

FPGA 阵列芯片的自然统一研究

孙 璐, 沈绪榜

(西安微电子技术研究所, 西安 710054)

摘 要: FPGA 是 1984 年 Rose Freeman 发明的一种支持 OLP 计算模式的阵列芯片, 实现了非数据并行算法计算的高效性. 但是, OLP 计算模式是采用 VHDL/Verilog 等硬件描述语言, 通过逻辑设计将算法映射到 FPGA 阵列芯片上的, 应用的抽象层次低, 而且 FPGA 阵列芯片体系结构, 与实现数据并行算法计算的 GPU 系统芯片体系结构是不统一的. 为了克服 FPGA 阵列芯片的这两个缺点, 本文研究了能够等效替代 OLP 计算模式及其 FPGA 阵列芯片的 ILP 计算模式及其 APU 系统芯片, 并自然统一了 ILP 与 DLP 两种计算模式的系统芯片体系结构.

关键词: OLP; FPGA; ILP; APU

中图分类号: TP39

文献标识码: A

文章编号: 1000-7180(2013)01-0001-04

The Study of Natural Unification for FPGA Array Chip

SUN Lu, SHEN Xu-bang

(Xi'an Microelectronic Technology Institute, Xi'an 710054, China)

Abstract: FPGA is invented by Rose Freeman in 1984, it is a kind of array chip for OLP computing paradigm, which has high efficiency of non-data parallel algorithm. But the algorithm is mapped by the logic design to the FPGA array chip with hardware describe language, such as VHDL or Verilog, that lower the level of applied abstraction. In addition, the architecture of FPGA array chip is not uniformed with GPU SoC for DLP computing. To overcome these two shortcomings, we studied the ILP computing paradigm and its APU SoC which can replace the OLP computing paradigm and its FPGA array chip equivalently, realized the natural unification of two architecture of SoC for ILP and DLP computing paradigms.

Key words: OLP; FPGA; ILP; APU

1 引言

1984 年 XILINX 公司的创始人之一 Ross Freeman 发明的现场可编程门阵列 (Field-Programmable Gate Array, FPGA) 芯片, 是通过逻辑设计在同一块芯片上实现非数据并行的不同算法, 具有可反复使用和高效实现并行计算的特点. 例如, 对于哈夫 (Hough) 变换和反哈夫变换的计算, 采用 3-4 GHz 的处理器, 需要 12 min 的处理时间, 而用 FPGA 作为处理器的协处理加速器, 只需要 12 s, 加速倍数为 370 倍^[1]. 但是, 在每次使用时, FPGA 阵列芯片中总有一部分逻辑单元处于闲置状态. 因此,

FPGA 在发明之初被称为“奢侈”的创意^[2]. 然而, FPGA 阵列芯片能够通过逻辑设计实现多种算法的特点, 在仿真计算和嵌入式计算等应用中, 可以节省昂贵的流片费用, 并缩短研发周期. 如果采用 ASIC 电路, 每种算法都需要一种特定的 ASIC 芯片来实现, 流片次数多, 研发周期长.

不仅如此, FPGA 阵列芯片对非数据并行算法计算的高效性与 GPU 系统芯片对数据并行算法计算的高效性正好是互补的, 因此, FPGA 阵列芯片成了一种不可缺少的硬件加速器, 能与 GPU 系统芯片一起实现高性能计算的高效性. 然而, 由于 FPGA 阵列芯片是采用硬件描述语言通过逻辑设计实现算

法的映射,而不是像 GPU 系统芯片那样,采用软件描述语言通过程序设计实现算法的映射,这就大大降低了 FPGA 阵列芯片应用的抽象层次.同时,FPGA 阵列芯片与 GPU 系统芯片体系结构的不统一带来了两种芯片协同工作的障碍.本文研究了 FPGA 阵列芯片与 GPU 系统芯片的自然统一问题.提出了一种基于 APU 系统芯片的 ILP 计算模式,以等效替代 FPGA 阵列芯片的 OLP 计算模式,并实现了 FPGA 阵列芯片到 APU 系统芯片的自然统一.

