第 5 章　并行处理机和多处理机

5.1　概述

并行处理机是单一控制部件控制下的多个处理单元构成的阵列，也称为阵列处理机，Flynn 分类法中的单指令流多数据流(Single Instruction Stream Multiple Data Stream，SIMD)结构就属于并行处理机。并行处理机主要适用于要求高速向量或矩阵运算的场合。

多处理机是指由两个或两个以上处理机组成，具有统一的操作系统，共享输入/输出子系统，通过共享主存或高速通信网络进行通信，共同求解复杂的问题。Flynn 分类法中的多指令流多数据流(Multiple Instruction Stream Multiple Data Stream，MIMD)结构就属于多处理机结构，多个处理机之间按某种方式互连，实现程序之间的数据交换和同步。

5.2　并行处理技术与发展

并行处理技术是四十几年来在微电子、印刷电路、高密度封装技术、高性能处理机、存储系统、外围设备、通信通道、语言开发、编译技术、操作系统、程序设计环境和应用问题等研究和工业发展的产物，并行计算机具有代表性的应用领域有：天气预报建模、VLSI 电路的计算机辅助设计、大型数据库管理、人工智能、犯罪控制和国防战略研究等，其应用范围还在不断地扩大。并行处理技术主要是以算法为核心，并行语言为描述，软硬件作为实现工具的相互联系而又相互制约的一种结构技术。

在计算机发展过程中，提高计算机性能的重要手段是增加并行性。并行性可以定义为在同一时刻或同一时间间隔内完成两种或两种以上性质相同或不相同的任务。

并行性有两种含义：一是同时性，指两个或多个事件在同一时刻发生在多个资源中；二是并发性，指两个或多个事件在同一时间间隔内发生在多个资源中。

计算机系统中的并行性可从不同的层次上实现，从低到高大致可分为以下层次。

(1) 指令内部的并行：是指指令执行中的各个微操作尽可能实现并行操作。

(2) 指令间的并行：是指两条或多条指令的执行是并行进行的。

(3) 任务处理的并行：是指将程序分解成可以并行处理的多个处理任务，而使两个或多个任务并行处理。

(4) 作业处理的并行：是指并行处理两个或多个作业，如多道程序设计、分时系统等。另外，从数据处理上，也有从低到高的并行层次。

(5) 字串位并：同时对一个二进制字的所有位进行操作。

(6) 字并位串：同时对多个字的同一位进行操作。

（7）全并行：同时对许多字的所有位进行操作。

5.2.1 并行处理技术的开发途径

并行处理技术有三种形式：时间并行、空间并行、时间并行与空间并行组合。

1. 时间并行

时间并行指时间重叠，在并行性概念中引入时间因素，让多个处理过程在时间上相互错开，轮流重叠地使用同一套硬件设备的各个部分，以加快硬件周转而赢得速度。时间并行性概念的实现方式就是采用流水处理部件。这是一种非常经济而实用的并行技术，能保证计算机系统具有较高的性能价格比。目前的高性能微型计算机几乎无一例外地使用了流水技术。

2. 空间并行

空间并行指资源重复，在并行性概念中引入空间因素，以“数量取胜”为原则来大幅度提高计算机的处理速度。大规模和超大规模集成电路的迅速发展为空间并行技术带来了巨大生机，因而成为实现并行处理的一个主要途径。空间并行技术主要体现在多处理器系统和多计算机系统，在单处理器系统中也得到了广泛应用。

3. 时间并行＋空间并行

指时间重叠和资源重复的综合应用，既采用时间并行性又采用空间并行性。显然，第三种并行技术带来的高速效益是最好的。

提高计算机系统的并行性措施主要有时间重叠、资源重复和资源共享三种方法。

时间重叠（Time Interleaving）：在并行性概念中引入时间因素。让多个处理过程在时间上相互错开，轮流重叠地使用同一套硬件设备的各个部分，以加快硬件周转而赢得速度。如指令内部各操作步骤采用重叠流水的工作方式。一条指令的解释分为取指、分析、执行三大步骤，分别在相应的硬件上完成。只要不出现相关，则每过一个 Δt 时间，就可以流出结果。这种执行方式加快了程序的执行速度。这种时间重叠技术原则上不需要增加更多的硬件设备就可以提高计算机系统的性能价格比。

