

489123

微处理器 Microprocessor 与 微计算机 Microcomputer

第十辑

(多微处理器系统 (MMS) 设计研究)



兵器工业部第二〇七研究所

489123



90030845

目 录

前 言	1
1.0 MMS 分析	3
1.1 目 的	3
1.2 结论预示	3
1.3 五个典型步骤	3
2.0 MMS 系统结构	25
2.1 作为一个系统的MMS	25
2.2 MMS 设想模型与目前概念性设计之间的差别	25
2.3 MMS 总线——一种统一的途径	27
2.4 S B S 协议的说明	28
2.5 MMS 使用的 S B S 结构的体系描述	30
2.6 MMS 总线结构的功能描述	30
2.7 MMS 总线结构的操作要求和物理要求	32
2.8 S B S 如何满足MMS 的物理要求和操作要求	33
2.9 另一种总线结构	33
2.10 S B S 的MMS 接口的详细说明	34
3.0 处理部件的初步设计	35
3.1 P E 软件初步设计	36
3.2 处理单元硬件	38
4.0 MMS 中 P E 的时间校准	47



4.1 MMS 中 P E 的时间校准的实现	47
4.2 时间校准控制器	48
4.3 局部伪时间累加器	48
4.4 时间校准的作用	49
5.0 I/O 仿真初步设计	50
5.1 概 述	51
5.2 系统软件目标	51
5.3 IOP 接口	51
5.4 P E 执行程序支援	52
5.5 I/O 仿真码举例	52
6.0 共享资源控制器	56
6.1 任务检查	56
6.2 固体(微程序存贮器)使用情况	57
7.0 性能监视子系统(PMS)	58
7.1 PMS 特点	58
7.2 实 现	60
7.3 提 要	61
8.0 初步的机械设计	61
8.1 FCP	62
8.2 MMS	62
10.0 MMS 功能介绍	64
10.1 术语表	66
10.2 功能—主层序级	67
10.3 功能介绍—程序段级	73

10.4	功能介绍—P E级	73
10.5	第10章使用的缩写词	78
11.0	总论—M M S 软件设计	80
11.1	第11章使用的缩写词	83
12.0	M M S 诊断程序	83
12.1	第12章使用的缩写词	88
13.0	E 代码编译程序	89
13.1	第13章使用的缩写词	89
14.0	微代码汇编程序	90
14.1	第14章使用的缩写词	97
15.0	P E 执行程序	98
15.1	第15章使用的缩写词	106
16.0	系统生成装入程序	107
16.1	第16章使用的缩写词	115
17.0	运行时间执行程序	115
17.1	第17章使用的缩写词	134
18.0	算后转储 (P M D)	135
19.0	P M S 脱机分析	138
19.1	数据显示	139
19.2	统计分析	140
19.3	线拟合	142
19.4	通用程序	143
19.5	参考文献	143
19.6	自顶向下的设计	143

19.7 第19章使用的缩写词	146
20.0 目标系统	146
20.1 仿真程序模块	146
20.2 拟用的目标系统	153
20.3 P D P 11/45的仿真	154
20.4 I N T E L 8080的仿真	157
20.5 寄存器仿真	161
20.6 第20章所使用的缩写词	162
21.0 用户情况	163
21.1 “辅助”文件	166
21.2 S G L 系统生成装入程序	168
21.3 R T E (运行时执行程序)	179
21.4 P M D (算后转储)	183
21.5 P M S (P M S 脱机分析)	183
21.6 第21章使用的缩写词	183
22.0 F C P 说明	184
23.0 几个一般的论题	185
23.1 同步总线结构	186
23.2 I O P 执行程序	187
23.3 P M S I 执行程序	187
23.4 S R C 执行程序	188
23.5 第23章使用的缩写词	189
附录A 性能监控子系统	190
1.0 P M S 目标/特点	190