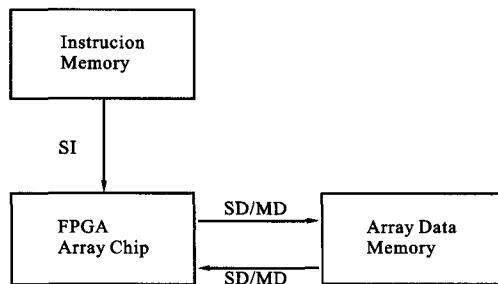
2 高效 FPGA 阵列芯片的不统一性

目前来看,提高计算机运算速度的主要途径就是并行计算,而运算并行度的提高归根结底都是以算法为基础来实现的.可以将所有算法分为两类,对多个数据能进行同一操作的算法称为数据并行算法,是用数据级并行(Data Level Parallel, DLP)计算模式的图形处理器(Graphic Processing Unit, GPU)系统芯片实现的;对多个数据不能进行同一操作的算法称为非数据并行算法,是用操作级并行(Operation Level Parallel, OLP)计算模式的 FPGA 阵列芯片实现的. OLP 计算模式是通过逻辑设计对阵列芯片的逻辑单元进行配置,实现非数据并行算法的功能的.如图 1(a)所示,数据从阵列数据存储器(Array Data Memory)中,流入到运算单元相互邻接的 FPGA 阵列芯片中,直接完成算法所要求的多操作后流出阵列芯片,存回到阵列数据存储器中.例如,为了完成哈夫(Hough)变换的计算公式:

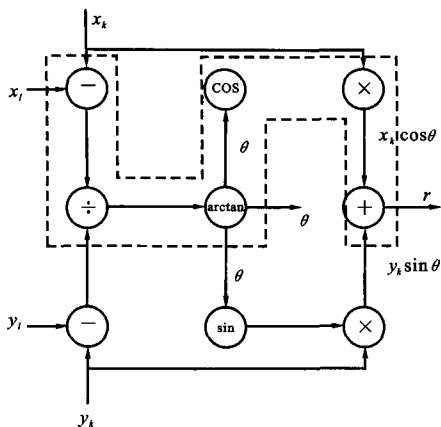
$$\theta = \arctan\left(\frac{x_l - x_k}{y_k - y_l}\right)$$

$$\gamma = x_k * \cos\theta + y_k * \sin\theta$$

FPGA 器件需要 9 个处理元,由 3×3 的阵列处理元组成,如图 1(b)所示.输入为直角坐标系中的两个任意非零点 (x_k, y_k) , (x_l, y_l) ,输出为参数空间中的一个点 (γ, θ) .计算出 θ 值需要 4 个处理元,包括 2 个减法处理元,1 个除法处理元,1 个反正切处理元.在得到 θ 值的基础上计算出 γ 值需要增加 5 个处理元,包括 1 个正弦处理元,1 个余弦处理元,2 个乘法处理元,1 个加法处理元.在有些 FPGA 中,反正切、正弦、余弦函数是用查找表的方法实现,而在 ARM 处理器中,这些函数是用一条指令来实现的,好处是能够简化 FFT 等运算的程序设计.在 Hough 变换的实现结构中,如图 1(b)中所示,两个减法操作,正弦和余弦操作以及两个乘法操作,都是



(a) OLP 计算模式实现的概念图



(b) FPGA 阵列芯片的 OLP 计算举例

图 1 OLP 计算模式的概念及计算举例

并行执行的,这些并行操作的长度随应用算法的不同都是可扩展的.数据流操作如图 1(b)虚线框中所示,包括减法、除法、反正切、正弦/余弦、乘法、加法等六级数据流,数据流的深度也是随应用算法的不同而扩展的.也就是说,阵列芯片的规模可以根据用途,随流水线深度及并行长度的增加而扩展,为算法开发者提高非数据并行算法的性能创造了条件.

