

用于图像识别的高能效脉冲神经网络加速器设计

张剑¹, 刘佳², 万贤杰², 俞宙², 韩传余¹, 张国和¹

(1. 西安交通大学电子与信息学部, 710049, 西安;

2. 中国电子科技集团第二十四研究所, 400060, 重庆)

摘要: 针对基于通用处理器解决方案在图像识别应用中速度慢、功耗高的问题,提出了一种高能效的脉冲神经网络加速器设计方案。首先,采用神经形态学计算中的高并行设计思想,设计了多核并行结构来实现硬件加速;然后,根据脉冲数据传输稀疏性的特点,采用基于事件驱动的数据传输与处理方式,设计了一对一的核间传输机制,减小了用于通信的硬件资源并提高了数据传输效率;其次,提出了按行的数据存放方式来加快膜电压数据在存储器的存取效率;最后,设计了结合查找表与异或的电路结构,可以快速地将事件向量转变为地址事件表达(AER)格式。采用所提加速器设计方案在现场可编程逻辑门阵列(FPGA)开发板上进行优化和部署。实验结果表明:当时钟频率采用 100 MHz 时,识别单张手写数字图像所需能量为 1.04 mJ,仅为 2.2 GHz 通用中央处理器(CPU)上的串行软件程序的 1/1 453.8。该加速器设计方案适用于实时性要求高和能量受限的实际场景。

关键词: 图像识别;脉冲神经网络;加速器;并行结构

中图分类号: TM301 **文献标志码:** A

DOI: 10.7652/xjtuxb202301020 **文章编号:** 0253-987X(2023)01-0211-10

Design of High Energy Efficient Spiking Neural Network Accelerator for Image Recognition

ZHANG Jian¹, LIU Jia², WAN Xianjie², YU Zhou², HAN Chuanyu¹, ZHANG Guohe¹

(1. Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

2. No. 24 Institute, China Electronics Technology Group Corporation, Chongqing 400060, China;)

Abstract: A high energy efficiency spiking neural network (SNN) accelerator is proposed to solve the problems of low-speed and high-power consumption in image recognition application based on the general processor. Firstly, a multi-core parallel structure is designed for hardware acceleration by adopting the concept of high parallel design in neuromorphic computation. Secondly, considering the sparsity of spike data transmission, the one-to-one inter-core transmission mechanism is designed based on event-driven data transmission and processing, which reduces the hardware resources used for communication and improves the data transmission efficiency. Thirdly, a data arrangement scheme is proposed to speed up the access efficiency of membrane in memory. Finally, a circuit structure combining lookups and XOR is designed, which can quickly transform the event vectors into address-event-represent (AER) format. The proposed design is optimized and implemented on the field programmable logic gate array (FPGA) development board. The experi-

收稿日期: 2022-07-29。 作者简介: 张剑(1989—),男,博士生;张国和(通信作者),男,教授,博士生导师。 基金项目: 国家自然科学基金资助项目(62174130);国防基础加强项目(2019-JCJQ-JJ-566)。

网络出版时间: 2022-09-06

网络出版地址: <https://kns.cnki.net/kcms/detail/61.1069.T.20220905.1429.008.html>

mental results show that when the clock frequency is 100 MHz, the energy required to recognize a handwritten digital image is 1.04 mJ, which is only 1/1 453.8 of the serial software program on the 2.2 GHz universal central processing unit (CPU). The proposed accelerator design scheme is suitable for the real scenarios with high real-time requirements and limited energy.

Keywords: image recognition; spiking neural network; accelerator; parallel architecture

随着万物互联的快速发展,在近数据端移动设备上进行高能效的数据分析可以解决数据传输至云端导致的高延时、高传输带宽和低安全性等问题。最流行的深度学习具有较大的计算量和高能耗的缺点,使其不能有效部署至边缘设备上^[1]。因此,高能效的类脑智能计算逐渐受到众多学者的关注,对其的研究不仅可以解决实际问题,比如图像识别^[2]和故障检测^[3]等,还可以以另一种方式来侧面了解生物学大脑。脉冲神经网络(SNN)作为生物学启发的类脑神经网络^[4],通过模拟大脑神经元和采用生物神经元之间的信息传输方式,具有高能效和硬件友好的优点^[5]。尽管脉冲神经网络的应用比如对图像传感器采集的图像进行识别可以在基于冯诺依曼结构的软件模型中实施,但是软件仿真普遍以串行的方式进行处理,处理速度慢导致不能完成数据的实时分析,并且通用处理器具有体积大、能耗高等缺陷,限制了其在能耗和空间体积受限环境中的使用。

为了获得实时低能耗的图像识别解决方案,有必要设计专用的硬件电路来替代基于通用处理器的软件仿真。脉冲神经网络的专用硬件电路设计通常分为模拟电路和数字电路两种实现方式。悉尼大学的 Schaik 和瑞士苏黎世大学的 Mitra 分别设计了模拟电路^[6-7]来实现类生物学神经元功能,但是模拟电路具有对电压温度等环境敏感的缺点,针对不同的应用条件需要对参数进行调整。数字电路由于对工艺节点不敏感等原因,更适合大规模的设计集成。数字电路的实现平台包括高灵活性的现场可编程逻辑门阵列(FPGA)和高性能的专用集成电路(ASIC)。IBM公司的TrueNorth芯片^[8]是基于ASIC的脉冲神经网络加速器,支持上百万个神经元并行工作。浙江大学的“达尔文”芯片^[9]是一款支持可配置的神经形态学协处理器,专为资源受限的嵌入式应用所设计。曼彻斯特大学的SpiNNaker芯片^[10]利用ARM核来实现脉冲神经网络的可编程性,支持高并行和高配置性的神经元处理。清华大学团队开发研制的“天机芯”^[11]支持SNN和ANN的混合芯片结构,并在自行车上成功进行了无人驾驶的试验。斯坦福大学的设计的NeuroGrid