1.1 PMS特点	190
1.2 用户交互	191
1.3 开 销	191
2.0 PMS设备	191
2.1 微编码结构	191
2.2 硬件要求	193
2.3 存贮器接口处理的 PMS 检测和共享资源控制器...	194
2.4 开 销	191
2.5 用户交互	194
3.0 PMS设备摘要	195
附录 B 处理器在分析中的定义.....	196
附录 C 参数估计.....	196
附录 D 基本的计算.....	197
附录 E MMS结构的定义.....	198
附录 F MMS设备.....	198
附录 G 缩写词汇.....	199
附录 H	200
1.0 为降低成本改进的MMS的设计	200
1.1 重新设计的目的及其在MMS上的效果	200
1.2 设计改进	200
附录 J MMS硬件规范.....	204
1.0 范 围.....	204
2.0 可引用文件	204
3.0 要 求.....	205

3.1 系统定义	205
3.2 特 性.....	211
3.3 设计与制造	214
3.4 文 件.....	215
3.5 后勤保障	215
3.6 人员及训练	215
4.0 质量保证措施	220
4.1 概 述.....	220
4.2 质量一致性检查	221
5.0 准备交货	221
6.1 附录A	221
附录K MMS 软件规格.....	222
1.0 范 围.....	222
1.1 鉴 定.....	222
1.2 功能概述	222
2.0 可应用文件	223
3.0 条 件.....	223
3.1 计算机程序限定	223
3.2 详细的功能条件	225
3.3 适应性	241
4.0 质量保证措施	241
4.1.1 计算机子程序测试	241
4.1.2 计算机程序测试	241
4.1.3 计算机程序验收试验 (C P A T)	241

多微处理器系统设计要求研究

前 言

在本MMS的最终报告中提出的多微处理器系统(MMS)的概念性设计,是两年来按罗马航空发展中心规定的要求进行研究的成果。这份最终报告不打算将最初的要求及对MMS的约束条件或MMS所经历的设计的反复作一一地介绍,只是详细描述一下MMS所选用的体系结构、MMS的能力、MMS部件以及这些部件是如何相互作用的。

读者要想全面地理解这份有关MMS的最终报告,就必须对促使MMS发展的原因以及MMS发展起来之后的一些概念有所了解。

为了了解多处理器系统的性能和操作,我们构思并设计了这个MMS。具体地说,这个系统已被设计成为实时多处理器系统的系统设计工具。

要达到这个目的,MMS必须具备:

- (1) 为进行仿真而设置的一套处理部件;
- (2) 一个广泛的机内性能监视子系统;
- (3) 一个保持仿真器时间协调的时间校准控制机构。

此MMS设计要满足的全部要求均包含在1978年10月1日颁布的“SAEF(系统结构评价装置)中间产品性能技术要求”文件里。读者审阅这份报告都应以该文件为背景材料。

MMS的最终报告分四部分。这四部分提供MMS的概念性设计和详细说明。第一部分是MMS硬件的概念性设计。第二部分是MMS软件的概念性设计。第三部分除对硬件和软件进行详细说明之外,还包括第一、第二部分里所注明的附录。第四部分包括MMS硬件和软件的详细说明。

