以块随机存取存储器换取查找表：面向基于FPGA的脉冲神经网络的混合位级权重存储

Dimitrios Sekertzis 和 Giorgos Dimitrakopoulos 集成电路实验室，电气与计算机工程系，色雷斯德谟克利特大学（DUTH），希腊

arXiv:2611.32288v2 [cs.AR] 25 Nov 2026

摘要—**脉冲神经网络（SNN）为能效高的人工智能推理提供了一种颇具吸引力的范式，而将其部署于可重构硬件上，则能够在不牺牲模型可重编程性的情况下实现面向具体应用的加速。然而，在现场可编程门阵列上实现这一潜力，往往会受到存储突触权重所需的大量片上存储器的严重限制。现有压缩技术虽然能够降低存储需求，但会以权重精度或准确率为代价。本文提出一种混合位级权重存储方案，在保持原始权重不变的同时，以现场可编程门阵列逻辑资源换取块随机存取存储器容量。该方案使用查找表实现选定的权重位，其余权重位则存储在块随机存取存储器中，并在推理过程中联合重构。我们将这一方法集成到空间数据流脉冲神经网络架构中，并引入一种实用的位映射启发式方法，在不修改权重的情况下，高效地将权重位划分到块随机存取存储器和基于查找表的逻辑中。实验评估表明，该方法可使块随机存取存储器利用率降低超过26%，同时仅小幅增加查找表和寄存器用量，最终显著降低整体功耗。**

I. 引言

传统机器学习日益增长的能量需求重新引发了人们对脉冲神经网络（SNN）的兴趣，其稀疏的事件驱动计算为实现高能效的人工智能推理提供了颇具吸引力的路径。在单芯片上部署不同脉冲神经网络模型的灵活性得以保留的同时，实现这一潜力需要作出关键的架构选择。可编程神经形态专用集成电路（如 Loihi2 [1] 和 SpiNNaker 2 [2], ）提供了通用灵活性，但其复杂的路由网络和类似软件的可编程性引入了大量硬件开销，可能限制能效。现场可编程门阵列通过利用可重构逻辑结构，而无需将通用神经形态基础设施嵌入加速器本身，从而在效率与可编程性之间实现了有吸引力的平衡 [3], [4]⁋[5]。这使开发人员能够综合完全针对具体应用的脉冲神经网络架构，使其适配目标网络的确切拓扑结构，以实现最大效率，同时还可通过 FPGA比特流重配置部署不同模型 [6], [7]。

尽管结构效率与硬件灵活性实现了理想的结合，但在实践中，基于现场可编程门阵列的脉冲神经网络加速器的扩展通常受到片上存储器有限容量的瓶颈制约 [7], [8]。除了跟踪神经元状态和中间数据外，加速器还必须容纳数量庞大的突触权重。将完整的网络拓扑结构保存在片上至关重要，因为从外部存储器获取权重会显著增加功耗和推理延迟，进而抵消脉冲神经网络的核心优势，从而抵消脉冲神经网络的核心优势 [9]。因此，最小化突触权重的片上存储器占用，已成为实现可扩展且高效的基于现场可编程门阵列的脉冲神经网络推理的关键要求。

现有方法通常通过减少表示权重所需的信息量来解决这一限制。剪枝 [10], 量化 [11],稀疏权重 [12], 和向量量化 [13] 最初主要针对深度神经网络开发，如今也已应用于脉冲神经网络。剪枝会移除被认为不那么重要的权重；量化使用更少的位表示权重；稀疏权重则通过适当的元数据仅编码非零权重；向量量化则将一组权重映射为共享码本中的紧凑索引。尽管这些技术能够降低存储需求，却是以推理准确率为代价换取存储器节省。

本文提出一种不同的方法，在不修改权重或牺牲精度的前提下，减少存储在 FPGA 块随机存取存储器中的权重位数。其核心思想是利用 FPGA 资源的异构特性，将每个权重的位分配到存储器和可编程逻辑中：选定的位由查找表在线生成，其余位则存储在块随机存取存储器中。推理过程中，这两个来源的数据会重构出完整权重，从而有效地以查找表资源换取块随机存取存储器容量，同时保留原有的权重表示。虽然将参数和运算嵌入查找表已应用于传统的基于逻辑的神经网络 [14]⁋[15],但这些无状态的组合逻辑技术无法支持具有状态的膜电位和时间反馈，这些特性是脉冲神经网络所固有的——用于脉冲神经网络 [16]。因此，专门将查找表用于在线权重生成是一种针对脉冲神经网络的新颖优化。

我们将所提出的混合权重存储方案集成到空间数据流脉冲神经网络架构 [4], [7], [17], 中，其中网络层和神经元被物理映射到现场可编程门阵列上。在设计阶段，我们采用一种实用的启发式方法，按网络层独立地在块随机存取存储器和基于查找表的逻辑之间高效划分权重位。本文的主要贡献概括如下：