资源重复（Resource Replication）：并行性概念中引入空间因素。通过重复设置的硬件资源来提高系统可靠性或其他性能。例如，通过使用两台或多台完全相同的处理器或计算机完成同样的任务来提高性能。

资源共享（Resource Sharing）：利用软件的方法让多个用户按一定时间顺序轮流地使用同一套资源，以提高其利用率，这样相应地提高整个系统的性能。例如，多道程序分时系统是利用共享 CPU 和主存资源以降低系统价格、提高设备利用率的一个实例。

在一个计算机中，可以通过多种技术途径，采取多种并行措施，既有执行程序的并行性，又有处理数据的并行性，它是一种信息处理的有效形式。

5.2.2 并行处理技术发展

计算机系统并行处理的发展体现了计算机系统结构的演变。并行处理技术的发展可以从两个方向上进行讨论，一是从单处理机上实现系统的并行，二是从多计算机系统向并行处理系统发展。具体发展过程为在单处理机范围内采取上述时间重叠、资源重复和资源共享三大计算机结构学的措施来发挥并行性，以提高处理速度和系统使用效率。其主要技术只是停留在功能部件一级，即在单处理机内部千方百计地改进各种功能部件(例如，使用流水线处理部件、多处理单元、相联存储器等)。实现系统并行性的进一步提高，需开发程序、任务、作业一级的并行。因此要摆脱单处理机的束缚，把多台相互独立离散的计算机相连，相互协调和配合，发展各种不同耦合度的多计算机系统(Multi-computer System)，达到更高的并行处理水平，获得更高的系统效率和处理速度，即采用功能专用化、机间互连和网络化三项基本技术措施，促使多计算机系统向并行处理系统进一步发展。多计算机采取各种措施，实现不同类型的多处理机系统(Multiprocessor System)，包括同构型多处理机、异构型多处理机和分布处理系统。

对于单处理机的发展，主导作用的是时间重叠这个途径。实现时间重叠的基础是部件功能专用化，即不断地对功能部件进行分离和细化以及平衡好它们之间的频带，尤其是注意克服信息流运行过程中影响速度的“瓶颈”来发展出高并行度的系统。为了取得主存和中央处理器的速度匹配，先后发展了指令重叠、先行控制、并行主存系统等，在 CPU 内部设置较多的通用寄存器、指令和数据缓冲寄存器、高速缓冲存储器 Cache 等。

在单机系统中，如果把功能专用化深入到处理机的执行部件内部，将部件再分成多个专用功能段进行流水处理，也可以提高处理机的并行度。

向量处理机(Vector Computer)是面向向量型并行计算，以流水线结构为主的并行处理计算机。采用先行控制和重叠操作技术、运算流水线、交叉访问的并行存储器等并行处理结构，对提高运算速度有重要作用。但在实际运行时还不能充分发挥并行处理潜力。向量运算很适合于流水线计算机的结构特点。向量型并行计算与流水线结构相结合，能在很大程度上克服通常流水线计算机中指令处理量太大、存储访问不均匀、相关等待严重、流水不畅等缺点，并可充分发挥并行处理结构的潜力，显著提高运算速度。向量处理机的发展方向是多向量机系统或细胞结构向量机。实现前者须在软件和算法上取得进展，解决如任务划分和分派等许多难题；后者则须采用适当的互连网络，用硬件自动解决因用户将分散的主存当作集中式的共存使用而带来的矛盾，才能构成虚共存的细胞结构向量机。它既具有阵列机在结构上易于扩大并行台数以提高速度的优点，又具有向量机使用方便的优点。

把时间重叠原理应用于任务一级，对各任务设置专用处理机，按流水线方式工作，就构成了宏流水线(Macro-Pipeline)，即由单处理机发展成多处理机系统。

多处理机通常分为同构型和异构型多处理机系统。同构型多处理机系统是指每个处理(器)机是同类型的，而且完成同样的功能，能同时处理同一作业中能并行执行的多个任务，这种多机系统称为对称型或同构型多处理机系统。这种同构型多处理机系统可以是

