带注释的近存计算（PIM）
参考文献初步版本 0

Peter
Kogge 2025年2月

arXiv:2601.09002v1 [cs.AR] 13 Jan 2026

目录

1 引言 **2**
1.1 组织结构.2 1.2 特定设计论文.2 1.3 变更日志.2

2 综述 **3**

3 特定系统设计研究 3
3.1 1969年：存储器中的细胞逻辑.3 3.2 1970年：缓存内计算.4 3.3 *1998年：主动页.4 3.4 *1999年：Linden DAMM.4 3.5 *2019年：面向数据密集型应用的 DigitalPIM.4 3.6 *2019年：DRAM存内处理.4 3.7 *2021年：MIMHD.4 3.8 *2021年：可重构存内计算阵列.4 3.9 *2022：MultPIM.4

4 达到原型阶段的设计 **达到原型阶段 4**
4.1 *1987年：J-Machine.5 4.2 *1992年：内容寻址存储器.5 4.3 1992年：RTAIS - 实时人工智能系统.5 4.4 1997年：IRAM/VIRAM.5 4.5 *1999年：DIVA.6 4.6 *2003年：Yukon.6 4.7 *2004年：PIM Lite.6 4.8 *2023年：DIANA.6

5 商用及近商用系统 6
5.1 1972年：STARAN.6 5.2 *1972年：ICL DAP.7 5.3 1981年：ASPRO - 机载关联处理器.7 5.4 1982年：MPP - 大规模并行处理器.7 5.5 1982年：GAPP - 几何算术并行处理器.7 5.6 1985年：CM-1：连接机器-1.7 5.7 1988年：CM-2：连接机-2.8 5.8 1992年：EXECUBE.8 5.9 *1993年：Cray T3D.9 5.10 1995年：Terasys.9 5.11 1997年：Mitsubishi M32R/D.9

5.12 2004年：迁移线程.10 5.13 *2011年：混合内存立方体.10 5.14 *2014年：Micron Automata.11 5.15 *2024年：UPMEM.11

6 神经形态应用的近存计算研究 11
6.1 使用传统技术的神经形态处理器存内计算.11 6.2 使用电阻式随机存取存储器的神经形态处理器存内计算.11 6.3 使用其他技术或混合技术的神经形态处理器存内计算.12

7 其他 **12**
7.1 *2016年：计算高速链路.12 7.2 2024年：FastOverlaPIM.12

8 待排序 **12**

1 引言

存内处理（PIM）以及类似术语，例如 存内计算（CIM）、存内逻辑（LIM）、存内计算（IMC）和近存计算（NMC），最近作为一种可能具有“革命性的新型”技术而受到关注。然而，事实上，这项技术的许多实例可以追溯到 60 多年前。本文旨在提供一份涵盖整个时间范围的近存计算技术带注释的参考文献目录，并按照即将发表的一篇文章的结构进行组织。

1.1 组织结构

鉴于论文数量庞大 ，我们尝试以有组织且分层的方式对它们进行归类
方式。

最高层级的分类依据是论文内容被视为对近存计算设计或近存计算相关技术的某种“综述”（第2节）、对特定近存计算相关技术的讨论（但并非“完整”系统，第7节），还是完整系统（第3至第5节）。

第8节列出了尚待分类的论文，以便在本文未来版本中将其归入并添加到相应的前述章节。

在大多数章节中，子章节按时间顺序排列，依据是每个子章节所涵盖的最早参考文献。

1.2 特定设计论文

第3至第5节涵盖讨论特定近存计算设计的论文。每节中均设有分别讨论特定设计的子章节。所有讨论同一特定设计的论文都列在同一子章节中。子章节按其中引用的最早论文的发表时间顺序排列。

论文归入哪个章节，取决于其中讨论的近存计算达到实际应用的程度。第3节涵盖没有实际实现、仅进行论文研究或仿真研究的设计。第4节涵盖最终建成某种原型机的系统，但其实现并非为超出“实验室”评估的用途而设计。第5节涵盖已达到一定商业化程度的论文，例如建成了多个副本，或系统向多位用户开放。

第6节专门关注近存计算类人工智能应用系统，因为这一领域近年来迅速发展。此处引用的论文也可能在前面章节中引用过；本节重点在于这些论文所涉及的人工智能处理环节。

1.3 变更日志

以下数字指文档版本。

1. 2026年1月13日：首次发布，用于为即将发表的论文建立链接。并非所有列出的参考文献都已记录。

2 综述

本节 列出了近存计算技术的综述，和/或对近存计算属性或特征的讨论 s.

- [Singh 等人， 2019]
- [Asifuzzaman 等人， 2023]
- [Peccerillo 等人， 2022]
- [Houshmand 等，2023b]
- [Mutlu 等， 2023]
- [Khan 等， 2024]

3 特定系统设计研究

以下各节讨论针对特定应用的近存计算设计或近存计算系统研究，这些设计或研究从未达到原型机或商业化水平。一般性的近存计算研究见第 2 节。

待添加：

- 内容寻址存储器 1955： [Buck， 1963]
- SpaceRam 2000年： [Margolus， 2000]
- 智能存储器 2000年： [Mai等人， 2000]
- 混合单指令多数据 2022： [Coluccio等人， 2022]
- 事务型： [McDonald等人， 2007]
- 面向消息传递接口的近存计算 2003年： [Rodrigues等人， 2003]
- 2022 NAND近存计算： [Kim等人， 2022]
- 2024Piccolo： [Shin等人， 2024]
- 2022RNA序列： [Chen等人， 2022]
- 1997 为何采用近存计算： [Burger 和 Goodman， 1997]

3.1 1969年：存储器中的细胞逻辑

早在很久以前，人们就认识到，在采用大规模集成技术制造时，器件中相同单元的二维重复排列具有多种优势。虽然当这些单元支持存储器时，这一点显然成立；但即使单元还包含逻辑电路，这一点同样成立。 [Kautz 1969] 特别研究了二维数组，其中每个单元都包含逻辑电路，但其行为类似于典型的可寻址存储器，还能实现排序、关联查找、下推存储器及其他函数。

Kautz 讨论了两种单元设计，两种设计中的每个单元都只与其四个邻居相连。与行选择线对应的寄存器既可以向一行中的所有单元发送选择信号，也可以根据该行单元的逻辑电路发出的信号进行置位或复位。

在他的设计中，该阵列的工作方式有些类似于相联存储器：数据值作为输入提供（概念上是“从阵列底部”输入），所有单元行都会将其值与输入进行比较。如果某行的值较大，就会置位一个共用的行选择位。随后，所有行选择位置位的行都会并行接收其下方一行的值，也就是变为下方一行的值。接着，被选中但其下方一行未被选中的那一行，将接收作为输入提供的值。如果数据值以这种方式逐个输入，最终阵列中的数据就会按顺序排列，最低值在底部，最高值在“顶部”。

一项改进是加入掩码字段，使每个字中只有特定的位位置参与排序。这还使得能够实现类似于PLA的功能，即该阵列可以对输入执行任意逻辑函数。将选定的一组行向上或向下移动的能力，还使得下推栈或FIFO队列等其他结构得以实现。

3.2 1970年：ComputeinCache

到1970年，将缓存 纳入商用计算机的概念才刚开始受到考虑。[Stone 1970] 建议在缓存中嵌入计算能力，其设计类似于当时刚应用于IBM 360 Model 85的缓存。用现代术语来说，这种缓存设计是具有16个“扇区”条目的全相联缓存，每个条目表示存储器中连续的1024个字节。发生未命中时采用最近最少使用机制。存储器系统由四个交叉存储体组成，每个存储体在一次访问中返回16个字节，因此每次存储器访问可提供一个64B“块”。发生未命中时，需要16个这样的块才能填满一个扇区。

Stone 的建议是在中央处理器的指令集中增加指令，用于对缓存中的一个扇区执行操作，并在缓存中加入逻辑电路来执行这些操作。这些逻辑电路会为寄存器中的每个字配备一个“标记位”。每条此类指令都会提供一个内存地址，用于从存储器中选择一个扇区，以及若干单字操作数。建议的操作包括：

- 掩码相等搜索：一个操作数是包含位掩码的寄存器，另一个是比较值。扇区中所有匹配的字都会设置其标记位。

- 掩码阈值搜索：与上述操作类似，但测试条件为大于或等于。
- 复制标记位：将匹配扇区的标记位移入操作数中指定的位。
- 标记位与运算：两个操作数指定扇区中每个字内应进行与运算的位位置。
- 扇区加法：将两个扇区相加。
- 扇区缩放：将扇区中的每个字乘以某个值。

.

3.3 *1998年：ActivePages

[Oskin 等人， 1998]：1998-Oskin-ActivePages，

3.4 *1999年：Linden DAMM[Lipovski

和 Yu， 1999]：1999-Lipovski-DAAM

3.5 *2019年：面向数据密集型的 DigitalPI

M[Imani 等人，2019b]

3.6 *2019年：DRAM存内

处理[Lee 等人， 2019]

3.7 *2021年：MIMHD[

Kazemi 等人， 2021]

3.8 *2021年：可重构存内计算阵列[Lu 等人，

2021]

3.9 *2022：MultPIM[

Leitersdorf 等人， 2022]

4 个已达到原型机阶段的设计

以下介绍的是已开发出原型实现、但未达到商业化水平的存内处理设计。其中一些设计可能曾有早期研究提出。

4.1 *1987年：J-Machine

[Dally 等人, 1987][Dally 等人, 1992]: 1992年-Dally-多处理器数据路径[Noakes 等人, 1993]: 1993年-Noakes-多处理器数据路径[Spertus, 1990] 1990-J-Machine 数据流[Dally，1988] 1988-Dally-J-actors

4.2 *1992年：CAM

[Stormon 等人， 1992]: 1992-Stormon-CAM

4.3 1992年：RTAIS - 实时人工智能系统

该$RTAIS$（实时人工智能系统）[Kogge 等人，1995a] 是IBM联邦系统部于20世纪80年代末在纽约州奥韦戈开发的一套原型系统。当时专家系统正日益普及，人们希望能够足够快速地执行此类计算，从而将其用于嵌入式实时环境。

该系统实际实现时由七个模块构成，所有模块共用一条共享内存总线。其中四个模块各配备一个MIPS R-3000 处理器；另外两个模块各配备64MB存储器。最后一个模块称为ACP（关联控制处理器），是一种存内处理结构，包含可通过共享内存总线像普通存储器一样进行地址访问的存储器，同时还能够执行“内存中”计算。