- 一种面向现场可编程门阵列脉冲神经网络的混合权重存储方案，将突触权重位分配在块随机存取存储器与基于查找表的在线权重生成逻辑之间，同时保持原有权重精度。实用的启发式方法可确定这两种资源之间的位分配。
- 一种支持所提出分拆存储范式的空间数据流脉冲神经网络架构，其中实际权重

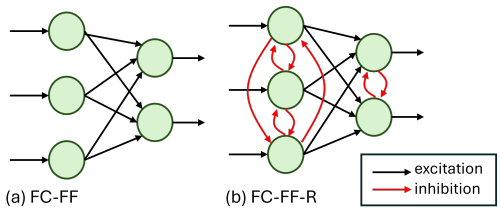

图 1. 脉冲神经网络拓扑结构：(a) 全连接（FC）前馈（FF）网络；(b) 带循环（R）侧向抑制的全连接（FC）前馈（FF）网络。

由存储在块随机存取存储器中的位和逻辑生成的位即时重构。

- 在现场可编程门阵列上实现脉冲神经网络，并在具有代表性的脉冲神经网络模型上与最先进的加速器进行评估。结果表明，所有情况下块随机存取存储器均显著节省超过 26%，代价是查找表和寄存器利用率略有增加。虽然所有测试案例的功耗均有所改善，但若仅比较在同一现场可编程门阵列器件上实现的脉冲神经网络，功耗平均降低 45%。

II. 脉冲神经网络基础与空间数据流现场可编程门阵列架构

脉冲神经网络通过离散、异步的二进制脉冲（称为脉冲）进行通信，以时间方式处理信息。要处理连续的现实世界数据，必须先将输入值映射为一定数量时间步内的脉冲序列。输入编码通常采用频率编码（脉冲频率与输入强度成正比）、时间编码（由精确的脉冲时序承载信息）或直接编码（将模拟输入直接施加到第一层的膜电位）。编码完成后，这些脉冲序列沿网络拓扑结构传播，驱动神经元的内部动态。

A. 漏积分发放神经元模型

漏积分发放（LIF）模型因兼顾生物学合理性与计算效率，被广泛用于硬件实现。该模型省去了霍奇金–赫胥黎模型和伊日凯维奇模型等高保真模型难以承受的复杂性 [18], [19]。LIF 神经元通过将输入脉冲对应的突触权重累加到膜电位中来进行积分，同时膜电位会随时间持续泄漏一部分，并逐渐趋向静息状态。

在 LIF 模型中，神经元 j 在时间步 t 的膜电位 $V_j[t]$ 离散时间更新公式为：

$$V_j[t] = V_j[t-1] + \sum_i w_{ji} s_i[t] - \lambda V_j[t-1] \qquad (1)$$

其中，λ表示泄漏因子，第二项是突触权重 w_{ji} 与来自突触前神经元的输入二进制脉冲 $s_i[t]$的内积。为尽量降低硬件复杂度，泄漏因子 λ 可以使用二的负幂（例如 $\lambda = 2^{-L}$）进行近似，在精度损失最小的情况下，如 [17] 所示，使衰减能够

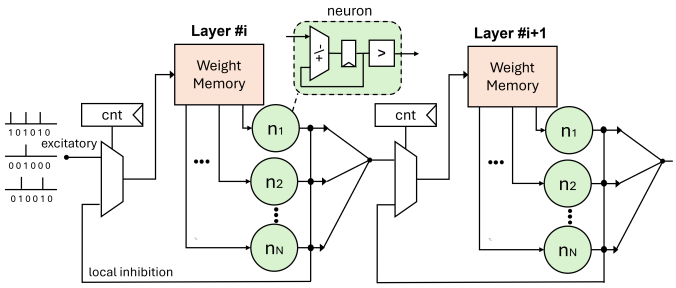

图 2. 采用专用神经元数据通路和各层权重存储器的空间数据流脉冲神经网络架构模板。

通过简单的按位右移而非代价高昂的乘法器高效实现。

状态更新后，将膜电位与发放阈值 V_{th} 进行比较。当膜电位达到或超过该阈值时，便会产生输出脉冲 $s_j[t]$：

$$s_j[t] = \begin{cases} 1, & V_j[t] \geq V_{\text{th}}, \\ 0, & V_j[t] < V_{\text{th}}. \end{cases} \qquad (2)$$

在脉冲发放后，必须重置膜电位，以防止神经元持续发放。在减法（软）复位中，只需从当前膜电位中减去阈值，从而保留任何残余电荷。或者，硬复位会将膜电位强制设定为预定义的静息状态。尽管复位机制的选择会影响脉冲序列的时间动态，但本文所有模型均采用将膜电位硬复位至固定的可训练复位参数。

