面向数字信号处理的神经形态 FPGA 设计

Justin London电气工程与计算机科学系，北达科他大学大福克斯，北达科他州
justin.london@und.edu

arXiv:2601.07069v1 [cs.NE] 11 Jan 2026

摘要—**本文分析并讨论了神经形态计算、脉冲神经网络（SNN）和忆阻器的基础。随后，将神经形态计算应用于数字信号处理（DSP）的 FPGA 设计。使用 Verilog 硬件描述语言在 Vivado 中实现了采用和不采用神经形态计算的有限脉冲响应（FIR）滤波器和无限脉冲响应（IIR）滤波器。结果表明，神经形态计算能够提供低延迟和突触可塑性，从而实现持续的片上学习。由于其并行和事件驱动的特性，神经形态计算可以通过消除冯·诺依曼瓶颈来降低功耗并提高效率，但代价是数值精度降低。**

I. 引言

使用忆阻器进行神经形态计算，无需大量重新训练即可持续进行片上学习，并适应新信息。忆阻器或电阻开关是具有可变非易失性电阻状态的无源非线性电路元件，与生物突触相似。它们与 CMOS 技术兼容。基于忆阻器的脉冲神经网络（SNN）采用时间脉冲编码，可实现超低能耗计算，因此非常适合电池供电的智能设备 [1, 21]。脉冲神经网络是神经形态计算的构建模块。在脉冲神经网络电路实现中，忆阻器使计算与存储一体化成为可能，从而缓解存储瓶颈问题；这一问题是数据密集型人工神经网络（ANN）工作负载面临的主要障碍。它们可定义为底层神经形态物理系统执行的计算。这些系统通过在物理时间内运行的硬件实现，进行稳健而高效的神经计算。通常，它们由事件或数据驱动，并采用低功耗、大规模并行的混合模拟、数字或混合型超大规模集成电路；其计算所依据的物理原理与神经系统所采用的相似[12]。

具体而言，神经形态计算着重于模拟生物脑结构和功能的硬件系统，通常涉及神经网络和脉冲神经元。神经形态 FPGA 使用 CMOS 器件与电路来模拟神经元和突触，并将数据处理为脉冲，而不是连续输入 [11]。忆阻器和脉冲神经网络已被用于深度学习神经形态 FPGA 设计 [8]。忆阻型脉冲神经网络的主要组件包括时间脉冲编码器、双开关、忆阻交叉阵列（MC）、漏电积分发放（LIF）神经元，以及各种已提出的控制模块，例如侧向抑制电路（LIC）、突触控制电路（SCC）、双开关控制电路（DCC）和更新控制电路（UCC） [20]

在空间上分布的脉冲可以并行处理，而在不同时间触发的脉冲则可以串行化。在没有脉冲事件的情况下，所需的内存访问较少，但处理基于脉冲的工作负载仍能受益于并行处理。

基于脉冲的激活函数会导致张量中的大型子块在图形处理器中保持非激活状态，而图形处理器既未针对稀疏性进行有效优化，也无法重新编程以实现这种优化 [10]。中央处理器和图形处理器处理中普遍采用的冯·诺依曼架构，使处理单元与存储单元之间的数据传输可能导致较高的功耗。现场可编程门阵列能够提供高度并行的处理能力和高度灵活性，并且可以重新编程；因此，它们适用于用于神经形态计算的脉冲神经网络，而后者可应用于数字信号处理和滤波。

有限脉冲响应（FIR）滤波器是非递归的（仅前馈），具有线性相位特性，这是许多数字信号处理应用中的一大优势。无限冲激响应（IIR）滤波器是递归的（采用反馈路径），与同等的有限冲激响应滤波器相比，能够以更少的资源实现更陡峭的频率响应 [25]。现场可编程门阵列因其固有的并行性和专用数字信号处理切片（例如，Xilinx DSP48E 元件）而非常适用于数字信号处理应用；这些切片能够高效执行乘累加（MAC）运算。Verilog 可用于描述这些滤波器的数字逻辑。忆阻器能够在同一物理位置同时执行数据存储和计算（存内计算），从而克服限制传统计算机架构速度和能效的“冯·诺依曼瓶颈”，因此特别适用于加速许多数字信号处理和人工智能任务中的核心运算——乘累加（MAC）运算。

本研究将使用 Verilog 硬件描述语言进行 FPGA 设计与综合，并使用 LTSpice 进行电路仿真，以实现数字信号处理有限脉冲响应和无限脉冲响应低通滤波器。设计中将采用基于多路复用器的乘法器、加法器或分布式算术技术，以优化面积、功耗和速度。为提高效率和速度，设计中还可采用基于多路复用器的截断乘法器和近似加法器。 [3] 随后将重新设计有限脉冲响应和无限脉冲响应滤波器，使其具备神经形态特性。这可能包括将输出级替换为脉冲神经网络，例如漏积分发放神经元模型。滤波器的输出可作为该神经元的突触输入电流。滤波器可以不再采用直接数字卷积，而是实现为由数字脉冲神经元构成的网络，其中数据由脉冲时间或脉冲频率表示。输入和输出数据随后将分别表示脉冲事件或编码信号。神经元只有在接收到脉冲时才会处理信息。与传统 DSP 滤波器相比，这可以大幅降低功耗；传统滤波器会持续使用时序逻辑进行计算，并通过移位寄存器存储所需的先前输入和输出。

标准有限脉冲响应和无限冲激响应滤波器可作为神经形态组件集成到 FPGA 设计中。我们首先尝试使用 Verilog 为 FPGA 设计标准有限脉冲响应和无限冲激响应滤波器。随后将实现一种受神经形态启发的数字信号处理滤波器架构，该架构采用脉冲神经网络，模仿神经网络的结构来执行滤波任务，其处理效率可能高于传统有限脉冲响应/无限冲激响应设计。本工作的目标是展示神经形态处理如何提升 FPGA 中数字信号处理滤波器的效率和处理能力。传统数字信号处理采用标准乘法器。基于神经形态忆阻器的设计可能采用高度可扩展的并行基于脉冲的数字乘法器或其他新颖的算术单元，以降低功耗和芯片面积。

与标准DSP滤波器不同，神经形态滤波器可以使用自适应抽头系数，并且通常集成片上学习，这意味着 COEFF值会通过最小均方算法（LMS）等算法动态变化，因此需要将系数存储在可重写的随机存取存储器中，而不是只读存储器或参数中。脉冲神经网络能够实现较高的能效并降低存储需求，因为它们使用稀疏的二进制（0-1）信号，而非实值激活和特征图。由于某些基于脉冲神经网络的滤波器可以采用自适应时间窗口设计，因此原则上，与传统滤波器相比，它们能够更稳健地处理噪声。