ACP 上的物理内存被划分为 30 个独立的 128KB 字节宽 SRAM 内存组。每个内存组的行缓冲区处都集成了一个小型 8 位算术逻辑单元，该单元采用现场可编程门阵列实现，可执行多种比较操作。模块上的一小块控制存储器可以加载程序，供嵌入内存的处理逻辑执行。

ACP 收到一条特殊写入指令后，就会触发一系列单指令多数据访问和处理。这相当于 30 组彼此独立但协同工作的向量操作。

虽然也能执行数值运算，但主要的向量操作基于比较。每次比较都包含一个用于比较的参考数据、一个比较运算符，以及一个要检查的数据值。如果前一对数据来自 R3000，那么该操作相当于一次顺序数据库搜索，数据库位于内存中。不过，更重要的配置是将参考数据和运算符存储在内存中，而比较数据则通过内存总线从 R3000 传入。这种情况下，操作相当于一个由模式组成的“数据库”，整体操作要回答的问题是：某组数据匹配了哪个模式。这种与数据库搜索互为对偶的方式是编程语言 $Linda$ 的核心，它使输入数据流能够与一组触发规则进行比较，并在匹配时生成警报，指出匹配的是哪条规则。

4.4 1997年：IRAM/VIRAM

IRAM 的近存计算方案是在存储器裸片上集成多个向量处理器，并利用全部内存带宽，对单次访问返回的所有数据执行单指令多数据操作，然后再进行多次此类访问。早期研究 [Patterson 等人、 1997], [Kozyrakis 等人、 1997],[Patterson 1999] 提出了一种存储器裸片架构：中心位置设有一个传统核心，多个向量单元通过交叉开关连接到存储器阵列。

Gebis 等人 [于 2004]2002 年采用 180纳米 CMOS 实现了一款原型芯片，包含 1.25 亿个晶体管。芯片上有八个动态随机存取存储器（DRAM）存储体，每个容量为 13 Mbit，具有 256 位接口，并通过交叉开关连接到四个向量通道，每个通道都实现了一个向量核心。每个这样的核心都包含一个 8 KB 向量寄存器和两个算术单元，其中一个支持单精度浮点运算。芯片还包含一套面向媒体的操作。MIPS M5Ke 核心则充当片上控制处理器。

在 200兆赫兹下，功耗约为 2 瓦；与多种其他嵌入式处理器相比，VIRAM 的性能优势十分显著。

4.5 *1999年：DIVA

“该项目 DIVA 制造了一款 5500万晶体管PIM芯片。DIVA团队于 2002年 7 月 31 日在加利福尼亚州阿纳海姆举行的 DARPA Tech 2002 研讨会上，通过一套开发系统展示了这款芯片。”

[Hall等人， 1999]：1999-Hall-DIVA[Kang和Draper， 2000]：2000-Kang-DIVA[Chiueh等人， 2002]：002-Chiueh-DIVA [Draper等人， 2002]：2002-Draper-DIVA[Mediratta等人，2004a]：2004-Mediratta-DIVA[Mediratta等人，2004b]：2004-Mediratta-DIVA-exceptions[Kwon等人，2004，Barrett等人， 2006]：2004-Kwon-DIVA,2006-Barrett-DIVA

4.6 *2003年：Yukon

[Kirsch， 2003]：2003-Kirsch-Yukon

4.7 *2004年：PIMLite

[Brockman等人， 2003]：2003-Brockman-PIMLite[Brockman，2003a]：2003-Brockman-PIMLite-Asm[Brockman，2003b]：2003-Brockman-PIMLite-编程[Brockman等人， 2004]：2004-Brockman-PIMLite[Thoziyoor等人， 2005]：2005-Thoziyoor-PIMLite[?]：2007-Li-PimLite

4.8 *2023年：DIANA

[Houshmand 等，2023a]：2023-Houshmand-DIANA

5 商业和近商业系统

以下介绍已发展成熟、最终推出商业版或近商业版的近存计算系统。

- 相联存储器： [Pagiamtzis 和 Sheikholeslami， 2006]
- MASPAR 1990年： [Blank，1990年，MasPar， 1990]
- Mitsubishi M32： [Shimizu等，1996年，Nunomura等，1997年，Nunomura等， 1998]
- Micron Automata： [Wang等人， 2016]

5.1 1972年：STARAN

20世纪70年代初，固特异航空航天公司交付了一系列名为STARAN [Batcher， 1974]的计算机，用于空中交通管制等高度并行的实时应用。STARAN的独特组件是一种多维存储器模块 [Batcher， 1977] ，既可按字访问，也可按位片访问（即同时访问所有字中的第“i”位）。

STARAN的存储器容量为256字、每字256位，配备了当时相当常规但宽度较大的（256位）读写总线，以及地址和命令总线。不过，与相联存储器类似，它也配有掩码总线。该掩码可控制写入时访问中的哪些位应被修改。更特别的是，它还具有一种翻转网络 [Batcher， 1976] ，能够以多种不同方式访问存储器中的位，尤其是按常规的“字”或“位片”访问，后者由每个字中的一个位组成。

存储器模块中集成了一个由 256 个单比特处理单元（*PEs*）组成的阵列，这些处理单元可以接收翻转网络的输出，并对其执行任意布尔运算。

典型的 STARAN 系统有四个这样的模块，由关联处理器（*AP*）控制，该处理器可直接访问 PEs 中的寄存器。主程序由一台传统计算机运行，并将 AP 作为专用外设访问。

[Batcher, 1974]介绍了多个基准测试，其中 STARAN 的速度比当时的传统处理器快了几个数量级。

[Davis, 1974] 1974-Davis-staran-sw

5.2 *1972年：ICLDAP

[Reddaway, 1973]

5.3 1981年：ASPRO - 机载关联处理器

1971年的$STARAN$（第5.1节）是在超大规模集成电路技术问世之前建成的。20世纪70年代后期，同一团队采用超大规模集成电路技术设计了STARAN的新版本，称为机载关联处理器 （$ASPRO$）[Batcher, 1982]。

每个存储器阵列模块包含16个1K×4位存储器和4个超大规模集成电路逻辑芯片，每个芯片包含32个处理单元和一个32位翻转网络。因此，每个模块有128×4096个存储位和128个PEs。ASPRO系统中共有17个这样的阵列模块（其中一个是备用模块）。

同样，主要应用重点是实时机载交通管制。

5.4 1982年：MPP - 大规模并行处理器

1971年的 $STARAN$ （第5.1节）是在超大规模集成电路技术尚未普及之前建造的。其后续版本 $ASPRO$ （第5.3节）于1981年为相同的应用领域——实时机载交通管制而建造。建造这些机器的同一团队还受委托为美国国家航空航天局设计一台用于图像处理的类似机器。该系统 [Batcher, 1982] 配有16K个 处理单元，排列成128行×132列的矩形阵列（另有4列用于冗余）。

每个处理单元现在都能够执行更丰富的操作，包括32位浮点运算，并且在二维排列中与其四个相邻单元建立了通信链路。

5.5 1982年：GAPP - 几何算术并行处理器

几何算术并行处理器 （$GAPP$） [Cloud, 1988] 是马丁·玛丽埃塔电子系统部的一个项目，旨在为航空电子问题开发一套大规模单指令多数据流系统。它采用由2微米互补金属氧化物半导体芯片组成的阵列，每个芯片内含一个单元阵列，而每个单元又包含存储器和逻辑电路。建成的系统规模最高可达8.3万个单元。

每个单元包含 128 个存储位，以及由四个 1位寄存器和一个 1位算术逻辑单元组成的逻辑电路。共有一条 13 位控制通路，用于控制算术逻辑单元和单元内的数据移动；另有 7 位地址用于存储器。

每个芯片包含 12×6 个单元。每个单元中的一位组成一个“南北方向”移位寄存器，另一位则支持“东西方向”移位。第三位只支持向“北”移位。芯片的运行频率为 10 兆赫，功耗约为 1/2 瓦。

处理器阵列作为外围设备连接到一台 DEC VAX 处理器。单块阵列板包含 88 个芯片，排列成 $48 \times 132 = 6336$ 个单元。

另有一个 *GAPP* 模块控制器 （GMC）位于主机与阵列之间，并保存用于执行阵列操作的底层例程。

另一项研究 [Signorini, 1988] 考察了相关扩展方案，这些方案能够对 GAPP 阵列中的调用进行虚拟重配置，并在任意单元之间提供非局部通信。

5.6 1985年：CM-1：ConnectionMachine-1

连接机-*1* （$CM\text{-}1$） [Tucker 和 Robertson, 1988] 是 Daniel Hillis 在麻省理工学院于20世纪80年代初完成的博士论文的商业化发展成果，该论文后来以书籍形式出版 [Hillis, 1989]。第一台原型机于1984年交付，商业版本则于1986年推出。

在体系结构上，它采用了$STARAN$ （第5.1节）、DAP （第5.2节）、$ASPRO$ （第5.3节）和MPP （第5.4节）的风格。存储器被划分为最多 65K 个 4K 位分区，每个分区都与一个小型

称为处理单元 （*PE*）的 1位算术逻辑单元相关联，其中包含若干个 1位寄存器。多个处理单元的逻辑电路被集成在一块超大规模集成电路芯片上。

每组 16K 个 PEs 由一个序列器 (*i*)控制，该序列器可以同时向组内所有 PEs 广播命令。这些命令由指向每个 PE 存储器的多个地址以及要对其执行的操作组成。指令通常采用存储器到存储器操作的形式——将两个具有独立地址的存储位送入 PR 中的算术逻辑单元进行处理，然后写回存储器中可能不同的存储位。

系统中最多可以将四个序列组合在一起，每个序列拥有自己的 16K 个 PEs。这些序列又可以由最多四台主机处理器控制，通常是*DEC VAX*。

CM-1 的指令集（称为*PARIS* [Machines, 1989] 的一个独特方面是，程序可以针对“虚拟 CM-1”编写，其 PEs 数量远多于硬件提供的物理 PEs 数量。在程序开始时，可以定义所需的 PEs 数量，并为其分配每个物理 PE 存储器中的一个区块。随后，当发出对每个虚拟 PE 执行操作的命令时，每个物理 PE 都会对其所持有的每个虚拟副本重复执行该操作。