B. 网络拓扑与量化

脉冲神经网络要实现较高的推理准确率，通常需要具有较强表达能力的网络拓扑。一种常见选择是采用带有循环侧向抑制的全连接前馈网络，使活跃神经元能够抑制其邻近神经元（图 1）。尽管脉冲神经网络通常使用全精度浮点运算进行训练，但推理可以采用有限精度整数或定点表示，从而避免浮点运算显著更高的硬件成本 [11]。

量化通过将网络变量转换为有限精度表示来实现这一转变，包括突触权重、膜电位、发放阈值和泄漏计算。对于硬件实现，量化最终会将每个突触权重表示为固定宽度的二进制值，在降低存储需求的同时简化算术数据通路。通过剪枝 [10] 和稀疏权重表示 [12], 等技术，还可以进一步降低权重存储需求，因为这些技术减少了必须存储的权重值数量。

C. 同步空间数据流 *FPGA* 脉冲神经网络架构

为了将脉冲神经网络模型高效映射到现场可编程门阵列上 [4], [5], [7], ，我们采用类似于 [17] 的同步空间数据流架构。如图 2 所示，完整的网络拓扑结构在现场可编程门阵列中实例化，每一层都分配有

专用的计算数据通路和权重存储器。该架构遵循时钟驱动执行模型，每一层在一个时间步内依次处理其输入脉冲，随后进入状态评估阶段。

通过这种设计，每个时钟周期恰好处理一个激活脉冲；一旦所有输入中都不再有激活脉冲，就可以提前退出。对于每个选定的激活脉冲，都会从存储器中读取对应的突触权重行，并将其广播给目标层中的所有神经元，使它们能够在同一周期内并行更新膜电位。首先处理来自前一层的兴奋性脉冲，随后使用相同的数据通路处理来自同一层内部的抑制性脉冲。由于这种顺序处理方式保证不会有两个输入脉冲同时到达同一个神经元，因此神经元的物理架构得以大幅简化。实际上，每个神经元都简化为一个基本的加法器或减法器，在每个激活周期执行一次算术运算：兴奋时加上突触权重，抑制时减去突触权重。

处理完当前时间步的所有激活输入脉冲后，该层进入状态评估阶段。将膜电位与 V_{th} 进行比较，以生成输出脉冲。发放的神经元会执行硬复位，随后通过前述二进制右移实现膜电位泄漏。

只有在所有层都完成状态评估阶段后，执行才会进入下一个时间步。

III. 混合权重存储架构

为减少脉冲神经网络加速器庞大的片上存储器占用，所提出的架构以更小的存储器取代传统的逐层权重存储器，该存储器仅存储部分权重位。其余位则通过组合逻辑动态生成，从而减少块随机存取存储器的使用量，同时保留原始权重表示。

A. 组合权重生成逻辑

考虑一个简化的层，其中包含三个输入脉冲（x_1、 x_2、x_3）和两个神经元（n_1、 n_2）。在图 3(a)所示的传统存储器设计中，该层使用一个包含三行的权重存储器。在处理阶段，当x_1 激活时，读取第一行，并将其权重广播至 n_1 和 n_2；类似地，当 x_2 和 x_3 分别激活时，读取第二行和第三行。

对于神经元 n_1，激活 x_1 时选择 4 位权重 0111，激活 x_2 时选择 0100，激活 x_3 时选择 0010。检查三个可能输入对应的各个位，可以发现直接的逻辑关系：

- 位 3（最高有效位）对所有输入均为“0”，因此可以直接连接到地。
- 位 2 仅在 x_1 或 x_2 处于激活状态时为“1”。
- 位 1 在输入 x_1 或 x_3 处于激活状态时为“1”。
- 位 0（最低有效位）仅在 x_1 处于激活状态时为“1”。

如图 3(b) 所示，这些位级关系可直接转换为组合逻辑，用于生成

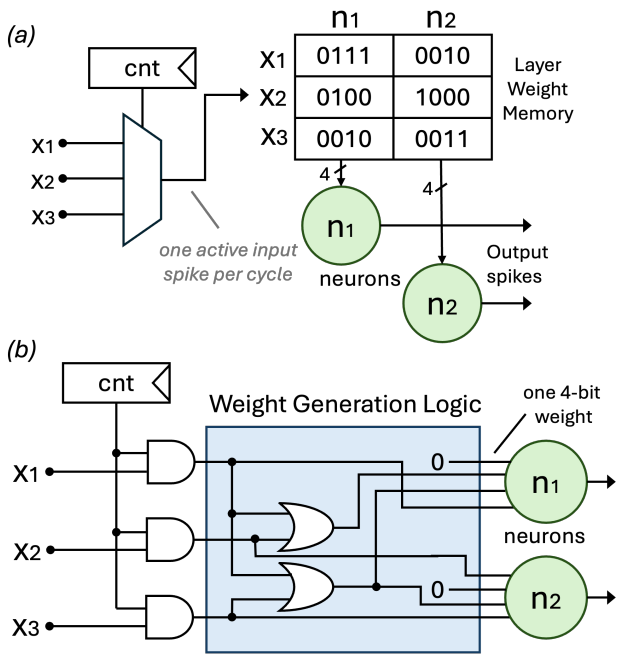

图 3. 含有三个输入脉冲和两个神经元的层的权重访问：(a) 基于块随机存取存储器的传统权重读取；(b) 由至多一个激活输入脉冲驱动的组合式权重生成。

两个神经元的权重。与掩码门和环形计数器确保每个时钟周期仅选择一个输入脉冲，因此逻辑只需生成与所选输入相对应的权重。这样，无需将权重存储在存储器中，即可为所有神经元生成所需的权重。

