

基于关键路径和面积预测的软硬件划分方法

彭艺频, 凌 明, 杨 军, 时龙兴

(东南大学国家专用集成电路系统工程技术研究中心, 江苏南京 210096)

摘 要: 本文提出了一种基于关键路径和面积预测的软硬件划分方法, 这种划分方法将软硬件映射和任务调度合而为一, 在调度过程中同时完成软硬件的映射, 充分发挥了任务调度的作用. 在实验过程中, 我们对比了基于模拟退火算法的软硬件划分方法(SA)和基于路径分析的软硬件划分方法(PA). 实验结果表明, 我们提出的方法在成功率以及结果的优化程度上都能取得更好的效果.

关键词: 软硬件划分; 任务调度; 关键路径; 面积预测; 通信综合

中图分类号: TN4 **文献标识码:** A **文章编号:** 0372-2112 (2005) 02-0249-05

Hardware/ Software Partitioning Method Based on Critical Path and Area Prediction

PENG Yi-pin, LING Ming, YANG Jun, SHI Long-xing

(The National ASIC Center, Southeast University, Nanjing, Jiangsu 210096, China)

Abstract: The paper proposes a hardware/ software partition method based on critical path and area prediction. It integrated hardware/ software mapping into task scheduling and make schedule more helpful to hardware/ software partitioning. The experiments show it is very efficient compared to SA - based method and Path Analysis method.

Key words: hardware/ software partitioning; task scheduling; critical path; area prediction; communication synthesis

1 引言

嵌入式系统的设计不仅仅需要考虑功能的正确性, 在很多情况下还要求系统达到一定的性能要求, 如 MP3, MPEG 系统的设计. 由于对系统性能的高要求给设计带来了很大的困难. 对于部分系统采用纯软件的实现方式可以达到系统性能的要求; 但对于大部分的系统是无法通过纯软件的实现方式来达到要求的系统性能的. 通常的做法就是采用纯硬件的实现方式. 尽管这种方式能够带来很好的性能, 但同时也带来了很大的硬件开销, 使得最终系统的代价很高, 在很大程度上降低了产品的竞争力. 采用混合软硬件的实现方式, 通过合理的软硬件划分, 以较小的硬件开销来获得系统性能的满足成为一个新的研究热点.

国内外学术界对软硬件划分的研究由来已久, 也提出了一些可行的方法. 从任务调度的角度, 这些划分方法可以概括成两类: 一类是先通过优化算法如模拟退火算法^[1]、禁忌搜索^[2]、遗传算法^[3]来确定参与划分结点的软硬件映射, 通过任务调度对获得的软硬件映射方案进行全局性能如系统执行时间的估计, 检查是否满足设计约束, 通过这种映射和估计的不断迭代, 以最小的代价获得设计约束的满足文[1~5], 如图 1

所示.

这一类方法也是研究较多的一类. 但是这类方法有一个明显的缺陷: 由于软硬件的映射过程



和任务调度过程是分离的, 任务调度过程仅仅是作为估计手段和进行约束检查使用, 无法对映射过程进行更直接的指导, 造成从映射到调度的迭代很频繁, 极大地增加了计算复杂度. 为了解决这个问题, 有些研究已经开始将软硬件的映射过程集成到任务调度过程, 在任务调度过程中同时完成软硬件的映射, 如上图 2 所示.

Bjorn Jorgensen 提出了一种基于关键路径优化的方法^[6]同时处理映射和调度过程, 但是这种方法的目标只是最小化关键路径, 而不考虑因此而带来的其它开销, 因此无法解决类似于软硬件划分这一类问题. Lunc Bianco^[7]提出了一种文[6]方法的改进方法, 增加了对硬件开销的优化过程, 解决在性能约束下的硬件开销最小化问题. 但是这种方法只是从资源共享方面进行硬件开销的优化, 优化的程度非常有限.

本文提出了一种新的基于关键路径的软硬件划分方法, 这种方法在关键路径的优化过程中不以路径最小化为目标,

而是在满足时序约束的前提下,通过硬件面积预测来引导结点的软硬件映射选择,解决系统性能约束下的硬件面积最小化的问题.