除了广播总线外，处理单元还通过全局或（每个处理单元都可以将一个位进行逻辑或运算并写入其中，所有处理单元都能看到结果）、*NEWS* 网络（每个处理单元都可以与二维数组中的四个邻居传输数据，并连接到12维超立方体网络。后一种网络是 CM-1 设计中较为独特的方面之一：每个处理单元都可以（并行）准备一条消息，发送给系统中的任意其他处理单元（目标处理单元的地址可由该处理单元动态计算），并将消息（并行）发送到网络中，再（并行）传递给目标处理单元。当大量虚拟处理单元需要交换数据时，这种方式尤其有用。

这台机器适配了多种编程语言，包括 FORTRAN、Lisp 和 C语言。

[Tucker 和 Robertson, 1988]
[Kahle 和 Hillis, 1989]

5.7 1988年：CM-2：连接机-2

连接机-*2* (*CM-2*)是对*CM-1* （第5.6节）的升级版本，速度更快、容量更大、可靠性也更高。此外，它还新增了输入/输出系统和硬件浮点运算能力。

每个处理单元 的容量是 CM-1 的 16 倍（64K 位），16 个处理单元的逻辑电路封装在一块超大规模集成电路芯片上。随后，每两块处理单元芯片（共 32 个处理单元）连接到一个专用的浮点内存接口和一个独立的浮点加速器。因此，对浮点芯片而言，32 个处理单元“看起来就像存储器”。

每块矩阵板容纳 32 个处理单元，以及各 16 个浮点内存接口和加速器芯片，总计 512 个处理单元和 32M 位，

CM-2“机箱”的物理设计刻意融入了一些美学元素，以突显其新颖的架构 [Thiel, 1994]。其标志性的前面板后来因电影《侏罗纪公园》而广为人知。

CM-2 [的性能——Sato 和 Swarztrauber 在 1988] 广泛的应用领域中取得的性能——在当时相当出色。

[Thinking Machines 公司, 1990] 1990-TMC-CM2-sw

5.8 1992年：EXECUBE

EXECUBE 是 IBM 联邦系统部在纽约州奥韦戈开展的一个项目名称，旨在开发一种同时集成存储器和计算功能、且可扩展至大型并行计算系统的单芯片类型。其目标是设法有效利用 DRAM 阵列提供的带宽，避免将任何数据传输到片外。最终成功的芯片设计采用了当时常见的 4 兆位 DRAM 器件，将子阵列划分为 8 个区段，并把计算逻辑集成到各分区的感应放大器和行缓冲区中。

每个 64KB 分区旁边都有一个专属的 16 位核心。每个芯片中的 8 个核心可以以多指令多数据（MIMD）模式运行（所有核心运行各自独立的程序），也可以以单指令多数据（SIMD）模式运行（所有核心执行从芯片外广播的指令），或者动态混合这两种模式。

每个核心有两种接口。第一种是所有核心都会监视的通用广播总线，外部处理器可通过它向任意核心组合发送命令。这些命令可以是针对特定分区的 MIMD 程序设置命令，也可以是由部分核心同步并发执行的单指令多数据指令。

其次，每个分区都配有四条全双工分区间链路。每个分区的其中三条链路连接到其他芯片上分区，从而构成一个完整的超立方体。第四条链路连接到片外，可直接连接到任何其他 EXECUBE 芯片上的任意端口。每条链路都支持双向的类似 DMA 的协议。通信可以采用“分组模式”，将一个分区存储器中的数据移至另一端分区的存储器；也可以采用“电路交换”模式，命令分区将从一条链路进入的数据流转发到另一条链路，而无需使用分区内存储器。

可用芯片于 1992年首次生产，最多 64 个芯片可集成到一台 IBM RISC-6000 工作站中。所有芯片的广播总线端口都连接到一个集群控制器，该控制器以 I/O 卡的形式安装在主机上主机上；外部链路则连接成一个四维超立方体。

[Kogge, 1994] 首次概述了该芯片及基于该芯片构建的系统。 [Kogge 等，1995b]将该芯片与当时的其他系统进行了比较。 [Sunaga 等人， 1996] 详细讨论了实现该芯片所用的技术。 [Sheliga 等人， 1997] 讨论了可能使用此类芯片的图像压缩算法。这种通用架构促成了 1995 年提出的三种拍级计算机架构之一[Sterling 等人， 1995]。

5.9 *1993年：CrayT3D

[Kessler 和 Schwarzmeier， 1993]

5.10 1995年：Terasys

Terasys 项目的目标 [Gokhale 等人， 1995] 是设计一种标准存储器芯片，在每个存储器位列下方集成一个 1 位算术逻辑单元，能够对存储器访问返回的数据执行单指令多数据操作。该设计最终制成了一款包含 2K 个 64 位字和 64 个 1 位算术逻辑单元的 128 Kbit 静态随机存取存储器。其接口采用标准静态随机存取存储器协议，并配有 4位数据总线。

虽然目标系统是为 Cray-3 超级计算机提供存储器，但还是制造了六台 SUN Sparcstation-2 工作站，每台都配备了上述芯片中的 8 MB。

算术逻辑单元包含三个 1位寄存器，可保存从存储器读取的数据或从相邻算术逻辑单元传输的数据。全局或将来自每个算术逻辑单元的信号合并起来。分区或则在系统算术逻辑单元的一个大小为 2 的幂的子集上提供类似的信号。并行前缀网络允许将信号从每个算术逻辑单元发送到另一个算术逻辑单元，后者的索引与前者相差一个 2 的幂（这支持对数时间加法）。

与主处理器的接口是标准的地址-命令对。如果最高位地址位为 0，则该地址选定存储器中的一个字，而命令指定常规的读取或写入操作。如果最高位地址位为 1，则命令会被重新解释为算术逻辑单元命令，而低位地址位会被重新解释为片上表中微编码指令的索引。

一种 C语言变体被开发为一种编程语言，并包含多种并行扩展。[Gokhale 等人 1995] 详细介绍了该芯片。

5.11 1997年：三菱M32R/D

从 EXECUBE（第5节 5.8）开始，将处理器内核嵌入采用以DRAM为中心的技术设计的芯片这一做法，在三菱 *M32R/D* [Nunomura 等人 1997]中得以延续。这款芯片配备了一个 2 MB DRAM 阵列，其行缓冲区充当缓存，供一个更强大的 66MHz 32位RISC内核使用；该内核采用五级执行流水线实现。其目标应用是便携式多媒体系统，在这些系统中，嵌入式功耗至关重要。外部总线支持访问外部存储器、设备和外设。

[Nunomura 等人 1998]提供了多种多媒体内核的性能详情。

第二代设计 [Shimizu， 1998] $M32Rx/D$ 采用 250 纳米互补金属氧化物半导体工艺，使用的金属层仅有 3 层，封装在一块 100 mm^2 裸片上。它将存储器容量增加了一倍，支持 100 兆赫的双发射核心，并新增了数字信号处理 (*DSP*)指令。此外，该芯片对外部设备而言可以表现得像一块存储器芯片，而片上核心则异步运行。因此，该芯片可以表现为带存储器的单芯片处理器、智能内存，和/或多处理器中的一个节点。

5.12 2004年：MigratingThreads

现代并行计算机几乎都被定义为分布式存储系统，由若干节点组成，每个节点都配有传统处理器和存储器。节点中的核心只能对同一节点中的存储器进行传统的加载/存储访问。如果某个核心中运行的程序希望访问其他节点中的存储器，就必须先（通过软件）确定目标对象所在的节点，然后（通过软件）向该节点中的其他处理器创建一条消息，再（通过软件）将该消息交给本节点上的网络接口控制器（*NIC*）。随后，NIC 将消息注入网络，由网络将消息路由至目标节点上的 NIC；在目标节点上，软件还必须再次被调用，以读取、解释消息，然后（全部通过软件）对相应的存储器执行所需操作。目标节点上的软件随后必须准备响应消息，并将其注入网络，使其沿相反方向返回。

从*PIM Lite*第4.7节开始，人们探索了一种替代概念：对于在某个核心中运行的软件程序而言，物理“节点”的名称，甚至节点内具体是哪一个核心，都无关紧要。相反，所有节点中的全部存储器都位于一个共同的分区全局地址空间（*PGAS*）中。如果某个线程在某个核心中执行，并访问了当前核心无法访问的内存地址，那么由硬件而非软件检测到这一点。随后，硬件（而非软件）暂停该线程，封装线程状态，并将其注入网络，以便传送到正确的物理节点。在另一端，同样由硬件（而非软件）解包线程状态，并将其插入该节点中任何一个能够访问所需内存地址的核心。然后线程继续执行，无需知道自己已经从一个物理节点移动到了另一个物理节点。物理节点和核心的“名称”变得无关紧要，唯一重要的是存储器。

[Kogge 2004] 正式提出了这种架构风格。初创公司 *Emu LLC.* 随后使用现场可编程门阵列中实现的逻辑电路，构建了一个名为 *SPIDERS* 的概念验证系统。传统个人计算机充当主处理器，可向此类系统注入初始线程，以便在“特定内存位置”执行计算。这些线程可以生成其他能够独立迁移的子线程。程序既可以存储在存储器中（在各节点间复制），也可以在某些情况下嵌入线程状态中。在大多数情况下，程序实现了可针对存储在存储器中的对象并行执行的函数。

后来的一个原型机名为 *Chick* [Dysart 等人 2016], ，支持 8 个物理节点，每个节点配有 8 个多线程核心和 8 个物理内存库。独立的传统处理器负责处理与外界的连接，同样可以向系统注入父线程以启动计算。此外，还定义了一种独立的超轻量级线程，可由常规线程中的单条指令生成，迁移至目标内存，并在内存控制器中执行。这类近内存计算对于实现远程原子操作等功能至关重要。与 SPIDERS 不同，Chick 的架构已足够成熟，能够运行完整程序，而不仅仅是函数。

一种更强健的实现称为 *Pathfinder*， [Page 和 Kogge， 2022] 其规模扩展至 8 个以上节点，并增强了功能，尤其是内存控制器中执行的超轻量线程。佐治亚理工学院 CRNCH 中心的一种实现由 [Young， 2022] 拥有 32 个节点，每个节点配有 16 个存储器组和 16 个多线程核心。

The Chick 和 Pathfinder 系统都使用一种名为 *Cilk* 的语言版本编程 .