硬件的概念性设计在第一部分里分为八节介绍。

第1节包括MMS和目标系统的分析。作为研制MMS基础的许多概念和要求都是这种分析的结果。

第2节包括MMS的系统结构体系,所研究的问题包括如总线结构、广播通讯,将MMS的各个部件与MMS总线相接的各种接口。

第3节提供MMS使用的64个处理部分的硬件的概念性设计。

第4节叙述MMS的时间校准机构如何实现保持仿真过程时间相关这一功能的。

第5节涉及执行目标系统的输入/输出功能仿真的过程。

第6节讨论如何保持被仿真的共享资源的控制问题。

第7节讨论如何执行MMS性能监视机构的功能。

第8节论述MMS的机械配置和构造。

在第一部分的这八节中所注明的任何附录都可在第三部分找到。MMS硬件的详细说明在第四部分。

10~23节论述软件的概念性设计，它的构成在第二部分。

第10节是MMS硬件的功能描述。这一节指出硬件支援所需要的软件/固件。

第11节叙述的是与MMS硬件和固件接口的软件包。这一节是MMS软件的引言。

第12节论述对测试和检验硬件操作能力的诊断程序的要求。

第13节指出对E代码编译程序的要求。

第14节叙述写入和开发仿真包所要使用的微码汇编程序。

第15节介绍MMS结构的所有PE所需的执行程序。

第16节介绍作为一组仿真包与用户接口构成用户硬件设计的系统生成装入程序。

第17节介绍执行规定的TS与用户接口的实时执行程序。

第18节介绍算后转贮软件包以及用户如何利用该软件包来研究停机时TS的状态。

第19节介绍PMS脱机分析软件包，以及用户如何利用这个包来分析所收集的性能数据。

第20节介绍MMS应用的目标系统的要求和范围。它将检验MMS的操作，并为MMS未来的发展提供一个例证。

第21节介绍用户情况 (User Scenario) 及在执行SGL和RTE过程中，与MMS接口的方法问题。

第22节介绍FCP计算机及与若干台小型计算机的比较评定。

第23节讨论四个一般论题。第一个论题是MMS总线结构的识别问题。其余3个论题是主区段各单元所需的下列执行程序：

- I/O处理器的执行程序；
- PMS接口的执行程序；
- 共享资源控制器的执行程序。

评 价

依照本合同所进行的这项工作，透彻地论证了产生一个通用的、面向仿真的工具来对多处理器计算机结构进行建模和评价的可行性。合同承包商以敏锐的技术洞察力，设计了一个高性能的面向用户的多微处理器系统 (MMS)。这一成果，对罗马航空发展中心 (RADDC) 广泛地调查研究整个系统设计 (TSD) 方法，为系统设计师提供更有有效的开发工具方面，是一个成功的开端 (TPOR3D, FY78/79)。在今后继续进行的工作中，本合同的成果将直接用于实际地产生一个进行工作的MMS。

1.0 MMS 分析

本节主要是叙述分析工作的目的和简要结论。分析工作包括五个不同的步骤。这些步骤将按顺序加以叙述。每一步骤又分为目的、输入、方法、结果和结论五个部分说明。最后一个步骤是分析的结果和结论，它最长而且也最复杂。

1.1 目的

分析的首要目的是对设计可行性提供保证并对本项工作初期阶段设计范围进行衡量。在开始阶段，要从成本和性能的角度来评价设计。然后指出设计应遵循的方向。MMS分析通过使系统性能灵敏度与MMS中每个部件联系起来，而有助于对系统中主要部件进行技术说明。这是权衡系统性能和成本两方面的一个有效的方法。系统分析可以作为检验MMS结构构思的一种手段，有时它还可提出一些能增强MMS性能的结构方法。最后的目的是表明哪个目标系统可逼真的被仿真，并估计这些系统相应的仿真效率。这样，分析就可以作为说明MMS对任一特定目标系统仿真效率的工具。

1.2 结论预示

下面的各点是由这个分析业已证明的简单设想：

- 1.2.1 并行存贮器总线。
- 1.2.2 单系统总线。
- 1.2.3 存贮器的配置：分布式和局部式。
- 1.2.4 分布式I/O的必要性。
- 1.2.5 可编微程序的肖特基逻辑选择。
- 1.2.6 对于8位、16位、32位和64位处理器，EE范围分别为1.8、3.7、11.5、17（EE=仿真效率）。
- 1.2.7 对于MMS总线来说500万次/秒的传输速度是足够的。
- 1.2.8 总线上数据字长为16或32位。
- 1.2.9 局部存贮器具有由字节寻址或字寻址的128k字节。
- 1.2.10 ALU字长为16位。
- 1.2.11 共享资源控制器将是可编程序肖特基逻辑。