基于处理机一级冗余的容错多处理机，让多个处理机中的一部分作为备用处理机以随时顶替出故障的工作处理机，从而提高系统工作的可靠性。在此类系统中，平时几台机器都正常工作，像通常的多处理机一样，如果某个处理机出故障就被“切”掉，让系统重新组织，降低规格继续运行，直到故障排除为止。

非对称型或异构型多处理机系统是指由多个不同类型担负不同功能的处理机构成，按照作业要求的次序，利用时间重叠原理，依次执行多个任务，各自实现规定的操作。

并行处理机是指通过重复设置多个相同的处理单元，在一个控制器的指挥下，按照同一指令（一条向量指令）要求，各处理机同时对向量各元素进行操作。它在指令内部实现了数据处理的全并行。如果并行处理机普遍采用阵列结构形式，也称为阵列机。

相联存储器是一种按内容寻址的、具有信息处理功能的存储器，能按字并位串或全并行方式对所有存储单元的内容进行操作。以相联存储器为核心，加上中央处理器、指令存储器、控制器和I/O接口，就可以构成以存储器并行操作为特征的相联处理机。它将并行处理机思想运用于相联存储器内部。发展到相联处理机（Associative Processor）和并行处理机（Parallel）等多种按单指令流多数据流方式工作的多处理（器）机系统，就进入了并行处理的领域。如果要进一步提高到任务级并行，则每个处理单元配备自己的控制器，能独立地解释、执行指令而成为一台处理机，就进入了多机系统范畴。

通过资源共享的途径进行并行性的开发最初是在单处理机上采用多道程序和分时操作，其实质是单处理机模拟多处理机功能，发展形成了虚拟存储器、虚拟处理机。分时系统适用于多终端情况，对于远地用户，可配接远程终端。随着远程终端、计算机网络和微型计算机、小型计算机的发展，采用真正的处理机代替虚拟处理机，构成以分散为特征的多处理机系统。如果在终端内配上微处理器，使其不仅有I/O功能和通信功能，还具有一定的信息存储、分析、处理的能力，就成为智能终端。智能终端的出现，使原来“集中”的形态向“分布”形态方向发展。这里将这种有大量分散、重复的处理机资源（一般是具有独立功能的单处理机）相互连接在一起，在操作系统（可以是集中的也可以是分散的）的全盘控制作用下统一协调地工作而最少依赖于集中的程序、数据或硬件的系统称为分布处理系统（Distributed Processing System）。

综上所述，遵循不同的技术途径，采用不同的并行措施，在不同的层次上实现并行性的过程，反映了计算机体系结构向高性能发展的自然趋势。

在单处理机系统中，主要的技术措施是在功能部件上，即改进各功能部件，按照时间重叠、资源重复和资源共享形成不同类型的并行处理系统。在单处理机的并行发展中，时间重叠是最重要的。把一个任务分成若干相互联系的部分，把每一部分指定给专门的部件完成，然后按时间重叠措施把各部分执行过程在时间上重叠起来，使所有部件依次完成一组同样的工作。例如，将执行指令的过程分为三个子过程：取指令、分析指令和执行指令，而这三个子过程是由三个专门的部件来完成，它们是取指令部件、分析指令部件和指令执行部件。它们的工作可按时间重叠，如在某一时刻第 i 条指令在执行部件中执行，第 $i+1$ 条指令在分析部件中分析，第 $i+2$ 条指令被取指令部件取出。三条指令被同时处理，从而提高了处理机的速度。另外，在单处理机中，也较为普遍地运用了资源重复，如多操作部件和多体存储器的成功应用。

多机系统是指一个系统中有多个处理机，它属于多指令流多数据流计算机系统。按多机之间连接的紧密程度，可分为紧耦合多机系统和松耦合多机系统两种。在多机系统中，按照功能专用化、多机互连和网络化三个方向发展并行处理技术。

功能专用化经松散耦合系统及外围处理机向高级语言处理机和数据库机发展。多机互连是通过互连网络紧密耦合在一起的、能使自身结构改变的可重构多处理机和高可靠性的容错多处理机。计算机网络是为了适应计算机应用社会化、普及化而发展起来的。它的进一步发展，将满足多任务并行处理的要求，多机系统向分布式处理系统发展是并行处理的一种发展趋势。