[Kogge 等人， 2025]详细介绍了这种架构风格的发展历程，并列举了多种算法，这些算法相较于传统并行系统展现出更强的可扩展性。

5.13 *2011年：混合内存立方体

能够通过系统架构提升这类技术的典范，是美光的混合内存立方体（HMC） [Pawlowski, 2011]。这是一款底层芯片全部由逻辑电路构成的产品，并支持内存控制器、多个网络端口、消息路由器和本地处理单元。动态随机存取存储器堆叠

在这块裸片上方，并通过多个端口连接到底层逻辑电路。大量此类堆叠可以互连成一片存储海，多个主机处理器则可以向阵列注入消息，以访问任意存储器。逻辑芯片的内存控制器支持一组与 T3D 中实现的操作类似的操作。

5.14 *2014年：MicronAutomata

[Wang等人， 2016[
Dlugosch等人， 2014]

5.15 *2024年：UPMEM

[Ortega等人， 2024]: 2024-ortega-PIM-AI[Das等人，
2022]: 2022-Das-UPMEM SDK:
https://sdk.upmem.com/2021.3.0/[Upmem， 2023]:
2023-upmem-abumpim

6 近存计算在神经形态应用中的研究

以下内容与第3节一样，仅介绍相关研究，但研究对象特定为神经形态（AI）应用。由于采用不同技术的此类研究数量庞大，因此按技术用途对这些研究进行分类。

6.1 使用传统技术的神经形态处理器存内计算

[Gupta 等人， 2019]: 2019年-Gupta-NNPIM[Kwon 等人，
2019]: 2019年-Kwon-VehicularAI[Zhao 等人， 2019]: 2019
年-Zhao-BNN-PIM[Imani 等人，2019a]: 2019年-
Imani-FloatPIM[Roy 等人， 2021]: 2021年-Roy-PIM-DRAM[
Zhou 等人， 2021]: 2021年-Zhou-PIM 深度学习编译器 for
PIMs[Ansari 等人， 2022]: 2022-Ansari-SPP2D[Sutradhar 等
人， 2022]: 2022-Sutradhar-LUT-PIM[Mandal 等人， 2022]:
2022-Mandal-COIN[Herzog 等人， 2022]:
2022-Herzog-TPU4[Kiningham 等人， 2023]: 2023年-
Kiningham-GRIP[Singh 等人， 2023]: 2023年-Si
ngh-PARAG[Deepa 等人， 2023]: 2023年-Deepa-数字存内计
算[Wu 等人， 2023]: 2023年-Wu-DE-C3[Zhao 等人， 2023]:
2023年-Zhao-2T-DRAM[Jeong 等人， 2023]: 2023年-Jeong-三
值[Fang 等人， 2023]: 2023年-Fang-NN-Rel-PSP[Afifi 等人，
2024]: 2024年-Afifi-Artemis[Wang 等人，2025a]: 2025年-
Wang-Fast-OverlaPIM[Wang 等人，2025b]: 2025年-
Wang-TensorCIM[Smagulova 等人， 2023]: 2023年-
Smagulova-ReNeural

6.2 使用电阻式随机存取存储器的神经形态处理器存内计算

[Yu 等人， 2011]: 2011年 _Yu-
阻变存储器[Chi等人， 2016]: 2016-Chi-Prime[
Long等人， 2018]: 2018-Long-RRAM-RNN[
Agrawal等人， 2019]: 2019-Agrawal-SPARE[Peng等
人，2019a]: 2019-Peng-RRAM-RNN

[Long等人，2019a]：2019-Long-FeFET-DNN[Long等人，2019b]：2019-Long-ReRAM-DNN[Yang等人， 2020]：2020-Yang-ReTransformer[Wan等人， 2020]：2020-WAN-NN-RRAM[Fei等人， 2021]：2021-Fei-XB-SIM[Wan等人， 2022]：2022-Kubendran-ReRAM[Zhang等人， 2022]：2022-Zhang-Re-FeMAT[Zhang等人， 2023]：2023-Zhang-FeFET-CIM[Kaushik等人， 2023]：2023-Kaushik-MRAM-GAN[Song等人， 2023]：2023-Song-Refloat[Chen等人，2024a] ：2024-Chen-PointCIM[Chen等人，2024b]：2024-Chen-NN-RRAM

6.3 使用其他或混合技术的神经形态处理器存内计算

[Choi等人， 2020]：2020-Choi-闪存-BNN[Samiee等人， 2021]：2021-Samiee-S-LIMXNN[Wu等人，2022b]：2022-Wu-NN[Nasab等人， 2023]：2023-Nasab-MTJ-CNTFET[Sunny等人， 2024]：2024-Sunny-光学-PIM

7 其他

7.1 *2016年：Compute Express Link

[Hofstee， 2016][CXL， 2025][Huang等人， 2023]

7.2 2024年：FastOverlaPIM

Fast-OverlaPIM [Wang等人，2024年b] 是一个框架，旨在识别神经网络中可利用连续层重叠执行的部分，专门面向存内处理应用，以提升运行时性能。这项工作是 OverlaPIM [Zhou 等人 2023], 的后续研究；后者是最早引入计算重叠概念的研究之一，甚至可能是首个相关研究。OverlaPIM 依靠将输入、输出和网络权重分解为更小的数据空间，使其符合给定时间步下每个存储单元的需求；这会带来评估连续层之间所有数据空间的高昂瓶颈，以获得最佳性能。为解决这一问题，FastoverlaPIM 提供了用于配置硬件和优化流程的新接口，并配备了能够准确评估映射效率的存内处理性能模型。为增强数据组织和优化效果，该框架包含细粒度数据空间生成算法，以及一种可解析跨层数据依赖关系、从而提升重叠性能的计算分析算法。此外，一种变换算法扩展了深度神经网络优化的搜索能力。

8 待整理

以下论文已被确定与近存计算相关，但尚未完成充分审阅，因而无法将摘要归入适当的子章节。

- 计算内存 [Elliott 等人， 1992]
- [Nowatzyk 等人， 1996]：1996-Nowatzyk-wall
- [Burger 和 Goodman， 1997]：1997-Burger-WhyPIM
- 1999媒体： [Rixner 等人， 1998]
- VSIA [Birnbaum 和 Sachs， 1999]

- 2017 单磁通量子快速电路 [Sato等人， 2017]
- [Mohammedali和Agyeman, 2018]: 2018-Mohammedali-云端加速
- 2019自旋电子学 计算型随机存取存储器: [Zabihi等人, 2019]
- [Angizi等人, 2019]: 2019-Angizi-按位运算
- [Peng等人, 2019b]: 2019-Peng-权重映射 权重映射
- [Angizi和Fan, 2019]: 2019-Angizi-深度神经网络-综述 自我综述
- [Krishnan等人，2020]: 2020-Krishnan-面积优化 面积优化
- [Wu等人，2022a]：2022-Wu-新技术比较
- [Ding等人， 2023]：2023-Ding-二值神经网络-不同阵列上的二值神经网络
- [Sayed等人， 2023]：2023-Sayed-二值神经网络-综述
- [Wang等人，2024年a]：2024-存内处理仿真神经网络
- [Andrulis等人， 2024]：2024-Andrulis-存内计算循环
- [Eicken等人， 1992] 1992-Eicken-主动消息
- [Hewitt 等人， 1973] 1973-Hewitt-actor
- [Nickolls 等人， 2008] 2008-Nickolls-cuda
- [Kogge， 2019] 2019-Kogge-pvn-semantics
- [Paul 等人， 2023] 2023-Paul-actor-pgas

参考文献

[Afifi等人， 2024] Afifi, S., Thakkar, I. 和 Pasricha, S.（2024年）。Artemis：面向Transformer神经网络的混合模拟随机存内加速器。*IEEE*集成电路与系统计算机辅助设计汇刊，43(11):3336–3347。[Agrawal等人， 2019] Agrawal, A., Ankit, A. 和 Roy, K.（2019年）。SPARE：利用嵌入ROM的RAM作为存内计算原语实现脉冲神经网络加速。*IEEE*计算机学报，68(8):1190–1200。[Andrulis等人， 2024] Andrulis, T., Emer, J. S. 和 Sze, V.（2024年）。CiMLoop：灵活、准确且快速的存内计算建模工具。在*2024*年*IEEE*系统与软件性能分析国际研讨会（*ISPASS*），第10–23页，美国加利福尼亚州洛斯阿拉米托斯。IEEE计算机学会。[Angizi和Fan， 2019] Angizi, S. 和 Fan, D.（2019年）。非易失性存储器中的深度神经网络加速：一种数字方法。在*2019*年*IEEE/ACM*纳米架构国际研讨会（*NANOARCH*），第1–6页。[Angizi等人， 2019] Angizi, S., He, Z., Reis, D., Hu, X. S., Tsai, W., Lin, S. J. 和 Fan, D.（2019年）。存内处理平台中的深度神经网络加速：模拟方法还是数字方法？在*2019*年*IEEE*计算机学会超大规模集成电路年度研讨会（*ISVLSI*），第197–202页。[Ansari等人， 2022] Ansari, A., Shelton, A., Ogunfunmi, T. 和 Panchbhaiyye, V.（2022）。一种用于深度神经网络的快速压缩硬件架构。在*2022*年*IEEE*电路与系统国际研讨会（*ISCAS*），第370–374页。[Asifuzzaman等人， 2023] Asifuzzaman, K., Miniskar, N. R., Young, A. R., Liu, F. 和 Vetter, J. S.（2023年）。存内处理技术综述：进展与挑战。《存储器：材料、器件、电路与系统》，4:100022。[Barrett等人， 2006] Barrett, T., Mediratta, S., Kwon, T.-J., Singh, R., Chandra, S., Sondeen, J. 和 Draper, J.（2006）。一种具有宽字浮点能力的双倍数据速率（DDR）存内处理（PIM）器件。在*2006 IEEE*国际电路与系统研讨会（*ISCAS*），第4页。[Batcher， 1982] Batcher（1982年）。位串行并行处理系统。*IEEE*计算机学报，C-31(5):377–384。