然而，对于具有数百或数千个输入的实际层，在逻辑中生成所有权重位并不切实际，因为由此产生的查找表网络可能会占用大量资源，造成布线拥塞，并带来较大的组合延迟。这种扩展性限制促使我们采用混合实现，将基于查找表的生成与传统块随机存取存储器存储相结合。

B. 混合权重存储架构

如图 4 所示，所得混合数据通路在推理过程中结合了两种权重信息来源。当选中一个激活脉冲时，其位置会被编码，并行提供给权重生成逻辑和块随机存取存储器。编码后的位置用于确定相应的块随机存取存储器行，该行提供目标层中所有神经元的存储位；与此同时，权重生成逻辑生成其余位。随后，这两个位子集会在神经元接口处拼接，以重构原始权重表示。

同一机制也天然支持用于侧向抑制的层内循环连接，这类连接需要负权重。为避免存储或生成冗余的符号位，块随机存取存储器和基于查找表的逻辑都只提供无符号权重幅值。符号则由数据通路中的连接类型决定：兴奋性脉冲会使膜电位增加，而抑制性

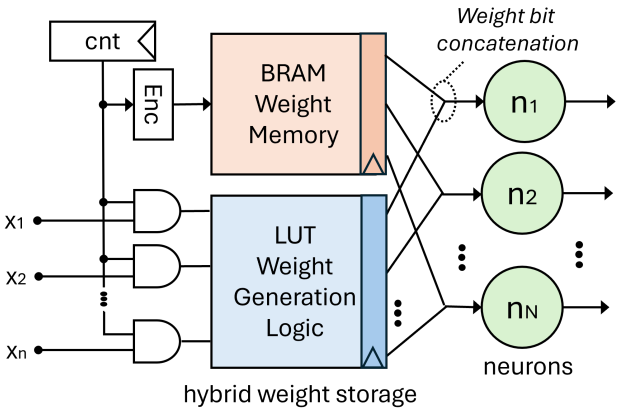

图 4. 脉冲神经网络层权重的混合权重存储（块随机存取存储器）与在线生成（查找表）。

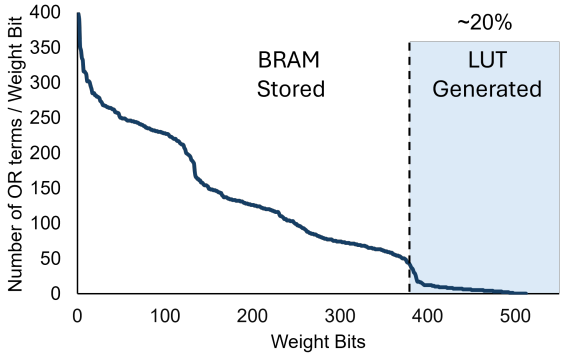

图 5. MNIST层中 $N \times W = 128 \times 4 = 512$ 权重位的逻辑生成成本，以所需或项的数量衡量。各位按成本降序排列，其中成本最低的位映射到查找表，其余位存储在块随机存取存储器中。

脉冲会使膜电位减小。这样便将权重幅值与符号解耦，同时使用同一权重存储机制支持前向连接和循环连接。

C. LUT/BRAM间权重位的启发式分配

为了在存储器和逻辑之间有效划分神经网络权重，我们采用了一种实用的分配策略。该策略根据设计者指定的存储器缩减目标，通过启发式方法评估并分配各个权重位。它将逻辑综合成本最低的位映射到查找表，其余位则存储在块随机存取存储器中。

一个包含 N 个神经元、 L 个可能输入脉冲和 W 位权重的层，总计包含 $N \times L \times W$ 位。每个周期会选中 L个输入脉冲中的一个，每个N 个神经元都会接收该输入对应的、取值独立的 W 位权重。因此，每个周期需要 $N \times W$个权重位，这些位可分别由块随机存取存储器、基于查找表的逻辑，或两者组合提供。

每个 $N \times W$ 个权重位的逻辑生成成本，取决于生成该位所需的输入相关或项数量。由于每个周期只有一个输入脉冲处于激活状态，生成的位是所选输入信号的逻辑或；当相应权重中的该位为“1”时，该输入信号会被选中。因此，或项数量直接反映输入依赖关系，并可用于衡量逻辑成本。