并行技术的发展还可以从并行软件的角度进行分析。并行软件可分成并行系统软件和并行应用软件两类，并行系统软件主要指并行编译系统和并行操作系统，并行应用软件主要指各种软件工具和应用软件包。在软件中所牵涉的程序的并行性主要是指程序的相关性和网络互连两方面。

1. 程序的相关性

程序的相关性主要分为数据相关、控制相关和资源相关三类。

数据相关：说明的是语句之间的有序关系，主要有流相关、反相关、输出相关、I/O 相关和求知相关等。这种关系在程序运行前就可以通过分析程序确定下来。数据相关是一种偏序关系，程序中并不是每一对语句的成员都是相关联的。可以通过分析程序的数据相关，把程序中一些不存在相关性的指令并行地执行，以提高程序运行的速度。

控制相关：是语句执行次序在运行前不能确定的情况。它一般是由转移指令引起的，只有在程序执行到一定的语句时才能判断出语句的相关性。控制相关常使正在开发的并行性中止，为了开发更多的并行性，必须用编译技术克服控制相关。

资源相关：与系统进行的工作无关，而与并行事件利用整数部件、浮点部件、寄存器和存储区等共享资源时发生的冲突有关。软件的并行性主要是由程序的控制相关和数据相关性决定的。在并行性开发时往往把程序划分成许多的程序段——颗粒。颗粒的规模也称为粒度，它是衡量软件进程所含计算量的尺度，用细、中、粗来描述。划分的粒度越细，各子系统间的通信时延越低，并行性就越高，但系统开销也越大。因此，在进行程序组合优化的时候应该选择适当的粒度，并且把通信时延尽可能放在程序段中进行，还可以通过软硬件适配和编译优化的手段来提高程序的并行度。

2. 网络互连

将计算机子系统互连在一起或构造多处理机或多计算机时可使用静态或动态拓扑结构的网络。静态网络由点-点直接相连而成，这种连接方式在程序执行过程中不会改变，常用来实现集中式系统的子系统之间或分布式系统的多个计算结点之间的固定连接。

动态网络是用开关通道实现的，它可动态地改变结构，使之与用户程序中的通信要求匹配。动态网络包括总线、交叉开关和多级网络，常用于共享存储型多处理机中。在网络上的消息传递主要通过寻径来实现。常见的寻径方式有存储转发寻径和虫蚀寻径等。在存储转发网络中，以长度固定的包作为信息流的基本单位，每个结点有一个包缓冲区，包

从源结点经过一系列中间结点到达目的结点。存储转发网络的时延与源和目的之间的距离(段数)成正比。而在新型的计算机系统中采用虫蚀寻径,把包进一步分成一些固定长度的片,与结点相连的硬件寻径器中有片缓冲区。消息从源传送到目的结点要经过一系列寻径器。同一个包中所有的片以流水方式顺序传送,不同的包可交替地传送,但不同包的片不能交叉,以免被送到错误的目的地。虫蚀寻径的时延几乎与源和目的之间的距离无关。在寻径中产生的死锁问题可以由虚拟通道来解决。虚拟通道是两个结点间的逻辑链,它由源结点的片缓冲区、结点间的物理通道以及接收结点的片缓冲区组成。物理通道由所有的虚拟通道分时地共享。虚拟通道 虽然可以避免死锁,但可能会使每个请求可用的有效通道频宽降低。因此,在确定虚拟通道数目时,需要对网络吞吐量和通信时延折中考虑。

5.3 并行处理机结构

并行处理机是通过重复设置大量相同的处理单元(Processing Element,PE),将它们按一定的方式互连,在统一的控制部件(Control Unit,CU)控制下,对各自分配来的不同数据并行地完成同一条指令所规定的操作。它依靠操作一级的并行处理来提高系统的速度。SIMD 计算机的操作模型如图 5-1 所示。

