

TTA 流水线 Petri 网模型

朱 勇

(武汉纺织大学 数学与计算机学院, 湖北 武汉 430073)

摘 要: Petri 网作为一种系统建模工具, 特别适合描述系统的控制流、并发和异步特性, 与状态机行为模型相比更具优势. 论文探讨了 TTA 流水线 Petri 网建模方法, 给出了 P/T 模型和 CPN 模型, 并用 PNML 语言进行描述.

关键词: 传输触发构架; Petri 网; Petri 网标记语言; 流水线

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2013)02-0029-04

TTA Pipeline Model Based on Petri Net

ZHU YONG

(College of Mathematics and Computer, Wuhan Textile University, Wuhan 430073, China)

Abstract: Petri net as a modeling tool is particularly suitable for describing the system control flow, concurrency and asynchronous features, and has more advantages compared with the state machine behavior model. The TTA pipeline model based on Petri net, the P / T model and the CPN model, and the description the by PNML language are discussed in the paper.

Key words: TTA(Transport Triggered Architecture); Petri net; PNML(Petri Net Markup Language); pipeline

1 引言

ASIP(专用指令集处理器)的最主要的优点体现在能够根据应用特征针对关键路径进行优化, 解决系统瓶颈, 实现最优的系统平衡(性能、功耗、成本等). 在 20 世纪八十年代, 国外就开展了 ASIP 的研究和应用. 近年来, ASIP 处理器又在网络与通信领域获得了广泛的应用. 当前许多公司推出了 ASIP 处理器产品以及开发环境, 比如: Tensilica 公司的 Xtensa^[1]、ARC 公司的 ARCTangent^[2]、Siemens 公司的 Carmel DSP^[3] 以及 CoWare 公司的 LISATek^[4]等. 这些产品的特点大都是在高可配置的处理器核基础上, 能针对特定应用进行优化, 并提供较为完备的开发工具支持.

芬兰坦佩雷理工大学、荷兰代尔夫特理工大学 Henk Corporaal 团队提出了基于 TTA(Transport Triggered Architecture, 传输触发体系结构)构架的 ASIP 及其原型 MOVE32INT^[5], 其构架突破了传

统处理器以操作为核心的编程理念, 突出操作/传输独立并行的思想, 且使得编译器可见所有的传输过程, 进而可以用软件优化调度指令^[6-7].

在计算机体系结构中, 可以用 Petri 网中的位置用来表示系统的局部状态, 如队列、缓冲、资源等; 变迁则用来描述引起系统状态改变的事件, 如信息处理、存储器读写、数据收发等操作; 弧表明状态和事件之间的关系^[8]. 给出了基于 Petri 网的磁盘阵列串控制器模型 HDL 描述. 进一步还能凭借位置容量和弧权表述其中的数量属性, 文献^[7]描述了不同制式的数字视频处理高速通道. 论文借鉴 Petri 网上述特性描述了 TTA 流水线行为.

2 TTA 体系结构

如果把 RISC(精简指令集计算机)看成 SISO(单指令单操作)类型的体系结构, VLIW(超长指令字)看成 SIMO(单指令多操作)类型的体系结构, 那么 TTA 可以看成传统 VLIW 体系结构的一个超

集,即 SIMT(单指令多传输)类型的体系结构。

TTA 体系结构核心思想是利用数据传输触发具体操作,即功能单元操作均以数据传输作为触发激励,即任何数据源中的数据经过互连网络的传输后,在写入功能单元数据寄存器的同时会触发一次完整的操作。互连网络、寄存器文件与功能单元三者间的互连结构非常适合于开发三者流水线之间的并行性,从而大幅度提高硬件利用率。

TTA 具有以下特征:

- 所有的功能单元都是流水的,不同的功能单元延迟表示其流水级数;
- 编译器对功能单元的延迟是可见的,可以据此对指令进行调度,以消除数据相关和资源相关;
- 提取应用程序特征参数不依赖于任何体系结构。

系统体系架构文件^[8]对处理器组件特性进行定义,主要对象如下:

- 传输总线(Transport Buses)
- Sockets
- 桥(Bridges)
- 功能单元(Function Units)
- 寄存器堆(Register File)
- 流水线(Execution Pipeline)
- 全局控制单元(Global Control Unit)

3 流水线状态机

在 TTA 体系结构中,数据传输网络和功能单元是完全分开的,其流水线行为也是互相独立的(这种方式能够加强编译器的调度优化)。

3.1 传输网络流水线

TTA(Transport Network pipelining)指令流水线执行机制如图 1 所示。

指令 i	取指	译码	数据传输	FU 操作	写结果
指令 i+1		取指	译码	数据传输	FU 操作
指令 i+2			取指	译码	数据传输

图 1 TTA 指令流水线

与传统的 RISC 流水线相比有较大的不同:源寄存器和目的寄存器的传输分离为不同的流水阶段;寄存器取数、写回以及功能单元间的数据旁路均在传输阶段完成,并且要显式编程;move 操作于数据传输段完成。