图 5 展示了 MNIST 层（$N = 128$,$W = 4$）的这些开销。每个点代表一个 $N \times W$ 权重位， y轴表示生成该权重位所需的或项数量。按照开销对这些权重位排序，右侧是开销最低的权重位，左侧是开销最高的权重位。例如，当目标查找表分配比例为 20% 时，该启发式方法会从分布中开销最低的一端选取 20% 的权重位，采用基于查找表的生成方式，其余权重位则存储在块随机存取存储器中。选出的子集随后作为权重生成逻辑的输入。尽管跨位位置共享逻辑有可能降低依赖度较高的权重位的开销，但我们的实验表明，选择或项最少的权重位能够实现有效映射。将更多权重位分配给查找表可能会加剧布线拥塞，并影响时序收敛。当前实现仅考虑单周期、非流水线的权重生成逻辑。

选择用于基于查找表的生成的位后，使用 Paar 算法 [20]对其布尔表达式进行优化。该算法识别各位表达式之间的公共输入项，并递归地将其组合成更大的子表达式，从而减少所需逻辑门的数量。在标准逻辑综合之前应用此优化，以获得高效的硬件实现。

IV. 实验评估

A. 评估设置

为了评估所提出架构的整体资源利用率和效率，我们在三个数据集上对实现进行了基准测试：MNIST [21]，、AudioMNIST [22]，和 SHD [23]。所有网络模型均使用 SNNTorch 库 [24],进行训练，完整的训练细节、超参数和网络拓扑具体信息均在 [25]中公开提供。

我们将所提出的设计与针对各个具体模型案例选取的、资源效率高且达到最先进水平的 FPGA 脉冲神经网络加速器进行比较。对于这些基线架构，所有硬件利用率和推理准确率指标均直接按照其各自论文中发表的结果报告：

- 小型 MNIST 模型：我们的实现与 Spiker+ [17]和 ModNef [7],进行了基准测试；二者在相同的 FPGA 平台上采用了类似的网络架构，此外还包括 TripleBrain [26] 和 MorphBungee-Tiny [27]。
- 大型 MNIST 模型：我们将其与 ModNef [7]以及 Chen 等人提出的加速器 [28]进行了比较。
- 时序工作负载（AudioMNIST [22] 和 SHD [23]）：为了进一步评估该架构在多样化时变数据集上的有效性，我们扩展了评估范围，并将结果直接与 Spiker+ [17]进行比较。

在这些比较中，我们评估脉冲神经网络模型的准确率及相关硬件指标，包括资源利用率（查找表数量、触发器数量、块 RAM 数量）、功耗、能效

表 I 使用所提出的混合 BRAM/LUT 权重存储的 MNIST 脉冲神经网络模型结果与最先进的 FPGA 加速器的比较。

数据集	MNIST					大型 MNIST		
实现	所提出的方法 L1：23% L2：100%	Spiker+ [17]	ModNEF [7]	TripleBrain [26]	MorphBungee Tiny [27]	所提出的方法 L1：23% L2：22% L3：100%	ModNEF [7]	[28]
Zynq FPGA	7020	7020	7010	7045	7010	7020	7010	7045
SNN拓扑结构	784-128-10	784-128-10	784-128-10	784-256	256-256-10	784-392-196-10	784-392-196-10	784-512-100
准确率（%）	95.60	93.85	95.76	95.10	97.55	94.50	96.96	97.21
网络类型	FC-FF-R	FC-FF	FC-FF	FC-FF	FC-FF	FC-FF-R	FC-FF	FC-FF
势比特	11	6	N.R.[a]	24	16	12	16	L1(13), L2(21)
权重位	4	4	10	16	12	4	8	L1(8), L2(16)
LUTs	7794	4314	5947	10052	8755	31263	20321	17578
FFs	4466	3298	3060	8505	2970	14521	28284	3877
DSPs	-	-	-	32	-	-	3	-
36 Kb 块 RAM	11	18	42	131	25	84	113	137
18 Kb 块 RAM	-	-	1	-	1	-	1	1
功耗 (mW)	134	180	141	938	139	193	336	398
能量 (mW/MHz)	1.34	1.80	1.12	3.75	1.39	1.93	2.68	3.98
频率 (MHz)	100	100	125	250	100	100	125	100
每张图像的延迟 (ms)	0.259	0.780	0.115	0.370	0.253	0.467	0.904	0.467

[a]N.R.：未报告

（毫瓦/兆赫兹）、实现的时钟频率以及每个输入的延迟。所提出的 FPGA 脉冲神经网络采用 SystemVerilog 寄存器传输级设计，在 [25] 下以允许使用的许可证开源，并针对 AMD Zynq-7020 片上系统进行综合。综合、实现和功耗评估均使用 Xilinx Vivado 完成。报告的功耗是通过实现后仿真得出的平均功耗，仿真使用了各脉冲神经网络模型测试集中的代表性、类别均衡子集。

