、Power, SPARC, X86, 可以在各种平台下进行模拟
3. Gem5 作为一个开源项目，允许所有人为其贡献

缺点：

1. 现在还不能完美的支持所有处理器架构
2. 在模拟运行时的性能还有待提升
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？

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约 3~4 倍

Gem5 不是一个本机的完全的的操作系统，无法得到使用所有计算机资源的权限，同时，其中的模拟处理器不能与本机操作系统完全兼容，因此实际的运行中受限，导致其程序性能下降。