芯片^[12],只需要5W的能量即可模拟上百万的神经元实现对哺乳动物大脑皮层工作机制的研究。上述设计优先考虑的是设计的通用性和神经元的大规模实现,复杂的通信系统需要高硬件资源和高通信带宽。电子科技大学的周军团团队提出了一款可配置SNN处理器^[13],通过采用多对多通信的路由电路来控制脉冲数据的分发与传输,具有低能耗高速度的优势,并在FPGA上进行了验证。作为一种可编程设备,FPGA支持互补金属氧化物半导体(CMOS)数字逻辑的灵活开发与设计,具有丰富的计算资源和比ASIC更快的开发周期。另外,在FPGA上成功进行验证与实现也是完成低能耗的ASIC芯片前的重要一步。本文的设计方案结合FPGA的特点进行硬件设计与优化,实现了高性能的脉冲神经网络应用。

本文受文献[14]中采用软件仿真来完成图像识别的启发,从计算单元并行工作实现加速的角度出发,以高能效和低硬件资源为目的,采用多核并行结构来实现数据的加速运算,提出了一种新的脉冲神经网络硬件加速器设计方案。该加速器设计方案通过可综合硬件描述语言Verilog在寄存器级上(RTL)进行电路描述,并将电路实施在Virtex-7 FPGA开发板上进行了验证。

1 脉冲神经网络模型

脉冲神经网络的计算单元、数据传输以及网络模型来自对生物神经网络的模拟。本节构建应用于图像识别的脉冲神经网络,主要研究内容包括图像编码、脉冲神经网络神经元模型建模、网络模型结构及事件驱动型数据传输等4个部分。

1.1 图像编码

由于脉冲神经网络的输入和神经元间的通信都是以1位的脉冲来传输,编码的目的是为了将图像传感器采集到的图像表示成脉冲神经网络接收的脉冲序列。编码方法一般有时间编码^[15]、频率编码^[16]和群体编码^[17]等,其中最常用的是频率编码。本文适用于频率编码。

对图像像素进行频率编码时利用数学概率分布

模型,采用泊松分布的方式来产生脉冲序列,在每个时间点 t ,若输入神经元满足式(1)条件,则需要在该时间点产生一个脉冲,若不符合式(1)则在该时间点不产生脉冲

$$\text{rand}() < ax \quad (1)$$

式中: $\text{rand}()$ 表示在 $(0,1)$ 范围内均匀分布中随机抽取一个数; a 是缩放因子,用于调整神经元的脉冲个数发放率; x 是像素标准化后的值。

采用频率编码机制对图像像素的编码示意图如图 1 所示,不同明暗程度的图像像素在总的编码区间内产生了不同的脉冲个数。

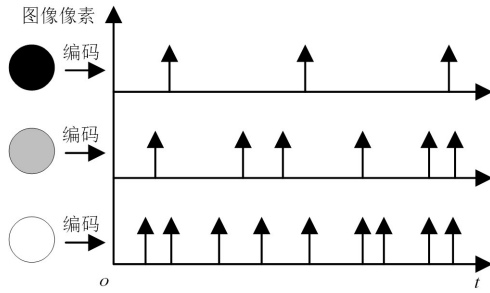


图 1 频率编码机制示意图

Fig. 1 Schematic diagram of rate coding mechanism

1.2 神经元建模

研究学者们提出了众多神经元模型用于模拟生物学大脑中神经元的膜电压变化,最常见的有 SRM 神经元^[18]、IF 神经元^[19]、LIF 神经元(积分泄露点火模型)^[20]、Hodgkin-Huxley 模型^[21]和 Izhikevich 模型^[22]等。本文使用的 LIF 神经元模型,是受到生物神经元启发的模型,具有计算简单、硬件友好的特点。LIF 神经元的物理模型示意如图 2 所示,体细胞的膜电压值在接收到树突传输的脉冲后会上升,当膜电压超过阈值时会产生一个脉冲,此时神经元的膜电压会下降到复位值并通过轴突向外发放脉冲。

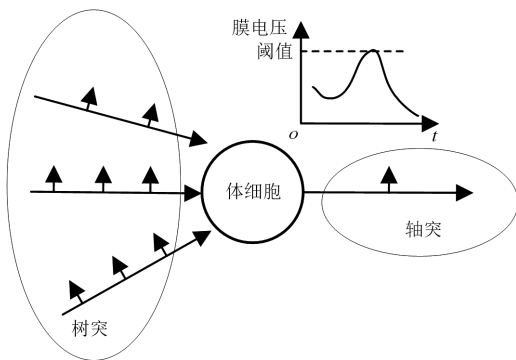


图 2 LIF 神经元的物理模型示意图

Fig. 2 Schematic diagram of the physical model of LIF neuron

LIF 神经元的膜电压变化可以总结如下

$$C_m \frac{dV}{dt} = g_l (V_{\text{reset}} - V) + I \quad (2)$$

式中: C_m 是膜电容; g_l 是电导率; V_{reset} 是复位膜电压; V 是神经元的膜电压; I 是输入电流。

为了以数字电路实施神经元模型,需要将 LIF 模型进行离散化,表示如下

$$\begin{cases} V_j(t) = V_j(t-1) + \sum_i w_{ij} S_i(t) - v_{\text{leak}} \\ S_j = 1 \text{ and } V_j(t) = V_{\text{reset}} \text{ if } V_j(t) \geq V_{\text{th}} \end{cases} \quad (3)$$