B. 比较

表 I 总结了小型和大型 MNIST 模型的结果，其中所提出的方法列于第一列，其余列列出最先进的 FPGA 加速器。所提出的设计采用 L1:23%、L2:100% 配置，这意味着第一层中 23% 的权重位和第二层中的全部权重位均使用查找表在线生成。该配置代表了一种实用的设计方案，在保持 FPGA 资源利用率均衡的同时，可显著节省块随机存取存储器。

对于每种实现，表中均报告了完整的脉冲神经网络拓扑结构，包括输入连接数、主层和次层的大小，以及膜电位和权重的数据位宽。Spiker+ 和ModNEF采用与所提出设计相同的网络拓扑，其余实现则采用复杂度相当的拓扑结构。尽管硬件组织方式不同，所提出的设计在所有设计中均保持了具有竞争力的推理准确率。

所提出方法的主要优势是显著降低了块随机存取存储器的使用量。在所选配置下，与 Spiker+ 相比，块 RAM 数量减少了 38%；与 TripleBrain 相比，最多减少了 91%。这一减少是以 LUT/FF 平均利用率比最先进的 FPGA 加速器高 17% 为代价的；当片上块随机存取存储器是主要资源瓶颈时，这种权衡是有利的。

内存占用的减少也降低了功耗；在所有评估的实现方案中，所提出的设计功耗最低。对于采用相同或更小型现场可编程门阵列器件的实现方案，例如分别面向 Zynq 7020 和 7010 的方案，其绝对功耗值可以直接比较。能效（mW/MHz）也呈现出类似趋势，不过 ModNEF 的表现略好。就每次输入的延迟而言，ModNEF 的平均延迟最低，而所提出的设计紧随其后，性能与 MorphBungee-Tiny 相当。

随着脉冲神经网络规模增大，所提出架构的优势愈发明显，因为片上块随机存取存储器逐渐成为一项限制。表 I 也报告了更大型 MNIST 模型的结果。所提出的设计与 ModNEF 使用相同的拓扑结构，由三个大小相同的层组成，而 [28] 使用两个较大的层。与其他模型相比，所提出的设计准确率略有下降，这主要归因于训练过程，而非硬件架构。与 ModNEF 相比，所提出的设计将块随机存取存储器利用率降低了 26%，与 [28], 相比则降低了 39%，同时使用的查找表和触发器总量也少于 ModNEF。此外，该设计的功耗最低、能效最高，且延迟与所比较的设计相当或更低。

表 II 对比了所提出的架构与 Spiker+在 AudioMNIST 和 SHD 数据集上的表现。这一比较尤为相关，因为两种设计面向相同的 FPGA

表 II 所提出的脉冲神经网络与 Spiker+ 在 AudioMNIST 和 SHD 数据集上的实验结果。

数据集	AudioMNIST		SHD	
实现	所提出的方法 L1：28% L2：100%	Spiker+ [17]	所提出的 L1：20% L2：100%	Spiker+ [17]
Zynq FPGA	7020	7020	7020	7020
SNN拓扑结构	40-150-10	40-150-10	700-200-20	700-200-20
准确率（%）	92.60	89.82	72.20	72.99
网络类型	FC-FF-R	FC-FF	FC-FF-R	FC-FF-R
势比特	12	8	14	8
权重位	6	5	7	触发器（6）/ 寄存器（5）
LUTs	8603	10124	14725	18268
FFs	3598		6675	
DSPs	-	-	-	-
36 Kb 块 RAM	9	16	31	51
18 Kb 块 RAM	-	-	-	-
功耗（mW）	146	290	151	430
能量（mW/MHz）	1.46	2.90	1.51	4.30
频率（MHz）	100	100	100	100
延迟/录音（毫秒）	0.329	0.080	0.574	0.540

平台，时钟频率均为 100 MHz，且采用相似的网络拓扑。所提出的设计在 AudioMNIST 上的准确率高出 2.78%，而在 SHD 上仅比 Spiker+ 低 0.8%。除 AudioMNIST 的 Spiker+模型外，所有配置均使用循环连接。

所提出的架构在两个数据集上均实现了最低的块随机存取存储器利用率，其中 AudioMNIST 和 SHD 分别降低了 43% 和 39%。这些节省分别需要增加 20% 和 17% 的查找表/触发器。内存占用的降低也使两个数据集的功耗均达到最低。在延迟方面，Spiker+在 AudioMNIST 上略胜一筹，而在 SHD 上，两种设计实现了几乎相同的延迟。

总体而言，评估结果证实，所提出的混合权重存储方法成功地以存储容量换取可编程逻辑。这种架构转变实现了高内存效率的脉冲神经网络加速，同时在不同网络规模和数据集上保持了具有竞争力的准确率、功耗和延迟。

C. 混合权重存储的局限性

为评估这种权衡能够推进到何种程度，表 III 报告了针对每个脉冲神经网络模型、在保持可布线性和目标时钟频率的同时实现最大块随机存取存储器削减的配置。超过这些限度后，进一步削减块随机存取存储器将导致设计无法布线，或无法在 100 MHz 下实现时序收敛。