与需要持续进行时钟周期运转和数据移动的传统处理器不同，神经形态芯片仅在接收到信号时才进行计算，并在更靠近数据存储位置的地方处理数据（存内计算

），并利用模拟计算更高效地执行数学运算。因此，神经形态计算凭借其事件驱动、大规模并行且以内存为中心的架构，能够提升数字信号处理的性能，并在传感器处理和推理等特定任务中降低功耗、提高效率。

有限脉冲响应和无限脉冲响应算法使用有符号整数来表示包含整数位和小数位的Q格式定点数。默认参数（$DATA\ W = 24$，FRAC ≈ 16）提供了一个较好的折中值。必须为各个信号（系数、数据、阈值）一致地选择 FRAC。忆阻器可用于数字信号处理和神经计算中的非易失性存储器，因为它们无需供电即可保持其电阻状态（存储器），从而实现高能效的存内计算，并模拟生物突触可塑性。

本文的组织结构如下。在第 II 节中，详细介绍所采用的方法。该方法涵盖对泄漏积分发放模型、有限脉冲响应滤波器和无限脉冲响应二阶滤波器的背景讨论。在 III 节中，讨论作为神经形态计算基础要素的忆阻器。此外，还介绍数字信号处理中使用的时间寄存器、运算逻辑及时域模块。在第 IV 节中，介绍用于数字信号处理的FPGA设计。在第V 节中，给出并分析仿真结果。最后，第 VI 节总结全文。

II. 方法

本研究使用 Xilinx Vivado/Verilog 硬件描述语言和 LTSpice，实现并仿真用于数字信号处理的有限冲激响应滤波器、无限冲激响应滤波器以及忆阻器/神经形态滤波器模型的波形和原理图。我们采用漏积分发放神经元（LIF）模型。漏积分发放神经元模型通过对输入电流进行积分来模拟神经元的行为；当膜电位超过阈值时，神经元会“发放”一个脉冲并重置。“漏”指的是膜电位会呈指数衰减并趋向静息电位，这与 RC电路类似。来自其他神经元的信号被建模为电流脉冲，并叠加到神经元的总膜电位上；此时，膜电位具有泄漏特性。与此同时，膜电位会自然地“泄漏”并回落至静息电位。描述神经元膜电位随时间变化的 LIF 基本方程为：

$$\tau \frac{dV}{dt} = -(V(t) - V_{rest}) + RI_{in}(t) \tag{1}$$

其中，$V(t)$为膜电位，τ 为膜时间常数，V_{rest} 为静息电位，R为膜电阻，$I_{in}(t)$为输入电流。令 $C = \frac{\tau}{R}$ 为膜的等效电容，则 LIF 神经元可表示为：

$$I_{in}(t) = \frac{V(t) - V_{rest}}{R} + C\frac{dV}{dt} \tag{2}$$

在 FPGA 中，可以使用欧拉积分将这一连续函数离散化，从而得到适用于硬件的差分方程。脉冲神经网络模型及其变体有多种，例如自适应指数积分发放（AdEx）模型、Hodgkin-Huxley模型和 Izhikevich模型 [10] 图1 对神经元模型进行了比较：

Neuron Models

Name	Formula	Bio. plausibility	Complexity
Integrate-and-Fire (IF)	$\frac{dv}{dt} = RI$	V. Low	5 FLOPs
Leaky IF (LIF)	$\frac{dv}{dt} = -(v - V_{rest}) + RI$	Low	10 FLOPs
Izhikevich (IZH)	$\begin{cases} \frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I \\ \frac{du}{dt} = a(bv - u) \end{cases}$	Low	60 FLOPs
FitzHugh-Nagumo (FHN)	$\begin{cases} \frac{dv}{dt} = v - \frac{v^3}{3} - w + I \\ \frac{dw}{dt} = 0.08(v + 0.7 - 0.8w) \end{cases}$	Medium	100 FLOPs
Adaptive IF (AdEX)	$\begin{cases} \frac{dv}{dt} = \frac{-g_L(v - E_L) + g_L\Delta_T exp(\frac{v - v_T}{\Delta_T}) - w + I}{C} \\ \frac{dw}{dt} = a(v - E_L) - w \end{cases}$	Medium	200 FLOPs
Hodgkin-Huxley (HH)	$\begin{cases} \frac{du}{dt} = \frac{-g_K n^4 (U - V_K) - g_{Na} m^3 h (U - V_{Na}) - g_{leak}(U - u_{leak}) + I}{C_m} \\ \frac{dn}{dt} = \alpha_n(U)(1 - n) - \beta_n(U)n \\ \frac{dm}{dt} = \alpha_m(U)(1 - m) - \beta_m(U)m \\ \frac{dh}{dt} = \alpha_h(U)(1 - h) - \beta_h(U)h \end{cases}$	High	1200 FLOPs

图1：神经元模型比较。 [22]

图2（左）展示了 LIF脉冲神经元的行为。该脉冲神经元接收来自多个输入端的脉冲，对其进行处理，并从输出节点生成输出脉冲。右侧展示了接收输入脉冲时神经元状态随时间的变化。当达到阈值时，它会生成一个输出脉冲。

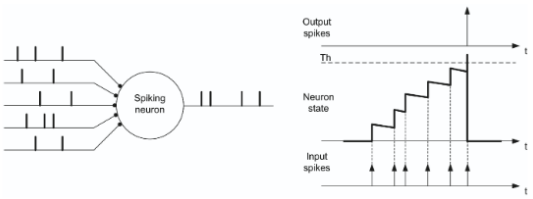

图2：（左）漏积分发放脉冲神经元；（右）神经元接收到输入脉冲时其状态随时间的变化。 [4]

图3 展示了 RC 漏积分神经元的硬件实现。

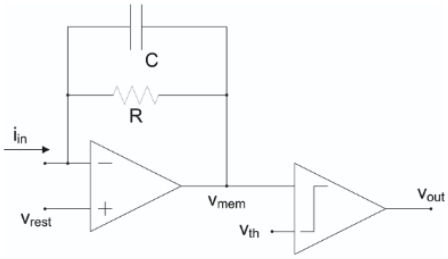

图3：RC LIF 神经元实现。 [4]

神经元的衰减由膜时间常数（τ）决定，该常数决定了电位下降的速度。如果积分后的电位达到特定的阈值电压，神经元就会“触发”一个动作电位（脉冲）。重置：触发后，神经元的膜电位会立即重置为较低的数值（通常为静息电位）。脉冲之后会紧接着一个短暂的绝对不应期，在此期间神经元无法再次触发，不过通常只需通过重置机制即可实现。较大的输入电流会使膜电位更快达到阈值，从而提高发放率。

