多级存储器与模拟内存内计算完美融合，人工智能边缘处理难题迎刃而解

作者：Microchip Technology Inc
嵌入式存储器产品开发总监
Vipin Tiwari

机器学习和深度学习已成为我们生活中不可或缺的部分。利用自然语言处理（NLP）、图像分类和物体检测实现的人工智能（AI）应用已深度嵌入到我们使用的众多设备中。大多数 AI 应用通过云引擎即可出色地满足其用途，例如在 Gmail 中回复电子邮件时可以获得词汇预测。

虽然我们可以享受到这些 AI 应用带来的益处，但这种方法导致隐私、功耗、延时和成本等诸多因素面临挑战。如果有一个能够在数据来源处执行部分或全部计算（推断）的本地处理引擎，那么这些问题即可迎刃而解。传统数字神经网络的存储器功耗存在瓶颈，难以实现这一目标。为了解决这一问题，可以将多级存储器与模拟内存内计算方法结合使用，使处理引擎满足更低的毫瓦级（mW）到微瓦级（µW）功率要求，从而在网络边缘执行 AI 推断。

通过云引擎提供服务的 AI 应用面临的挑战

如果通过云引擎为 AI 应用提供服务，用户必须将一些数据以主动或被动方式上传到云，计算引擎在云中处理数据并提供预测，然后将预测结果发送给下游用户使用。下面概述了这一过程面临的挑战：

图 1：从边缘到云的数据传输

1. ***隐私问题：***对于始终在线始终感知的设备，个人数据和/或机密信息在上传期间或在数据中心的保存期限内存在遭受滥用的风险。
2. ***不必要的功耗：***如果每个数据位都传输到云，则硬件、无线电、传输装置以及云中不必要的计算都会消耗电能。

3. ***小批量推断的延时：***如果数据来源于边缘，有时至少需要一秒才能收到云系统的响应。当延时超过 100 毫秒时，人们便有明显感知，造成反响不佳的用户体验。
4. ***数据经济需要创造价值：***传感器随处可见，价格低廉；但它们会产生大量数据。将每个数据位都上传到云进行处理并不划算。

要使用本地处理引擎解决这些挑战，必须首先针对目标用例利用指定数据集对执行推断运算的神经网络进行训练。这通常需要高性能计算（和存储器）资源以及浮点算数运算。因此，机器学习解决方案的训练部分仍需在公共或私有云（或本地 GPU、CPU 和 FPGA Farm）上实现，同时结合数据集来生成最佳神经网络模型。神经网络模型的推断运算不需要反向传播，因此在该模型准备就绪之后，可利用小型计算引擎针对本地硬件进行深度优化。推断引擎通常需要大量乘-累加（MAC）引擎，随后是激活层（例如修正线性单元（ReLU）、Sigmoid 函数或双曲正切函数，具体取决于神经网络模型复杂度）以及各层之间的池化层。

大多数神经网络模型需要大量 MAC 运算。例如，即使是相对较小的“1.0 MobileNet-224”模型，也有 420 万个参数（权重），执行一次推断需要多达 5.69 亿次的 MAC 运算。此类模型中的大多数都由 MAC 运算主导，因此这里的重点是机器学习计算的运算部分，同时还要寻找机会来创建更好的解决方案。下面的图 2 展示了一个简单的完全连接型两层网络。输入神经元（数据）通过第一层权重处理。第一层的输出神经元通过第二层权重处理，并提供预测（例如，模型能否在指定图像中找到猫脸）。这些神经网络模型使用“点积”运算计算每层中的每个神经元，如下面的公式所示：

$Y_i = \sum_i W_i X_i$ （为简单起见，公式中省略了“偏差”项）。

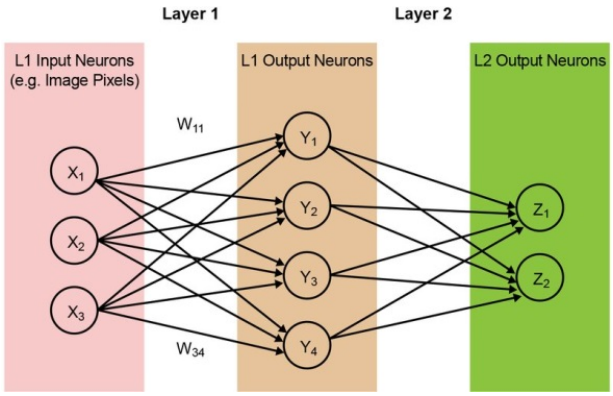

图 2：完全连接的两层神经网络

在数字神经网络中，权重和输入数据存储在 DRAM/SRAM 中。权重和输入数据需要移至某个 MAC 引擎旁以进行推断。根据下图，采用这种方法后，大部分功耗都来源于获取模型参数以及将数据输入到实际发生 MAC 运算的 ALU。从能量角度来看，使用数字逻辑门的典型 MAC 运算消耗约 250 fJ 的能量，但在数据传输期间消耗的能量超过计算本身两个数量级，达到 50 皮焦（pJ）到 100 pJ 的范围。公平地说，很多设计技巧可以最大程度减少存储器到 ALU 的数据传输，但整个数字方案仍受冯·诺依曼架构的限制。这就意味