所实现的块随机存取存储器减少幅度显著，与表 I–II 中报告的均衡配置相比，内存使用量减少了约一半；而这些均衡配置相较最先进水平已实现显著节省。然而，这种激进的削减不可避免地增大了权重生成逻辑的规模，部分抵消了降低内存使用量带来的功耗优势

表 III 在 100MHz 下 ZYNQ 7020 上针对所有经检验的脉冲神经网络模型实现的设计方案及其最大块随机存取存储器节省量。

数据集	MNIST	大型 MNIST	AudioMNIST	SHD
实现	L1：61% L2：100%	L1：31% L2：31% L3：100%	L1：80% L2：100%	L1：36 % L2：100%
LUTs	10561	32450	12433	17423
FFs	4663	14719	4067	6886
36 Kb 块 RAM	5	75	2	25
18 Kb 块 RAM	1	-	1	-
功耗（mW）	137	193	146	153

。尽管表 III 中的设计在大多数情况下仍比最先进水平功耗更低，但其功耗比我们的均衡配置略高，不足 2.5%。因此，必须谨慎权衡存储器与逻辑之间的取舍，以避免组合逻辑过度增长。

更广泛地说，表 III 表明，所提出的架构能够通过在块随机存取存储器与查找表之间转移权重存储需求，适应可用的 FPGA 资源。这种灵活性对于资源受限的设备尤其有价值：即使仍有可编程逻辑可用，块随机存取存储器的容量也可能限制可部署脉冲神经网络的规模。因此，所提出的方法为扩展特定 FPGA 可支持的脉冲神经网络模型范围提供了一种实用手段。

V. 结论与未来工作

在这项工作中，我们为数字 FPGA 脉冲神经网络引入了一种新颖的架构范式，可在可编程逻辑中直接动态生成一部分指定的突触权重。该方法采用启发式划分策略评估逻辑综合成本，并将各个权重位高效分配至可用硬件资源（即块 RAM 和查找表）。由此实现的设计显著降低了内存占用和整体功耗，同时不影响推理准确率或处理吞吐量。

展望未来，这一混合框架为进一步改进提供了明确的方向。未来研究将探索硬件与算法协同设计，重点关注定制权重编码和专门的训练约束。例如，将权重调整为具有天然稀疏位模式的取值（如二的幂），可以主动降低所需的逻辑 1 密度。此外，还可以扩展分配算法，将物理综合和布线指标直接纳入选择循环，进一步优化逻辑与存储器之间的分配。

参考文献

[1] S. B. Shrestha、J. Timcheck、P. Frady、L. Campos-Macias 和 M. Davies，“利用 Loihi 2 高效处理视频与音频”，载于*IEEE*国际声学、语音与信号处理会议 *(ICASSP)*，2024年，第 13 481–13485 页。