式中: $V_j(t)$ 和 $V_j(t-1)$ 是后突触神经元 j 在 t 时刻和 $t-1$ 时刻的膜电压;前突触神经元标记为 i ; w_{ij} 为前突触后突触神经元间的权重; V_{leak} 为神经元泄露值; V_{th} 为神经元阈值。离散化表示的神经元膜电压随输入脉冲的变化如图 3 所示,在每个时间点 t 上,神经元的膜电压若接收到脉冲,膜电压则对输入的脉冲进行集成并判断是否大于阈值。若大于阈值,则此刻膜电压复位为复位膜电压并发放出脉冲。若小于阈值,则实施泄露操作,膜电压此时减去固定的泄露值 V_{leak} 。

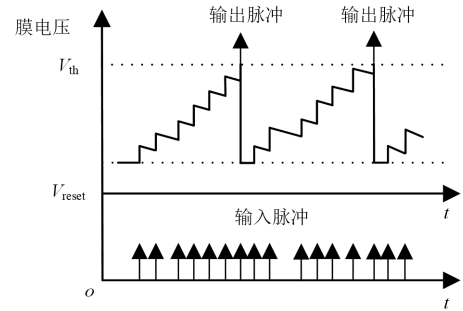


图 3 神经元膜电压的变化曲线示意图

Fig. 3 Schematic diagram of neuron membrane change curve

1.3 网络模型结构

图 4 为本文构建的用于图像目标识别的脉冲神经网络结构示意图,由 7 层网络层组成,包括 1 个输入层、2 个卷积层、2 个池化层和 2 个全连接层。输入层用于对图像进行编码,共包括 784 个神经元,分别代表图像大小为 28×28 中的一个像素。卷积层用于对上一层的输出进行特征提取,随着层数的增加,提取的特征也逐渐由局部到全局变化。每个卷积层后都紧跟着一个池化层,池化层被用于下采样并对卷积层提取的特征进行降维操作来减小数据信息和提高网络结构的容错率。第一个卷积层对输入图像采用 32 个 3×3 大小的卷积核进行卷积,卷积层后是采用 2×2 大小的平均池化层。第二个卷积层用于对第一层池化层的输出进行滤波,卷积核大小

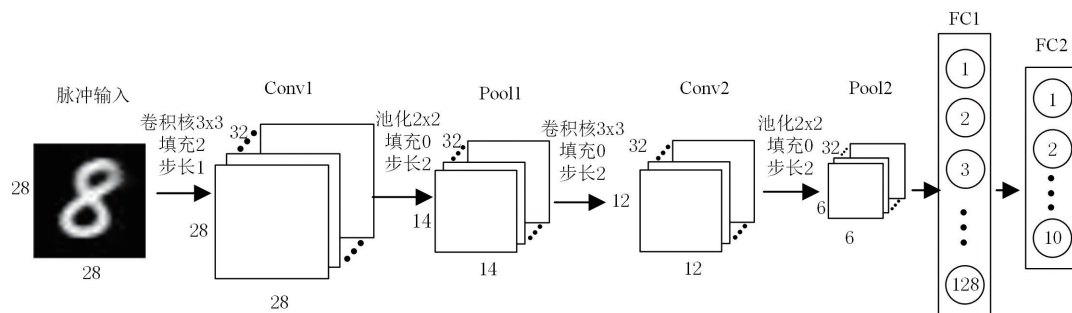


图 4 本文构建的脉冲神经网络结构示意图

Fig. 4 Schematic diagram of the proposed spiking neural network structure

为 $32 \times 3 \times 3$, 后面紧接着是一个 2×2 的平均池化层。全连接层用于对提取的特征进行分类, 2 个全连接层各包括 128 个神经元和 10 个神经元, 第二层池化层中 $32 \times 6 \times 6$ 个神经元与全连接层的 128 个神经元一对一连接, 第一个全连接层神经元与最后一层的 10 个神经元也一对一连接。最后一层的 10 个神经元分别代表着一个类别, 神经网络最终的识别结果根据最后一层 10 个神经元的个数来决定: 在整个编码的时间区间内, 输出个数最多的神经元所标记的类别作为该图像的最终识别结果。

1.4 基于事件传输型的数据传输

以输入层 28×28 个神经元为例, 由于 SNN 中加入了时间属性, 输入的数据可看作是三维的矩阵。考虑到脉冲的稀疏性, 在同一时间点上, 只有很少输入神经元产生脉冲, 若采用基于帧驱动的卷积操作需要计算大量的 0 值, 造成硬件资源与运算时间的浪费。本文采用基于事件驱动的数据传输, 将数据表示为地址事件表达 (AER) 的方式串行传输^[23]。根据基于事件传输型的数据传输方式, 上一层神经元与卷积层神经元的连接关系如图 5 所示, 每个上一层神经元都与当前卷积层的多个神经元连接。对于当前卷积层中连接的神经元称为接收野, 接收野的大小为 $q \times w_k \times h_k$, 其中 q 为卷积层中特征图的个数, w_k, h_k 为卷积核的宽度和高度。

事件驱动型的卷积操作使处理器可以在接收到第一个脉冲时就进行数据处理, 同时只要有卷积层中的神经元达到产生脉冲条件就可以输出脉冲, 提供给下一层神经元进行数据处理与运算。事件驱动型的计算避免了基于帧驱动型传输方式中需要等待当前帧所有数据到达后才能运算的缺点, 这种伪同步传输提升了运算速度。