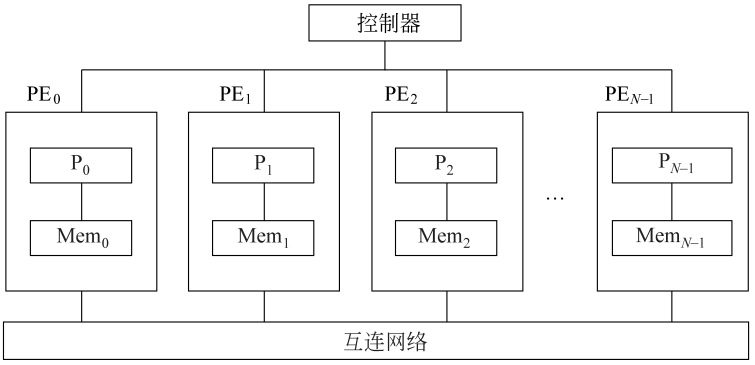

图 5-1 SIMD 计算机的操作模型

并行处理机根据存储器采用的组成方式不同分成两种基本构成：分布存储的并行处理机和共享存储的并行处理机。下面分别介绍这两种结构。

1. 分布式存储器结构

各个 PE 设有局部存储器存放分布式数据,它们只能被本处理单元直接访问。此种局部存储器称为处理单元存储器(Processing Element Memory,PEM)。在 CU 内设有一个用来存放程序的主存储器 CUM。整个系统在 CU 统一控制下运行系统程序和用户程序。执行主存中的用户程序时,所有指令都在 CU 中进行译码。将译码结果中属于标量或控制类的指令交由 CU 自己直接执行,将属于向量类的指令“播送”给各个 PE,控制 PE 并行地执行。为了有效地对向量数据进行高速处理,这种结构形式要求能将数据合理地

分配到各个处理单元的局部存储器中，以使 PE_i 主要取自己的局部存储器 PEM_i 中的数据来进行运算。运算过程中处理单元之间的数据交换可通过设置于处理单元之间的互连网络(Interconnection Network，ICN)完成。ICN 的工作也受 CU 的统一控制。其示意图如图 5-2 所示。

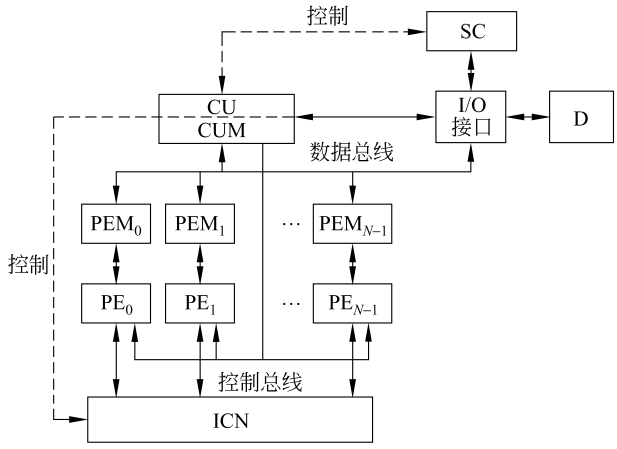

图 5-2 分布式存储器并行计算机

此类系统中处理器阵列一般是通过 CU 接到一台管理处理机(SC)上。SC 一般是一种通用计算机，用于管理整个系统的全部资源，完成系统维护、输入/输出、用户程序的汇编及向量化编译、作业调度、存储分配、设备管理、文件管理等操作系统的功能。这里 D 是大容量的磁盘存储器，通过输入/输出接口(I/O 接口)和系统相连。

采用这种结构方式的并行处理机有 ILLIAC-Ⅳ阵列处理机，1972 年由 Burroughs 公司开始生产并投入使用；美国 Goodyear 宇航公司 1979 年研制成的巨型并行处理机(Massively parallel Processor，MPP)；英国 ICL 公司 1974 年开始设计、1980 年生产的分布式阵列处理机(Distributed Array Processor，DAP)，等等。

其中，ILLIAC-Ⅳ阵列处理机是比较典型的分布式存储器并行计算机，它比当时传统的计算机速度要快得多，在一些要求高速运算的部门得到了广泛应用。总体来说，ILLIAC-Ⅳ是由三种处理机联合组成的多处理机系统，一种是专门进行数组运算的处理单元阵列，一种是进行标量运算同时又是处理单元的控制器，还有一种是管理机 B6700，担负 ILLIAC-Ⅳ输入/输出系统和操作系统管理，具体结构如图 5-3 所示。