[2] S. Scholze、J. Partzsch、S. H¨oppner、F. Kelber、A. Dixius、M. Stolba、S. Arfa、M. Berthel、G. Ellguth、J. Garside等，“SpiNNaker2芯片：用于灵活且可扩展的脑启发计算的众核平台”，*IEEE*开放电路与系统期刊，2026年。[3] J. Li等，“FireFly v2：采用时空现场可编程门阵列加速器，为高性能脉冲神经网络推进硬件支持”，*IEEE*计算机辅助设计汇刊，第 43 卷，第 9 期，第 2647–2660 页，2024年。[4] A. Khodamoradi、K. Denolf 和 R. Kastner，“S2n2：用于流式脉冲神经网络的现场可编程门阵列加速器”，载于*ACM*国际现场可编程门阵列研讨会，2021年，第 194–205 页。[5] M. Karamimanesh等，“现场可编程门阵列上的脉冲神经网络：方法与最新进展综述”，*Neural Networks*，第 186 卷，第 107256 页，2025年。[6] A. Carpegna、A. Savino 和 S. Di Carlo，“Spiker：面向脉冲神经网络的现场可编程门阵列优化硬件加速器”，载于*2022*年*IEEE*计算机学会年度*VLSI*研讨会 *(ISVLSI)*，2022年，第 14–19 页。[7] A. Saulquin等，“ModNEF：面向低功耗边缘人工智能的开源现场可编程门阵列模块化神经形态仿真器”，*ACM TACO*，第 22 卷，第 2 期，第 1–24 页，2025年。[8] M. A. Lebdeh、K. S. Yildirim 和 D. Brunelli，“面向存储器高效脉冲神经网络的逐层专用化”，载于国际神经形态系统会议 *(ICONS)*，2024年，第 169–176 页。[9] D. Le、B. Arig、M. Isik、I. C. Dikmen 和 T. Karadag，“神经形态计算中的存储墙综述”，载于*IEEE*国际计算与机器智能会议 *(ICMI)*，2025年，第 1–6 页。[10] T. Liang等，“用于深度神经网络加速的剪枝与量化：综述”，*Neurocomputing*，第 461 卷，第 370–403 页，2021年。[11] F. Asim、S. Aung 和 J. Lee，“FlowQ：用于高效、准确脉冲神经网络推理的定点低精度训练后量化框架”，载于*ASP-DAC*，2026年，第 119–125 页。[12] M. Li、Y. Kan、M. Wu、R. Zhang 和 Y. Nakashima，“Fpspike：一种利用结构化稀疏性加速脉冲神经网络的全并行可重构架构”，*IEEE*电路与系统汇刊*I*辑：正规论文，2026年。[13] D. Sekertzis 和 G. Dimitrakopoulos，“VQ4SNN：用于存储器高效现场可编程门阵列脉冲神经网络的向量量化”，载于*IEEE*国际电子电路与系统会议 *(ICECS)*，2026年。[14] M. Nazemi、G. Pasandi 和 M. Pedram，“通过布尔逻辑最小化实现能效高、低延迟的神经网络”，载于亚洲及南太平洋设计自动化会议论文集 *(ASPDAC)*，2019年，第 274–279 页。[15] M. Andronic 和 G. A. Constantinides，“Neuralut：将神经网络密度隐藏于可综合布尔函数中”，载于国际现场可编程逻辑与应用会议 *(FPL)*，2024年，第 140–148 页。[16] M. Dampfhoffer、T. Mesquida、A. Valentian 和 L. Anghel，“脉冲神经网络真的比人工神经网络更节能吗？一项深入的硬件感知研究”，*IEEE* 新兴计算智能主题汇刊，第 7 卷，第 3 期，第 731–741 页，2022 年。[17] A. Carpegna、A. Savino 和 S. Di Carlo，“Spiker+：用于边缘端推理的高效脉冲神经网络 FPGA 加速器生成框架”，*IEEE* 新兴计算主题汇刊，第 13 卷，第 3 期，第 784–798 页，2024年。[18] C. D. Schuman、T. E. Potok、R. M. Patton、J. D. Birdwell、M. E. Dean、G. S. Rose 和 J. S. Plank，“神经形态计算与硬件中的神经网络综述”，*arXiv* 预印本 *arXiv:1705.06963*，2017 年。[19] E. Z. Farsa、A. Ahmadi 和 O. Keszocze，“面向神经形态计算的可重构数字 FPGA 实现：最新进展与未来方向综述”，*IEEE* 新兴计算智能主题汇刊，第 9 卷，第 5 期，第 3210–3232 页，2025 年。[20] C. Paar，“里德–所罗门编码器的优化算术”，载于*IEEE* 信息论国际研讨会论文集，1997 年，第 250 页。[21] Y. LeCun 等，“将基于梯度的学习应用于文档识别”，*IEEE* 论文集，第 86 卷，第 11 期，第 2278–2324 页，1998 年。[22] S. Becker 等，“AudioMNIST：在简单基准测试中探索用于音频分析的可解释人工智能”，富兰克林研究所杂志，第 361 卷，第 1 期，第 418–428 页，2024年。[23] B. Cramer 等，“用于系统评估脉冲神经网络的海德堡脉冲数据集”，*IEEE* 神经网络与学习系统汇刊，第 33 卷，第 7 期，第 2744–2757 页，2020 年。[24] J. K. Eshraghian 等，“借鉴深度学习经验训练脉冲神经网络”，*IEEE* 论文集，第 111 卷，2023 年，第 1054 页，第 9 期，第 1016 页。[25] IC-Lab-DUTH，“采用混合 BRAM-LUT 权重存储的 FPGA 脉冲神经网络”，www.github.com/ic-lab-duth/hybrid-weight-storage-snn.git，2026 年。[26] H. Wang、Z. He、T. Wang、J. He、X. Zhou、Y. Wang、L. Liu、N. Wu、M. Tian 和 C. Shi，“TripleBrain：一种紧凑型神经形态硬件核心，具备快速片上自组织和强化脉冲时序依赖可塑性”，*IEEE* 生物医学电路与系统汇刊，第 16 卷，第 4 期，第 636–650 页，2022 年。[27] Z. Zhong、J. Yang、H. Wang、Y. Jiang、Y. Lin、P. Gan、L. Liu、L. Liang 和 C. Shi，“MorphBungee-Tiny：一种超低成本 FPGA 神经形态架构实现，支持高精度片上 E-prop 学习”，*IEEE* 电路与系统 *II*：简报，第 72 卷，第 12 期，第 2012–2016 页，2025 年。[28] G. Chen、M. Chen、T. Wang、H. Wang、X. Fu、Y. Lin、L. Liu 和 C. Shi，“一种支持高精度片上聚合标签学习的边缘神经形态处理器”，*IEEE* 电路与系统 *II*：简报，第 72 卷，第 3 期，第 509–513 页，2025 年。