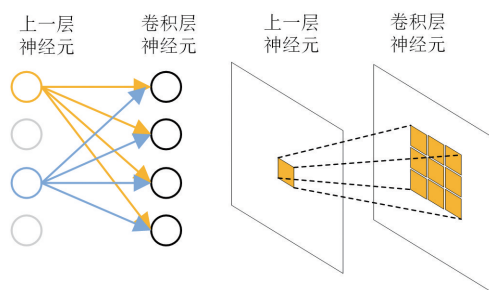


图 5 事件驱动型的神经元连接关系示意图

Fig. 5 Schematic diagram of event-driven neuronal connectivity

2 脉冲神经网络加速器设计方案

SNN 硬件设计具有高仿生学和高能效的优点, 适用于硬件资源和能量受限场景的解决方案。本节从加速器的总体架构出发, 分别对各模块电路结构进行设计, 实现了一种基于 SNN 模型的图像识别硬件电路, 提高了图像数据的处理效率。

2.1 结构总览

图 6 描述了本加速器的架构, 包括 4 个卷积工程机、一个全局控制器和一个数据接口。卷积工程机用于执行单层的卷积计算。在每个卷积工程机内都设计了一个局部控制器, 用于控制本工程机的数据流向、与其他工程机的通信以及存储器 (Block RAM) 的存取操作。全局控制器既要 SNN 加速器所有模块进行控制, 还要对输入的脉冲事件的时间点进行记录, 当时间点 t 变化时, 卷积工程机里的神经元需要执行泄露操作。数据接口是一个串行外设接口 (SPI), 用于接收上位机输入的脉冲事件并将加速器的产生的脉冲输出到上位机。每个卷积工程机里都设计了一个地址计算器、32 个神经元单元、存储器、事件产生器以及输入输出 FIFO, 存储器包括一个神经元状态存储器和一个卷积核存储器,

神经元状态存储器用于存储神经元的膜电压和神经元的泄露时间点,卷积核存储器用于存储卷积核的值。

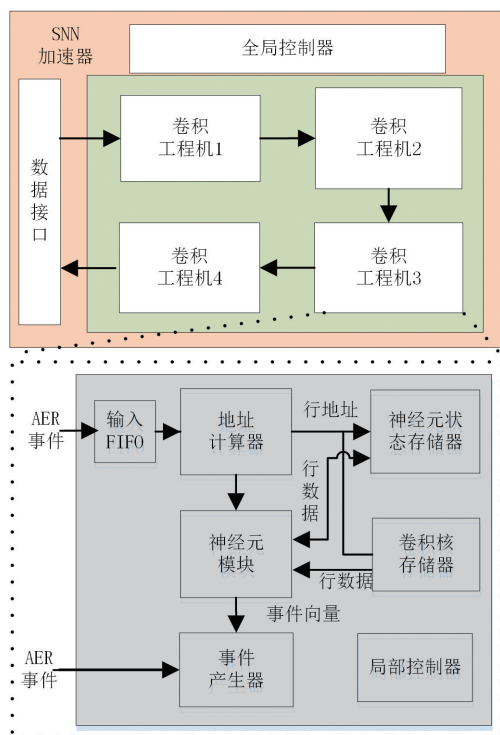


图 6 SNN 加速器结构总览图与卷积工程机电路框图

Fig. 6 Overview of SNN accelerator and block diagram of convolution engine circuit

为了实施多层的 SNN,不同层神经元之间的通信需要满足低延时传输。常见的 SNN 多核电路设计中^[13],采用的是多对多总线结构。但是,多对多的总线结构有 3 个缺点:一是需要复杂的路由器电路结构,导致高硬件资源需要;二是高扇入和高扇出的总线负载导致大的信号延时;三是多对多通信导致高数据冲突的概率。本设计设计了一对一的数据通信,串行的数据传输避免了数据冲突,不需要仲裁电路,而且保证电路的低扇入扇出。由于 SNN 的脉冲数据具有稀疏性的特点,既每个时间点上只有很少神经元产生脉冲^[24],因此加速器的输入脉冲以 AER 编码方式传输^[25],脉冲表示为 (t, y, x) ,其中 t 表示时间点, y 和 x 表示脉冲在图像中的位置。加速器的输出是图像的识别分类结果,由于是根据输出层神经元的个数进行判断,因此加速器的输出脉冲表示为 c ,即发放脉冲的神经元编号。

2.2 卷积工程机

卷积工程机由 7 个并行模块组成,电路框图在

图 6 里展示。为了加快数据处理,存储器的数据按行来存取。地址计算器根据输入的脉冲计算出神经元状态存储器和卷积核存储器的地址,两个存储器的行数据发送到神经元模块用于更新神经元膜电压,并判断膜电压是否大于神经元阈值。32 个神经元计算单元根据膜电压与阈值比较的结果输出 32 bit 的事件向量给事件产生器。事件向量中的每一位为 1 或 0,表示对应的神经元是否有脉冲产生。事件产生器接收到事件向量后仍旧将其编码为 AER 格式进行输出。当卷积工程机中没有数据处理时,该卷积工程机使用钟控技术使电路保持静态,减小电路的动态功耗。