[Batcher, 1976] Batcher, K.（1976）。STARAN中的翻转网络。《IEEE计算机学报》- TC。[Batcher, 1974] Batcher, K. E.（1974）。STARAN并行处理器系统硬件。在《1974年5月6日至10日全国计算机会议暨博览会论文集》、AFIPS '74中，第405–410页，美国纽约州纽约。计算机协会。[Batcher, 1977] Batcher, K. E.（1977）。STARAN中的多维访问存储器。《IEEE计算机学报》，C-26(2):174–177。[Birnbaum 和 Sachs, 1999] Birnbaum, M. 和 Sachs, H.（1999年）。VSIA如何应对片上系统困境。《计算机》，32(6):42–50。[Blank, 1990] Blank, T.（1990年）。MasPar MP-1架构。在《Compcon Spring '90论文摘要——第三十五届IEEE计算机学会知识杠杆国际会议》中，第20–24页。[Brockman等人, 2003] Brockman, J., Kogge, P. M., Thoziyoor, S. 和 Kang, E.（2003年）。PIM Lite：迈向大规模并行系统中的持续多线程。技术报告，圣母大学，计算机科学与工程系。[Brockman, 2003a] Brockman, J. B.（2003a）。PIM Lite架构与汇编语言编程指南。技术报告，圣母大学，计算机科学与工程系。[Brockman, 2003b] Brockman, J. B.（2003b）。PIM Lite编程。技术报告，圣母大学，计算机科学与工程系。[Brockman等人, 2004] Brockman, J. B., Thoziyoor, S., Kuntz, S. K. 和 Kogge, P. M.（2004）。一种低成本多线程存内处理系统。在《第三届存储器性能问题研讨会论文集：与第三十一届国际计算机体系结构研讨会联合举办》中，WMPI '04，第16–22页，美国纽约州纽约。计算机协会。[Buck, 1963] Buck, D. A.（1963）。关于舰载武器控制系统中计算机与计算技术最佳利用的第一份中期报告。技术报告 BuWeps-Project RM1004 M88-3U1，TRW。[Burger 和 Goodman, 1997] Burger, D. 和 Goodman, J.（1997）。以存储器为中心的架构：原因以及可能的形式。在《面向未来一代高性能处理器与系统的创新架构》论文集，第92页起。[Chen等人, 2022] Chen, L.-C., Yu, S.-Q., Ho, C.-C., Chang, Y.-H., Chang, D.-W., Wang, W.-C. 和 Chang, Y.-M.（2022）。基于存内处理架构的RNA测序定量：观察与特征分析。在《2022年IEEE第11届非易失性存储器系统与应用研讨会（NVMSA）》中，第26–32页。[Chen等人, 2024a] Chen, X.-J., Chen, H.-P. 和 Yang, C.-L.（2024年）。PointCIM：用于加速深度点云分析的存内计算架构。在《2024年第57届IEEE/ACM微体系结构国际研讨会（MICRO）》中，第1309–1322页，美国加利福尼亚州洛斯阿拉米托斯。IEEE计算机学会。[Chen等人, 2024b] Chen, Y., Han, L., Zhang, Y., Zhou, Z., Huang, P., Li, C., Liu, L. 和 Kang, J.（2024年）。基于阻变存储器阵列的注意力机制神经网络内积加速方案。《IEEE电子器件快报》，45(3):360–363。[Chi等人, 2016] Chi, P., Li, S., Xu, C., Zhang, T., Zhao, J., Liu, Y., Wang, Y. 和 Xie, Y.（2016年）。PRIME：一种用于基于ReRAM主存中神经网络计算的新型存内处理架构。在《2016年ACM/IEEE第43届年度国际计算机体系结构研讨会（ISCA）》中，第27–39页。[Chiueh等人, 2002] Chiueh, H., Draper, J., Mediratta, S. 和 Sondeen, J.（2002）。数据密集型架构（DIVA）系统的地址转换单元。在《第28届欧洲固态电路会议论文集》中，第767–770页。

[Choi 等人， 2020] Choi, W. H., Chiu, P.-F., Ma, W., Hemink, G., Hoang, T. T., Lueker-Boden, M., and Bandic, Z. (2020). 基于闪存的二值神经网络加速器，采用单层单元 NAND 闪存阵列。收录于 2020 IEEE 国际电路与系统研讨会（ISCAS），第 1–5 页。[Cloud， 1988] Cloud, E. (1988). 几何算术并行处理器。收录于《大规模并行计算前沿研讨会第二届会议论文集》，第 373–381 页。[Coluccio 等人， 2022] Coluccio, A., Casale, U., Guastamacchia, A., Turvani, G., Vacca, M., Roch, M. R., Zamboni, M., and Graziano, M. (2022). 混合 SIMD：一种面向后冯·诺依曼计算的模块化可重构方法。《IEEE计算机学报》，71(9):2287–2299。[CXL， 2025] CXL (2025). CXL 3.2 规范。https://computeexpresslink.org/.[Dally， 1988] Dally, W. (1988). J-Machine：对参与者的系统支持。https://apps.dtic.mil/sti/citations/ADA204210.[Dally 等人， 1992] Dally, W., Chien, A., Fiske, J., Fyler, G., Horwat, W., Keen, J., Lethin, R., Noakes, M., Nuth, P., and Wills, D. (1992). 消息驱动处理器：一种集成式多计算机处理单元。收录于《1992 IEEE计算机设计国际会议论文集：计算机中的超大规模集成电路与处理器》，第 416–419 页。[Dally 等人， 1987] Dally, W. J., Chao, L., Chien, A., Hassoun, S., Horwat, W., Kaplan, J., Song, P., Totty, B., and Wills, S. (1987). 消息驱动处理器的架构。收录于第 14 届年度国际计算机体系结构研讨会论文集，ISCA '87，第 189–196 页，纽约，美国纽约州。计算机协会。[Das 等人， 2022] Das, P., Sutradhar, P. R., Indovina, M., Dinakarrao, S. M. P., and Ganguly, A. (2022). 在商用存内处理硬件中实现并评估深度神经网络。收录于 2022 IEEE 第 35 届片上系统会议（SOCC），第 1–6 页。[Davis， 1974] Davis, E. W. (1974). Staran 并行处理器系统软件。收录于 1974年5月6日至10日全国计算机会议暨博览会论文集，AFIPS '74，第 17–22 页，纽约，美国纽约州。计算机协会。[Deepa 等人， 2023] Deepa, A., Nirmalkumar, P., Jose, D., and Prasad, M. (2023). 使用静态随机存取存储器和位串行算法实现可重构存内计算。收录于 2023年第5届计算应用创新研究国际会议（ICIRCA），第 1637–1644 页。[Ding 等人， 2023] Ding, A., Qiao, Y., and Bagherzadeh, N. (2023). 二值神经网络：利用阻变存内计算加速的理想架构。《IEEE 新兴计算主题汇刊》，11(2):281–291。[Dlugosch 等人， 2014] Dlugosch, P., Brown, D., Glendenning, P., Leventhal, M., and Noyes, H. (2014). 一种用于并行自动机处理的高效可扩展半导体架构。《IEEE 并行与分布式系统汇刊》，25(12):3088–3098。[Draper 等人， 2002] Draper, J., Chame, J., Hall, M., Steele, C., Barrett, T., LaCoss, J., Granacki, J., Shin, J., Chen, C., Kang, C. W., Kim, I., and Daglikoca, G. (2002). DIVA 存内处理芯片的架构。收录于第 16 届国际超级计算会议论文集，ICS '02，第 14–25 页，纽约，美国纽约州。计算机协会。[Dysart 等人， 2016] Dysart, T., Kogge, P., Deneroff, M., Bovell, E., Briggs, P., Brockman, J., Jacobsen, K., Juan, Y., Kuntz, S., Lethin, R., McMahon, J., Pawar, C., Perrigo, M., Rucker, S., Ruttenberg, J., Ruttenberg, M., and Stein, S. (2016). 基于 EMU 系统架构、采用迁移线程的高可扩展近存处理。收录于 2016年第6届非规则应用：体系结构与算法研讨会（IA3），第 2–9 页。[Eicken 等人， 1992] Eicken, T., Culler, D., Goldstein, S., and Schauser, K. (1992). 主动消息：一种集成通信与计算的机制。收录于 *[1992]* 第 19 届年度国际计算机体系结构研讨会论文集，第 256–266 页。

[Elliott 等人， 1992] Elliott, D., Snelgrove, W., and Stumm, M. (1992). 计算内存：一种存储器-SIMD混合架构及其在 DSP 中的应用。收录于 1992年 IEEE 定制集成电路会议论文集，第 30.6.1–30.6.4 页。[Fang 等人， 2023] Fang, W., Xuan, Z., Chen, S., and Kang, Y. (2023). 一种采用 rel-PSP函数和存内计算的 1.38 纳焦/次推理无时钟混合信号神经形态架构。收录于 2023年 IEEE生物医学电路与系统会议（BioCAS），第 1–5 页。[Fei 等人， 2021] Fei, X., Zhang, Y., and Zheng, W. (2021). XB-SIM：用于建模和探索基于阻变存储器的卷积神经网络加速设计的仿真框架。《清华科学与技术》，26(3):322–334。[Gebis 等人， 2004] Gebis, J., Williams, S., and Patterson, D. (2004). VIRAM1：一种面向媒体、带嵌入式动态随机存取存储器的向量处理器。收录于 2004年设计自动化会议（DAC）。[Gokhale 等人， 1995] Gokhale, M., Holmes, B., and Iobst, K. (1995). 存内处理：Terasys 大规模并行近存处理阵列。《计算机》，28:23–31。[Gupta 等人， 2019] Gupta, S., Imani, M., Kaur, H., and Rosing, T. S. (2019). NNPIM：一种用于神经网络加速的存内处理架构。《IEEE计算机学报》，68(9):1325–1337。[Hall 等人， 1999] Hall, M., Kogge, P., Koller, J., Diniz, P., Chame, J., Draper, J., LaCoss, J., Granacki, J., Brockman, J., Srivastava, A., Athas, W., Freeh, V., Shin, J., and Park, J. (1999). 将非规则应用映射到 DIVA：一种基于存内处理的数据密集型架构。收录于 SC ’99：1999年 ACM/IEEE超级计算会议论文集，第 57–57 页。[Herzog 等人， 2022] Herzog, B., Reif, S., Hemp, J., H¨onig, T., and Schr¨oder-Preikschat, W. (2022). 边缘张量处理单元的资源需求估算。《ACM嵌入式计算系统汇刊》，21(5)。[Hewitt 等人， 1973] Hewitt, C., Bishop, P., and Steiger, R. (1973). 一种用于人工智能的通用模块化 Actor形式体系。收录于第 3 届国际人工智能联合会议论文集，IJCAI’73，第 235–245 页，美国加利福尼亚州旧金山。Morgan Kaufmann出版社。[Hillis， 1989] Hillis, W. D. (1989). 连接机器。麻省理工学院出版社，剑桥，美国马萨诸塞州。[Hofstee， 2016] Hofstee, P. (2016). 主题演讲：面向大数据和云的可重构加速器。第 30 届 IEEE国际并行与分布式处理研讨会上的第 23 届可重构架构研讨会。[Houshmand 等人，2023a] Houshmand, P., Sarda, G. M., Jain, V., Ueyoshi, K., Papistas, I. A., Shi, M., Zheng, Q., Bhattacharjee, D., Mallik, A., Debacker, P., Verkest, D., and Verhelst, M. (2023a). DIANA：一种面向边缘的端到端混合数字与模拟神经网络片上系统。《IEEE固态电路期刊》，58(1):203–215。[Houshmand 等，2023b] Houshmand, P., Sun, J., and Verhelst, M. (2023b). 模拟和数字静态随机存取存储器存内计算架构的基准测试与建模。arXiv, abs/2305.18335。[Huang 等人， 2023] Huang, Y.-H., Hsieh, Y.-C., Lin, Y.-C., Chih, Y.-D., Wang, E., Chang, J., King, Y.-C., and Lin, C. J. (2023). 面向先进微控制器应用的高密度嵌入式3D可堆叠通孔阻变随机存取存储器。收录于 2023年 IEEE超大规模集成电路技术与电路研讨会（VLSI Technology and Circuits），第 1–2 页。[Imani 等人，2019a] Imani, M., Gupta, S., Kim, Y., and Rosing, T. (2019a). FloatPIM：以高精度对深度神经网络训练进行存内加速。收录于 2019年 ACM/IEEE 第 46 届年度国际计算机体系结构研讨会（ISCA），第 802–815 页。[Imani 等人，2019b] Imani, M., Gupta, S., and Rosing, T. (2019b). 基于数字的存内处理：一种面向数据密集型应用的高度并行加速器。收录于国际存储器系统研讨会论文集，MEMSYS ’19，第 38–40 页，纽约，美国纽约州。计算机协会。