着，有大量的机会可以减少功率浪费。如果执行 MAC 运算的能耗可以从约 100 pJ 减少到若干分之几 pJ，将会怎样呢？

消除存储器瓶颈同时降低功耗

如果存储器本身可用来消除之前的存储器瓶颈，则在边缘执行推断相关的运算就成为可行方案。使用内存内计算方法可以最大程度地减少必须移动的数据量。这反过来也会消除数据传输期间浪费的能源。闪存单元运行时产生的有功功率消耗较低，在待机模式下几乎不消耗能量，因此可以进一步降低能耗。

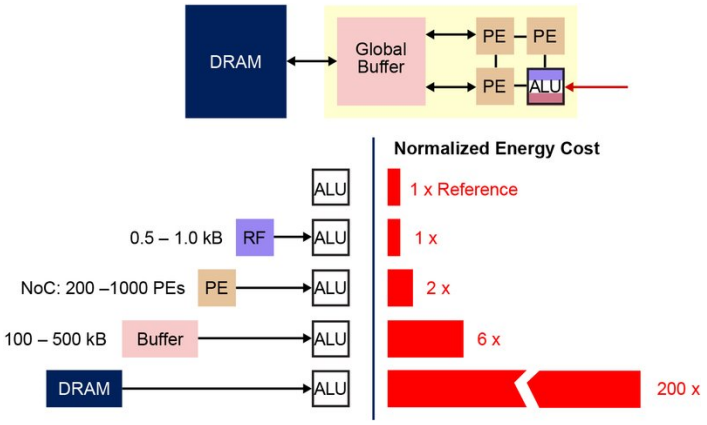

图 3：机器学习计算中的存储器瓶颈

来源：Y.-H. Chen、J. Emer 和 V. Sze 于 2016 国际计算机体系结构研讨会发表的“Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks”。

该方法的一个示例是 Microchip 子公司 Silicon Storage Technology（SST）的 memBrain™技术。该解决方案依托于 SST 的 SuperFlash®存储器技术，这项技术已成为适用于单片机和智能卡应用的多级存储器的公认标准。这种解决方案内置一个内存内计算架构，允许在存储权重的位置完成计算。权重没有数据移动，只有输入数据需要从输入传感器（例如摄像头和麦克风）移动到存储器阵列中，因此消除了 MAC 计算中的存储器瓶颈。

这种存储器概念基于两大基本原理：(a)晶体管的模拟电流响应基于其阈值电压（Vt）和输入数据，(b)基尔霍夫电流定律，即在某个点交汇的多个导体网络中，电流的代数和为零。了解这种多级存储器架构中的基本非易失性存储器（NVM）位单元也十分重要。下图（图 4）是两个 ESF3（第 3 代嵌入式 SuperFlash）位单元，带有共用的擦除门（EG）和源线（SL）。每个位单元有五个终端：控制门（CG）、工作线（WL）、擦除门（EG）、源线（SL）和位线（BL）。通过向 EG 施加高电压执行位单元的擦除操作。通过向 WL、CG、BL 和 SL 施加高/低电压偏置信号执行编程操作。通过向 WL、CG、BL 和 SL 施加低电压偏置信号执行读操作。

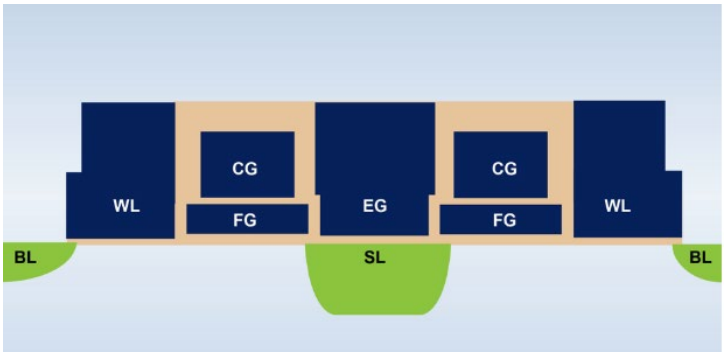

图 4：SuperFlash ESF3 单元

利用这种存储器架构，用户可以通过微调编程操作，以不同 Vt 电压对存储器位单元进行编程。存储器技术利用智能算法调整存储器单元的浮栅（FG）电压，以从输入电压获得特定的电流响应。根据最终应用的要求，可以在线性区域或阈下区域对单元进行编程。