2.3 数据存储

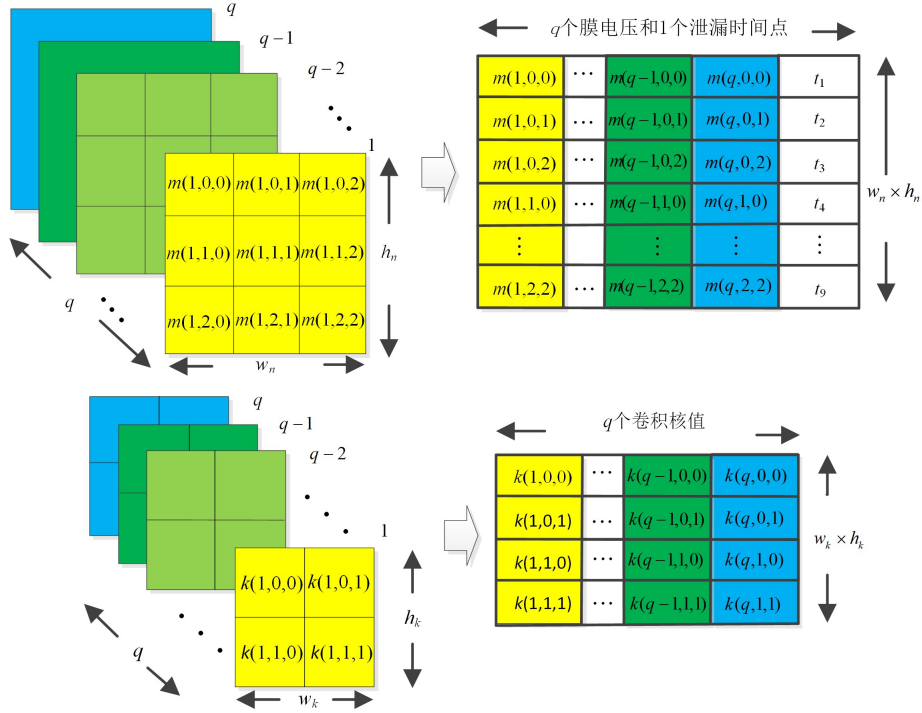
数据的存储方式对卷积处理的速度有着重要影响,对于基于事件驱动的脉冲卷积处理器,脉冲是逐个串行输入和处理。对于每个输入脉冲,需要更新 $q \times w_k \times h_k$ 个神经元膜电压和存取 $q \times w_k \times h_k$ 个卷积核值。以 $1 \times 7 \times 7$ 的卷积核为例,49 次的存储器读写增加了读写延时以及数据通信的瓶颈,但是可以通过改变数据存取的方式来改变,比如每个时钟周期内按行对存储器进行读取。在同一时钟周期内如何最大化的取出这些神经元膜电压是需要解决的问题。在神经网络中的神经元可看作是三维矩阵,卷积核看作是多个三维矩阵,需要将三维核多维的矩阵排放在二维的存储器中。考虑到卷积神经网络中的 q 都远大于卷积核尺寸 h_k ,因此将卷积层不同特征图中具有相同 (y, x) 坐标的神经元排列到存储器的同一行中,如图 7 所示。由于神经元在每个时间点上都需要实施泄露操作,为了减小膜电压的存取次数,在神经元状态存储器中增加了一列数据,每个数据用于记录本行的膜电压数据在上一次实施泄露的时间。卷积核存储器中的卷积核值以类似的方式进行数据排列,每个三维卷积核矩阵排列为二维存储器矩阵中,再将所有排列后的二维存储器矩阵堆叠在统一存储器中。

2.4 神经元模块

神经元模块采用流水线技术来实现数据的高速处理,为了保证数据流水线处理,神经元状态存储器选用双端口存储器。神经元模块包括 32 个神经元计算单元,神经元计算单元的电路结构如图 8 所示。图中 t 来自全局控制器,泄露时间点 (T) 和 32 个膜电压 $(m_1, m_2, \dots, m_{32})$ 在神经元状态存储器的第一

个端口输出,32 个卷积核值($k_1, k_2, \dots, k_{32}$)在卷积核存储器输出。根据当前时间点 t 与存储器中膜电压上一次实施泄漏的时间点 T 的差值($t - T$)乘以每个时间点需要泄漏值 V_{leak} , 得到实施泄漏要减掉

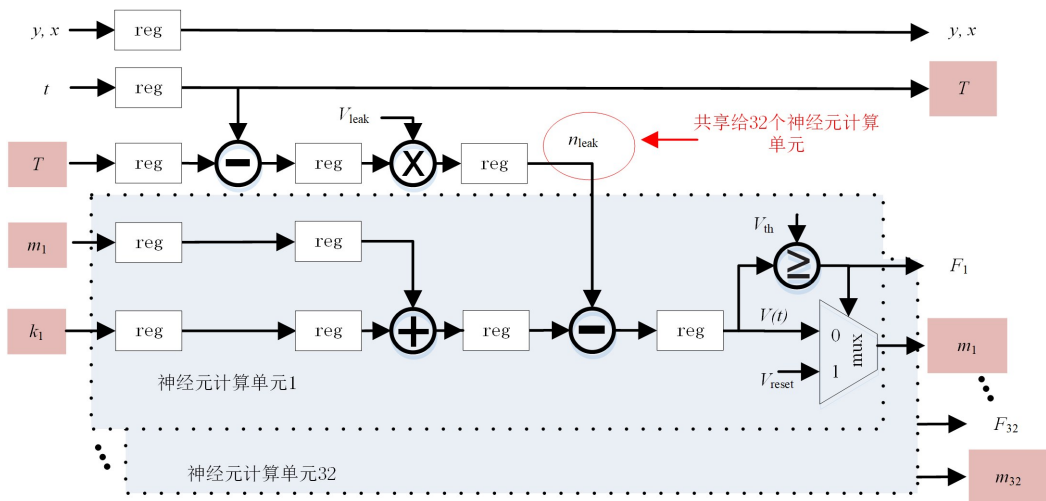
的泄漏值 n_{leak} 。泄漏值被所有神经元计算单元共享来减小硬件资源。在更新该行所有的膜电压后,将膜电压与当前时间点写入到神经元状态存储器的第二个端口。