脉冲时序依赖可塑性（STDP）是一种常见的片上学习规则。突触权重的更新取决于突触前脉冲和突触后脉冲的相对时序。其数学表达式通常根据时间差对突触权重进行加减，且通常采用简单的掩码与累加操作，而非复杂的乘法累加（乘累加，MAC），如传统人工智能中所用。硬件算术实现采用定点运算、可重构乘法器和脉冲编码。

大多数基于 FPGA 的神经形态系统采用定点数表示（例如 16 位或 18 位）来表示权重和膜电位，因为与浮点表示相比，这种方式所需的硬件资源（例如查找表（LUT）和触发器（FF））更少，功耗也更低。FPGA 内置硬件乘法器（数字信号处理器），但其使用经过优化，以满足神经形态工作负载对精度的多样化要求；有时还会采用近似计算技术，以平衡准确性和资源占用。信息主要通过二进制脉冲（事件驱动）传递，而非连续值，因此在脉冲到达时，运算可以简化为“掩码并累加”或加法操作。

A. 有限冲激响应滤波器

长度为 M 的有限脉冲响应滤波器，输入为 x（n），输出为 y（n）：

$$y(n) = \sum_{k=0}^{M-1} b_k x(n-k) \tag{3}$$

或者，输出序列也可以表示为系统的单位样本响应 h（n）与输入信号的卷积，得到

$$y(n) = \sum_{k=0}^{M-1} h(k) x(n-k) \tag{4}$$

其中 $b_k = h$（k）。滤波器可以用其系统函数来表征

$$H(z) = \sum_{k=0}^{M-1} h(k) z^{-k} \tag{5}$$

其中 z^{-k} 是 k 阶的延迟

频率响应为 $H(e^{j\omega})$ $\sum_{k=0}^{N-1} h[k]e^{-j\omega k}$ $h[k]$ = 其中，ω 是滤波器系数。 π 是弧度每采样的频率（范围从 0 到 π弧度，其中

是奈奎斯特频率。）要计算有限脉冲响应滤波器的增益，我们使用频率响应公式 $H(\omega) = \sum_{k=0}^{N-1} h[k]e^{-j\omega k}$，其中 $h[k]$ 是滤波器系数，N 是滤波器长度，ω 是频率。这个复数值的模 $|H(\omega)|$，就是该特定频率下的线性增益。对于直流增益（0 Hz）的特定情况，增益就是所有系数的总和：$G_{DC} = \sum_{k=0}^{N-1} h[k]$。要以分贝表示增益，我们使用以下公式：$Gain_{dB} = 20\log_{10}(|H(\omega)|)$。

FIR数字信号处理器（数字信号处理器）主要使用专用的乘累加（MAC）单元，以及加法器、减法器、寄存器（用于流水线处理和延迟线）和多路复用器。这些组件通常会集成到FPGA中的专用高效“DSP模块”中，或作为独立DSP芯片中的硬连线组件。乘法器通常采用48位或64位寄存器，可执行一系列乘加运算（乘累加，即MAC运算），这是有限脉冲响应滤波器运行的核心。流水线寄存器/移位寄存器对于有限脉冲响应滤波器中的抽头延迟线至关重要。输入数据会经过一系列寄存器，以产生各抽头所需的延迟，从而通过流水线处理实现高性能。

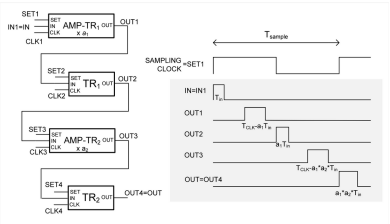

图4：（a）时间模式 z^{-1} 乘法器电路，（b）脉冲时序图。 [18]

在数字信号处理中，时域 z^{-1} 运算符乘法器，其中 $T_{in}(k)z^{-1} = T_{out}(k) = T_{in}(k-1)$，需要生成一个输出脉冲，其脉冲 T_{out}宽度等于 $a \cdot T_{in}$，其中 T_{in} 是输入脉冲，a 是乘法系数，且输出脉冲将与采样信号同步，并延迟一个时钟周期。图4展示了时域 z^{-1} 乘法器电路及其时序脉冲。

如图 z^{-1} 所示，时域乘法器由两个 AMP-TR 和两个 TR 电路串联组合实现5。

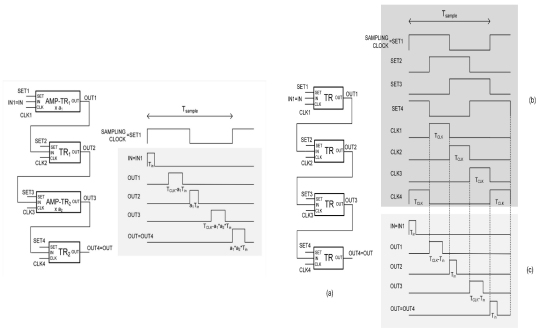

图5：（左）（a）时间模式 z^{-1} 乘法器电路；（b）脉冲时序图；（右）（a）时域z^{-1} 电路，（b）时钟时序图，以及（c）脉冲时序图 [18]

.

*B. IIR*双二阶节

无限脉冲响应（IIR）滤波器的一般公式为以下差分方程：

$$y[n] = \sum_{k=0}^{M} b_k x[n-ki] - \sum_{k=1}^{N} a_k y[n-k] \qquad (6)$$

表示当前输出 y[n] 是当前/过去输入与过去输出的加权和，其中使用前馈（b）和反馈（a）系数；其传递函数为：

$$H(z) = \frac{B(x)}{A(z)} = \frac{\sum_{k=0}^{M} b_k z^k}{1 + \sum_{k=1}^{N} a_k z^{-k}} \qquad (7)$$

“双二阶节”（biquad）是 bi-quadratic 的简称，也是双极点双零点数字滤波器的常见名称。双二阶节的传递函数可定义为：

$$H(z) = g\frac{1 + \beta_1 z^{-1} + \beta_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \qquad (8)$$