2 软硬件划分问题的数学描述

2.1 目标架构

研究主要针对于如下的硬件结构:

这种单处理器 + 硬件 ASIC 的目标结构常常用于数据密集型的应用如 MP3, MPEG 采用基于共享内存的通信模型^[8]. 为了便于进行研究,对目标结构进行了如下的假设:



(1) 假设内部 memory 的容量固定,而且能够满足软件存储和运行的需要.当结点在微处理器上运行时只有执行时间的开销,而忽略 memory 的开销.

(2) 假设结点的硬件执行时间总是小于结点的软件执行时间.

(3) 简单认为所有的通信开销为一个常数,而且不参与任务调度过程.

2.2 系统建模及数学描述

数据密集型的应用通常可以用有向非循环图(DAG: Direct Acyclic Graph) $G_T = (V_T, E_T)$ 来表示,如下图 4 所示.

其中,结点 $(i \in V_T)$ 表示系统描述中的计算部分,边 $(e_{ij} \in E_T)$ 表示结点之间的通信.基于上面的通信假设,在划分过程中并没有考虑通信调度,因此 DAG 图中的边只表示结点之间的相关性,而不作量化处理.



对系统 DAG 模型进行了如下参数化考虑:

$$i(T_{sw}^i, T_{hw}^i, A_{hw}^i, I^i, T_s^i, T_e^i)$$

其中:

T_{sw}^i : 表示结点 i 软件实现时的执行时间; T_{hw}^i : 表示结点 i 硬件实现时的执行时间;

A_{hw}^i : 表示结点 i 硬件实现时的面积开销; I^i : 表示结点的映射选择(0:软件实现, 1:硬件实现); T_s^i : 表示结点在调度中的开始执行时间; T_e^i : 表示结点在调度中的执行结束时间.

根据上面的系统模型,将满足系统性能约束前提下,系统硬件开销最小的软硬件划分问题提炼成如下的数学描述: 软硬件划分问题中的目标函数:

$$\text{Min} \quad (A_{hw}^i I^i) \tag{1}$$

约束条件 1:

$$\text{Max}(T_e^i) \leq T_C \tag{2}$$

约束条件 2:

假设 $\text{pred}(i)$ 为结点 i 的所有父结点的集合; $\text{succ}(i)$ 为结点 i 的所有子结点的集合,那么有:

$$T_c^k \leq T_s^i \leq T_s^k \tag{3}$$

其中: $i \in \text{succ}(j), k \in \text{pred}(j)$.

3 基于关键路径和面积预测的软硬件划分方法

3.1 关键路径定义

由于采用的方法是在调度过程中确定软硬件的映射选择,在调度完成之前并不知道所有结点的映射情况,因此在关键路径搜索时考虑最差情况,对 ALAP(as later as possible) 调度算法进行了重新定义,具体定义如下:

$$(1) \text{ if succ}(i) = : \begin{aligned} pl_{hw}^i &= T_c - T_{hw}^i \\ pl_{sw}^i &= T_c - T_{sw}^i \end{aligned} \tag{4}$$

$$(2) \text{ if succ}(i) : \begin{aligned} pl_{hw}^i &= \text{Min}_j \text{succ}(j) (pl_{hw}^j) - T_{hw}^i \\ pl_{sw}^i &= pl_{hw}^i + T_{hw}^i - T_{sw}^i \end{aligned} \tag{5}$$

另外,通过 ASAP 调度算法(as soon as possible),确定结点最早可以开始时间,假设当前结点将映射到硬件:

$$est_{hw}^i = \text{Max}_k \text{pred}(j) \{ est^j + T_{sw}^j (1 - I^j) + T_{hw}^j I^j \} \tag{6}$$