1.3 五个典型步骤

1.3.1 最坏情况分析

1.3.1.1 目的

最坏情况分析的目的是取得一个总信息传送速率的初始估计。

1.3.1.2 输入

四个参数被用作此分析步骤的输入。这些参数及其最坏情况值是NP, 处理器的个数为64; BS, 处理器的位数为64; IMA, 对于一个64位处理器, 每条指令的访问存贮器的次数为1.36; 以及NIPS, 一个64位处理器每秒执行指令数1.5兆。

1.3.1.3 方法

计算传送速率的方法是这些输入参数的连乘, 通过 $NP \times BS \times IMA \times NIPS$ 得到串行总线结构, 存贮器存取的传送速率是每秒8.4位。

1.3.1.4 结果

结果只表示原始的存贮器传送, 不包括与信息格式有关的开销, 对于一一对应的仿真效率而言, 需要8.3千兆赫的速率。目前的技术状况, 实现这一速率是不现实的。即使设计标准取100:1仿真延迟这一最低点, 这个速率也要达到83兆赫。

1.3.1.5 结论

根据这些结果得到的初步结论是, 由于速度问题, 串行存贮总线是不能采用的。因此, 选择了并行总线。

1.3.2 目标系统分析

1.3.2.1 目的

为了确定MMS系统的技术条件, 必需了解要仿真的目标系统的规模和种类。为了使目标系统性能定量参数化, 还必须对通信结构和速率进行分析。直接将大量的现有目标系统硬件转换成MMS硬件的固定设备是不可能的。然而, 可将每个目标系统的物理转换、归类或者归纳为一些小的功能转换组。通过转移总速率, 这些功能性转移可以分配于实际的MMS网络中。因此, 这些总的平均速率是重要的。

1.3.2.2 输入

目标系统分析的输入是一组统计参数。这些参数及其估值分别列于附录B和附录C。

1.3.2.3 方法

多处理器网络的检测是以一个处理器为基础来完成的。通过对指令的检测来细致地分析每个处理器。通过对指令混合指数, 特别是对DIS MIX的研究, 可以得到一条平均指令的存贮器存取次数的估算。采用目前流行的微处理器及计算机上已应用的DIS MIX, 只要分别确定了不同机器的位长, 就要决定平均数有效值。尽管指令和指令格式可能很不相同, 在同样位长的机器中, 存贮器存取量相同。现在讨论的存贮器的存取参数称作存贮器存取指数(IMA)。军用计算机由数字集成子系统混合而成, 它按下表来完成计算任务。

类 型	举 例	重 量	机 器 字 长			
			8	16	32	64
取 操 作 数	取 数 、 存 数	36%	4	3	2	2
长 指 令	转 移	30%	3	2	1	1
短 指 令	寄 存 器 操 作	34%	1	1	1	1

结果如下:

IMA (8) =2.68

IMA (16) =2.02

IMA (32) =1.36

IMA (64) =1.36

描述目标系统中处理器的其它许多参数,可根据这些参数的作用频度来选择,而作用频度与执行的指令条数有关。这些参数可用于提供一种原理简单的方法来观察处理器的性能。这种分析的目的之一就是要得到各种功能转换的目标系统总速率。描述有关输入/输出(I/O)、处理器间通信以及共享资源请求的频度的一些参数在达到这个目的中是非常有用的。这些参数的完整表格及其定义以及这些参数的选定值列于附录B和C。

四种操作速率一要求在多处理器系统部件中间进行与/或控制转移的目标系统中有四个逻辑操作。这些操作是:存贮、输入/输出、处理器间通信、以及共享资源控制逻辑。对于最后转变成给定的MMS结构确定这些操作的总速率。这些总速率根据执行时间的每秒传送次数来确定。这里不规定传输的宽度。任何时间传输的数据,不管它在目标系统中并行传送的位数多少,每次传输都要加入总的计数。