实际应用表明,FPGA 阵列芯片支持非数据并行计算的能力,比传统 CPU 的计算能力高百倍以上.因此,FPGA 阵列芯片的 OLP 计算模式是一种不可缺少的并行计算模式.然而,FPGA 阵列芯片是在设计人员了解电路/逻辑结构的基础上,采用硬件描述语言 VHDL/Verilog 的逻辑设计来实现 OLP 计算模式的.而不是像 GPU 系统芯片那样,设计人员不需要了解硬件结构,就能采用 C 等软件描述语言的程序设计来实现 DLP 计算模式.编程语言的不统一,使 FPGA 阵列芯片通常只能用作芯片设计的模拟器和 MPP 计算机的协处理加速器.例如,2007 年底 Gray 公司推出的 XT5 超级计算机,每个节点就是由一个 AMD 的 Opteron 处理器,和两个 DRC 的 RPU(Reconfigurable Processing Unit)组成,通过 Hyper Transport 总线互连^[3].

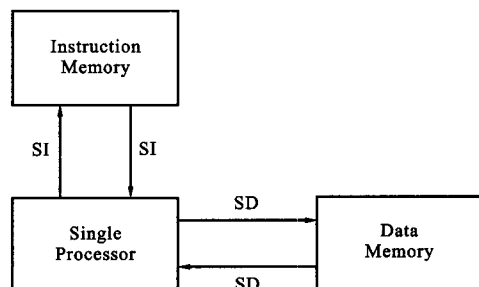
随着工艺的发展,FPGA的集成度越来越高,为了将其应用于更广阔的嵌入式领域,人们想到了在FPGA上集成处理器、控制器以及常用外设的输入输出电路来增强特性降低功耗^[4].例如,2011年Altera公司发布的基于ARM的SoC FPGA系列产品,在单芯片中集成了28 nm Cyclone V/Arria V FPGA架构、双核ARM Cortex-A9 MPCore处理器、纠错码(ECC)保护存储器控制器、外设和宽带互联电路等.处理器系统的峰值性能达到4 000 DMIPS(Dhrystone Million Instructions executed Per Second).同时,由于SoC FPGA中使用硬核嵌入式电路,不必在FPGA中实现这些IP核的功能,这样就有更多的可编程逻辑资源来实现专用定制逻辑,从而降低了系统功耗(其功耗不到1.8瓦)^[5].目前,SoC FPGA主要应用于“绿色”或节能技术的工业领域、视频处理、无线基础设施及交换机和路由器、计算机和存储等四大应用市场中^[6].但是,无论FPGA与CPU/GPU怎样融合,编程语言类型的不统一问题始终限制着FPGA的应用.于是,在2008年,人们想到了用软件编程语言(C/C++、system C)来进行芯片的RTL级优化.Cadence公司的C-to-Silicon编译器已在日本开始应用^[7].现在的超级计算机中,通常会使用FPGA阵列芯片作为加速器来完成非数据并行的计算以达到加快整体运算速度的目的.然而,由于FPGA阵列芯片的体系结构与GPU系统芯片体系结构是不同的,它们组成异构计算机时,存在着节点比例分配和数据传输的问题.而且,由于算法的不同,不同芯片的节点处于等待状态,很难达到持续的峰值,这些因素都会影响异构计算机的计算效率.

因此,从系统结构的角度考虑,就需要提出一种新的软件编程语言的并行计算模式及其系统芯片,以等效地替代硬件描述语言的OLP计算模式及其FPGA阵列芯片.