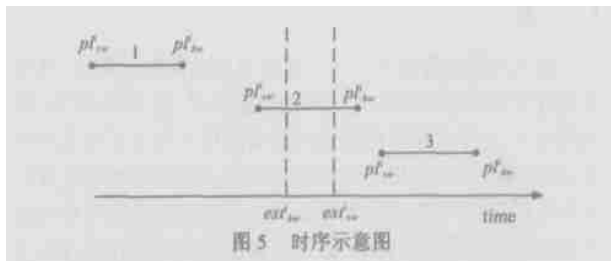
假设当前结点将映射到软件,由于微处理器的是串行执行设备,需要考虑当前微处理器是否处于 idle 状态,且 idle 时间足够长 $idle \geq T_{sw}^i$,因此对上式进行了如下的修正:

$$est_{sw}^i = \text{Max} \{ \text{Max}_j \text{pred}(j) \{ est^j + T_{sw}^j (1 - I^j) + T_{hw}^j I^j \}, \text{cpuidle} \} \tag{7}$$

注:cpuidle 为 cpu 当前可用的且 $idle \geq T_{sw}^i$ 的 idle 起点时间.

3.2 软硬件划分方法

根据上面的定义,很容易对结点的软硬件映射进行时序约束满足的判断,如下图 5 所示:



根据 2.1 的假设 2,我们知道 $pl_{hw}^i < pl_{sw}^i$, 因此 pl_{sw}^i 和 est_{sw}^i 的关系可以分为下面三种情况:

情况 1

$$pl_{hw}^i < est_{hw}^i, pl_{sw}^i < est_{sw}^i \tag{8}$$

对于这种情况,不论结点 i 映射到软件还是硬件,都无法使得最终的系统性能满足给定的约束,即无法获得合理解.

情况 2

$$pl_{hw}^i \leq est_{hw}^i, pl_{sw}^i < est_{sw}^i \tag{9}$$

对于情况 2,结点 i 只能映射到硬件实现才能满足最终的时序约束.

情况 3

$$pl_{hw}^i \leq est_{hw}^i, pl_{sw}^i \leq est_{sw}^i \tag{10}$$

对于情况 3,结点 i 可以映射到软件实现,也可以映射到硬件实现,都能够满足时序约束,对于这种情况,硬件面积的优化将成为重点,引入了面积预测方法来指导当前结点的映射,使

得系统的硬件开销最小。

根据上面三种情况,提出了软硬件划分方法,其伪代码如下所示:

```

CPA ( DAG ( V, N ) {
    H = ;
    S = { all . nodes } ;
    While exist tasks V_T not schedule
        Get Ready node list R ;
        Foreach i R
            Compute  $pl_{hw}^i, pl_{sw}^i, est_{sw}^i, est_{hw}^i$  ;
            Caseof
                 $pl_{hw}^i < est_{hw}^i, pl_{sw}^i < est_{sw}^i$  : Fail ; exit ;
                 $pl_{hw}^i < est_{hw}^i, pl_{sw}^i < est_{sw}^i$  : Add i to H, schedule ; re-
move i from S ; break ;
                 $pl_{hw}^i < est_{hw}^i, pl_{sw}^i > est_{sw}^i$  :  $I^i = 0$  ; // if node i mapping
to soft
                    Area1_i = Area . Estimate ( DAG ) ;  $I^i = 1$  ;
                    Area2_i = Area . Estimate ( DAG ) +  $A_{hw}^i$  ;
                    area_i = | Area1 - Area2 | ;
                    Break ;
            Endcase ;
        Endfor ;
        Select node j with Max area_j ;
        if Area1_j > Area2_j, add j to H, schedule ;
        else add j to S, schedule ;
    Endwhile ; }
    
```

3.3 面积预测技术

3.3.1 局部特性量化 对于上述情况 3, 结点可以映射到软件实现,也可以映射到硬件实现,都可以使得未来的时序约束得到满足.这同时也是获得硬件面积最小化的一个关键优化点.当前结点的映射决定将对未来面积的影响情况是决定当前结点映射的关键.开发了一种面积预测技术(针对还没有完成调度结点的预测)来驱动当前结点的映射.这种面积预测技术结合了结点本身的局部特性,也考虑了全局面积增加.具体描述如下:

结点局部特性量化 1 (Local_1)

$$Local(i) = (T_{sw}^i - T_{hw}^i) / A_{hw}^i \quad (11)$$

这种结点的局部特性用来表征结点本身特性对于软件实现和硬件实现的倾向性.Local 值越小表明结点本身更倾向于软件实现.这种结点局部特性的分布有时差别很大.

