

# 动态计算机的思想方法

姜咏江

对外经济贸易大学

关键词：动态核 CMP 多核 软件消耗 动态关系总线

## 引言

长期以来，人们将计算机的程序调度和资源管理定格在操作系统的软件方法上。实际上，软件管理计算机的方法为计算机的应用带来了巨大的复杂度，不仅在各种系统软件的编写上耗费了计算机专业人员大量的时间，而且在计算机任务完成的过程中也消耗了大量的机时，因而极大地影响了计算机效率的提高，特别是在片上多处理器计算机系统中。

近几年片上多处理器CMP（Chip Multicore Processor）发展迅速，片上“多核”已经被商家炒得热火朝天，但多核的计算机效率并未像人们想象的那样，获得成倍的增长，其原因主要有：

**多核不能实现完全的并行工作** 因为多核本质上是一个局域网，核间通过网络形式连接，任务的分配和执行是非一致性的，特别是片上多核的结构不能像片外网络那样，每个节点都有任务输入机制。它的任务装填方式是有限的一个或几个端口，故而难以保证各个核能够分配到充分的任务工作。

**多核不能充分发挥芯片结构的特点** 由于高集成度的芯片体积小，线延时极短，并且时钟有较好的“同时性”，非常适合“同步数据传输”。但网络数据传输方式是异步的，需要打包、解析、确定路由等一系列对于同步传输是多余的过程，因而多核会耗费许多时间来进行数据转移传输，而不能很好利用芯片集成结构的优势。

**多核系统管理方式落后** 目前，多核结构中程序执行的调度管理采用软件方式，依据操作系统进程调度或网络协议，频繁地进行任务分配和程序

调度。由于操作系统和协议本身就是需要到处理器上执行的程序，它们的执行不但占用处理器资源，而且还无法实现系统对各核运行的同步管理，这样在“管态”下，就会出现处理器停顿的现象，不可避免地会形成“软件管理瓶颈”。

**多核无法避免“存储墙”问题** 由于多核延续继承了现代操作系统的共享内存的程序存储方式，因而对内存的要求越来越大，而访存带宽不能随着核的增多而增加，反而使处理器对存储器访问时间加长的矛盾愈加突出，虽然采用多级缓存结构勉强可以保证正常运转，但增加缓存所带来的时间消耗，抵消了多核并行工作提高效率。这种“存储墙”问题一直伴随多核结构。

**多核结构并行编程困难** 由于多核在片上采用的是网络结构，而网络结构不适合统一编写并行程序，这样就加重了软件编程人员的负担。软件人员惊呼：“免费的午餐没有了！”

超大规模集成电路技术的发展，不仅在方寸之间汇集了大量的硬件资源，而且为信息的高效传输、硬件逻辑设计等都提供了极为方便的条件。CMP架构设计能否充分利用集成芯片的特点，将是衡量其好坏的重要标志。国内率先发展起来的动态计算机核心设计，不仅在CMP核心架构上有了很大的改造，而且在多核引发的一系列问题上有望找到化解的方法。

## 动态核的思想

所谓动态计算机是指在任务执行过程中，计



序地调度多个程序并发执行。

$n$ 个MU是有优先顺序的，因而与PU的连接不会发生冲突。高层逻辑可以依据MU的连接请求信号，PU的空闲信号来发出连接控制信号，必要时还可以辅助限时执行的方法来保证程序任务的均衡完成。这种均衡性的保证还可以配备硬件“限位数”计算，从而使MU整体具有“循环优先级”的特点，保证优先级方式不出现“服务沉淀”。