基本运算一建立模型以便为获得每种逻辑速率必要进行一次基本运算。选择这些参数,它们将一种操作频率表示成测得的与指令有关的量。这些参数称作指数,用“I”表示,并放在其它参数标志之前。这些参数的缩写为IMA、IIO、IIPC和ISRR(见附录B的定义)。每个参数表示每个已执行指令操作出现一次。另外一个参数是NIPS,代表每秒执行指令的平均数。NIPS也称作KOP(千次运算/秒)由指令混合指数确定的处理器额定值。将NIPS和系统中所有处理器所总计的每个指数相乘,就得到了一个操作的总速率。决定总速率的方程列于附录D。

每个参数值在统计范围内决定。根据对于每个参数所规定的分布,随机地选择那个参数的一个值,就产生了样本目标系统。成千上万的目标系统被采样,并且标出四种操作速率的每个直方图。每个直方图都标有三个点,以表示结果的精确图形。第一点“A”代表这种传送速率,以这个速率寻找有较少传送器的目标系统的概率为0.1。“B”点代表中等传送速率。最后,“C”点,代表这样的传送速率,以此速率寻求具有较少的转移次数的概率为0.9。

1.3.2.4 结 果

样本直方图示于图1.3.2.4。标有A、B和C各点分别代表上节讨论的10%、50%和

90%的值。对每个操作速率所计算的“A”、“B”、“C” 传送速率列于表1.3.2.4。

目 标 系 统 结 果

存贮器传送速率 (MTR)	A	B	C
共享资源请求速率 (SRRR)	77.7K	320.4K	679.6K
I/O 传送速率 (IOTR)	130.2K	269.0K	468.8K
处理器间的信息速率 (IMR)	12.1K	24.9K	45.8K

(单位：传送次数/秒)

1.3.2.5 结 论

与存贮器存取有关的大的传送速率进一步确认了宽并行总线结构。从理论上讲，这一速度代表了必须由MMS 总线结构支援的存贮器传送的数量。假定总线结构的宽度能满足用户要求，单个地逻辑传送最多的位数。

存贮器传送速率的倒数表明大约30毫微秒时间总线就应将一个请求送到一个存贮器内，访问一遍这个存贮器，并送出一个应答信号到这个单元。这种要求显然是超出了现有技术水平。这里介绍两种可能减少MMS总线结构带宽的设计方案。