$m(z, y, x)$ —坐标为 (z, y, x) 神经元的膜电压; $k(z, y, x)$ —坐标为 (z, y, x) 的卷积核的值;
 z —卷积层中特征图的个数; y, x —行标签和列标签。

图 7 神经元膜电压以及卷积核值在存储器中的排列方式示意图

Fig. 7 Schematic representation of neuron membranes and kernel values arranged in memory



T —泄漏时间点; F_i —事件标记, 32 个 F_i 组成事件向量; m_i, k_i —神经元膜电压和卷积核值 ($i = 1, 2, \dots, 32$)。

图 8 神经元模块电路结构图

Fig. 8 Circuit structure diagram of neuron module

2.5 事件产生器

事件产生器用于将事件向量中的脉冲编码为 AER 格式。对事件向量中的每一位逐位判断是否有脉冲是低效率的,因此采用了基于查找表的方法,实现了优先编码器的功能。事件产生器的电路结构图如图 9 所示,将事件向量输入查找表,下个时钟则输出事件向量中最高位为 1 的位置,再将该位置与事件向量进行异或操作,将对应位置所在的位数置 0。结合查找表与异或操作查找事件向量中的脉冲过程如图 10 举例所示。假设事件向量的第 29 位为 1,表示在此处有一个脉冲,此时将事件向量输入到

查找表中,下个时钟查找表输出 $f' = 29$ 以及独热码。将其与事件向量进行异或,则得到清除了第 29 位标记的事件向量。依此循环,直到事件产生器中所有的事件都被编码。

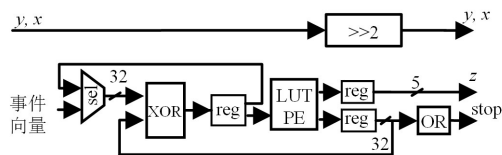


图 9 事件产生器电路结构图

Fig. 9 Circuit structure diagram of event generator

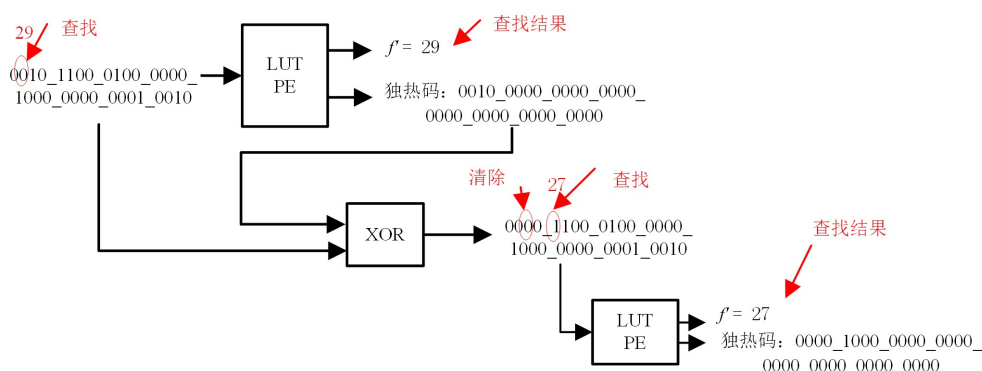


图 10 事件产生器查找脉冲位置举例示意图

Fig. 10 Schematic illustration of the event generator finding the spike position

虽然脉冲具有稀疏性,但是时间向量中也会有多个脉冲事件的情况,此时事件产生器处理每个事件向量时可能需要多个时钟。若事件产生器对当前事件向量还未处理完毕,又有新的事件向量输入,此时会造成数据冲突。为了避免数据冲突,当前的事件向量的脉冲事件未全部找到时,事件产生器会产生一个“停止”(stop)信号,使地址计算器和神经元模块中的寄存器值保持不变,直到脉冲事件全部处理完,再使寄存器继续工作。产生脉冲的卷积层神经元坐标为 (z, y, x) , z 通过结合查找表与异或操作求出。池化层对卷积层进行下采样, 2×2 的下采样在硬件中可以通过移位来实现,避免了硬件昂贵的乘法器的使用。

3 实验验证与结果分析

本文的设计通过可综合的硬件描述语言进行实现,在 Vivado2016 软件上进行综合,并实施在 Xilinx Virtex-7 VC707 FPGA 开发板中,该开发板包含 485 000 个逻辑单元,2 800 个 DSP 资源以及 37 Mb 的

存储单元。本文提出的设计在开发板上的工作频率为 100 MHz。本设计使用了 MNIST 手写数字图像数据集进行测试,该数据集包括 7 万张手写数字图像,每张图像大小为 28×28 ,其中的 5 万张为训练集,其余 2 万张为测试集,该数据集的示意图如图 11 所示。

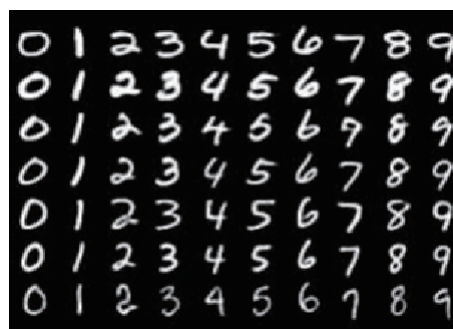


图 11 MNIST 数据集示意图

Fig. 11 Schematic of the MNIST dataset

为了保证准确率和硬件资源的平衡,首先在软件上采用不同的权值位数测试识别结果。如图 12 所示,随着卷积核位数增大,识别准确度也逐渐提高,并

在 8 bit 后稳定,最终卷积核的位数选择为 8 bit。

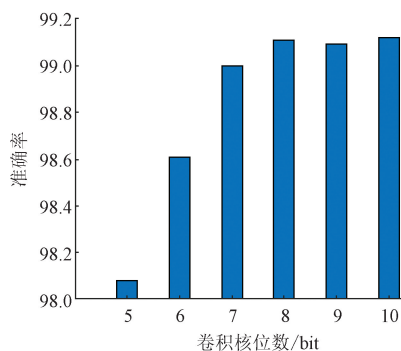


图 12 卷积核位数与加速器分类识别性能关系

Fig. 12 Relationship between the kernel bit width and the recognition performance of the accelerator