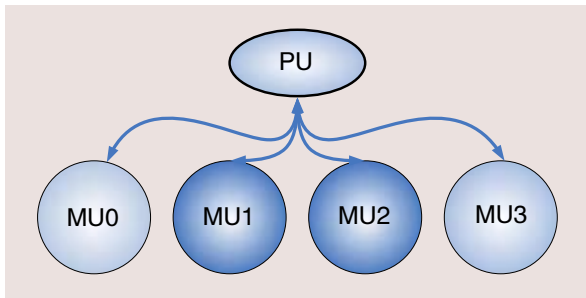


图3 多程序动态执行结构

假设图3的PU- $n$ .MU结构的每个MU中有一个程序，那么必然是一个程序执行时，另外的几个程序都在等待，这就是单处理器计算机的特色。由于MU中都具有其中程序执行的环境，因而在与PU交换连接时，不用像进程运行那样要访问进程控制块，施行“现场保护”和“现场恢复”，这样就可以节省软件调度所需要的大量时间。这种硬件线路的切换一般在一个机器节拍中就可以完成，因而程序的中断或继续执行也是在一个节拍之内就能转化完成。

如果每个MU中有固定的存储器容量，那么整个计算机存储容量的大小是由MU的数量来决定的。由于程序执行时访存的时间只由单一的MU来决定，因此我们可以将MU中存储器的容量，限制在与处理器速度要求匹配的条件下，这样就不会产生“存储墙”问题了。

MU的存储容量的限制并不会影响超大程序的执行，我们仍然可以用虚拟存储的方式，分段将程序送到MU中执行。

## 多程序同时执行结构

若要真正实现多程序“同时”并行执行，就必须有多个处理器存在。为了使任务程序能够快速完成，一般每个任务程序中又可以分成若干个可独立执行的线程，当某个线程无法执行时，另外的线程能快速启动执行。图4是多处理器与多程序同时执行计算机核结构。

由于每个MU中放置一个程序，而程序又以线程的方式划分，那么MU中要为每个线程设置一个执行环境。也就是说，线程的地址计数器和其他环境设备，在执行时会成为当前执行的程序计数器和环境条件。MU内部线程执行也应该采用线路切换机制，快速实现线程交换，即被称为“同时多线程”的机制。显然这种线程切换适合“较稳定”的PU-MU连接方式，如果频繁实现线程在多种异构处理器间转换，那么必然带动任务程序在相应处理器间转换。因而MU中过多地设置“硬线程”未必是最佳选择。不过由于无论是在MU中还是在MU之外，线路切换的速度都是一样的，因此不会产生效率降低的问题。

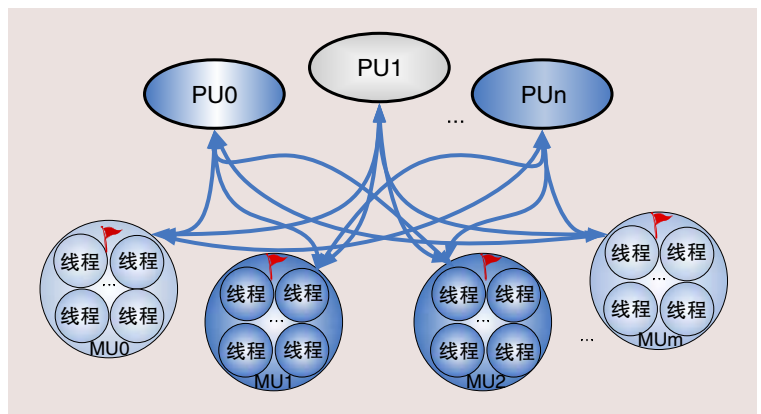


图4  $n$ .PU- $m$ .MU多处理器计算机核

## 如何控制PU与MU的连接与断开

如果有 $k$ 种不同类型的处理器PU，那么在MU中要有一个 $k$ 位的连接标志寄存器，每个标志位对应一类PU。当某连接标志位是1时，表示该MU要求与

那一类PU连接，此时若该类PU有空闲的，那么高层逻辑控制就会立即按优先级实现连接，否则MU就处于等待状态。PU一方是靠“闲标志”来与MU连接的，如果PU正忙，那么需要与之连接的MU就会仍然处于等待状态。