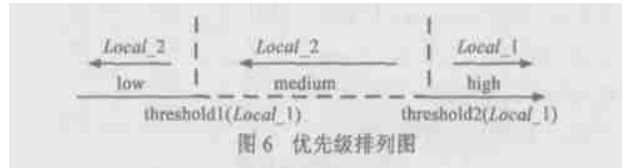
结点局部特性量化 2 (Local_2)

$$Local(i) = A_{hw}^i / T_{sw}^i \quad (12)$$

在面积预测过程中,将采用背包问题^[9]的解决方法,在调度长度为 T_c 的范围内,尽量使映射到软件的结点面积 A 最大,反之也就是用硬件实现的结点面积最小.定义了局部特性 2(单位软件时间的硬件面积值),局部特性 2 越大,则当前结点在预测过程中倾向于软件实现.

3.3.2 优先级分配规则 通过上述的两个局部特性量,定义

未调度结点中将映射到硬件结点的面积总开销,对还未被调度的结点通过下面的优先级分配来预测其未来的映射情况,优先级分配规则如下图所示:



将未调度的结点分为三个优先级段:

(1) 高优先级段 High:

$$\{ i | i \in U, Local_1(i) > threshold2 \}$$

(2) 中优先级段 medium:

$$\{ i | i \in U, Local_1(i) < threshold2, Local_1(i) > threshold1 \}$$

(3) 低优先级段 low:

$$\{ i | i \in U, Local_1(i) < threshold1 \}$$

注: U 表示 unscheduled node list.

不同段的优先级关系如下:

$$i \text{ high}, j \text{ medium}, k \text{ low} \Rightarrow Priority(i) < Priority(j) < Priority(k)$$

对于高优先级段内的结点,按照局部特性 1 进行排序,越大,优先级越高.对于中、低优先级段按照局部特性 2 进行排序,越小,优先级越高.如图 6 所示.

3.3.3 面积预测算法 根据上面的优先级分配规则,给出面积预测方法的伪代码,如下所示:

```

Area . Estimate ( DAG ( V, E ) ) {
    Area = 0 ;
    Foreach ( i U )
         $I^i = 0$  ;
        Compute Local_1 ( i ), Local_2 ( i ) ;
    Endfor ;
    Foreach ( i U )
        Priority . assign ( i ) ;
    Endfor ;
    While ( U ! = )
        K = node with max priority ;
         $I^K = 1$  ;
        T = Compute . system . time ( ) ;
        if ( T > T_c )
            Area = Area +  $A_{hw}^K$  ;
            H = H + k ; U = U - k ;
        else
             $I^K = 0$  ; S = S + k ; U = U - k ;
        Endif ;
    Endwhile ; Return ( Area ) ; }
    
```

4 实验结果

采用 C/C++ 语言开发完成了上述提出的基于关键路径优化的软硬件划分方法.为了验证其可行性和有效性,进行了对比实验.

实现了软硬件划分研究中常常使用的基于模拟退火算法的划分方法^[1,2],以及文[8]提出的基于路径分析的软硬件划分方法,通过 Princeton 大学开发的 TGFF^[10]产生 benchmark 作为实验对象,进行了大量的实验,下面给出部分实验结果.

表 2 实验参数表

		EX1	EX2	EX3	EX4
SA	初温	200	200	300	400
	末温	1	1	1	1
	降温系数	0.96	0.98	0.96	0.98
PA	T. reuse	0.2	0.2	0.2	0.2
	T. new	0.9	0.9	0.9	0.9
CPA	Threshold1	0	0	0	0.04
	Threshold2	1.5	0.2	0.8	0.17

注:SA-Simulated Annealing,请参考文[2];PA-Path Analysis,请参考文[8];CPA-本文提出的划分方法.

根据上面的实验参数,进行了如下实验,通过控制系统性能约束 T_c 值(从当所有的结点都采用软件实现时的系统执行时间作为 T_c 初始,然后使得 $T_c = T_c - \dots$,直到 T_c 等于所有结点都采用硬件实现时的系统执行时间为止.通过约束的移动控制,获得在满足当前约束 T_c 时的硬件面积开销曲线.下面给出实验 EX4 的实验结果,如图 7 所示.