图 5 说明了在存储器单元中存储多个电压的功能。例如，我们要在一个存储器单元中存储一个 2 位整数值。对于这种情况，我们需要使用 4 个 2 位整数值（00、01、10、11）中的一个对存储器阵列中的每个单元进行编程，此时，我们需要使用四个具有足够间隔的可能 Vt 值之一对每个单元进行编程。下面的四条 IV 曲线分别对应于四种可能的状态，单元的电流响应取决于向 CG 施加的电压。

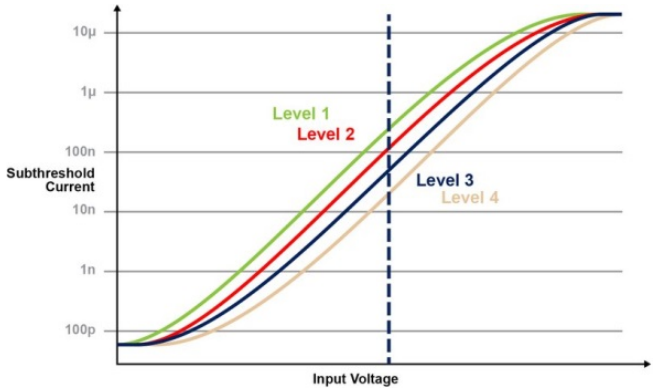

图 5：ESF3 单元中的编程 Vt 电压

受训模型的权重通过编程设定为存储器单元的浮栅 Vt。因此，受训模型每一层（例如完全连接的层）的所有权重都可以在类似矩阵的存储器阵列上编程，如图 6 所示。对于推断运算，数字输入（例如来自数字麦克风）首先利用数模转换器（DAC）转换为模拟信号，然后应用到存储器阵列。随后该阵列对指定输入向量并行执行数千次 MAC 运算，产生的输出随即进入相应神经元的激活阶段，随后利用模数转换器（ADC）将输出转换回数字信号。然后，这些数字信号在进入下一层之前进行池化处理。

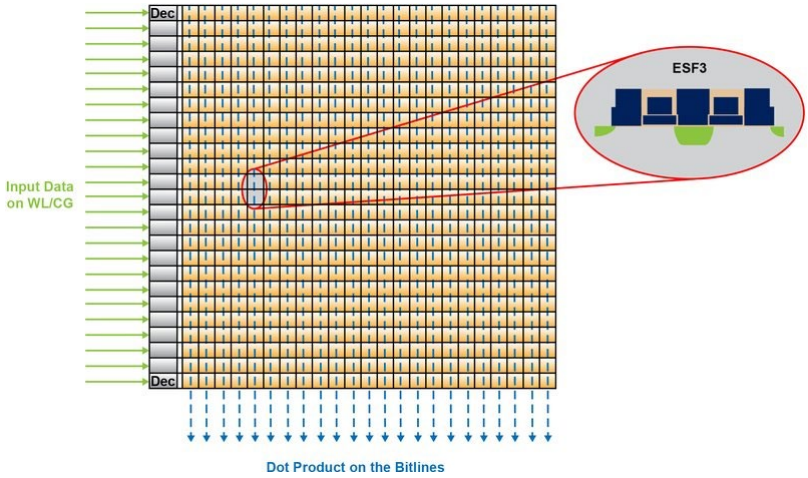

图 6：用于推断的权重矩阵存储器阵列

这类多级存储器架构模块化程度非常高，而且十分灵活。许多存储器片可以结合到一起，形成一个混合了权重矩阵和神经元的大型模型，如图 7 所示。在本例中，MxN 片配置通过各片间的模拟和数字接口连接到一起。