根据计算机任务的需求，处理器的功能将被细划，这样可以提高执行任务的效率。图5是多种PU与MU动态连接示意图，其中CPU是程序执行处理器，T是程序通信处理器，NET是网络处理器，CHL是输入输出处理器。空心圆表示MU中的程序正在执行，实心圆表示MU中的程序正在等待执行或其中没有程序。

### 动态核心系统运行概述

如图5所示，系统初始，除了固化有程序的MU之外，所有的空MU都将请求连接标志指向CHL，如果没有外部任务程序请求执行，CHL会拒绝与这些MU连接。外设与MU的连接也是动态的，如

果外设中有任务程序请求执行，空闲的CHL将与之连接，并且立即与空闲的MU连接，将任务传输给MU，同时将MU中的程序上下文寄存器填写好，以备后继程序段能够连续执行。装填好程序段的MU与CHL断开连接后，将连接申请标志将指向CPU，进入程序动态执行。

MU之间会依据一定的优先顺序与各处理器连接，各类处理器也有连接优先顺序，以避免连接冲突。装载了程序段的MU会依据程序执行需要，设立标志，凭借标志自动寻找处理器连接或断开。在转换或完成程序段执行时，通过指令填写好应该连接的处理器标志，以达到选择处理器或周而复始执行程序的过程。

MU的存储容量是有限的，超长的程序需要分段进入MU执行。为不使任务错位，保证上下文共享数据资源，编译程序要在每个装填段的末尾添加设置连接CHL标志指令，并用保存在MU中的程序段编号来控制次序，在全部程序段执行完成后，清空程序上下文寄存器，申请指向CHL，以便新的程序能够进入MU执行。

由于处理器PU无痕的特点，在同类型处理器中，MU与任何一个处理器连接都可以执行程序，这样可以充分调动全体处理器工作，最大限度地实现程序同时执行。动态核心中同时执行的多个程序，不必刻意去编写专门的并行程序，普通传统编程方法得到的程序，都可以投入程序同时并行执行状态，因而这种程序并行执行可以称为超并行程序执行。同一程序的线程要同处于一个MU中，通过环境转换，实现“同时多线程”运行。

相互关联的多个程序装入不同的MU中执行时，如果它们之间需要数据通信，可以通过指令将地址信息填入MU的通信地址寄存器保存，

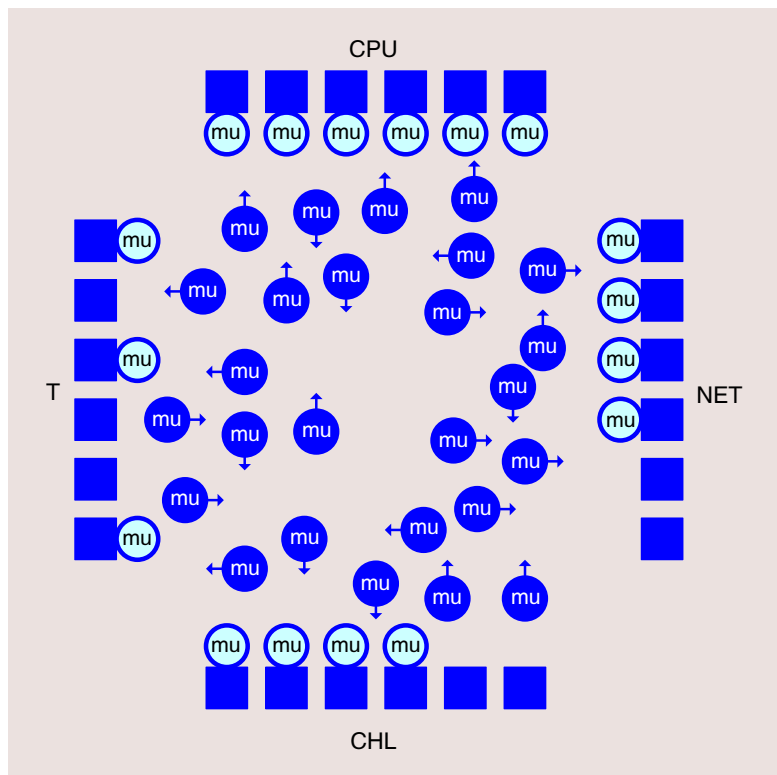


图5 PU与MU动态连接示意图

然后转向与通信处理器T连接。发信的MU在与T连接后，通过T将发送地址和数据送到邮箱。接收数据的MU再通过T访问邮箱的信件，如果邮箱没有需要的信件，相隔一定时间，可以再次通过T查找。