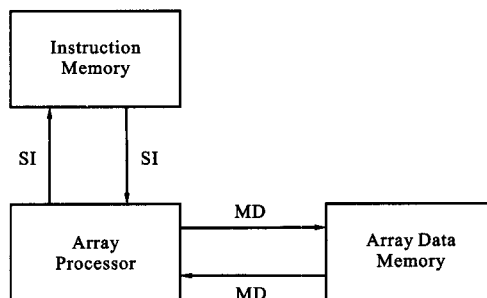
3 高效FPGA阵列芯片的可统一性

FPGA阵列芯片的OLP计算模式的应用,只能采用硬件描述语言,没有软件描述语言的编程灵活性,是由于在OLP计算模式的FPGA阵列芯片中,根据其非数据并行的算法特点,每个处理单元是依靠逻辑设计时生成的多操作来互连的,并不是通过指令来实现的.因此,要用一种软件编程语言的计算模式来等效取代OLP计算模式,才能实现FPGA阵列芯片的可统一性.

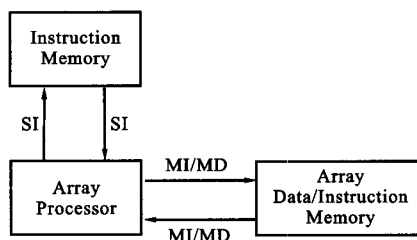
从体系结构上来说,串行计算模式的计算机概念,如图2(a)所示,是由指令存储器,单处理器以及数据存储器三部分组成.由于指令是由操作码和操作数的地址码两部分组成,也就是说,指令流包括了“操作流”和地址码控制的“数据流”.在现在已有的处理器中,指令的地址码,只对寄存器/存储器中的数据访问.但在DLP计算模式中,GPU系统芯片是由指令存储器、阵列处理器(Array Processor)和阵列数据存储器(Array Data Memory)三部分组成的,如图2(b)所示.由于阵列处理器中的处理器核是相互邻接的,为了用指令形成处理器核之间的操作互联,指令是具有邻接寻址模式的,指令中操作数的地址码不仅可以对寄存器/存储器中的数据进行访问,也可以对相邻处理器核中的数据进行访问.



(a) 串行计算模式的计算机概念图



(b) DLP计算模式的计算机概念图



(c) DLP和ILP计算模式的计算机概念图

图2 统一芯片体系结构统一的概念图

为了用指令实现OLP计算模式,提出了指令级并行(Instruction Level Parallel, ILP)计算模式,并将其对应的系统芯片称为阵列处理器(Array Processing Unit, APU)系统芯片.ILP计算模式的

APU 系统芯片是由指令存储器、阵列处理器(Array Processor)和阵列数据/阵列指令存储器(Array Data/Instruction Memory)三部分组成的,如图 2(c)所示. ILP 计算模式中的指令也是具有邻接寻址模式的,指令中操作数的地址码对相邻处理器核中数据的访问,可实现 FPGA 阵列芯片中多操作形成的运算单元之间数据的流动. 这样,就可以用抽象层次高的 ILP 计算模式的程序设计,取代抽象层次低的 OLP 计算模式的逻辑设计,实现两种计算模式的统一. 与 DLP 计算模式的 GPU 系统芯片不同之处在于,APU 系统芯片中,是将阵列数据存储器也当作阵列指令存储器来使用,所以图 2(c)称为 DLP+ILP 计算模式的计算机概念图. APU 系统芯片可以实现所有 DLP 计算模式的数据并行算法以及 ILP 计算模式的非数据并行算法. 所以,ILP 计算模式的系统芯片与 DLP 计算模式的系统芯片是统一的. 同时,由于单处理器是 1×1 的阵列处理器,因此,APU 系统芯片是串行计算模式和并行计算模式系统芯片体系结构的统一.