C. 巴特沃斯滤波器

考虑图中所示的二阶巴特沃斯（萨伦-基）低通滤波器6。

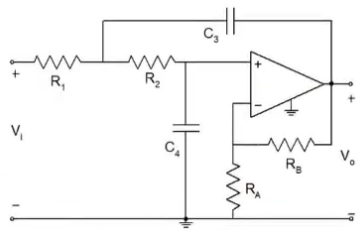

图6：二阶滤波器

假设 $\alpha = 7$ 和 $\beta = 6$。直流增益为 $K = \frac{4+\alpha}{2} = \frac{4+7}{2} = 5.5$ $= f_c =$ 。截止 $\frac{4+\beta}{2} = \frac{4+6}{2} = 5KHz$ $f_c = \frac{1}{2\pi\sqrt{R_1 R_2 C_1 C_2}}$ 频率为 。令 为截止频率，并假设 $R = R_1 = R_2$和 $C = C_1 = C_2 = 15nF$，从而得到 $f_c = \frac{1}{2\pi RC}$，使得 $R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi \cdot 5kHz}()(15nF) = 2.2k\Omega$。传递函数为

$$\frac{V_o(s)}{V_i(s)} = \frac{K\frac{1}{R_1R_2C_3C_4}}{s^2 + s\left(\frac{1}{R_1C_3} + \frac{1}{R_2C_3} + \frac{1}{R_2C_4} - \frac{K}{R_2C_4}\right) + \frac{1}{R_1R_2C_3C_4}}$$

or

$$\frac{K\frac{1}{R^2C^2}}{s^2 + s\left(\frac{1}{RC + \frac{1}{RC} + \frac{1}{RC} - \frac{K}{RC}}\right) + \frac{1}{R^2C^2}}$$

代入数值后得到

$$\frac{V_o(s)}{V_i(s)} = \frac{5.05 \times 10^9}{s^2 - 75757.7s + 9.2 \times 10^8} \tag{9}$$

其中 $K = 1 + \frac{R_A}{R_B}$ 如果 $R_A = 1\ k\Omega$，那么 $R_B = 4.5k\Omega$。图中展示了带宽低通滤波器电路及其在 LTSpice 中的仿真结果，分别如图 7 和 8所示。

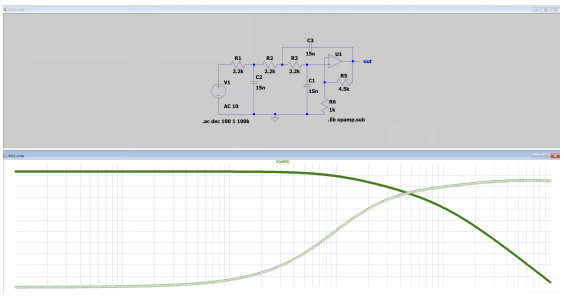

图7：LTSpice 中的带宽电路

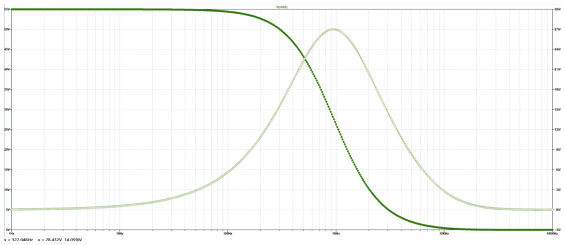

图8：LTSpice 带宽滤波器仿真

D. 高通滤波器

图 9展示了使用两个差分运算放大器的三阶有源高通滤波器的 LTSpice 仿真结果。

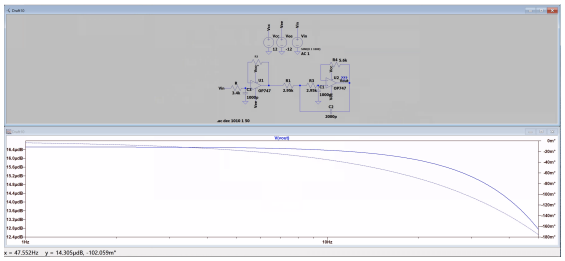

图9：高通滤波器

图 10 展示了无源带通滤波器的 LTSpice 仿真：

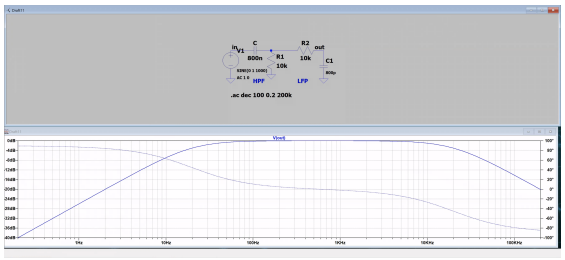

图 10：带通滤波器

III. 忆阻器

基于忆阻器的神经元由具有可变电导的双端／电极开关构成，其输入信号 x_1、……、 x_n 会被传送至基于忆阻器的突触，

并具有相应的权重 w_1、$\cdots$、w_N [16]。通过外加电压或电流脉冲改变相应的忆阻值，即可实现相应突触权重的变化。它们功耗低、开关速度快，并且与互补金属氧化物半导体（CMOS）技术具有良好的兼容性 [15]。图 11展示了忆阻器的示意图。顶部电极标记为 `te`，底部电极标记为 `be`。正弦电压源为忆阻器等效电路供电。忆阻器电流由电压控制电流源 G_1建模。该受控源的电流即为忆阻器电流。用于测量忆阻器状态变量的附加电极 x 标记为 Y。状态变量的时间导数与忆阻器电流成正比，并由受控电压控制电流源$G_2 \equiv G_Y$表示。电容负载用于对与忆阻器状态变量时间导数成正比的电流进行积分 x。相应的

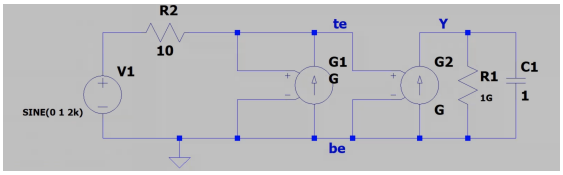

图11：忆阻器电路

忆阻器模型的 LTSpice 网表代码表示为$K1$：

```
1 .subckt K1 te be Y2          = 100      = 16 3   = 10 3   1 = 1 .params ron
```

大多数忆阻器基于过渡金属氧化物，例如二氧化钛、二氧化铪和氧化钽。已有许多二氧化钛忆阻器纳米结构模型，包括 Strukov-Williams、Joglekar-Biolek、Ascoli-Corinto 和 Lehtonen-Laiho。 [15] 忆阻元件呈现电压与电流之间的滞回关系。图12（a）展示了忆阻器纳米结构，其中展示了金属接触（顶部电极和底部电极）以及忆阻器的掺杂层和未掺杂层。图 12（b）展示了一个等效电路，其中掺杂区和未掺杂区的电阻串联连接。