如图6所示，邮箱、外部设备和网络端口设备都可以有多个，不执行程序任务时，它们与处理器总是断开的，只有程序任务需要连接这些处理器时，才会根据需要连接，并同时设置好忙标志，以防连接冲突。输入输出、通信、连网，一般要同时实现三种连接：MU与处理器连接；MU与使用的其他设备连接；处理器与这个其他设备连接。动态核结构要求，只要有一个连接不成立，就立即断开其他两方的连接。

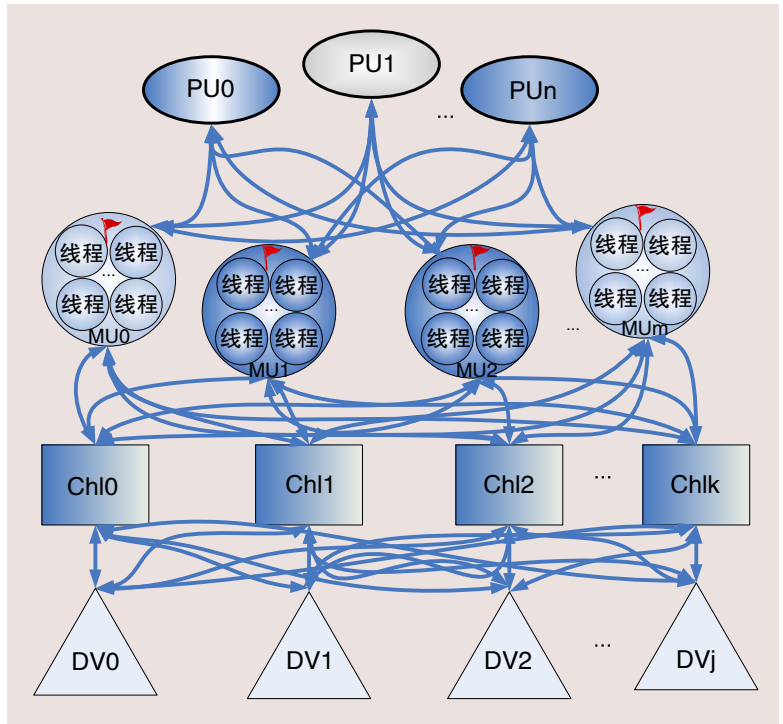


图6 动态计算机关系总线结构

## 动态关系总线

图6是包含多种类型处理器的动态计算机连接结构图， $PU_i$ 是单端连接的一类程序执行单位，它只连接MU； $Chl_i$ 是一类双端连接的执行单位， $Chl_i$ 的一端要连接MU，另一端和外设DV连接。每个不同种类的器件之间，都用总线连接，其中包括控制总线。这些总线都有可控开关控制通断。如果原来的两条或两条以上的总线相连，那么就组成了一条总线（见图7），也就是说不论何种情况，各设备都只能在这条总线上传输数据，而不能同时占用相互冲突的多条总线工作。我们称这种总线结构为“动态关系总线”。动态关系总线不分内部总线和外部总线，它是内外一致的总线。