[Jeong等人， 2023] Jeong, H.、Kim, S.、Park, K.、Jung, J. 和 Lee, K. J.（2023年）。采用16T1C位单元架构的三值神经网络存内计算处理器。《IEEE电路与系统学报II：快报》，70(5)：1739–1743。[Kahle和Hillis， 1989] Kahle, B. 和 Hillis, W.（1989年）。连接机模型CM-1架构。《IEEE系统、人和控制论学报》，19(4)：707–713。[Kang和Draper， 2000] Kang, C. W. 和 Draper, J.（2000年）。面向数据密集型架构（DIVA）系统的快速简易路由器。载于《第43届IEEE中西部电路与系统研讨会论文集》（目录号CH37144），第1卷，第188–192页。[Kaushik等人， 2023] Kaushik, P.、Gupta, A.、Kaushik, B. K. 和 Nehete, H.（2023年）。基于磁阻式随机存取存储器的存内计算，用于生成对抗网络的高效加速。载于《2023年IEEE第23届纳米技术国际会议（NANO）》，第1–5页。[Kautz， 1969] Kautz, W.（1969年）。细胞式存内逻辑阵列。《IEEE计算机学报》，C-18(8)：719–727。[Kazemi等人， 2021] Kazemi, A.、Sharifi, M. M.、Zou, Z.、Niemier, M.、Hu, X. S. 和 Imani, M.（2021年）。MIMHD：使用多比特存内计算实现准确高效的超维推理。载于《2021年IEEE/ACM国际低功耗电子与设计研讨会（ISLPED）》，第1–6页。[Kessler和Schwarzmeier， 1993] Kessler, R. 和 Schwarzmeier, J.（1993年）。Cray T3D：Cray Research的新维度。载于《论文摘要：Compcon Spring》，第176–182页。[Khan等人， 2024] Khan, A. A.、Lima, J. P. C. D.、Farzaneh, H. 和 Castrillon, J.（2024年）。近存计算与存内计算概览：研究与商业视角。[Kim等人， 2022] Kim, H.、Baek, S.、Song, J. 和 Song, T.（2022）。NAND闪存中的新型存内处理单元与架构。载于《2022年第19届国际SoC设计会议（ISOCC）》，第127–128页。[Kiningham等人， 2023] Kiningham, K.、Levis, P. 和 R´e, C.（2023年）。GRIP：图神经网络加速器架构。《IEEE计算机学报》，72(4)：914–925。[Kirsch， 2003] Kirsch, G.（2003年）。主动存储器：美光的Yukon。载于《国际并行与分布式处理研讨会论文集》，第11页。[Kogge， 2004] Kogge, P.（2004年）。从小猪到线程片段：面向自包含、移动式存储器编程的架构。载于《面向未来一代高性能处理器与系统的创新架构（IWIA’04）》，第130–138页。[Kogge等人，1995a] Kogge, P.、Giambra, T. 和 Sasnowitz, H.（1995a）。RTAIS：用于实时决策辅助的嵌入式并行处理器。NAECON。[Kogge， 1994] Kogge, P. M.（1994年）。EXECUBE——面向可扩展MPPS的新型架构。《并行处理国际会议》，1：77–84。[Kogge， 2019] Kogge, P. M.（2019年）。使用移动线程实现高度异构系统的多线程语义。载于《2019年高性能计算与仿真国际会议（HPCS）》，第281–289页。[Kogge等人， 2025] Kogge, P. M. 和等人（2025年）。支持迁移线程的理由。待发表。[Kogge等人，1995b] Kogge, P. M.、Sunaga, T.、Miyataka, H.、Kitamura, K. 和 Retter, E.（1995b年3月27–29日）。面向大规模并行系统的DRAM与逻辑芯片。载于《第十六届VLSI高级研究会议论文集》。[Kozyrakis等人， 1997] Kozyrakis, C.、Perissakis, S.、Patterson, D.、Anderson, T.、Asanovic, K.、Cardwell, N.、Fromm, R.、Golbus, J.、Gribstad, B.、Keeton, K.、Thomas, R.、Treuhaft, N. 和 Yelick, K.（1997年）。十亿晶体管时代的可扩展处理器：IRAM。《计算机》，30(9)：75–78。

[Krishnan等人， 2020] Krishnan, G.、Mandal, S. K.、Chakrabarti, C.、Seo, J.-s.、Ogras, U. Y. 和 Cao, Y.（2020年）。面向互连的深度神经网络存内加速面积与能耗优化。*IEEEDesign and Test*，37(6)：79–87。[Kwon等人， 2004] Kwon, T.-J.、Moon, J.-S.、Sondeen, J. 和 Draper, J.（2004年）。用于存内处理系统的0.18 /spl mu/m浮点单元实现。载于*2004 IEEE*国际电路与系统研讨会（*ISCAS*），第2卷，第II–453页。[Kwon等人， 2019] Kwon, Y.、Yang, J.、Cho, Y. P.、Shin, K.-S.、Chung, J.、Han, J.、Lyuh, C.-G.、Kim, H.-M.、Kim, C. 和 Choi, M.-S.（2019年）。采用纳米存内核心架构的功能安全车载人工智能处理器。载于*2019*年*IEEE*人工智能电路与系统国际会议（*AICAS*），第127–131页。[Lee等人， 2019] Lee, W. J.、Kim, C. H.、Paik, Y.、Park, J.、Park, I. 和 Kim, S. W.（2019年）。针对动态随机存取存储器行为优化的“存内”处理设计。*IEEE Access*，7：82633–82648。

[Leitersdorf 等人， 2022] Leitersdorf, O.、Ronen, R. 和 Kvatinsky, S.（2022）。MultPIM：用于近存计算的快速有状态乘法。*IEEE*电路与系统学报*II*：快报，69(3):1647–1651。

[Lipovski 和 Yu， 1999] Lipovski, G. 和 Yu, C.（1999年）。动态关联访问存储芯片及其在单指令多数据处理和全文数据库检索中的应用。载于*1999*年*IEEE*存储技术、设计与测试国际研讨会论文集，第24–31页。

[Long 等人，2019a] Long，Y.、Kim，D.、Lee，E.、Saha，P.、Mudassar，B. A.、She，X.、Khan，A. I. 和 Mukhopadhyay，S.（2004a）。基于铁电场效应晶体管的存内处理架构，用于深度神经网络加速。*IEEE*探索性固态计算器件与电路期刊，5(2):113–122。[Long 等人，2019b] Long，Y.、Kim，D.、Lee，E.、Saha，P.、Mudassar，B. A.、She，X.、Khan，A. I. 和 Mukhopadhyay，S.（2019b）。基于铁电场效应晶体管的存内处理架构，用于深度神经网络加速。*IEEE*探索性固态计算器件与电路期刊，5(2):113–122。[Long 等人， 2018] Long，Y.、Na，T. 和 Mukhopadhyay，S.（2018）。基于阻变存储器的存内处理架构，用于循环神经网络加速。*IEEE*超大规模集成电路系统汇刊，26(12):2781–2794。[Lu 等人， 2021] Lu，A.、Peng，X.、Luo，Y.、Huang，S. 和 Yu，S.（2021年）。基于存内计算的硬件加速器运行时可重构设计。载于*2021*年欧洲设计、自动化与测试会议暨展览会（*DATE*），第932–937页。[Machines， 1989] Machines，T.（1989年）。Paris参考手册。技术报告Tech Report Version 6.0，Thinking Machines。[Mai等人， 2000] Mai，K.、Paaske，T.、Jayasena，N.、Ho，R.、Dally，W. 和 Horowitz，M.（2000年）。智能存储器：一种模块化可重构架构。载于第*27*届国际计算机体系结构研讨会论文集（*IEEE*目录号*RS00201*），第161–171页。[Mandal等人， 2022] Mandal，S. K.、Krishnan，G.、Goksoy，A. A.、Nair，G. R.、Cao，Y. 和 Ogras，U. Y.（2022）。COIN：面向图卷积网络的通信感知存内加速。*IEEE*电路与系统新兴及精选主题期刊，12(2):472–485。[Margolus， 2000] Margolus，N.（2000）。用于大规模空间晶格计算的嵌入式动态随机存取存储器架构。载于第*27*届国际计算机体系结构研讨会论文集（*IEEE*目录号*RS00201*），第149–160页。[MasPar， 1990] MasPar（1990年）。MasPar应用语言（MPL）参考手册。技术报告Pub-9302-0000，MasPar。