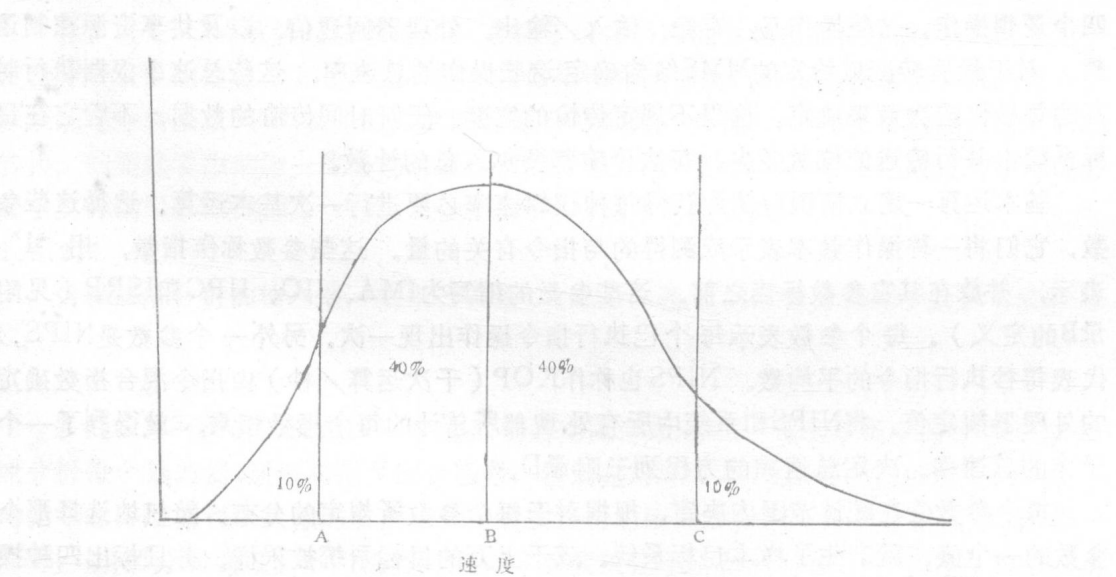


图1.3.2.4 出现的频率

第一，由于在PE中应用了局部存贮器，使得在大量存贮请求的存贮器存取通道中取消了总线。第二，这些局部存贮器作为独立的存贮模块存在，它们可以作为独立的智能功能模块通过存贮总线进行存取，并允许总线以与存贮周期无关的方式进行工作。这样就减少了总线的周期时间。

目标系统分析的显著结果是和I/O所计算的速率、处理器间通信速率、以及共享资源请求速率之间的差别很大。传送速率中的差别表明实现MMS功能总线有两种可能方案。第一,是对I/O和SRR(共享资源)通信使用较低带宽的并行总线,而对于IPC(处理器间传送)则使用串行总线。第二,是在单个总线上完成所有的通信任务,其总的传送速率约等于存贮器的传送速率。

当使用一些独立的物理总线来完成与每项相应功能有关的总速率时,性能会因此而降低。常常会出现这样一些情况,I/O执行码很繁忙。这时,I/O总线会产生超载并会降低总的系统性能。单总线有助于缓和这一问题。就时间而言,任何一种操作处于应用的高峰时,应正好是其余操作的稀疏应用时间。这些操作速度与其总的平均值的差异将会相互抵消。使用统一总线就不会使性能降低。

可能导致性能降低的第二个原因是总平均方法的易误解性。例如,在SRC总速率上独立总线不足以满足系统同步的速度要求。在目标系统中,共享资源裁决进行极快。然而,裁决频率不一定像实际裁决发生的那样快。总速率代表这一频率。对整个MMS中的同步来说,所完成的裁决速度应接近于目标系统动作的裁决速度。专门的SRC总线不一定很快。也可以论证以说明专门的SRC总线可以较高的速率来实现。但这一速率只会等于或超过所提出的统一总线的速度。由于与四个独立的总线相比,这个统一总线较接近于所要求的速率,因而,是一个较好的方法。所以建议采用统一的总线结构。

除处理器间信息传送速率外,数值的10%和90%中的变化不代表传送速率的一个数量级的变化。这说明两个主要问题,第一,目标系统中存在足够的一致性。这样可以保证建立一个硬件系统来仿真各种各样的目标系统。第二是可实现性,不是所有可以仿真的目标系统都由统计分析而产生的。

1.3.3 原始MMS模型

1.3.3.1 目的

建立原始MMS模型的目的是用来初步估算MMS通信速率。

1.3.3.2 输入

原始MMS模型的开发,是通过修改目标系统模型实现的。修改后的目标系统模型用来决定完成一个目标系统——对应性能仿真所需的MMS传送速率。

基本上有两种输入到目标系统分析的新参数。第一种参数是局部存贮容量(LMS)。LMS参数用来确定与MMS中每个PE有关的总存贮位数。第二种参数是总线的宽度(BWD)。BWD参数是用来确定MMS总线结构的数据部分的宽度。

这两个参数以及决定目标系统中与每个处理器有关的位长(BS)和存贮空间(MS)的一些参数,都被用来计算局部存贮器系数(LMF)和加权系数(NF)。这两个系数可用来将目标系统存贮器的传送速率变换成MMS的存贮器传送速率。

当PE直接对它本身的局部存贮器进行存取时,不必占用功能性MMS存贮器总线。LMF用来将正在由一个PE直接存取的局部存贮器效率换算成存贮器传送速率(MTR)的运算。LMF是这样计算的:

$$LMF = \frac{(MS)(BS) - LMS}{(MS)(BS)}$$

计算时, 如果 $(MS) \times (BS) < LMS$, $LMF = 1$

定义LMF时, 有如下假定:

1. 除了存贮宏代码所需的存贮空间大于单个局部存贮器的容量之外, PE 总是执行其局部存贮器给出的指令。

2. PE 访问其指定的MMS存贮空间的任何部分的概率相等。

NF将固定的MMS总线宽度的效率换算成MTR进行计算。NF的计算公式为:

$$NF = BS / BWD$$

计算时, 如果 $BS < BWD$, $NF = 1$

定义NF时, 可作如下假定:

1. 当仿真 $BS < BWD$ 的处理器时, 不使用MMS总线的整个带宽。

2. 仿真一个N位处理器, 则需要传送 N / BWD 次的N总位。

下面是计算MMS MTR的公式。

1.3.3.3 方法

$$MTR = \sum_{i=1}^{NP} IMA_i \times NIPS_i \times LMF_i \times NF_i$$

1.3.3.4 结果

已作成MMS的四种不同结构图, 它们对于每个目标系统具有相同的统计变量。共享资源控制器速率、输入/输出传送速率以及处理器间的信息交换速率不受这些变量的影响, 而且不包括分析这部分。新的存贮器传送速率列表如下。注意, 结果按照以上讨论过的直方图的“ABC”格式排列。

LMS	BWD	A	B	C
* 0	∞	17.8M	29.4M	41.7M
16K $\times$ 16	16	13.3M	24.4M	37.8M
16K $\times$ 32	32	6.94M	13.1M	20.0M
64K $\times$ 16	16	1.94M	6.11M	13.1M
64K $\times$ 32	32	46.3K	46.3K	2.69M

* 目标系统速率

1.3.3.5 结论

首先, 这里要强调的一个结论是, 局部存贮器确实可以降低为执行仿真任务而必须由存贮器总线完成的速率。重要的是MMS 传送速率可降低到不超过高速肖特基逻辑电路的极限值。

第二, 使用可以并行存取的小存贮块, 不管它们是否安装在PE 上, 其目的都是为

了获得高的总线吞吐量。随着可接收请求, 进行存取和可请求传输总线的存贮模块的生产, 存贮周期可以与总线周期分开。

通过给每个PE中加入一个存贮块, 从而减少访问该模块的用户数量。当这个存贮块放在一个模块的一个总线上时, 一个PE使用单个存贮模块的等待时间将比实际存贮器的存取时间大好多倍。

借助于最终MMS分析程序, 对MMS总线的存取时间和一个包括16个八位处理器的目标系统的仿真效率进行了计算。对于只使用一个存贮块及其存取时间包括在总线周期内的一个总线结构来说, 存贮器的等待时间为14.6微秒, 系统仿真效率(EE)为15.59比1。对64k局部存贮块而言, 其等待时间为1.67ns, EE为1.79比1。

从这一点进行分析, 决定将注意力集中于并行总线结构上, 此结构大约可达每秒10兆的传送速率。通过检查一个局部存贮容量为64k×16的存贮器传送速率数值的50%, 就确定了这个棒球场式的图形。

1.3.4 仿真效率和部件负载的计算

1.3.4.1 目的

为了对系统中部件的特性作出较好的估计, 必须较好地掌握这些PE相对实时而言的工作速度。这就是所谓的仿真效率(EE), 并由MMS仿真时间与伪时间(在目标系统上完成的实时目标系统运行所花的时间)之比确定。仿真效率从原来的1:1降低到专门计算出的效率, 表明降低了对MMS部件的要求。

在分析中还需要一种方法来确定系统中那个部件对系统中的其它部件正在起瓶颈作用。其目的是为了减少这种瓶颈现象并使系统中所有部件的性能要求达到平衡。

1.3.4.2 输入

分析这个阶段的主要输入是总线及其协议的概念的内涵。当存贮器周期与总线周期无关时, 所用的MMS总线结构是一个同步总线。在一个总线周期中就可把信息从一个装置送到另一装置。如果需要这个装置应答, 该装置的“考虑时间”应同其它总线周期一致。如果已准备好应答, 该装置就应请求总线并象其它装置或处理器那样在一个总线周期内的总线上完成传送任务。这就是说, 每个存贮器在总线上完成读出操作, 存在着请求总线时所导致的两个总线周期和两个等待时间的要求。这两种操作适于具有共享资源控制器的共享资源请求/应答动作, 也适用于必须具备两种通信手段的系统的任何其它动作。在分析中, 假定所有的存贮器存取操作的90%是读操作。这个假定是以所完成的数字超高速缓存器的研制为基础的。