为了比较软件仿真与硬件加速器的性能,该 SNN 模型也在计算机上使用 Python 软件进行了仿真,计算机的配置为 3.2 GB 的英特尔中央处理器 (CPU) 与 16 GB 的内存。软件程序对测试集的 2 万张图像进行测试。为了避免引入计算机外围设备的工作功耗,中央处理器 (CPU) 在软件程序运行时的功耗 P 由计算机的 AIDA64 软件进行测量。使用 Python 的 `time.clock()` 函数以浮点数来计算软件程序开始运行的 CPU 时间 t_1 和结束运行时的 CPU 时间 t_2 , 得到软件程序识别 2 万张图像的总运行时间 $(t_2 - t_1)$ 和识别单张图像的运行时间 t 。识别单张图像的能量 E 由功耗 P 与运行时间 t 相乘得到, 公式为 $E = Pt$ 。本文提出的加速器设计方案通过 Vivado2016 软件实施在 FPGA 上, 并使用 Xilinx 的功耗分析器 (Xpower Analysis tools) 对设计进行分析并得到功耗。在硬件设计中添加了一个计数器, 通过对时钟数进行计数, 最后得到了加速器识别单张图像的运行时间。由表 1 的性能对比可知, 本文的硬件设计与软件仿真相比, 具有更快的处理速度以及具有显著优势的能量效率。硬件的处理速度、功耗及能量为软件仿真速度的 $1/7.7$ 、 $1/186.2$ 和 $1/1453.8$ 。虽然本设计采用了定点数进行计算, 仍然获得了 99.11% 的准确率, 准确率和采用 32 位浮点数的软件的准确率相近。

表 1 硬件设计与软件仿真的性能对比

Table 1 Performance comparison between hardware design and software simulation

平台	时间/ms	功耗/W	能量/mJ	准确率/%
硬件	3.6	0.29	1.04	99.11
软件	28	54	1512	99.58

为了进一步比较本设计的优势, 本设计与其他采用 FPGA 实现的硬件加速器方案进行对比, 如表 2 所示。文献[9]是“达尔文”芯片, 采用单核 (8 个神经元) 复用计算技术, 相比于本设计中多核 (每个核 32 个神经元) 计算, 其低神经元并行度造成处理单张图像需要 160 ms 的时间。文献[13]采用多核结构, 但是核与核之间采用了复杂的多对多通信, 增加了硬件资源和通信延时; 采用基于帧驱动的计算增加了对无用数据 0 的存储与计算, 使电路的功耗高达 1.6 W。本方案通过设计多核并行工作方式实现了多神经元的高并行计算, 采用基于事件驱动的数据传输增加了数据效率, 只需 3.6 ms 的时间和 1.04 mJ 的能量来完成单张图像的识别, 能量分别是文献[9]与文献[13]的 $1/3.23$ 和 $1/4.85$ 。实验结果表明, 本方案的能效领先于主流基于 FPGA 实现的 SNN 硬件水平。

表 2 本文设计与其他文献的性能对比

Table 2 Performance comparison between the proposed design and other literatures

方案来源	器件型号	时钟频率 /MHz	时间 /ms	功耗 /W	能量 /mJ
本文	Virtex-7	100	3.6	0.29	1.04
文献[9]	Spartan-6	25	160	0.021	3.36
文献[13]	Virtex-7	100	3.15	1.6	5.04

4 结 论

为了解决传统冯诺依曼结构上软件仿真速度慢、能耗高和体积大等问题, 本文以神经形态学的脉冲神经网络为研究背景, 提出了一种用于图像识别的脉冲神经网络加速器设计方案。为了解决脉冲稀疏性导致的基于帧传输信息效率低的问题, 提出了一种基于事件驱动的脉冲传输方式。通过多核并行工作, 每个卷积工程机实现单层的卷积计算, 提高了网络的处理速度。提出了一种新的数据排列方式, 提高了数据的利用率和神经元计算的并行度。设计了结合查找表与异或的电路结构, 可以高效快速的对脉冲位置进行定位。

本文基于寄存器级电路设计, 全面仿真验证, 并在 FPGA 开发板上进行了实现, 使用经典的手写数字图像数据集进行测试。实验结果表明, 与基于冯诺依曼结构的软件仿真方案进行对比, 本设计具有高速度低能耗的优势。在与同类型硬件电路的比较中, 本设计的能效优势处于主流水平, 更适用于

能量体积等受限的边缘设备中。

参考文献:

- [1] GREENBERG A, HAMILTON J, MALTZ D A, et al. The cost of a cloud: research problems in data center networks [J]. ACM SIGCOMM Computer Communication Review, 2009, 39(1): 68-73.
- [2] TOGAÇAR M, ERGEN B, CÖMERT Z. Detection of weather images by using spiking neural networks of deep learning models [J]. Neural Computing and Applications, 2021, 33(11): 6147-6159.
- [3] 马新娜, 赵猛, 祁琳. 基于卷积脉冲神经网络的故障诊断方法研究 [J]. 广西师范大学学报(自然科学版), 2022, 40(3): 112-120.
MA Xinna, ZHAO Meng, QI Lin. Fault diagnosis based on spiking convolution neural network [J]. Journal of Guangxi Normal University(Natural Science Edition), 2022, 40(3): 112-120.
- [4] 胡一凡, 李国齐, 吴郁杰, 等. 脉冲神经网络研究进展综述 [J]. 控制与决策, 2021, 36(1): 1-26.
HU Yifan, LI Guoqi, WU Yujie, et al. Spiking neural networks: a survey on recent advances and new directions [J]. Control and Decision, 2021, 36(1): 1-26.
- [5] BOUVIER M, VALENTIAN A, MESQUIDA T, et al. Spiking neural networks hardware implementations and challenges: a survey [J]. ACM Journal on Emerging Technologies in Computing Systems, 2019, 15(2): 22.
- [6] VAN SCHAIK A. Building blocks for electronic spiking neural networks [J]. Neural Networks, 2001, 14(6/7): 617-628.
- [7] MITRA S, FUSI S, INDIVERI G. Real-time classification of complex patterns using spike-based learning in neuromorphic VLSI [J]. IEEE Transactions on Biomedical Circuits and Systems, 2009, 3(1): 32-42.
- [8] AKOPYAN F, SAWADA J, CASSIDY A, et al. TrueNorth: design and Tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2015, 34(10): 1537-1557.
- [9] MA De, SHEN Juncheng, GU Zonghua, et al. Darwin: A neuromorphic hardware co-processor based on spiking neural networks [J]. Journal of Systems Architecture, 2017, 77: 43-51.
- [10] FURBER S B, GALLUPPI F, TEMPLE S, et al. The SpiNNaker project [J]. Proceedings of the IEEE, 2014, 102(5): 652-665.
- [11] DENG Lei, WANG Guanrui, LI Guoqi, et al. Tian-jie: a unified and scalable chip bridging spike-based and continuous neural computation [J]. IEEE Journal of Solid-State Circuits, 2020, 55(8): 2228-2246.
- [12] BENJAMIN B V, GAO Peiran, MCQUINN E, et al. Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations [J]. Proceedings of the IEEE, 2014, 102(5): 699-716.
- [13] LI Sixu, ZHANG Zhaomin, MAO Ruixin, et al. A fast and energy-efficient SNN processor with adaptive clock/event-driven computation scheme and online learning [J]. IEEE Transactions on Circuits and Systems: I Regular Papers, 2021, 68(4): 1543-1552.
- [14] WU Yujie, DENG Lei, LI Guoqi, et al. Spatio-temporal backpropagation for training high-performance spiking neural networks [J]. Frontiers in Neuroscience, 2018, 12: 331.
- [15] 刘宙思, 李尊朝, 张剑, 等. 一种离散时间调度的图像自分类脉冲神经网络 [J]. 西安交通大学学报, 2021, 55(3): 57-64.
LIU Zhousi, LI Zunchao, ZHANG Jian, et al. Discrete-time-scheduling spiking neural network for image self-classification [J]. Journal of Xi'an Jiaotong University, 2021, 55(3): 57-64.
- [16] KOSTAL L, LANSKY P, ROSPARS J P. Neuronal coding and spiking randomness [J]. European Journal of Neuroscience, 2007, 26(10): 2693-2701.
- [17] 苏亚丽, 吴健行, 惠维, 等. 一种用于视觉颜色特征分类的脉冲神经网络 [J]. 西安交通大学学报, 2019, 53(10): 115-121.
SU Yali, WU Jianxing, HUI Wei, et al. A spiking neural network for classification of visual color features [J]. Journal of Xi'an Jiaotong University, 2019, 53(10): 115-121.
- [18] GERSTNER W, KISTLER W M. Spiking neuron models: single neurons, populations, plasticity [M]. New York, USA: Cambridge University Press, 2002: 20-25.
- [19] ABBOTT L F. Lapicque's introduction of the integrate-and-fire model neuron (1907) [J]. Brain Research Bulletin, 1999, 50(5/6): 303-304.
- [20] 程龙, 刘洋. 脉冲神经网络:模型、学习算法与应用 [J]. 控制与决策, 2018, 33(5): 923-937.
CHENG Long, LIU Yang. Spiking neural networks: model, learning algorithms and applications [J]. Control and Decision, 2018, 33(5): 923-937.
- [21] IZHIKEVICH E M. Simple model of spiking neurons [J]. IEEE Transactions on Neural Networks, 2003, 14(6): 1569-1572.

- [22] KUMAR A, ROTTER S, AERTSEN A. Spiking activity propagation in neuronal networks: reconciling different perspectives on neural coding [J]. *Nature Reviews Neuroscience*, 2010, 11(9): 615-627.
- [23] SERRANO-GOTARREDONA R, SERRANO-GOTARREDONA T, ACOSTA-JIMENEZ A, et al. On real-time AER 2-D convolutions hardware for neuromorphic spike-based cortical processing [J]. *IEEE Transactions on Neural Networks*, 2008, 19(7): 1196-1219.
- [24] SERRANO-GOTARREDONA R, OSTER M, LICHTSTEINER P, et al. CAVIAR: a 45 K neuron, 5 M synapse, 12 G connects/s AER hardware sensory-processing- learning-actuating system for high-speed visual object recognition and tracking [J]. *IEEE Transactions on Neural Networks*, 2009, 20(9): 1417-1438.
- [25] SON B, SUH Y, KIM S, et al. 4.1 A 640×480 dynamic vision sensor with a $9 \mu\text{m}$ pixel and 300Meps address-event representation [C]//2017 IEEE International Solid-State Circuits Conference (ISSCC). Piscataway, NJ, USA: IEEE, 2017: 66-67.

(编辑 荆树蓉 刘杨)