3.2 功能单元流水线

功能单元流水线(FU pipelining)采用混合方式

的流水线机制实现,即中间值保持上一个流水阶段的结果还被覆盖由有效位控制(这将为调度提供更大的自由空间)。处理结果可以在触发操作后的任意时间从功能单元得到,并且减少了中间寄存器存储临时数据。

典型的功能流水线如图 2 所示。

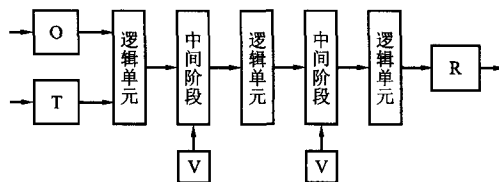


图 2 TTA 功能单元流水线

图中 O、T、R 分别表示 FU 的 3 类基本寄存器,即操作数寄存器、触发寄存器和结果寄存器。V 是控制流水线每个中间阶段的有效位,逻辑单元作为功能单元组合逻辑部分。

流水线包括以下 3 个主要阶段:

- (1) 取指 (Ifetch) 阶段:将指令从 ICache 置入 IR(指令寄存器);
- (2) 译码 (Decode) 阶段:源、目的操作数传输至相应的 socket;
- (3) 传输执行 (Move) 阶段:数据经 FU 处理后传输到目的寄存器;根据 FU 所实现的功能不同,需要一系列的流水线时序。

TR(触发寄存器)触发 FU 功能执行,结合传输网络流水线可以获得极大地并行度;在 FU 内部依据有效位 V 控制功能流水线执行,又获得了极大地灵活性。典型的 TTA 流水线状态机可以表示如图 3 所示。

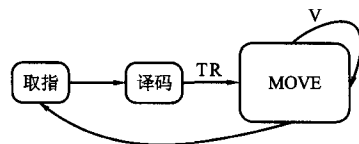


图 3 TTA 流水线状态机

利用有限状态机 FSM 建立 TTA 事件驱动模型,描述各功能实现过程中流水阶段所经历的状态序列,以及如何响应来自其他流水段的各种事件。

4 Petri 网模型

流水线的本质是部件、行为以及时序的集合,它们之间的映射关系便决定了流水线的结构。流水线结构中的元素可以分成两类:一类是寄存器,用于存储数据;另一类是执行部件,它反映了指令的功能行为。在 Petri 网模型中,可以用库所映射存储部件,

用变迁表示功能执行,用扩展 token 的流动来模拟指令在流水线中的流动。

4.1 TTA 流水线 Petri 网模型

基本的 Petri 网模型由一个三元组 $N = (P, T, F)$ 构成:

状态集合 $P = \{ PC, IR, RF, MEM, stall \}$ 为一系列的状态,称作库所,可以描述流水线工作状态和物理对象;

$T = \{ IF, ID, \{ MOVE \} \}$, $MOVE = \{ fu_k \}$ 是迁移集合;主要用于描述指令行为;

$F \in (P \times T) \cup (T \times P)$ 是关系集合,称作弧; P 中还有一些标识称为 Token,描述点火行为,使 Petri 网具有执行语义。

参照 TTA 流水线状态机行为的 Petri 网模型如图 4 所示。

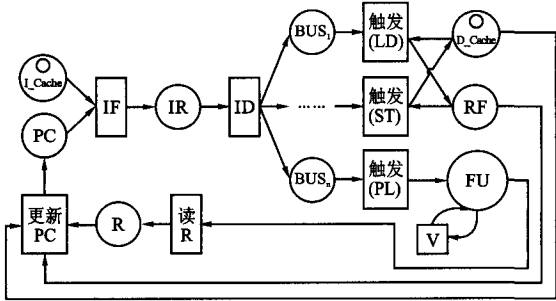


图 4 TTA 流水线 Petri 网模型

与状态机相比 Petri 网模型更加强调了变迁的实质,很好地兼顾了流水线控制器与数据路径的描述与设计。

4.2 TTA 流水线 CPN 模型

传统的 P/T 网比较适合描述控制流,但其本身不具有表示与之相应的处理器执行模式状态的数据作用。数据对象反映的不仅是其类型与操作的属性,而且还有它们之间的依赖信息(如超标量机环境、指令分支、TTA 网络总线选择)。CPN(Coloured Petri Nets, 着色网)具有表示数据属性的机制,它能够有效地将状态标识(通过标识 token)和处理器执行的数据依赖条件结合在一起。

TTA 流水线 CPN 模型如图 5 所示。

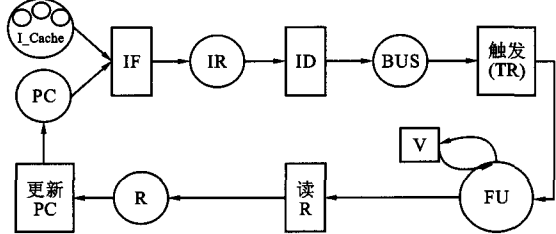


图 5 TTA 流水线 CPN 模型

CPN 使 token 着色,即具有数据属性,对于描述复杂对象非常有效。对于指令中源、目的寄存器采用统一的译码行为,通过让 token 具有不同的数据属性来描述指令的具体功能。指令颜色集定义如下:

Record guard : unsigned * 保护条件:范围 0 -7,表示“允许执行”、“不允许执行”、“条件 0 执行”、“条件 1 执行”等。

src : struct op * 源操作数。

des : struct op * 目的操作数。

op = struct 格式 1	op = struct 格式 2
{ flag : bit = 0 ;	{ flag : bit = 1 ;
addr : unsigned ;	addr : unsigned ;
opcode : string ;	imm : integer ;

5 PNML 描述

PNML^[9]是基于 XML 交换格式标准的 Petri 网模型描述的语言,它采用了 UML 元模型,即 PNML Core Model 来定义共享概念。它们之间映射的元素如表 1 所示。

表 1 PNML Core Model 元素表

Class	XML element	XML attributes
PetriNet Doc	<pnml>	—
PetriNet	<net>	id:ID type:URL
Place	<place>	id:ID
Trannsition	<trannsition>	id:ID
Arc	<arc>	source:IDRef (Node) target:IDRef (Node)
Page	<page>	id:ID
Graphics	<graphics>	—

参照图 4、图 5 TTA 流水线 Petri 网模型,得到 PNML 描述如下:

```
< pnml xmlns = " http://www. tta. org/
pnml">
  <net id=" ttanet" type="http://www. tta.
org/pnml/PTNet">
    <page id=" top-level">
      <name> <text>TTA Pipeline</text></
name>
      <place id=" p1 ">
        <name><text>I_Cache</text></name>
      >I_Cache 库所
      <initialMarking><... /></initialMarking>
    >Token 描述
  </place>
```

<place id="pi"><... /></place> 其它库所

<transition id="t1">

<name><text>IF</text></name>IF

迁移

</transition>

<transition id="ti"><... /></transition>

>其它迁移

<arc id="a1" source="p1" target="t1">

</arc>弧的描述

<arc id="ai" source="pj" target="tk">

</arc>

</page></net></pnml>

限于篇幅,未给出指令流水线细节和可视显示部分 PNML 描述。

6 结束语

处理器设计的核心问题之一是体系结构中的流水线状态机的描述与综合。Petri 网作为一种系统建模工具,特别适合描述系统的控制流、并发和异步特性,与状态机行为模型相比更具优势,并且可被逻辑综合与电路实现。

高级 CPN 模型中 token 可以带有信息,克服了传统 Petri 网中仅仅将其看作一般初始化的不足。在描述多种相关情形时,通过定义颜色集对进行模型简化和规整,加强了 Petri 网的建模能力。

PNML 描述语言就是一种极具潜力的设计方

法,它规范了各种 Petri 网的语义和行为,可以预期验证系统功能的正确性,还能作为综合工具的输入以及系统技术文档标准,并且可以通过第三方工具扩展其可执行描述功能。

参考文献:

- [1] Gonzalez R E, Xtensa. A configurable and extensible processor[J]. IEEE Micro, 2000, 20(2):60-70.
- [2] Segre Leef. A methodology for virtual hardware/software integration[M]. Mentor,1998.
- [3] Itoh M, Higaki S, Sato J, et al. PEAS2-III: an ASIP design environmen[J]. Computer Design, 2000: 430 - 436.
- [4] Oliver Schliebusch. Architecture implementation using the machine description language LISA[C]//Proceedings of the 15th International Conference on VLSI Design. India;Bangalore,2002;239 - 244.
- [5] Henk Corporaal. Microprocessor Architectures: from VLIW to SDTA[M]. New York:John Wiley, 1998.
- [6] 朱勇. 嵌入式 RAID 数据通道的逻辑描述[J]. 小型微型计算机系统,2004(2):254-256.
- [7] 朱勇. 嵌入式高速通道逻辑描述[D]. 武汉:华中科技大学,2007.
- [8] A. Cilio. Architecture definition file[D]. Finland: Tampere University of Technology,2007.
- [9] Billington J. The petri net markup language: concepts, technology, and tools[C]// 24th International Conference, LNCS 2679. Springer, 2003: 483 - 505.

(上接第 28 页)

- [3] Samsung Electronics. K9XXG08XXM Flash Miemory [EB/OL]. [2011-05-06]. www.samsung.com; 18-43.
- [4] National Semiconductor. DS92LV1224 30-66MHz 10 Bit Bus LVDS Deserializer[EB/OL]. [2009-10-06]. www.national.com. 2-3.
- [5] 李佳琦,李健,党相甫,等. 基于固态硬盘的闪存阵列并行结构设计[J]. 微电子学与计算机,2010, 27(4): 85-88.
- [6] 郭小兵,李圣昆,任勇峰. 基于双备份多路数据采集存储系统的设计与实践[J]. 电子技术应用,2012,38(2):

67-69.

作者简介:



郭小兵 男,(1986-),硕士研究生.研究方向为微系统集成、嵌入式系统、高速数据采集、测试技术。