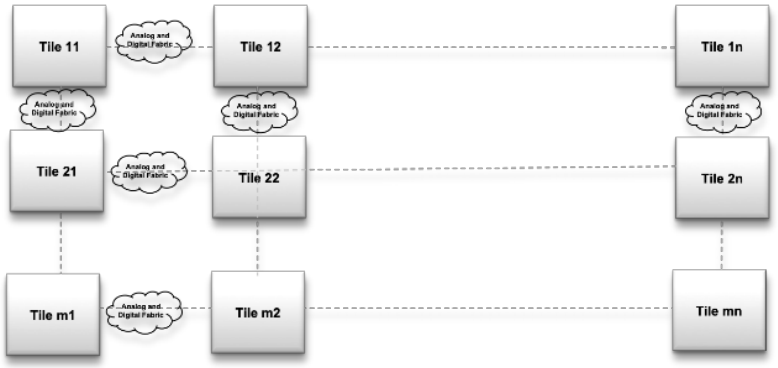

图 7：memBrain™的模块化结构

截至目前，我们主要讨论了该架构的芯片实施方案。提供软件开发套件（SDK）可帮助开发解决方案。除了芯片外，SDK 还有助于推断引擎的开发。SDK 流程与训练框架无关。用户可以在提供的所有框架（例如 TensorFlow、PyTorch 或其他框架）中根据需要使用浮点计算创建神经网络模型。创建模型后，SDK 可帮助量化受训神经网络模型，并将其映射到存储器阵列。在该阵列中，可以利用来自传感器或计算机的输入向量执行向量矩阵乘法。

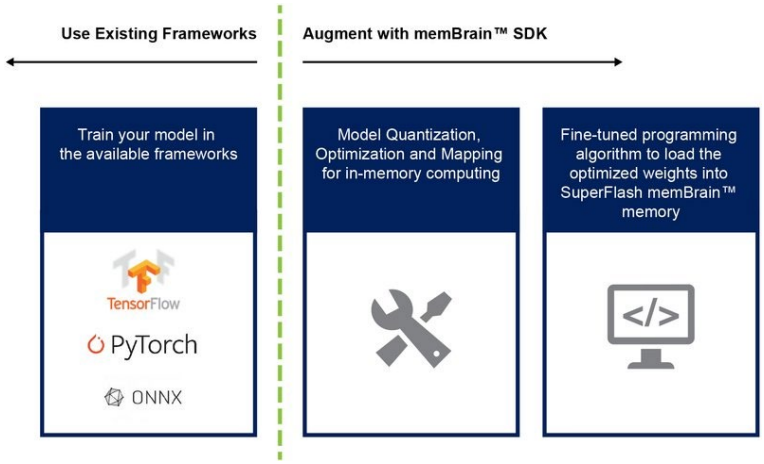

图 8：memBrain™ SDK 流程

多级存储器方法结合内存内计算功能的优点包括：

1. ***超低功耗：***专为低功耗应用设计的技术。功耗方面的第一个优点是，这种解决方案采用内存内计算，因此在计算期间，从 SRAM/DRAM 传输数据和权重不会浪费能量。功耗方面的第二个优点是，闪存单元在阈下模式下以极低的电流运行，因此有功功率消耗非常低。第三个优点是待机模式下几乎没有能耗，原因是非易失性存储器单元不需要任何电力即可保存始终开启设备的数据。这种方法也非常适合对权重和输入数据的稀疏性加以利用。如果输入数据或权重为零，则存储器位单元不会激活。
2. ***减小封装尺寸：***该技术采用分离栅（1.5T）单元架构，而数字实施方案中的 SRAM 单元基于 6T 架构。此外，与 6T SRAM 单元相比，这种单元是小得多。另外，一个单元即可存储完整的 4 位整数值，而不是像 SRAM 单元那样需要 4*6 = 24 个晶体管才能实现此目的，从本质上减少了片上占用空间。
3. ***降低开发成本：***由于存储器性能瓶颈和冯·诺依曼架构的限制，很多专用设备（例如 Nvidia 的 Jetsen 或 Google 的 TPU）趋向于通过缩小几何结构提高每瓦性能，但这种方法解决边缘计算难题的成本却很高。采用将模拟内存内计算与多级存储器相结合的方法，可以在闪存单元中完成片上计算，这样便可使用更大的几何尺寸，同时降低掩膜成本和缩短开发周期。

边缘计算应用的前景十分广阔。然而，需要首先解决功耗和成本方面的挑战，边缘计算才能得到发展。使用能够在闪存单元中执行片上计算的存储器方法可以消除主要障碍。这种方法利用经过生产验证的公认标准类型多级存储器技术解决方案，而这种方案已针对机器学习应用进行过优化。

作者简介

Vipin Tiwari 在产品开发、产品营销、业务开发、技术许可、工程管理以及存储器设计方面拥有 20 多年的丰富经验。目前，Tiwari 先生在 Microchip 的子公司 Silicon Storage Technology, Inc.担任嵌入式存储器产品开发总监。

###