NVPsim：应用于非易失处理器架构探索的仿真器

摘要

非易失处理器（NVP）应用了非易失内存技术，在发生断电时保存运行时的信息。在能量采集系统中，这个功能使得NVP能够在间断的电源供应下进行连续的工作进展。这篇论文基于gem5建立了一个NVP仿真器，NVPsim，这个仿真器被已经流片的原型芯片的测试结果所验证，拥有良好的误差。刚进一步，为了阐述这个仿真器进行架构探索的能力，我们探究了选取不同非易失内存作为片上缓存、不同备份策略和不同能量缓冲大小的非易失处理器设计的性能和能量消耗。

介绍

能量采集技术被广泛应用于无电池设备的供电。然而，能量源往往有着先天的不确定性，这导致了采集到的功率不可避免地在有些时候低于设备的需求，造成间歇的电源中断。频繁的电源中断造成了微处理器中寄存器、易失的缓存中运行状态的丢失，导致了系统性能的下降。为了解决这一问题，非易失处理器（NVP）在最近被广泛认为是一种需要从外界进行能量采集的节点的解决方案。

非易失处理器使用片上的非易失储存器来在发生断电时保存运行线程的状态。当供电恢复，线程状态从非易失储存器中被恢复。Wang等人将易失的触发器（Flip Flop）换成了铁电的触发器，从而实现了非易失处理器。这个处理器相比“MSP430FR系列”微处理器休眠和唤醒时间分别小30倍和103倍。

非易失处理器体系结构的探索在之前的工作中被实现，Ma等人探索了多种非易失处理器的设计，这些设计使用了不同观点微结构和不同的能量源来最大化非易失处理器的运行工作进展。然而，他们主要关注点在于备份非易失处理器的寄存器的策略，而忽视了片上缓存的设计。尽管NVP指令和数据的缓存可以使用非易失储存器，高的写入所需能量和延时会因为非易失储存器的特性而被引入。非易失SRAM（nvSRAM）被提出以替换传统缓存中的SRAM，以实现高性能和低能量消耗。Tsai等人分析了比特级的写入nvSRAM写入数据量降低技术，Li等人提出了块级的能量利用率高的nvSRAM备份技术。然而，这些备份技术的有效性缺少在不同架构下的系统探究。事实上，随着核心数和内存数的提升差距变大，NVP处理器的缓存设计变得越来越重要。它为非易失处理器的设计添加了新的维度。

NVP处理器的设计探索需要一个合适的仿真工具，Ma等人验证非易失处理器设计的方式是使用Verilog，此仿真方式仿真整个体系架构速度非常慢，Dong等人设计了NVSim，以仿真不同非易失存储器的设计技术，但是没有考虑整个体系架构。Binkert等人提出了gem5，一个高配置自由度仿真框架，但是不对非易失处理器结构仿真提供支持。我们拓展了这个仿真框架，使之成为一个非易失处理器设计探索的合适仿真器。

这篇文章的目标有两层。首先，建立在现存gem5仿真框架下的一个体系结构级的非易失处理器仿真器被编写；其次，使用被提出的这个仿真器，我们进行了非易失处理器的设计探索。我们着重探索了非易失处理器片上缓存的设计，这在前人的工作中没有被提到。综上，此论文给出了如下贡献：

* 一个建立在gem5仿真框架的非易失处理器仿真器（NVPsim）被提出，经过验证，它是一个拥有高灵活性和高准确度的非易失处理器体系架构的探索工具。
* 我们使用NVPsim验证了使用不同非易失储存器来进行片上缓存、不同能量缓存的非易失处理器设计。
* 使用NVPsim，我们验证了在不同缓存类型在不同策略下的备份运行状态的效率。

论文的其余部分如下：第二部分介绍了NVPsim基于的非易失处理器的模型买第三部分介绍NVPsim的设计，第四部分验证了NVPsim的准确性，第五部分进行了NVP的设计探索。

非易失处理器建模

我们参考一个已经流片的非易失处理器建立的非易失处理器模型，这个模型的概要在图A.1中给出，且由收集到的能量进行功能。在一个基于非易失处理器的系统，收集到的能量首先由一个DC-DC转换器进行蒸馏，接下来保存到一个电容中，作为能量的缓冲来保存收集到的能量并给非易失处理器功能，在非易失处理器中，寄存器和片上缓存都使用的是非易失储存器。我们假定主内存使用阻性非易失储存器，因为主内存技术不在本文探讨范围内。

A. 电压检测器建模

当发生一次断电时，非易失处理器在关机前保存线程运行状态，当电力恢复时，非易失储存器在重新运行前恢复线程运行状态。电压检测模块会通过检测电容的电压来寻找可能的电能中断点和电能恢复点。图A.2阐述了电压检测器在时间域的行为。当电容的电压超过阈值电压时，电压检测器在时间后，时发送一个“restore”信号给备份/恢复控制器。当这个恢复信号被备份/恢复控制器接收到时，非易失处理器将会在的时间内备份线程运行信息。非易失处理器在重新开始运行。当电容电压低于阈值电压时，电压检测器会发送“backup”信号给备份/恢复控制器并在时间后在时间完成备份。

