# 现代计算机体系架构第一次课程实验

——基于gem5仿真器的单核流水线处理器仿真

## 目标：

本次实验的目的是帮助同学们在学习了基本处理器特征的基础上，熟悉计算机体系架构仿真器gem5的安装与使用，并且分析基本流水线型处理器的性能，为之后的进一步探究打下基础。

## 实验准备：

1. 阅读前几节课的课件。
2. 阅读gem5 tutorial 课件。
3. 熟悉基本linux操作指令。
4. 在<http://www.gem5.org/> 阅读Get started, Documentation/Building, Documentation/CPU models和Documentation/Compiling Workload等部分。

## 实验内容：

1. **gem5的安装与测试：**
   1. 按照gem5 tutorial课件提供的安装教程，安装gem5。
   2. 编译arm指令集架构SE 模式下的处理器仿真程序build/ARM/gem5.debug，并采用se.py脚本执行测试程序: tests/test-progs/hello/bin/arm/linux/hello。
2. **交叉编译：**
   1. 交叉编译器的安装使用

安装ARM指令集交叉编译器gcc-aarch64-linux-gnu。将八皇后问题的c程序文件queens.c交叉编译成为在ARM架构下的可执行程序并在ARM仿真平台上运行，皇后个数为8。要求采用表1的参数，未涉及的其它参数使用默认参数。

(提示：可通过build/ARM/gem5.debug configs/exmaple/se.py --help观察参数设定方法)

表 1仿真参数

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Value | Parameter | Value |
| CPU-type | minor | l1 data cache size | 8kB |
| l1 instr. cache size | 8kB | l1 cache assoc | 4 |
| L2 cache size | 1MB | l2 cache assoc | 8 |
| Cache line size | 64 | Clock | 1GHz |
| CPU number | 1 |  |  |

* 1. 如何得到其他数量的皇后对应的解？如何得到指定皇后数对应的解的总个数？简述你的方法并展示结果。
  2. 回答：交叉编译的目的是什么？
  3. \***（选做）自行搭建交叉编译器**

下载安装面向其他架构的交叉编译器(RISCV等)重复步骤2.1，简述你采用的命令、方法并展示结果。

1. **流水线处理器的性能分析：**

gem5支持多种cpu类型，比如timing和minor等。timing型cpu每个周期计算一个指令，但是在访存阶段需要等待多个周期读取内存。第二步中使用到的minor 处理器类型是In order处理器类型到一种，详细介绍见：<https://www.gem5.org/documentation/general_docs/cpu_models/minor_cpu> , 类定义及配置文件在: src/cpu/minor/MinorCPU.py 。在阅读文档和代码到基础上，回答下列问题：

* 1. Minor流水线结构

Minorview.py是一个可以生成流水线GUI界面的工具，使用方法见附录。请按照教程生成minor流水线GUI界面，将运行截图放在下面，通过观察和参考doc文件回答：

1. Minor处理器由几级流水线构成？每一级的名称及作用是什么？请用下面的表格说明：

表 2流水线功能

|  |  |  |
| --- | --- | --- |
| Stage Number | 名称 | 功能 |
| 1 |  |  |
| 2 |  |  |
| … |  |  |

1. Excute stage具体包括哪些模块？各个模块的功能是什么？
2. Stage间的forward FIFO和backward FIFO的功能分别是什么？
   1. 处理器频率-性能特性

分别使用timing cpu和minor cpu在ARM仿真平台上运行八皇后问题，编写脚本文件改变处理器频率从1GHz到3GHz，中间取至少10个频率点，其它参数依旧采用表1中的参数，观察stats.txt文件在**一张图**中绘制出两种处理器的模拟运行时间(sim\_seconds)随频率的变化关系曲线。回答：

1. 随着频率增长，处理器运行时间变化趋势是什么样的？从1GHz->3GHz处理器速度提升了多少倍？尝试用流水线处理器理论和计算机性能评估方法解释上述现象。
2. 哪一种处理器模型的性能随频率上升提升更多？为什么？

（提示：可以参考附件sim\_freq.sh编写运行脚本）

3.3 处理器功能单元特性

在MinorCPU.py中定义了MinorFU为functional unit（功能单元）类，该类中定义了每种功能单元的延时参数opLat。假设采用新型的运算单元可以将整数运算或浮点运算延时缩减为原来的一半左右，请问针对八皇后问题应该选择优化整数运算单元还是浮点运算单元？设计实验方案并提供数据支持你的结论。

（提示：可以通过直接修改MinorCPU.py文件修改仿真参数，但是在修改后需要重新执行scons build/ARM/gem5.debug命令）

**附录： Minorview的使用**

1. 输出trace文件，执行：

build/ARM/gem5.debug --debug-flags=MinorTrace --debug-start=[StartIns] --debug-end=[EndIns] --debug-file=[filename] configs/example/se.py …

…表示省略了部分仿真参数设置命令，其中StartIns和EndIns分别是trace file记录的起始tick和结束tick序号，tick是仿真的最小时间单位，可以取有限长度的ticks减小trace文件大小。

1. 在gem5目录下执行：

python2 ./util/minorview.py ./m5out/[filename]

打开pipeline view的GUI界面，可以通过visible ids设置标记指令的颜色数量以便于观察。

注：如果无法输出请运行：

sudo apt install python2

wget <http://archive.ubuntu.com/ubuntu/pool/universe/p/pygtk/python-gtk2_2.24.0-5.1ubuntu2_amd64.deb>

sudo apt install ./python-gtk2\_2.24.0-5.1ubuntu2\_amd64.deb

根据提示安装缺少的python-is-python2、python-gtk2等包

并且，在使用完minorview之后，请运行

sudo apt install python-is-python3

以重新安装之前在安装python-gtk2时被自动卸载的程序包，以避免导致后续实验中重新编译gem5时出错。

**提交要求：**

1. pdf或word格式的实验报告，回答所有问题，不方便翻译的名词可以直接用英文，需要包括典型结果输出，可以截图或以文本形式粘贴，例如：

\*\*\*\* REAL SIMULATION \*\*\*\*

info: Entering event queue @ 0. Starting simulation...

info: Increasing stack size by one page.

Hello world!

hack: be nice to actually delete the event here

Exiting @ tick 13410500 because target called exit()

报告中也可以说明你在实验中遇到的问题以及解决方法。

1. 附上3.2中编写的运行脚本，如果有自行编写的其它脚本文件请附上并在报告中说明。
2. 将(docx或pdf格式)的报告和其他材料打包，命名为“姓名\_学号.zip”，在网络学堂上提交，请注意截止时间。