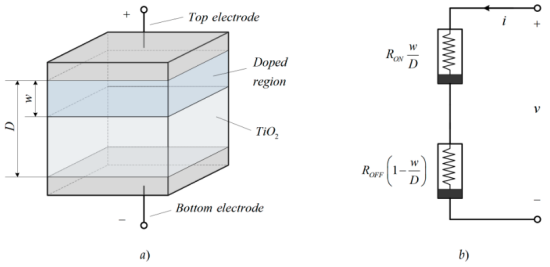

图12：（a）忆阻器纳米结构和（b）等效电路。 [15]

具有扩散动力学的忆阻器非常适合模拟突触的行为，因为它们可以缩小到低于 10 nm 的特征尺寸，可以将记忆状态保留数年，以纳秒时间尺度进行切换，并可根据脉冲时间依赖可塑性（STDP）等受生物启发的学习规则，实时进行脉冲学习 [21]。忆阻元件的特征 I-V 方程可近似表示为：

$$i_{MR} = G(w, v_{MR}) v_{MR} \tag{10}$$

$$dw/dt = f_{MR}(w, v_{MR}) \tag{11}$$

其中，i_{MR} 和 v_{MR} 分别是端子器件处的电流电压降，其中

$$f_{MR} = \begin{cases} I_0 \text{sign}(v_{MR})(e^{|v_{MR}|/v_0} - e^{v_{TH}/v_0}) \\ \qquad\qquad \text{if } |v_{MR}| > v_{TH} \\ 0 \text{ otherwise} \end{cases} \tag{12}$$

$G(w,\ v_{MR})$ 是器件的电导，其会随施加电压而变化（假设采用电压或磁通控制的器件模型）；而 w 是某种物理参数特性，其变化通常由施加电压的非线性函数 f_{MR} 控制，其中包含阈值势垒 [4]。它们通常是二元的

器件在高电阻态和低电阻态之间切换。

在硬件中，经过训练的神经逼近器可以用更少的专用固定滤波器抽头（更少的乘法器）来逼近滤波器响应；在忆阻器交叉阵列实现中，乘累加运算内在于欧姆定律（向量与电导相乘，并在模拟域求和）——与数字乘累加阵列相比，面积和功耗可能大幅降低。其权衡在于：逼近误差与资源/功耗节省之间的取舍。图13展示了忆阻电阻器神经网络的结构，而13（b）展示了基于忆阻电阻器的突触示意图。无源

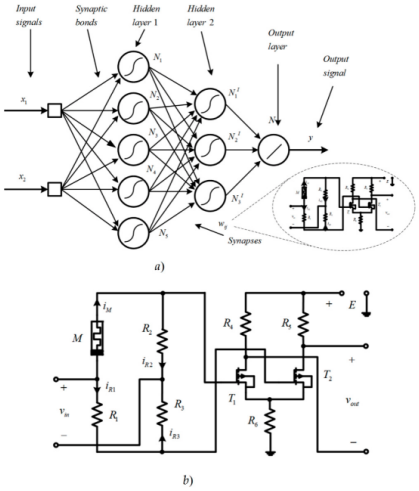

图13：（a）忆阻器神经网络；（b）基于忆阻器的突触。 [15]

忆阻器存储器交叉阵列如图14（a）所示，其等效电路如图14（b）所示。LTSpice 中的忆阻器存储器交叉阵列示意图如图14（c）所示。

A. 时间寄存器和运算逻辑

时间寄存器（TR）或移位寄存器是一串触发器，用于存储位，并在每个时钟脉冲到来时将位依次移动一个位置，从而产生时间延迟或执行串行转并行/并行转串行转换，充当由时钟同步的临时数据存储器。图 15显示了时间

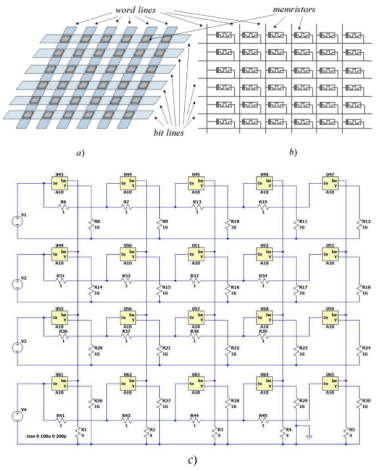

图14：（a）忆阻器存储器交叉阵列；（b）等效电路；（c）LTSpice 示意图。 [15]

寄存器及其符号。如果 SET = 0，晶体管 M_1（即 PMOS ）导通，电容器电压便会被设为电源电压 V_D。当晶体管 M_2（即 NMOS）导通时，电容器会放电，而晶体管的导通由或门 [18] 控制。

图 15 展示了一个时间寄存器的时序图，该寄存器用于跟踪同步过程。SET 信号的时间间隔 T_{CLK} 是一个脉冲宽度固定且 占空比为 25% 的脉冲。当 IN 和 CLK 均为 0 时，电容器电压保持不变。IN 信号的脉冲宽度越大 T_{in} ，由 T_{in}导致的放电时间就越长（CLK 导致的放电时间保持不变）。输出是一个宽度等于 $T_{CLK}-T_{in}$的脉冲，因此可以存储 T_{in}的值，同时输出脉冲与 CLK 信号同步。

该电路可以存储输入脉冲的时间间隔，并按增益因子放大脉冲宽度。这个新电路是一个时间放大器，如图 16左侧所示。按照 Felouris 等人 [18]，在此配置中，或门的运作由两个晶体管支路 M_{a1}–M_{b1}和 M_{a2}–M_{b2}实现。晶体管 M_{a1} 和 M_{a2} 具有相同的宽长比，充当开关。 M_{b1}和 M_{b2} 的宽长比不同，因此具有不同的

放电斜率。假设 M_{b2} 和 M_{b1} 的沟道宽度分别为 $W_{b,2}$ 和 $W_{b,1}$，且两个晶体管的沟道长度相同。那么直观地看，根据图 5(c)， T_{in} 和 T_{CLK} 之间的放电斜率会有所不同。对应于 T_{in} 的放电斜率可表示为

$$slope_{in} = a \cdot slope_{clk} \tag{13}$$

其中 a 是时间增益，其值为

$$a = \frac{W_{b,2}}{W_{b,1}} \tag{14}$$

并且 $slope_{clk}$ 是由 T_{CLK}引起的放电参考斜率。因此，输出脉冲宽度将等于

$$T_{OUT} = T_{CLK} - \frac{W_{b,2}}{W_{b,1}} T_{in} \tag{15}$$

图16右侧展示了一个基于时间寄存器的时间加法器电路。时间加法器只需将 n 个输入脉冲的脉冲宽度$T_{in,}1, T_{in,}2, \cdots, T_{in,}n$ 相加。晶体管$M_{b,}1, M_{b,}2, \cdots, M_{b,}n, M_{b,}n+1$ 具有相同的宽长比 [18]。因此，由$T_{in,}1, T_{in,}2, \cdots, T_{in,}n$ 引起的放电斜率为

$$slope_{in,1,in,2,\cdots,in,n} = slope_{in,1} + slope_{in,2} + \cdots + slope_{in,n} \tag{16}$$