为了确定这个系统中四个部分的能力以及负载, 提出了最大速率和附加参数。假定总线需每秒10兆的最大传送速率, 共享资源控制器每秒300万次请求/应答操作, I/O处理器要每秒万次输入/输出操作。输入/输出操作可划分并参数化为三种基本类型。第一, 有这样的简单的I/O操作, 它们可以完全被控制在PE内处理, 以致于完全不需I/O处理器进行工作。对于称作IIOC的这类I/O, 每条I/O指令赋予一个指数, 输入/输出控制器功能的指数是在PE中。这时, IIOC的估算值为0.2。第二种I/O是通过

总线从软盘或磁带传送到PE的局部存贮器或者某些存贮模块的字组传送。这可能要求I/O处理器进行DMA操作,但不需要处理时间。I/O仿真的实际处理重新在有I/O控制器功能的PE内发生。为了表示这种I/O而选择的参数是IMMIO、存贮器映象的输入/输出的指数。其估计值约为0.5。第三种I/O实质上是环境并要求I/O处理器参加。这几种I/O操作被认为是I/O处理器的重负载。此外,任何由IOWF(I/O权系数)定义的,并认为是共享的I/O,均受I/O处理器操作的支配。IOWF的值是0.95。

系统中需要最高速度的最后一个部件就是PE本身。对一个处理部件能仿真目标系统处理器规模和PE中的ALU的位长的每秒钟指令数进行了估算。下面是以每秒千条指令为单位的估算表。

目 标 系 统 位 长					
MMS ALU位长	8	16	32	64	
16	300	160	120	100	每秒千次运算
32	300	200	200	160	额定值

1.3.4.3 方 法

I/O、IPC、SRR和存贮器的四种操作速率被转换成表示MMS总线、共享资源控制器以及I/O处理器的三种实际速率。这三种实际速率为:

$$BTR=MTR\times1.9+SRRR\times2\ IMR+IOTR$$

$$SRC的速度=SRRR$$

$$IOPR=\sum_{i=1}^{NP} IIO_i\times[1-IOWF_i\times(IIOC_i+IMMIO_i)]\times NIPS_i$$

总线传送速率(BTR)包括所有操作速率的组合,这些操作速率由代表整个事务处理所需总线周期数的权系数进行修正。系数1.9是存贮器读出和读/写操作频率所需的两个总线周期的组合,共享资源控制器速率同以前讨论的共享资源请求速率是一致的。OPR是老的输入/输出传送速率,修改后用来只包含那些由I/O处理器进行的计算操作的I/O传送。

通过BTR、SRRR、IOPR和PE的仿真效率(EE)计算,就可以分三步来决定整个MMS的仿真效率。

第一步是确定每个MMS PE的EE。每个MMS PE的EE是根据被仿真的微处理器(NIPS)指令执行速度与估算的MMS PE每秒千次运算速度之比来决定的。通过所计算的EE比率可以看出MMS PE在执行指令方面是多么慢。

第二步是根据最大延迟比来确定PE。(延迟比是仿真效率的倒数)仿真效率的所有结果都以这种延迟形式表示。这一比率被用作这个系统的第一通道仿真效果。

第三部是决定总线、SRC和IOP上的负载。根据所计算的速度(BTR、STCR和IOPR)与1.3.4.2节中给定的部件所需的最高速度之比来计算每个负载。在检验这些负载时,如果确定所有的负载都小于或等于100%,那么,所求得仿真效率和部件负载就