## 核重组与执行核

动态计算机中的动态关系总线形似交叉开关，但实际上是计算机的“核重组”结构，与数据传输中的交叉开关线路结构有本质的不同。不论是单端

连接的程序执行单位PU还是双端连接的程序执行单位PU，都有自己的指令系统，它们都必须与MU连接才能够执行程序。这种处理器PU与MU连接之后才会形成能够执行程序的核，我们称之为“执行核”。动态计算机中，同时可以形成许多执行核，执行核还会随着程序执行的需要，不断地进行核重组。MU与单端处理器连接时，其他与MU连接的总线都会处于断开状态。当MU与双端连接处理器Chl的总线连接时，双端连接处理器Chl必须能够找到合适的外设，并与之连接，形成统一的总线结构（见图7），否则就断开与MU连接的所有总线。这样做的好处是设备不会在连接的状态下不工作，从而保证不出现死锁。

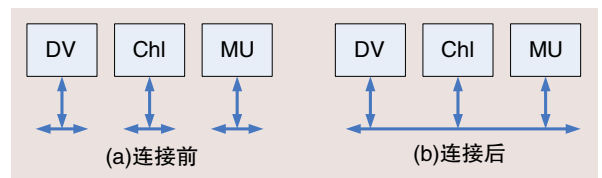


图7 动态关系总线连接前后

## 高层控制逻辑

在动态计算机中,所有高端设备的相互作用都是通过信号标志线来表达的。由于这些设备处于同一台机器,不会发生像网络那样的资源随机性,因而可以用信号通过逻辑电路来发出各种连接断开的控制。由于实现MU与PU的连接之后,程序的执行过程由“重组核”中的控制器来管理,所以程序执行无需高层控制逻辑干涉。

核内控制器与高层控制逻辑的分工使动态计算机既能像以往的计算机那样编写和执行程序,又能够不用软件进行程序调度和处理机管理。这种结构突破了现在各种非动态计算机最低层缓存(Cache)私有的界限,摆脱了操作系统管理程序、处理器、内存和数据分配等一系列模式。如同飞机从螺旋桨推动转到喷气推动一样,动态计算机结构效率的提高,是无法用原有的软件管理方式进行比拟的。

如果将操作系统软件管理方式比作“人治”的管理方法,那么动态计算机的硬件逻辑管理方式就是“法治”的产物。人治,服务元素和服务对象都是被动的,要靠能力强的管理者指挥来动;而法治,能够让服务元素和服务对象都能依据“法规”,主动地去完成任务,实现服务。

由于动态计算机中各种高层次的设备PU、MU和DV等都是靠标志信号来联络的,因而不论程序是否处于执行状态,都能接受来自高层逻辑的调度管理。高层逻辑是一个与所有设备同时工作的管理系统,不用像操作系统那样,在“管态”才能发挥作用,因而不论程序在何种状态,都能实现对硬件设备的管理。高层逻辑电路的存在及工作方式,不仅仅是一个“硬件加速”问题,更重要的是从根本上改变了复杂落后的软件管理方式,加快了整个计算机系统完成任务的进度。

## 动态计算机的应用与发展

动态计算机并不是一种全新的计算机思想产物,而是冯·诺依曼思想体系在多处理器计算机实践发展的一种解决方案。如果把以操作系统为标志的,以软件来管理程序并行执行的方法,看作是冯·诺依曼思想体系实现的一种软件解决方案,那么动态计算机相对操作系统来说,就是一种简约的、高效的硬件解决方案。

## 单片机领域

通用计算机人机交互是一个极其复杂的开发领

## 第27届中国数据库学术会议征文

由中国计算机学会数据库专业委员会主办,中国人民大学、北京大学、清华大学承办的第27届全国数据库学术会议(NDBC 2010)将于2010年10月14日至17日在北京举行。

本次会议录用论文将分A辑和B辑出版,其中A辑论文拟安排在《计算机学报(正刊)》、《计算机科学与探索(正刊)》出版;B辑论文拟安排《计算机研究与发展(增刊)》出版。会议将评选“萨师焯优秀研究生论文”和“最佳系统演示报告”。