[McDonald等人， 2007] McDonald，A.、Carlstrom，B. D.、Chung，J.、Minh，C. C.、Chafi，H.、Kozyrakis，C. 和 Olukotun，K.（2007）。事务性内存：软硬件接口。《IEEE Micro》，27(1):67–76。[Mediratta等人，2004a] Mediratta，S.、Sondeen，J. 和 Draper，J.（2004a）。面向数据密集型架构（DIVA）系统的面积高效路由器。载于第17届国际超大规模集成电路设计会议。论文集，第863–868页。[Mediratta等人，2004b] Mediratta，S.、Steele，C.、Singh，R.、Sondeen，J. 和 Draper，J.（2004b）。面向存内处理系统的面积高效精确异常处理单元的0.18/spl mu/m互补金属氧化物半导体实现。载于《2004年第47届中西部电路与系统研讨会，2004年（MWSCAS 2004）》，第3卷，第455–458页。[Mohammedali和Agyeman， 2018] Mohammedali，N. 和 Agyeman，M. O.（2018）。云计算可重构加速器研究。载于《第2届国际计算机科学与智能控制研讨会论文集》，ISCSIC '18，美国纽约州纽约市。计算机协会。[Mutlu等人， 2023] Mutlu，O.、Ghose，S.、G´omez-Luna，J. 和 Ausavarungnirun，R.（2023年）。存内处理现代入门，第171–243页。Springer Nature Singapore，新加坡。[Nasab等人， 2023] Nasab，M. T.、Amirany，A.、Moaiyeri，M. H. 和 Jafari，K.（2023年）。用于存内处理架构的混合磁隧道结/碳纳米管场效应晶体管二进制突触与神经元。《IEEE Magnetics Letters》，14:1–5。[Nickolls等人， 2008] Nickolls，J.、Buck，I.、Garland，M. 和 Skadron，K.（2008）。利用CUDA实现可扩展并行编程。《Queue》，6(2):40–53。[Noakes等人， 1993] Noakes，M. D.、Wallach，D. A. 和 Dally，W. J.（1993）。J-Machine多计算机：架构评估。《SIGARCH计算机体系结构新闻》，21(2):224–235。[Nowatzyk等人， 1996] Nowatzyk，A.、Pong，F. 和 Saulsbury，A.（1996）。错失内存墙：处理器/内存集成的必要性。载于第23届年度国际计算机体系结构研讨会（ISCA'96），第90–90页。[Nunomura等人， 1998] Nunomura，Y.、Shimizu，T.、Saitoh，K. 和 Tsuchihashi，K.（1998）。带嵌入式动态随机存取存储器的微处理器的多媒体应用。载于《1998年IEEE国际声学、语音与信号处理会议论文集》，ICASSP '98（目录号98CH36181），第5卷，第3157–3160页，第5卷。[Nunomura等人， 1997] Nunomura，Y.、Shimizu，T. 和 Tomisawa，O.（1997年）。M32R/D——集成动态随机存取存储器与微处理器。《IEEE Micro》，17(6):40–48。[Ortega等人， 2024] Ortega，C.、Falevoz，Y. 和 Ayrignac，R.（2024年）。PIM-AI：一种高效大语言模型推理的新型架构。[Oskin等人， 1998] Oskin，M.、Chong，F. 和 Sherwood，T.（1998）。主动页：一种面向智能内存的计算模型。载于《第25届年度国际计算机体系结构研讨会论文集》（目录号98CB36235），第192–203页。[Page和Kogge， 2022] Page，B. A. 和 Kogge，P.（2022）。一种新计算模型的演进。载于《2022年IEEE/ACM非规则应用：体系结构与算法研讨会》（IA3），第9–18页。[Pagiamtzis和Sheikholeslami， 2006] Pagiamtzis，K. 和 Sheikholeslami，A.（2006）。内容可寻址存储器（CAM）电路与架构：教程与综述。《IEEE固态电路期刊》，41(3):712–727。[Patterson， 1999] Patterson，D.（1999年）。IRAM：面向后PC时代的微处理器。载于《1999年国际超大规模集成电路技术、系统与应用研讨会：技术论文集》（目录号99TH8453），第39–41页。

[Patterson 等人， 1997] Patterson, D., Anderson, T., Cardwell, N., Fromm, R., Keeton, K., Kozyrakis, C., Thomas, R., and Yelick, K. (1997年). 智能 RAM（IRAM）：能够记忆和计算的芯片。发表于 1997年 IEEE 国际固态电路会议。技术论文摘要，第 224–225 页。[Paul 等人， 2023] Paul, S. R., Hayashi, A., Chen, K., Elmougy, Y., and Sarkar, V. (2023年). 面向分区全局地址空间应用的细粒度异步批量同步并行模型。《计算科学杂志》，69:102014。[Pawlowski， 2011] Pawlowski, J. T. (2011年). 混合内存立方体（HMC）。Hot Chips 23。[Peccerillo 等人， 2022] Peccerillo, B., Mannino, M., Mondelli, A., and Bartolini, S. (2022). 硬件加速器综述：分类体系、趋势、挑战与展望。《系统架构期刊》，129(C)。[Peng 等人，2019a] Peng, X., Liu, R., and Yu, S. (2019a). 面向基于阻变随机存取存储器的存内处理架构，优化卷积神经网络的权重映射与数据流。发表于 2019 年 IEEE 国际电路与系统研讨会（ISCAS），第 1–5 页。[Peng 等人，2019b] Peng, X., Liu, R., and Yu, S. (2019b). 面向基于阻变随机存取存储器的存内处理架构，优化卷积神经网络的权重映射与数据流。发表于 2019 年 IEEE 国际电路与系统研讨会（ISCAS），第 1–5 页。[Reddaway， 1973] Reddaway, S. F. (1973年). DAP——分布式阵列处理器。发表于第 1 届计算机体系结构年度研讨会（ISCA ’73），第 61–65 页，美国纽约。计算机协会。[Rixner 等人， 1998] Rixner, S., Dally, W., Kapasi, U., Khailany, B., Lopez-Lagunas, A., Mattson, P., and Owens, J. (1998年). 一种用于媒体处理的高带宽效率架构。发表于第 31 届 ACM/IEEE 微体系结构国际研讨会论文集，第 3–13 页。[Rodrigues 等人， 2003] Rodrigues, A., Murphy, R., Kogge, P., Brockman, J., Brightwell, R., and Under- wood, K. (2003年). 存内处理体系结构模型对消息传递接口的影响。发表于 2003 年 IEEE 集群计算国际会议论文集，第 259–268 页。[Roy 等人， 2021] Roy, S., Ali, M., and Raghunathan, A. (2021年). PIM-DRAM：利用商用 DRAM 加速机器学习工作负载。IEEE 电路与系统新兴及精选主题期刊，11(4):701–710。[Samiee 等人， 2021] Samiee, A., Borulkar, P., DeMara, R. F., Zhao, P., and Bai, Y. (2021年). 基于自旋霍尔效应的逻辑存内架构对二值化卷积神经网络进行低能耗加速。IEEE 新兴计算主题汇刊，9(2):928–940。[Sato 等人， 2017] Sato, R., Hatanaka, Y., Ando, Y., Tanaka, M., Fujimaki, A., Takagi, K., and Takagi, N. (2017年). 基于快速单磁通量子逻辑、采用最简指令集架构的嵌入随机存取存储器微处理器的高速运行。IEEE 超导学汇刊，27(4):1–5。[Sato 和 Swarztrauber， 1988] Sato, R. and Swarztrauber, P. (1988年). Connection Machine 2 基准测试。发表于《超算 ’88：1988 年 ACM/IEEE 超级计算会议论文集》，第 I 卷，第 304–309 页。[Sayed 等人， 2023] Sayed, R., Azmi, H., Shawkey, H., Khalil, A. H., and Refky, M. (2023年). 二值神经网络系统性文献综述。IEEE Access，11:27546–27578。[Sheliga 等人， 1997] Sheliga, M., Kogge, P., and Hsing-Mean Sha, E. (1997年). 使用 Execube 处理阵列进行视频压缩的硬件/软件协同设计。发表于 IEEE 1997 年国家航空航天与电子会议论文集（NAECON 1997），第 1 卷，第 279–286 页，第 1 卷。[Shimizu， 1998] Shimizu, T. (1998年). M32Rx/D——一种具有大容量 4MB 内部 DRAM 的单芯片微控制器。Hot Chips 10。