4 统一系统芯片上多进程的实现

ILP 计算模式及其 APU 系统芯片的提出,不仅减少了高性能计算芯片设计的种类,使所有的计算模式都可以用 DLP+ILP 计算模式来实现,而且能够实现这些计算模式的计算高效性. 例如,9 个多进程的执行,可以由 $P_{1 \times 1}$ 的单处理器的串行计算模式来实现,如图 3(a)所示,进程 $T_1, \dots, T_9$ 是串行执行的,假定每个进程执行的时间分别为 $t_1, \dots, t_9$,那么,这 9 个进程执行完成的总时间为 $t_s = t_1 + \dots + t_9$. 如果由 $P_{1 \times 9}$ 的 CPU 系统芯片的进程级并行(Tread Level Parallel, TLP)计算模式来实现,如图 3(b)所示,进程 $T_1, \dots, T_9$ 是分配到 9 个处理器核上并行执行的,但每个进程是串行执行的. 若进程之间存在数据相关, TLP 计算模式就会带来互斥(数据依赖)的问题,增加额外的时间开销. 那么,每个进程的执行时间 $t_1', \dots, t_9'$ 都可能大于串行计算模式中每个进程的执行时间 $t_1, \dots, t_9$. 因此,这 9 个进程执行完成的总时间 $t_t = \max(t_1', \dots, t_9') \geq t_s/9$. 如果用本文所提出的 APU 系统芯片的 DLP+ILP 计算模式实现,如图 3(c)所示,每个进程都是可以在 $P_{3 \times 3}$ 的阵列处理器上并行执行的,执行时间分别为 $t_1'' \geq t_1/9, \dots, t_9'' \geq t_9/9$, 执行完成的总时间 $t_d = t_1'' + \dots + t_9'' \geq t_s/9$. 这 9 个进程之间是一个接一个顺序执行的,不仅没有同步和互斥的问题,即 $t_d \leq$

t_t , 而且可以统一串行计算和并行计算的编程语言和程序设计方法.

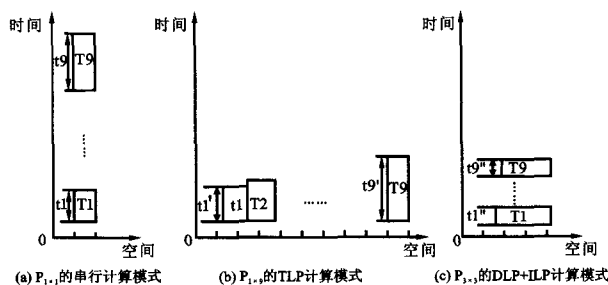


图 3 多进程执行的时空图

5 结束语

本文研究 FPGA 阵列芯片的自然统一,通过提出能够等效替代 OLP 计算模式及其 FPGA 阵列芯片的 ILP 计算模式及其 APU 系统芯片,统一了并行计算系统芯片的编程语言及其体系结构. 可实现同构计算机,解决多种系统芯片的异构计算机中不同系统芯片之间的节点分配比例和数据传输问题. 同时,能够解决 TLP 计算模式的 CPU 系统芯片并发计算的同步和互斥问题.

参考文献:

- [1] Alter 公司. 利用 FPGA 加速实现高性能计算[J]. 电子产品世界, 2008(1): 103-106.
- [2] 冯晓伟. 赛灵思:降低电子设计成本主推目标设计平台[N]. 中国电子报, 2009.
- [3] 原昊, 吴东. 高性能可重构计算机系统架构研究[J]. 高性能计算技术, 2010(3): 1-15.
- [4] Shen XuBang, Zhao CuiHua. An AP SoC for a unified architecture[J]. Chinese Science Bulletin, 2012, 55(35): 4089-4093.
- [5] 陆楠. SoC FPGA 进一步提高设计灵活性[J]. 电子技术设计, 2011(12): 24-25.
- [6] 李映. 嵌入式应用成 SoC FPGA 新引擎[N]. 中国电子报, 2012.
- [7] Martin G, Smith G. Tensilica. High-level synthesis: past, present, and future[J]. Design & Test of Computers, IEEE. 2009, 26(4): 18-25.

作者简介:

孙璐女, (1988—), 硕士研究生. 研究方向为计算机系统结构.

沈绪榜男, (1933—), 博士生导师, 中国科学院院士. 研究方向为嵌入式计算机及其国产芯片实现.