**征文范围** (详见网站<http://ndbc2010.ruc.edu.cn>)

**投稿要求** 投稿网址: <http://www.easychair.org/conferences/?conf=ndbc2010>  
会议网站<http://ndbc2010.ruc.edu.cn>

**重要日期** 论文提交截止时间: 2010年5月1日 论文录用通知时间: 2010年6月10日  
排版稿件截止时间: 2010年6月25日 提交出版社时间: 2010年7月1日

**联系信息** 电话: 010-62512334 传真: 010-62512334 邮件: [ndbc10@gmail.com](mailto:ndbc10@gmail.com)

域，而单片机的人机交互要相对简单，为此，动态计算机核心技术首先应用的领域必然是单片机。

单片机系统最重要的部件是微处理器MCU或者数字信号处理器（Digital Signal Processor, DSP），从本质上来讲它们都是CPU。由于单片机外部设备简单，需要执行的程序往往固化在存储器当中，因而它们是一个计算机的核心。由于执行程序固定，所以单片机有时就成为专用计算机的别名。

虽然单片机的功能简单，但在处理速度上要求很高，因而同时并行操作是单片机努力追求的行为，而简单直接又是其提速的一个重要方法。

虽然单片机可以看成专用计算机，但随着手机和机器人等高性能专用计算机系统的开发，其复杂程度也越来越高，功能和行为的处理单凭一个CPU已经跟不上高速高效的要求，因而异构多处理器结构必然也要成为单片机的核心架构。动态计算机核心正是这种多元异构计算机核心的体现，特别是片上动态计算机核心，不仅结构灵活，而且简单高效。动态关系总线结构异常方便多功能协调，能充分调动每一个处理器工作，因而动态计算机会在单片机领域率先显示出其强大优势。

## 通用计算机领域

通用计算机是能够运行各种应用程序的计算机。与单片机相比，由于通用计算机的功能具有复杂性和多样性，因此它在资源管理、任务调度和人机交互等方面都有很高的要求，主要包括3点：第一，要解决各种各样的程序都能够执行；第二，要能更多地处理输入输出事务；第三，要能灵活处理程序对处理器使用和资源的调度分配。

通用计算机的这些要求无疑会增加处理器功能。以往的作法是设计功能更加强大的处理器，让所有的程序都能够在这个通用处理器上执行。这种作法虽然可以满足各类程序执行的需要，但由于处理器对程序的互斥使用，造成了处理器内部功能资源的闲置。而在处理器外部，由于各程序要排队互斥使用处理器，又造成了程序执行等待的时间浪费。这两种情况的发生都对通用计算机的核心结构

提出了要求。首先，计算机核心中处理器要向功能单一的方向细化，要尽可能地在程序执行的过程中，减少处理器内部资源闲置；其次，要增加程序调度分配的灵活性，减少程序排队等待执行的时间。前者必然会推出异构的多处理器的计算机核心，而后者必然要打破共享庞大内存的存储结构，转而采用多存储器分散存储程序和数据，进而避开存储器互斥访问的瓶颈。

处理器细化和程序分散存储，都要求架构能够方便实现分散和结合，这就是动态关系总线的必然。同时，动态关系总线的快速变换也要求要有更高层次的逻辑控制来实现。

片上的动态计算机核心结构能够适应芯片发展的要求，因而会在人们普遍使用的各种计算机中发挥作用。

## 结语

动态计算机是计算机任务和效率需求发展的必然产物。而先前在硬件资源过于昂贵的情况下，先期发展的以现代操作系统为代表的软件管理调度方案，只适用于单处理器多功能核心架构。虽然在网络结构中这种软件管理方式可以完成各节点的异步任务操作，然而在片上的“空间紧致”和“时间一致”的环境中，难以发挥高效的作用。因此，现在流行的片上多核芯片设计，一定会让位于片上多处理器CMP结构，同时动态计算机核心结构又是CMP中最优秀的架构。

动态计算机架构可以实现对传统编程方式的支持，让传统的程序能够在多处理器上同时并行执行，因而对应用软件设计开发提供了有力的硬件支撑环境。■



姜咏江

CCF高级会员。对外经济贸易大学副教授。主要研究方向为计算机理论与设计方法、系统结构、操作系统等。  
accsys@126.com