结果分析:

从实验情况可得到如下结论:基于模拟退火算法的软硬件划分方法(SA)在结点数较少的情况下能够获得较优的结果,对于结点数较多的情况收敛较慢,当约束很紧时有时无法得到可行解.而 PA,CPA 方法收敛速度很快,当约束很紧时也能得到可行解.另外,随着实验结点数的增加,本文提出的方法(CPA)能够获得比 SA 和 PA 方法更优的结果,如上图 7 所示,在相同系统性能约束下,能够以更小的面积获得性能约束的满足.

5 总结

本文提出了一种基于关键路径和面积预测的软硬件划分方法,这种划分方法将软硬件映射和任务调度合而为一,在调度过程中同时完成软硬件的映射,充分发挥了任务调度的作用,实验证明,这种方法能够取得很好的效果.

然而,本文还存在一些不足之处,首先,在上述的方法中简单的将通信常数化,同时通信没有参与调度,这种简化对于通信延迟较小的应用可以适用,但是对于类似于无线通信系

统这样通信开销很大的应用会带来明显的问题;其次,提出的面积预测技术采用了类似于背包问题的处理方式,这种方式带有明显的贪婪本质,容易陷入局部优化;

在下一步工作中,我们将引入精确的通信模型(而不采用简单的基于共享内存的通信模型),将软硬件划分完成以后才进行的通信综合^[11,12]提前引入到划分过程中,使得通信部分能够更早、更精确地融入划分过程中;改进面积预测技术,使得面积预测能跳出局部优化,获得更加精确的估计结果.

参考文献:

- [1] Rolf Ernst, Thomas Benner. Hardware-software cosynthesis for micro-controllers[J]. Design & Test of Computers, IEEE. Dec. 1993, 10(4): 64 - 75.
- [2] Petru Eles, Zebo Peng, Krzysztof Kuchcinski. Hardware/ Software Partitioning with Iterative Improvement Heuristics [A]. System Synthesis, 1996. 9th International Proceedings [C]. California IEEE Computer Society, 1996. 71 - 76.
- [3] D Saha, R S Mitra, Anupam Basu. Hardware Software Partitioning using Genetic Algorithm[A]. VLSI Design, 1997. 10th International Proceedings[C]. Hyderabad India: Indian Institute of Technology, 1997. 155 - 160.
- [4] Daniel W Engels, Srinivas Devadas. A New Approach to Solving the Hardware-Software Partitioning Problem in Embedded System Design [A]. Integrated Circuits and Systems Design, 2000. 13th International Proceedings[C]. Manaus Brazil: IEEE Computer Society, 2000. 275 - 280.
- [5] J Hou, W Wolf. Pre-synthesis partitioning for hardware/ software cosynthesis[J]. IEEE Proc-Comput Digit Tech, 1998, 145(3): 61 - 65.
- [6] Bjorn Jorgensen, Jan Madsen. Critical Path Driven Cosynthesis for Heterogeneous Target Architecture [A]. Hardware/ Software Codesign, 1997. (CODES/ CASHE '97), Proceedings of the Fifth International Workshop [C]. Braunschweig Germany: IEEE Computer Society, 1997, 15 - 19.
- [7] Lunc Bianco, Michel Auguin, Guy Gogniat. A Path Analysis based Partitioning for Time Constrained Embedded Systems [A]. Hardware/ Software Codesign, 1998. (CODES/ CASHE '98) Proceedings of the Sixth International Workshop [C]. Seattle WA: IEEE Computer Society, 1998. 85 - 89.
- [8] J Henkel, Th Benner, R Ernst. Hardware Generation and Partitioning Effects in the COSYMA System [R]. Germany: Technical Paper, Technische Universitat Braunschweig, 1993.
- [9] 宋文, 吴晟, 杜亚军. 算法设计与分析 [M]. 重庆: 重庆大学出版社, 2001. 9.
- [10] Dick R P, Rhodes D L, Wolf W. TGFF: task graphs for free [A]. Hardware/ Software Codesign, 1998. (CODES/ CASHE '98) Proceedings of the Sixth International Workshop [C]. Seattle WA: IEEE Computer Society, 1998. 97 - 101.
- [11] Jear-Marc Daveau, Tarek Ben Ismail. Synthesis of System-level Communication by an Allocation Based Approach [A]. International Symposium on Systems Synthesis 1995 Proceedings of the 8th international symposium on System synthesis [C]. Cannes, France: SGS Thomson Microelectronic, 1995. 150 - 155.
- [12] Guy Gogniat, Michel Auguin. Communication Synthesis and HW/SW