而输出脉冲宽度将为

$$T_{OUT} = T_{CLK} - T_{in,1} - T_{in,2} - \cdots T_{in,n} \tag{17}$$

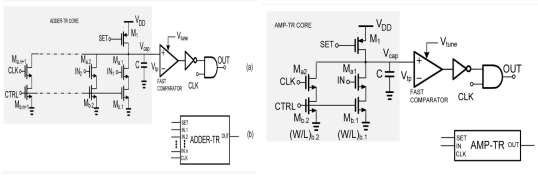

图16：（左）时间寄存器电路上的时间放大器及其符号（ADDER TR）；（右）基于时间寄存器的时间加法器及其符号（AMP TR） [18]

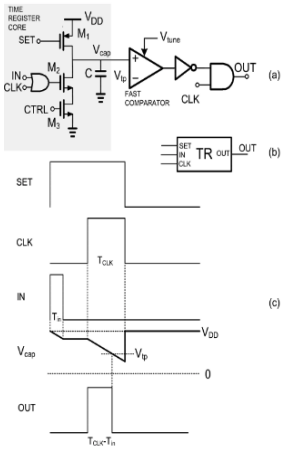

图15：（a）时间寄存器电路；（b）符号；（c）时序图 [18]

B. 时域模块

数字信号处理使用时间滞后/延迟操作的时间模式 z^{-1} 乘法器用于产生一个脉冲宽度为 T_{out} 的输出脉冲 aT_{in} 其中 T_{in} 是输入脉冲，而 a 是乘法系数，输出脉冲将与采样信号同步，并延迟一个时钟周期。所提出的 z^{-1} 电路如图 5左侧所示，右侧显示了波形。四个时间寄存器电路串联组合，实现了一个 z^{-1} 电路。假设采样时钟信号为TR1电路的SET1信号，输入信号IN等于TR1的输入IN1，而最终输出信号OUT是TR4 [18]的OUT4输出。

为与SET2脉冲同步，OUT1脉冲在CLK1的上升沿生成。随后，OUT1被用作时间寄存器输入端。OUT2与CLK2同步，并且 $T_{OUT2} = T_{CLK} - T_{OUT1} = T_{CLK} - (T_{CLK} + T_{in}) = T_{in}$

IV. 用于数字信号处理的FPGA设计

IBM 的 TrueNorth芯片 [2]和英特尔的 Loihi [7]是采用神经形态架构的专用集成电路芯片。TrueNorth核心由五个组件组成：

神经元模块、核心 SRAM、路由器、调度器和令牌控制器 [23]。单个 TrueNorth 芯片包含 4,096 个神经突触核心。Valancius 等人 [23]以 TrueNorth 架构为参考，为赛灵思 Zynq UltraScale+ MPSoC 上的现场可编程门阵列神经形态架构设计了一种参数化、可配置的仿真平台原型。他们使用 MNIST 数据集和向量矩阵乘法案例研究，验证了仿真环境的功能，并展示了在保持高准确率的同时提高计算效率、缩短计算时间和减少资源利用率的效果。文献 [22]对 FPGA 上实现的数字神经形态架构（NMA）进行了系统综述。

现场可编程门阵列提供了大量乘累加（MAC）功能，这是数字信号处理器的基本构建模块。相比之下，传统数字信号处理器仅提供一至四个此类乘累加功能。赛灵思基于 ASMBL 架构构建的 Virtex-4 系列包含一种名为 DSP48 切片的新元件，该元件将高性能算术与累加单元以及乘法器集成在一起。DSP48 切片可以级联成称为 XtremeSDP 瓦片的单元，每个瓦片包含两组。DSP48 切片由四个主要部分组成：输入/输出寄存器、 18×18 有符号乘法器、三个输入加减器模块，以及 OPMODE 多路复用器，支持在 500 MHz 下执行 40 多种独特运算。对于数字信号处理应用，可在现场可编程门阵列硬件上使用 DSP48 模块。图 17 展示了一个由两个 DSP48 切片组成的 DSP48 瓦片。

18 位 A 总线与 A bbus 拼接，其中 A 总线为最高有效位。X、Y 和 Z 多路复用器均为 48 位设计。选择任意 36 位输入时，都会得到符号扩展至 48 位的输出。乘法器输出两个 36 位部分积，并将其符号扩展至 48 位。部分积送入 X 和 Y 乘法器。OPMODE 选择乘法器时，会使用 X 和 Y 多路复用器，且加法器/减法器将部分积组合成有效的乘法器结果。P 的乘累加路径经过 Z 多路复用器。通过 X 多路复用器进行 P 反馈，可在未使用乘法器时实现 P 级联累加

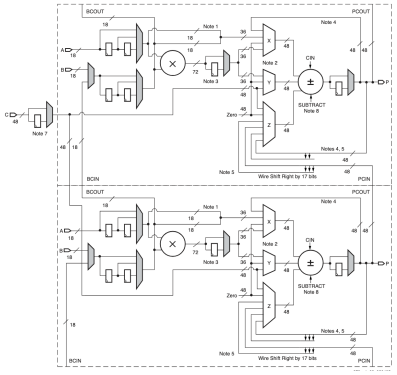

图17：包含两个 DSP48 切片的 DSP48 瓦片。 [26]

未使用乘法器时。灰色多路复用器位于 17 中，并在配置时进行编程。共享的 C 寄存器支持乘加、宽位加法或舍入。启用 SUBTRACT 后，加法器/减法器的输出为 Z - (X+Y+CIN) [26]。

这种数字信号处理架构可以通过基于 FPGA 的神经元阵列设计进行修改，例如 Cassidy 等人提出的设计。 [5] （如图 18所示），该设计集成了使用突触、延迟机制、累加器、阈值和脉冲时间依赖可塑性的脉冲神经元网络 [13]。

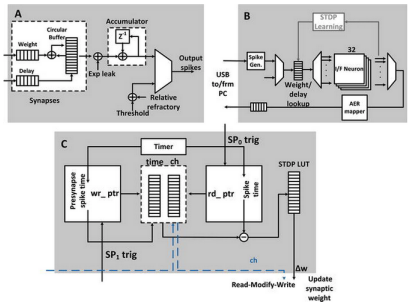

图18：基于 FPGA 的神经元。A：展示单个神经元架构的细节；B：展示整体网络架构，包括脉冲生成接口；C：脉冲时序依赖可塑性的实现 [5]

