

基于 GEM5-NVP 的 DFS 系统设计与仿真

无 42 陈誉博 林子恒

Abstract

非易失处理器（NVP）是一种不同于传统处理器的新型处理器，其结构由能量采集和存储模块、电压检测模块、系统状态管理模块、非易失存储器模块以及传统 CPU 中包含的处理器、寄存器、内存、缓存等模块。动态频率选择（DFS）系统是一种能够随着工作负载和外界能量变化而改变自身频率的处理器，其意义在于避免一般非易失处理器容易出现的频繁掉电上电、关机重启、影响效率的行为导致的系统效率低的现象。在本文中，我们使用 gem5 作为仿真工具。首先，我们对 gem5 进行分析，详细分析了 gem5 中的事件（Event）和事件队列（EventQueue）以及指令延时的作用机制；之后，我们对针对非易失处理器开发的 gem5-NVP 仿真工具进行了分析，详细分析了其中能量管理模块与 CPU 的通信机制、能量管理模块中的状态机等。最后，我们利用上面分析得到的结果设计了一个 DFS 系统，并在 gem5-NVP 平台上进行实现以及针对不同能量输入情况的仿真。最终我们得出结论：DFS 系统在能量供给较不足的时候，可以有效的减少系统的开机关机次数；并且，若考虑此开机关所带来的时间代价，DFS 系统相较于一般的非易失处理器可以带来十分可观的性能优化。

File structure

```
1  code/
2  |— configs
3  |   └─ example # 仿真的代码们
4  |       |— auto_script.py
5  |       |— my_dfs.py
6  |       |— my_two_thre.py
7  |       |— traverse_duty_ratio.py
8  |       |— traverse_duty_ratio_ori.py
9  |       |— traverse_high.py
10 |       |— traverse_high_ori.py
11 |       |— traverse_low.py
12 |       └─ traverse_low_ori.py
13 |— plot # 存放了输出的结果、画图的 matlab 代码、以及输出的图片
14 |— src # 修改的 gem5 代码们
15 |   └─ cpu
16 |       |— engy
17 |       |   |— DFS.py
18 |       |   |— dfs.cc
19 |       |   └─ dfs.hh
20 |       └─ simple
21 |           |— AtomicSimpleCPU.py
22 |           |— atomic.cc
23 |           └─ atomic.hh
```

- src/cpu 里存放了需要修改的gem5代码
 - engy/* 为 DFS 需要添加的代码
 - simple/* 为为了将状态机更改为 DFS 需要修改的代码
- configs/example 里存放了仿真的代码
 - my_two_thre.py 为第4问 仿真 TwothreSM 状态机的代码
 - my_dfs.py 为第5问 仿真 DFS 的代码
 - auto_script.py 为第六问 仿真 的基础代码
 - traverse_* 等 为第六问 仿真 不同高、低电平、占空比 性能、开关机次数的代码
 - *_ori.py 为仿真原系统的代码
 - 其余为仿真 DFS 的代码
- plot/ 里存放了仿真的结果以及画图的代码、输出的图片
 - *.txt 为仿真的结果们
 - my_plot.m 为画图的代码
 - *.eps 为输出的图片