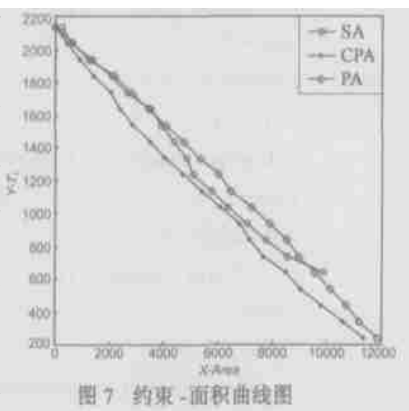


图 7 约束-面积曲线图

Integration for Embedded System Design[A]. 1998 Proceedings of the 6th international workshop on Hardware/ software Codesign[C]. Seattle

WA:IEEE Computer Society, 1998, 49 - 53.

作者简介:



彭艺频 男,东南大学电子工程系 2002 级博士研究生,目前在东南大学国家专用集成电路系统工程技术研究中心从事软硬件协同设计方法学以及快速原型的研究. E-mail: pyp96@seu.edu.cn



凌明 男,东南大学电子工程系讲师,目前在东南大学国家专用集成电路系统工程技术研究中心从事嵌入式软件开发以及嵌入式系统内存管理的研究工作.

2005 年全国开放式分布与并行计算学术会议 征文通知

由中国计算机学会开放系统专业委员会主办、上海大学计算机学院承办、上海计算机学会协办的“2005 年全国开放式分布与并行计算学术会议”将于 2005 年 10 月 27—29 日在上海召开,有关信息如下:

一、征文范围(论文范围包括但不限于下列方面):

- 开放式分布与并行计算模型及体系结构;
- 下一代开放式网络、数据通信、网络与信息安全、业务管理技术;
- 开放式海量数据存储与 Internet 索引技术,分布与并行数据库及数据/ Web 挖掘技术;
- 开放式机群计算、网格计算、Web 服务、P2P 网络及中间件技术;
- 开放式移动计算、自组网与移动代理技术;
- 分布式人工智能、多代理与决策支持技术;
- 分布与并行计算算法及其在科学与工程中的应用;
- 开放式虚拟现实技术与分布式仿真;
- 开放式多媒体技术,包括媒体压缩、内容分送、缓存代理、服务发现与管理技术.

二、征文要求:

论文必须是未正式发表的、或者未正式等待刊发的研究成果. 论文格式仿照《计算机应用与软件》刊物的格式,应包括

题目、作者、所属单位、摘要、关键词、正文和参考文献. 论文中、英文均可,一般不超过 5000 字,一律用 Word2002 格式排版,提供 A4 激光打印稿一式两份,并将论文电子版上传到会议网站上或发送 Email 至 bfzhang@staff.shu.edu.cn. 经程序委员会审查合格的论文,将收入论文集,在中国计算机学会会刊《计算机应用与软件》上发表.

三、重要日期:

会议时间:2005 年 10 月 27~29 日

截稿日期:2005 年 7 月 15 日

录用通知:2005 年 7 月 30 日

四、联系方式:

投稿地址:上海延长路 149 号上海大学计算机学院

缪淮扣 收(请在信封上注明 DPCS2005)

邮政编码:200072

电话:021-56331669

电子邮件:bfzhang@staff.shu.edu.cn(请在邮件主题中注明 DPCS2005)

会议主页:<http://www.cs.shu.edu.cn/DPCS2005> 可查询进一步的会议信息.

专委会联系人:南京大学计算机系 陈贵海,联系电话:025-58916715, 电子邮件:gchen@nju.edu.cn.