注意到和都由一些电压窗设置技术，比如MPPT来决定，因此是一定能得到满足的，是NVP操作电压的最小值，在最坏的情况下，电容电压在恢复过程中可能低于，这样的情况被认为是系统崩溃，系统将会回滚。

B. 备份/恢复控制器建模

备份/恢复控制器控制着寄存器和缓存的备份。在这个模型中，寄存器包含了非易失的触发器，寄存器中的内容如果不被备份到非易失触发器中将会丢失。当控制器接收到“backup”信号时，它将会向寄存器发送控制信号使所有的内容并行地被备份。

缓存的备份过程取决于非易失储存技术的选取，如果缓存使用的是需求是非易失储存器技术（比如nvSRAM），备份/恢复模块需要使用在章节3中提到的策略控制缓存备份过程，否则缓存中的内容将不会被备份，因为在断电时数据的完整性可以被保持。当“restore”信号到达备份/恢复控制器时，控制器将会使用和备份时相同的方式来恢复寄存器和缓存中的数据。

C． 非易失处理器状态机建模

图A.3给出了非易失处理器的状态机模型。“s1”和“s2”代表了“running”和“run-backup”状态，当电容电压小于阈值电压时，状态机将会从“running”状态转移到“run-backup”状态。当“run-backup”状态消耗的时间

大于时，状态机将会从“run-backup”状态转移到“backup”状态（“s3”），当非易失储存器在的时间里保存好线程运行数据后，状态机将从“backup”状态转移到“off”状态（“s4”）。状态机将维持在“off”状态，直到电容电压超过，这是状态机将会从转移到“off-restore”状态（“s5”），状态机在时间后将会从“off-restore”状态转移到“restore”状态（“s6”），当非易失处理器在时间完成状态的恢复后，状态机会从“restore”状态转移到“rollback”状态（“s7”）。

“rollback”状态对系统的回转过程（跳转到上一条被“backup”信号打断未完成操作）进行了建模，当状态机在“rollback”状态时，它可能根据电容中的电压转移到不同的状态。如果非易失处理器还没有完成回转时电容电压就低于，则状态机进入“run-backup”状态，如果NVP能够完成回转并且没有发生电能中断，则状态机进入“running”状态。

NVPsim仿真器设计方法

这一章主要介绍NVPsim的设计方法，我们首先给出NVPsim的概述，接下来介绍电压检测模块、备份/恢复控制器和各个模块间的公布方式。

A. 仿真器概述

图A.4中给出了NVPsim的整体设计，第二章中介绍的非易失处理器模型指导了这一设计。NVPsim是在gem5仿真框架的基础上建立的，TimingSimpleCPU模型（一个平衡了仿真准确性和仿真速度的CPU模型）被用作CPU的模型，下述几个模块被添加：i）电压检测器，ii）备份/恢复控制器。一些gem5模块，比如事件管理模块和缓存模块被修改以便支持NVPsim。

NVPsim将能量供给文件、系统参数和仿真程序作为输入。能量供给文件决定了输入的能量，NVP的一些设置被系统参数设定（比如和电容大小）。NVPsim使用Mibench基准程序系列、能量供给文件和系统设定进行仿真。NVPsim的输出是不同硬件模块的能量消耗以及一个能够帮助我们评估性能和能量效率的数据统计文件。

B. 电压检测器

给出当前电容的电压后，电压检测器会使用差分算法计算下一个状态的电压。电容中的电压由方程A.1决定。

对方程A.1的左右两端取时间的微分，得到：

使用简单的变换，方程A.2变为：

在方程A.3中，是在第i步输入的净能量，而是步间的时间间隔。等于，和是第i步的输入功率和输出功率。是第i步的电能供给，的动态计算是基于第2章中处理器的内部状态（如访问缓存）来计算的。比如，如果NVP在第i步的状态是“backup”，则为备份/恢复模块计算出的备份功率。如果NVP在第i步的状态是“running”，则输出功率被当前处理器的内部状态（如缓存访问）来决定。

C. 备份/恢复控制器设计

备份/恢复控制器计算备份线程状态所需要的能量和时间，我们认为线程状态可以被比特级地并行备份和恢复，这被现存的硬件所支持。

假定某一级内存（比如缓存）中的N个比特被并行地备份，所需的总功率为，所需时间是，和是备份一个比特所需的功率和时间，我们从NVSim的仿真结果与前人工作的文献中获得和。是整个系统的功率上限，如果超过了，这些字节将会分批次备份。令，这些比特被分成了个批次，这时，，为了备份N个比特使用的功率和时间取决于和如果，则。

令，如果，计算结果将会由决定，如果，则，如果，备份N比特所需的功率和时间将被分为2个阶段，如方程A.4所示。