图 5-3 中主要包括以下几部分。

(1) 处理单元阵列。

在 ILLIAC-Ⅳ计算机中，处理机阵列由 64 个结构完全相同的处理单元 PE_i 构成，每个处理单元 PE_i 字长为 64 位，PEM_i 为隶属于 PE_i 的局部存储器，每个存储器有 2k 字，全部的 PE_i 由 CU 统一管理，PE_i 都有一根方式位线，用来向 CU 传送每个 PE_i 的方式寄存器 D 中的方式位，使 CU 能了解各 PE_i 的状态是否活动，作为控制它们工作的依据。

PE_i 内部的主要寄存器有以下几种类型。

4 个 64 位寄存器：A 为累加器，存放第一个操作数和结果，B 是操作数寄存器，存放

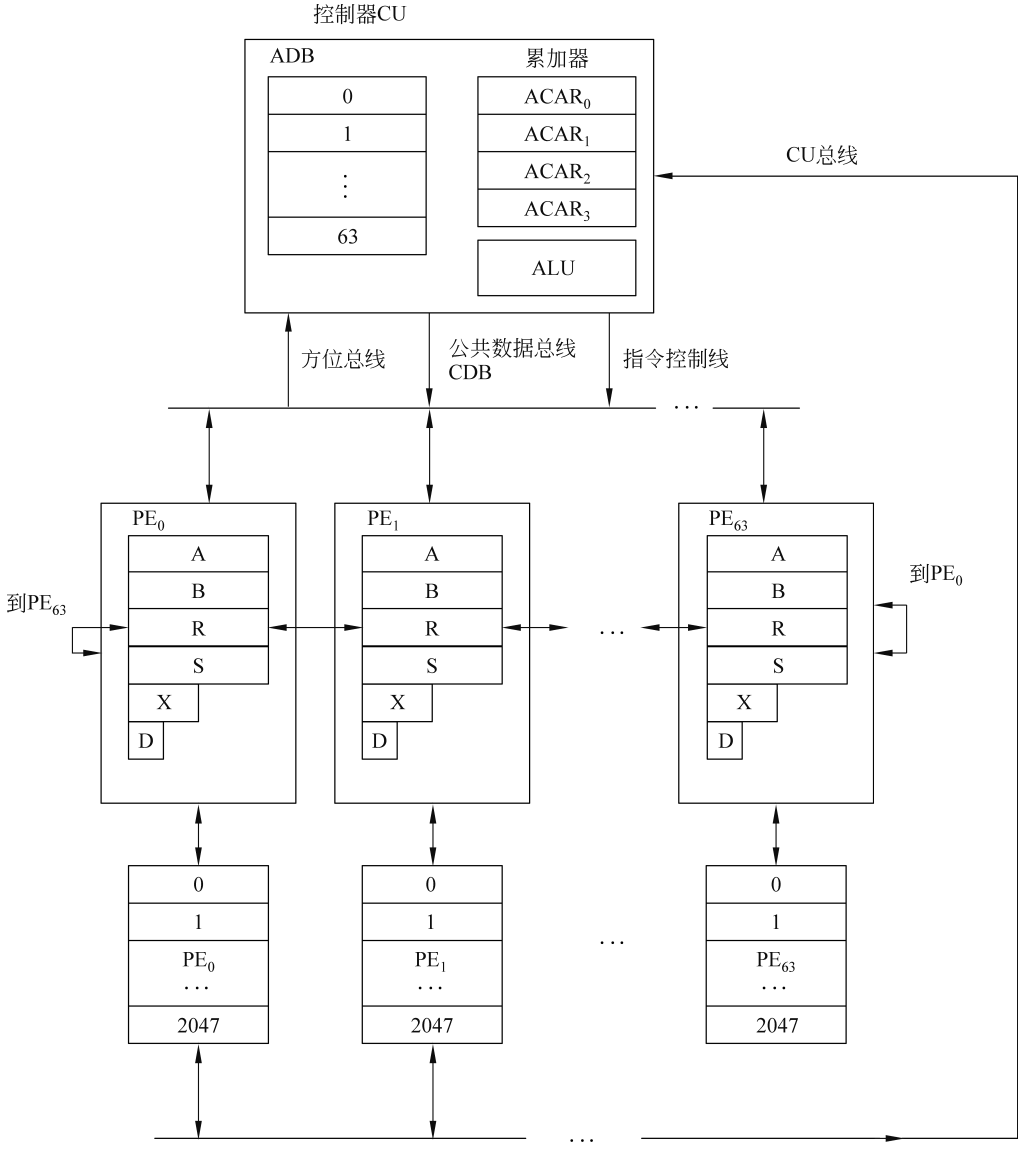

图 5-3　ILLIAC-Ⅳ阵列处理机结构图

加、减、乘、除等第二个操作数，R 是数据路由寄存器（在互连指令控制下，与相邻 PE_i 的路由寄存器相连），S 是通用存储寄存器（存放程序中间结果）。

1 个 16 位变址寄存器 X（用来形成有效地址的变址值）。

1 个 8 位方式寄存器 D，存放测试结果和 PE_i 屏蔽信息（活动标志位）。

PU 间互连状态：PU_i 代表 64 位处理单元 PE_i、所带局部存储器 PEM_i 及存储器逻辑部件总称。每台 PU_i 只能与它的 4 个近邻连接。PU_i 的 4 个近邻是 PU_{i-1}，PU_{i+1}，PU_{i-8}，PU_{i+8}（mod 64）。在这种连接称为闭合螺线阵列。这种互连网络中，当数据从一个 PU_i 传送到另一个 PU_i 要走好几步，中间经过其他 PU_i 传送。传送步数 $I \leqslant \sqrt{N-1}$

(N 为 PE_i 总数)。当 $N=64$ 时,最多步数为 7。在每次数据传送操作时由软件算出最短路径。如将 PU_{63} 传送到 PU_{10},最快可经 $PU_{63}\rightarrow PU_7\rightarrow PU_8\rightarrow PU_9\rightarrow PU_{10}$。处理单元存储器 PEM_i 分属每一个处理单元,各有 2048×64 位的存储容量和不大于 352ns 的取数时间。64 个 PEM_i 联合组成阵列存储器,存放数据和指令。整个阵列存储器可以接受控制器的访问,但是每一个处理单元只能访问到自己的局部存储器。各单元之间的连接如图 5-4 所示。