[Shimizu 等人， 1996] Shimizu, T., Korematu, J., Satou, M., Kondo, H., Iwata, S., Sawai, K., Okumura, N., Ishimi, K., Nakamoto, Y., Kumanoya, M., Dosaka, K., Yamazaki, A., Ajioka, Y., Tsubota, H., Nunomura, Y., Urabe, T., Hinata, J., and Saitoh, K. (1996). 一款具有 16MB DRAM 的多媒体 32 位 RISC 微处理器。发表于 1996年IEEE国际固态电路会议。技术论文摘要，ISSCC，第 216–217 页。[Shin 等人， 2024] Shin, C., Kwon, T., Song, J., Ju, J. H., Liu, F., Choi, Y., and Lee, J. (2024年). 探讨利用存内随机散聚实现快速图处理的必要性。IEEE计算机体系结构快报，23(1):73–77。[Signorini， 1988] Signorini, J. (1988年). NCR GAPP 的路由模型。发表于国际工业应用人工智能研讨会论文集，第 517–521 页。[Singh 等人， 2019] Singh, G., Chelini, L., Corda, S., Awan, A. J., Stuijk, S., Jordans, R., Corporaal, H., and Boonstra, A.-J. (2019年). 近存计算：过去、现在与未来。《微处理器与微系统》，71:102868。[Singh 等人， 2023] Singh, G., Kuppannagari, S. R., and Vrudhula, S. (2023年). PARAG：用于 GCN 实时加速的存内处理架构。发表于 2023年 IEEE第30届高性能计算、数据与分析国际会议（HiPC），第 11–20 页。[Smagulova 等人， 2023] Smagulova, K., Fouda, M. E., Kurdahi, F., Salama, K. N., and Eltawil, A. (2023年). 电阻式神经网络硬件加速器。IEEE会刊，111(5):500–527。[Song 等人， 2023] Song, L., Chen, F., Li, H., and Chen, Y. (2023年). ReFloat：利用 ReRAM 进行低成本浮点运算，以加速迭代线性求解器。发表于 SC23：高性能计算、网络、存储与分析国际会议，第 1–15 页。[Spertus， 1990] Spertus, E. (1990年). J-Machine 的数据流计算。http://hdl.handle.net/ 1721.1/7030。[Sterling 等人， 1995] Sterling, T., Messina, P., and Smith, P. H. (1995年). 千万亿次计算的使能技术。麻省理工学院出版社，美国马萨诸塞州剑桥。[Stone， 1970] Stone, H. S. (1970年). 一台逻辑存内计算机。IEEE计算机学报，C-19(1):73–78。[Stormon 等人， 1992] Stormon, C., Troullinos, N., Saleh, E., Chavan, A., Brule, M., and Oldfield, J. (1992年). 一种通用 CMOS 关联处理器芯片及系统。IEEE Micro，12(6):68–78。[Sunaga 等人， 1996] Sunaga, T., Miyatake, H., Kitamura, K., Kogge, P. M., and Retter, E. (1996). 一款带有嵌入式DRAM宏单元的并行处理芯片。IEEE 固态电路期刊，31(10):1556–1559。[Sunny 等人， 2024] Sunny, F., Shafiee, A., Balasubramaniam, A., Nikdast, M., and Pasricha, S. (2024年). OPIMA：用于卷积神经网络加速的光学存内处理。IEEE集成电路与系统计算机辅助设计汇刊，43(11):3888–3899。[Sutradhar 等人， 2022] Sutradhar, P. R., Bavikadi, S., Connolly, M., Prajapati, S., Indovina, M. A., Di- nakarrao, S. M. P., and Ganguly, A. (2022). 基于查找表、用于深度学习应用且具有可编程精度缩放功能的存内处理架构。IEEE 并行与分布式系统汇刊，33(2):263–275。[Thiel， 1994] Thiel, T. (1994). 连接机的设计。发表于《设计问题》，第 10 卷，第 5–18 页。[Thinking Machines 公司， 1990] Thinking Machines Corp. (1990). 连接机CM-2型号技术概要，第 6 版。https://people.csail.mit.edu/bradley/cm5docs/nov06/ConnectionMachineModelCM-2TechnicalSummary.pdf。

[Thoziyoor等人， 2005] Thoziyoor, S., Brockman, J., and Rinzler, D. (2005). PIM Lite：一种多线程存内处理器原型。在第15届ACM大湖区超大规模集成电路研讨会论文集，GLSVLSI '05，第64–69页，美国纽约州纽约市。计算机协会。[Tucker和Robertson， 1988] Tucker, L. and Robertson, G. (1988年). 连接机的架构与应用。《计算机》，21(8):26–38。[UPMEM， 2023] UPMEM (2023年). Abumpimp 2023。[Wan等人， 2020] Wan, W., Kubendran, R., Eryilmaz, S. B., Zhang, W., Liao, Y., Wu, D., Deiss, S., Gao, B., Raina, P., Joshi, S., Wu, H., Cauwenberghs, G., and Wong, H.-S. P. (2020). 33.1 一款74 TMAC/s/W互补金属氧化物半导体-阻变随机存取存储器神经突触核心，支持动态可重构数据流和用于概率图模型的原位可转置权重。载于2020年IEEE国际固态电路会议（ISSCC），第498–500页。[Wan等人， 2022] Wan, W., Kubendran, R., Schaefer, C., Eryilmaz, S., Zhang, W., Wu, D., Deiss, S., Raina, P., Qian, H., Gao, B., Joshi, S., Wu, H., Wong, H., and Cauwenberghs, G. (2022). 一款基于电阻式随机存取存储器的存内计算芯片。《自然》，608:504–512。[Wang等人， 2016] Wang, K., Angstadt, K., Bo, C., Brunelle, N., Sadredini, E., Tracy, T., Wadden, J., Stan, M., and Skadron, K. (2016年). 美光自动机处理器概述。载于2016年国际硬件/软件协同设计与系统综合会议（CODES+ISSS），第1–3页。[Wang等人， 2024a] Wang, X., Sun, X., Han, Y., and Chen, X. (2024a). PIMSIM-NN：一种面向存内处理加速器、基于ISA的仿真框架。载于2024年欧洲设计、自动化与测试会议暨展览（DATE），第1–2页。[Wang等人， 2024b] Wang, X., Zhou, M., and Rosing, T. (2024b). Fast-OverlaPIM：一种用于存内神经网络加速的快速重叠驱动映射框架。IEEE集成电路与系统计算机辅助设计汇刊。[Wang等人，2025a] Wang, X., Zhou, M., and Rosing, T. (2025a). Fast-OverlaPIM：一种用于存内神经网络加速的快速重叠驱动映射框架。IEEE集成电路与系统计算机辅助设计汇刊，44(1):130–143。[Wang等人，2025b] Wang, Y., Wu, Z., Wu, W., Liu, L., Hu, Y., Wei, S., Tu, F., and Yin, S. (2025b). TensorCIM：一种采用基于多芯片组件架构、用于超越神经网络加速的数字存内计算张量处理器。IEEE固态电路期刊，60(2):734–747。[Wu等人， 2023] Wu, G.-W., Chang, C.-Y., and Wu, A.-Y. A. (2023). DE-C3：一种面向基于存内计算的卷积神经网络加速的动态能效压缩方法。载于2023年IEEE第36届片上系统会议（SOCC），第1–6页。[Wu等人，2022a] Wu, J., Wen, H., Pang, W., and Chong, G. (2022a). 基于存内计算的阵列电路设计研究。载于2022年第4届国际自然语言处理会议（ICNLP），第537–541页。[Wu等人，2022b] Wu, J., Zhao, B., Wen, H., and Zhao, Q. (2022b). 基于存内计算理论的神经网络加速器设计。载于2022年第4届国际自然语言处理会议（ICNLP），第547–551页。[Yang等人， 2020] Yang, X., Yan, B., Li, H., and Chen, Y. (2020). ReTransformer：一种基于阻变存储器、用于Transformer加速的存内处理架构。载于2020年IEEE/ACM国际计算机辅助设计会议（ICCAD），第1–9页。[Young， 2022] Young, J. (2022). 近存与存内计算。https://crnch-rg.cc.gatech.edu/ near-memory-and-in-memory/。

[Yu等人， 2011] Yu, S., Wu, Y., Jeyasingh, R., Kuzum, D., and Wong, H.-S. P. (2011年). 一种用于神经形态计算的金属氧化物电阻式开关存储器电子突触器件。*IEEE*电子器件学报，58(8):2729–2737。[Zabihi等人， 2019] Zabihi, M., Chowdhury, Z. I., Zhao, Z., Karpuzcu, U. R., Wang, J.-P., and Sapatnekar, S. S. (2019年). 自旋电子CRAM上的存内处理：从硬件设计到应用映射。*IEEE*计算机学报，68(8):1159–1173。[Zhang等人， 2023] Zhang, S., Chen, J., Wang, Y., Jia, Z., Zhuo, C., and Yin, X. (2023年). 面向神经网络加速的铁电场效应晶体管存内计算设计与优化。载于*2023*年国际电子设计自动化研讨会（*ISEDA*），第225–229页。[Zhang等人， 2022] Zhang, X., Liu, R., Song, T., Yang, Y., Han, Y., and Chen, X. (2022). Re-FEMAT：一种可重构多功能铁电场效应晶体管存储架构。*IEEE*集成电路与系统计算机辅助设计汇刊，41(11):5071–5084。[Zhao等人， 2023] Zhao, Y., Shen, Z., Xu, J., Chai, K. C., Wu, Y., and Wang, C. (2023年). 一种用于片上深度神经网络训练与推理的全新转置型2T-DRAM存内计算架构。载于*2023*年*IEEE*第*5*届人工智能电路与系统国际会议（*AICAS*），第1–4页。[Zhao等人， 2019] Zhao, Y., Yang, J., Jia, X., Wang, X., Wang, Z., Kang, W., Zhang, Y., and Zhao, W. (2019年). 利用近存处理架构加速贝叶斯神经网络。载于*2019*年*IEEE*计算机学会超大规模集成电路年度研讨会（*ISVLSI*），第203–206页。[Zhou等人， 2021] Zhou, M., Chen, G., Imani, M., Gupta, S., Zhang, W., and Rosing, T. (2021年). PIM-DL：通过数据布局优化提升数字存内处理架构上的深度神经网络推理性能。载于*2021*年第*30*届并行架构与编译技术国际会议（*PACT*），第1–1页。[Zhou等人， 2023] Zhou, M., Wang, X., and Rosing, T. (2023年). OverlaPIM：面向存内神经网络加速的重叠优化。载于*2023*年欧洲设计、自动化与测试会议暨展览（*DATE*），第1–6页。IEEE。

索引

ACP，5，另见 关联控制处理器 机载关联处理器，7，另见 ASPRO AP，6，另见 关联处理器
ASPRO，7，另见 机载关联处理器 关联控制处理器，5，另见 ACP 关联处理器，6，另见 AP

广播总线，9

缓存，4 Chick，10 Cilk，10 集群控制器，9 CM-1，7，另见 连接机-1，8 CM-2，8，另见 连接机-2 连接机-1，7，另见 CM-1 连接机-2，8，另见 CM-2

DAP，7 DEC VAX，8 数字信号处理，10，另见 DSP 分布式存储系统，10 DSP，10，另见 数字信号处理

Emu LLC.，10
EXECUBE，8

翻转网络，6

GAPP，7，另见 几何算术并行处理器 GAPP 模块控制器，7，另见 GMC 几何算术并行处理器，7，另见 GAPP 全局或，8 GMC，7，另见 GAPP 模块控制器

Linda，5

M32R/D，9
M32Rx/D，10
MPP，7

网络接口控制器，10，另见 NIC NEWS 网络，8 NIC，10，另见 网络接口控制器 节点，10

PARIS, 8 分区全局地址空间, 10, 另请参阅 PGAS
Pathfinder, 10 处理单元, 7, 8,另请参阅 处理单元
PEs, 6, 另请参阅 处理单元
PGAS, 10, 另请参阅 分区全局地址空间 PIM Lite, 10 PLA, 3 处理单元, 8, 另请参阅 处理单元 处理单元, 6, 另请参阅 PEs

实时人工智能系统，5，另见 RTAIS
RTAIS，5，另见 实时人工智能系统

序列器，8，另见 t
SPIDERS，10 STARAN，7

t，8，另见 序列器
Terasys，9

向量通道，5