Pearson 等人 [19]提出了一种基于现场可编程门阵列的阵列处理器架构，用于模拟大型网络

支持超过一千个神经元进行实时信号处理。然而，其可扩展性受限于基于总线的通信协议。针对神经形态现场可编程门阵列，人们提出了更先进的脉冲神经网络架构，包括优化的深度事件驱动脉冲神经网络（ODESA） [14],多时序脉冲神经网络（PSNN） [9,24],面向可重构架构计算仿真的嵌入式多核构建模块（EMBRACE-FPGA） [17, 6, 12]

ODESA 是一种多层架构，采用无需梯度的阈值学习算法，根据局部突触行为和赢家通吃（WTA）机制动态调整突触权重和神经元阈值。每个神经元都以事件驱动方式运行，将输入作为离散的二进制脉冲进行处理，从而显著降低功耗和处理开销。在 PSNN 中，不同神经元发出的脉冲沿轴突传递，并经过精确延迟后同时到达目标神经元，使其触发，即使源神经元的触发时间并不同步。EMBRACE 不采用传统的模拟神经元，而是在现场可编程门阵列上使用软处理器对这些神经元进行建模，从而开创了一种利用数字系统可重构性进行神经计算的新方法。

V. 结果

我们使用信号幅度 $A = 0.6$、频率$f = 50$ Hz 和噪声幅度 $A_{noise} = 0.05$，对叠加高斯噪声的正弦信号进行仿真，然后将其量化为 Q15 定点数。

$x[n] = A$正弦$(2\pi fn)+2A_{noise}$(rnd%1000)$/1000-1)$

其中，rnd%1000 是对高斯随机数生成器的输出取模 1000， n 是来自 $[0, 2000$ 的时间步。信号先在训练阶段进行仿真，再在测试阶段进行比较，每个阶段均包含 2000 个时间步。我们实现了经典有限脉冲响应、经典无限冲激响应、神经形态有限脉冲响应和神经形态无限冲激响应模型。神经形态有限脉冲响应模型是一个小型前馈神经网络（单个隐藏层），用于近似线性有限脉冲响应卷积。我们采用经典有限脉冲响应：直接型、参数化抽头、定点格式（WIDTH 位，FRAC 个小数位），以及一个经典无限冲激响应转置直接 II 型（数值稳定性），阶数参数化并采用定点格式。神经形态无限冲激响应采用一个小型循环神经网络（Elman 型）来近似无限冲激响应滤波行为，并通过循环向量乘法和激活函数实现（双曲正切由查找表近似）。

神经形态版本在逻辑中模拟忆阻器权重阵列（权重存储在寄存器中），并在模块内部包含最小均方学习规则（用于在线自适应）。只有在训练 =1 且测试平台提供期望信号时，才会调用该学习规则。我们的无限脉冲响应采用直接 I 型二阶节（二阶），使用定点系数；神经形态无限脉冲响应采用自适应循环权重结构，并通过最小均方更新调整反馈／前馈抽头。测试平台向所有变体提供相同的激励信号（正弦波 +噪声、脉冲、阶跃信号），确保输出比较公平。LTSpice 网表使用忆阻器和运算放大器电流求和节点实现交叉阵列求和。忆阻器行为采用一种受物理原理启发的简单行为模型（适用于性能测试）。图 19 展示了在 Vivado 中为有限脉冲响应和二阶节无限冲激响应模型生成的原理图。19。以下是

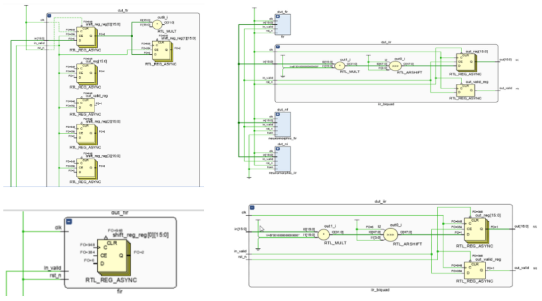

图19：原理图。（左）有限脉冲响应，（右）二阶节无限冲激响应

神经形态有限脉冲响应和神经形态双二阶无限脉冲响应如图20所示。输入为抽头采样。实际应用中，权重会离线训练并加载到硬件寄存器（或 LTSpice 中的片上忆阻器阵列）中。通过量化并重复使用权重，或利用稀疏性，可以减少乘法器的使用。下表比较了这些模型的均方误差

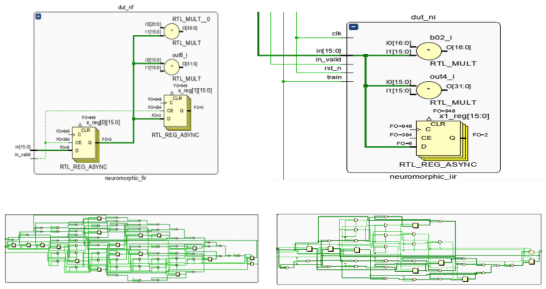

图20：原理图。（左）神经形态有限脉冲响应和（右）神经形态二阶节无限冲激响应

相对于经典有限冲激响应黄金参考而言。因此，经典有限冲激响应的均方误差为 0。图 21

表I：相对于经典有限脉冲响应黄金参考的均方误差

模型	MSE	单元	单元引脚	网表
经典有限冲激响应	0.0	173	36	921
神经形态有限脉冲响应	0.5209	335	53	1544
经典无限冲激响应	0.1374	192	36	1007
神经形态无限冲激响应	0.1533	163	53	760

展示了经典有限冲激响应（左）和神经形态有限脉冲响应（右）的仿真输出。如图所示，神经形态有限脉冲响应的变化幅度远大于经典有限冲激响应。与经典有限脉冲响应信号相比，神经形态有限脉冲响应信号的变化幅度大得多，这主要是因为底层神经形态硬件固有的噪声和元件变化。此类硬件通常采用模拟或混合信号电路来模拟生物过程。相比之下，经典数字信号处理（DSP）在高度受控且确定的环境中处理精确的数值。