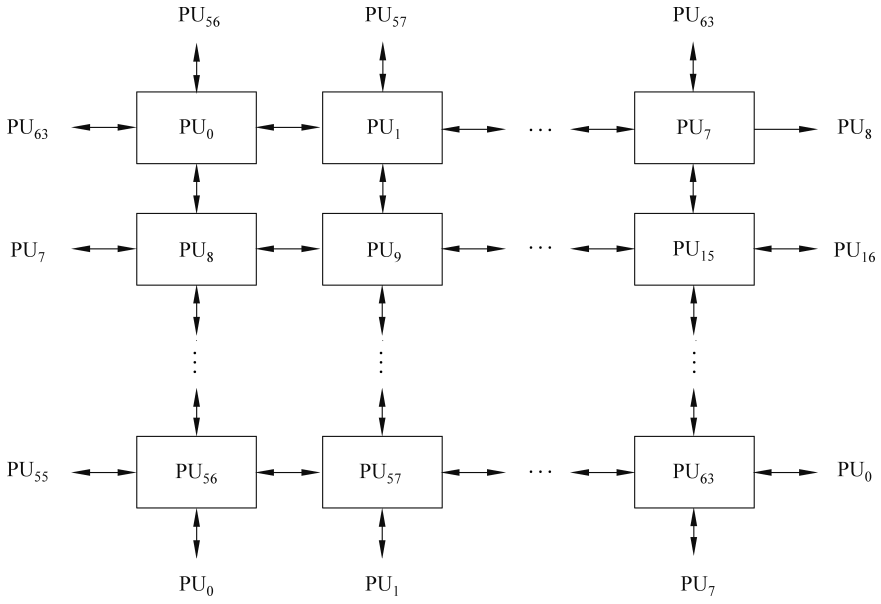

图 5-4 ILLIAC-Ⅳ各处理单元连接图

分布在各个处理单元存储器中的公共数据,只能在读至控制器后,再经过公共数据总线(CDB)广播到 64 个处理单元中来。这样,阵列存储器就如同一个二维访问存储器:如果把 64 个 PEM_i 看成列,把每一个 PEM_i 本身看成行,那么,CU 对它是按列访问,而 PE 对它是按行访问。阵列存储器的另一个特点是它的双重变址机构:控制器实现所有处理单元的公共变址,每一个处理单元内部还可以单独变址。最终的操作数有效地址对 PE_i 来说由式(5-1)决定:

$$a_i = a + (b) + (C_i) \tag{5-1}$$

式(5-1)中,a 是指令地址,(b)是 CU 中央变址寄存器内容,(C_i)是 PE_i 局部变址寄存器内容。

(2) 阵列控制器。

阵列控制器 CU 实际上是一台小型控制计算机。它除了对阵列的处理单元实行控制以外,还能利用本身的内部资源执行一整套指令,用以完成标量操作,在时间上与各 PE 的数组操作重叠起来。因此,控制器的功能有以下五方面。

① 对指令流进行控制和译码,包括执行一整套标量操作指令。

② 向各处理单元发出执行数组操作指令所需的控制信号。

③ 产生和向所有处理单元广播的公共地址部分。

④ 产生和向所有处理单元广播的公共数据。

⑤ 接收和处理由各PE、系统I/O操作以及B6700所产生的陷阱中断信号。

(3) 输入/输出系统。

ILLIAC-Ⅳ输入/输出系统由磁盘文件系统(DFS)、I/O分系统和B6700组成，完成输入/输出及其他管理功能。

ILLIAC-Ⅳ系统的操作系统，连同编译程序、汇编程序、输入/输出服务子程序等都驻留在宿主机B6700中，处理单元阵列就像是宿主机的一台专门作向量处理的后端机。或者说宿主机是处理单元阵列的一台输入/输出处理机。

ILLIAC-Ⅳ在64个PE_i并行工作时，系统最高速度可达2亿次/秒运算，可供气象预报、核工程研究等科学计算使用。

2. 集中式共享存储器组成的并行处理机结构

集中式共享存储器的结构如图5-5所示，其中大部分与分布式相同。

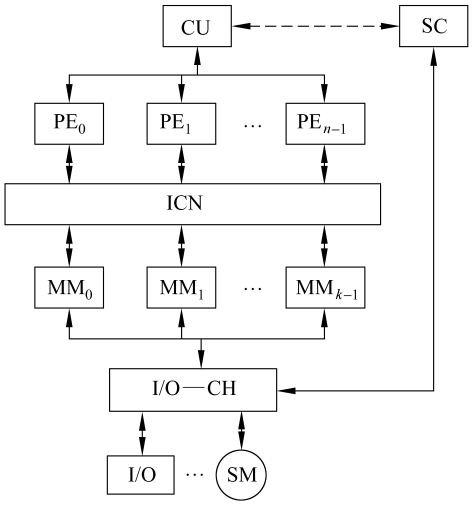

图5-5　集中式共享存储器结构

主要区别如下。

(1) 系统的存储器是由K个存储体($M_0 \sim M_{k-1}$)集中在一起构成，经过互连网络ICN为全部N个处理单元($PE_0 \sim PE_{n-1}$)所共享。在这种结构形式中，为使各处理单元对长度为N的向量中的各个元素都能同时并行处理，存储体的体数K通常总是等于或多于处理单元的个数N。为了使各处理单元在访问主存时，尽可能避免发生分体冲突，也要求有合适的算法能够将数据按一定规律合理地分配到各个存储体中。

(2) 互连网络ICN的作用不同。集中式主存的结构形式中，互连网络是用来连接处理单元和存储器分体之间的数据通路的，希望能让各个处理单元可以高速、灵活的方式连到不同的存储体上。因此有的并行处理机系统上将其称为对准网络(Alignment Network)。具