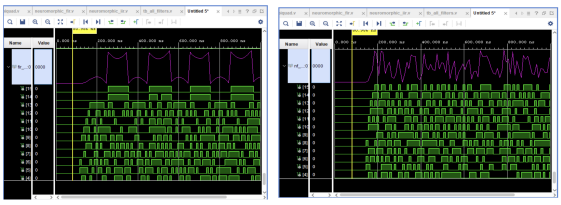

图21：（左）经典有限冲激响应和（右）神经形态有限脉冲响应

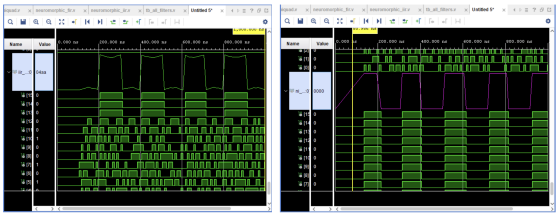

图22：（左）经典无限脉冲响应和（右）神经形态无限脉冲响应

VI. 结论

使用 Vivado 和 Verilog 硬件描述语言进行 FPGA 设计，仿真并比较了采用神经形态计算的有限脉冲响应和无限脉冲响应滤波器，以及标准的冯·诺依曼架构有限脉冲响应和无限脉冲响应滤波器。使用脉冲神经网络生成事件驱动信号。在实际硬件实现中，将使用 DSP48 模块中的忆阻器。与有限脉冲响应黄金参考相比，神经形态计算的无限脉冲响应滤波器和标准无限脉冲响应滤波器的均方误差相近。相较于标准有限脉冲响应黄金参考，神经形态有限脉冲响应模型的均方误差高得多。尽管神经形态计算可能无法像采用时序逻辑的标准数字信号处理那样提供精确的结果，但它在功耗、效率和成本方面具有显著优势。此外，神经形态网络的分布式特性使其天生更能抵御故障或错误。神经形态计算具有并行特性，并能进行自适应学习，同时将处理和存储器集成于同一位置，类似于大脑中神经元和突触的工作方式，因此消除了冯·诺依曼瓶颈。未来的工作将研究如何提高实时数字信号处理应用的精度，并使用赛灵思进行实际的 FPGA 实现。

参考文献

[1] S. Abreu 和 J. Pedersen。 神经形态编程：面向类脑硬件的新兴方向。2024 年。 网址：https://arxiv.org/html/2410.22352v1。

[2] F. Akopyan 等人。“TruthNorth：一款 65 毫瓦、拥有 100 万个神经元的可编程神经突触芯片的设计与工具流程”。载于：《IEEE 集成电路与系统计算机辅助设计汇刊》34.10（2015），第 1537–1557 页。[3] K. Bali 和 N. Kumar。“使用基于多路复用器的截断乘法器和近似加法器实现有限脉冲响应滤波器”。载于：2025年国际视觉分析与数据可视化会议（ICVADV），印度蒂鲁内尔维利（2025），第 718–723 页。[4] L. Camunas-Mesa、B. Linares-Barranco 和 T. Serrano-Gotarredona。“神经形态脉冲神经网络及其基于忆阻器-CMOS 的硬件实现”。载于：《Materials》12（2019）。[5] A. Cassidy 等人。“基于现场可编程门阵列的硅基脉冲神经阵列”。载于：IEEE 生物医学电路与系统会议。加拿大蒙特利尔，2007 年。[6] S. Cawley 等人。“脉冲神经网络硬件原型设计与应用”。载于：《遗传编程与可进化机器》12（2011），第 257–280 页。[7] M. Davies 等人。“Loihi：具有片上学习功能的神经形态众核处理器”。载于：《IEEE Micro》38.1（2018）。[8] T. Hong、Y. Kang 和 J. Chung。“InSight：面向深度神经网络的基于现场可编程门阵列的神经形态计算系统”。载于：《低功耗电子应用期刊》10.36（2020）。DOI：doi:10.3390/jlpea10040036。[9] E. Izhikevich。“多脉冲同步：基于脉冲的计算”。载于：《神经计算》18（2006），第 245–282 页。[10] M. Karamimanesh 等人。《基于现场可编程门阵列的脉冲神经网络：方法综述与最新进展》。2025。[11] Wei Lu。《基于忆阻材料与器件的神经形态计算》。2020。网址：https://www.youtube.com/watch?v=qfBlqCZdG6U&t=261s。[12] A. Mehonic 等人。《采用新兴技术实现神经形态计算路线图》。2024。网址：https://arxiv.org/abs/2407.02353v1。[13] A. Mehrabi 和 A. van Schaik。《基于现场可编程门阵列的脉冲神经网络》。载于：《神经形态计算最新进展》。IntechOpen，2024。URL：https://www.intechopen.com/chapters/1189625。[14] A. Mehrabi 等人。基于现场可编程门阵列的多层无梯度在线训练脉冲神经网络的高效实现。2023。[15] V. Mladenov。《统一且开放的 LTSpice 忆阻器模型库》。载于：《Electronics》10（2021）。[16] V. Mladenov 和 S. Kirilov。《基于采用MOS晶体管的简单对数-乙状线性传递函数的忆阻器神经网络》。载于：（2024）。[17] F. Morgan 等人。探索基于片上网络的脉冲神经网络在 FPGA 上的演进。2009。DOI：10 : 1109 / fpt . 2009 . 5377663。[18] O. Panetas-Felouris 和 S. Vlassis。《基于时域模式信号处理的三阶有限脉冲响应滤波器实现》。载于：《Electronics》11.6（2022）。DOI：https://doi.org/10.3390/electronics11060902。[19] M. Pearson 等人。《为实时信号处理和控制应用实现脉冲神经网络：一种经模型验证的现场可编程门阵列方案》。载于：《IEEE 神经网络汇刊》18.5（2007），第 1472–1478 页。[20] S. Prajapati、S. Sur-Kolay 和 S. Dutta。采用监督式原位脉冲时序依赖可塑性方法的高效忆阻型脉冲神经网络架构。2025。URL：https://arxiv.org/html/2507.20998。[21] J. Sune。用于神经形态电路和人工智能应用的忆阻器。MDPI，2020。[22] W. Szczerek 和 A. Podobas。《四分之一个世纪以来基于 FPGA 的神经形态架构——综述》。载于：arXiv:2502.20415v*3* [cs.AR] （2025）。[23] S. Valancius 等人。《基于 FPGA 的神经形态架构仿真环境》。载于：arXiv:2004.06061v*1* [cs.ET]（2020）。

[24] R. Wang 等人。《具有延迟自适应功能的多时序脉冲神经网络的 FPGA 实现》。载于：《*Frontiersin Neuroscience*》（2013），第 7–14 页。[25] R. Woods。《信号处理系统的 *FPGA* 实现》，第 *2* 版。威利，2017。[26] 赛灵思。《*DSP*：面向最佳结果的设计：使用 *Virtex-4 FPGA* 的高性能数字信号处理》。赛灵思，2005。URL：
htt
ps://www.xilinx.com/publications/archives/books/dsp.pdf。