基于脉动阵列的卷积计算模块硬件设计

王春林，谭克俊
（大连海事大学 信息科学技术学院，辽宁 大连 116026）

摘 要：针对FPGA实现卷积神经网络中卷积计算的过程中，高并行度带来长广播、多扇入/扇出的数据通路问题，采用脉动阵列来实现卷积神经网络中卷积计算模块，将权重固定到每个处理单元中，并按照输入和输出特征图的维度来设置脉动阵列的大小，最后通过Vivado高层次综合实现卷积计算模块的硬件设计。实验结果表明，本设计在实现1级流水化时序要求的同时，具有较低的资源占用和良好的扩展性。

关键词：FPGA；脉动阵列；卷积计算；高层次综合
中图分类号：TN402；TP391.41　　**文献标识码**：A　　**DOI**：10.16157/j.issn.0258-7998.191070

中文引用格式：王春林，谭克俊. 基于脉动阵列的卷积计算模块硬件设计[J].电子技术应用，2020，46(1)：57-61.
英文引用格式：Wang Chunlin，Tan Kejun. Hardware design of convolution calculation module based on systolic array[J]. Application of Electronic Technique，2020，46(1)：57-61.

Hardware design of convolution calculation module based on systolic array

Wang Chunlin，Tan Kejun
（Information Science and Technology College，Dalian Maritime University，Dalian 116026，China）

Abstract：Aiming at the long broadcast，much fan in/fan out data path problem brought by high parullelism in the process of the Field Programmable Gate Array(FPGA) to realize the convolution computation in convolutional neural network，this paper adopts pulse array to realize convolution calculation module of convolutional neural network，fixes weights to each processing unit，according to the dimension of the input and output characteristic figure sets to pulse array size，and finally by Vivado high level synthesis realizes convolution calculation module hardware design. The experimental results show that the design has low resource occupancy and good expansibility while realizing the time-series requirements of level 1 pipelining.

Key words：FPGA；systolic array；convolution computation；high level synthesis

0 引言

在过去的几年里，深度神经网络(Deep Neural Network，DNN)在图像分类、目标检测[1]及图像分割等领域起到十分重要的作用。这些使用的各种DNNs及其拓扑结构中，卷积神经网络(Convolutional Neural Network，CNN)是其中最为常见的实现方式。目前在硬件加速方案中，主要有基于CPU、GPU以及FPGA三种主流方案。考虑到CPU性能限制和GPU功耗高的问题，采用FPGA来实现卷积神经网络成为了一种可行的实现方式，例如文献[2]在FPGA中实现神经网络目标检测系统，其检测速度与能效均优于CPU。采用传统的Verilog HDL或者VHDL硬件描述语言实现卷积神经网络较为困难[3]，高层次综合(High Level Synthesis，HLS)将C/C++代码通过特定的编译器转化为相应的RTL级的代码，降低了卷积神经网络的开发难度，减少了卷积神经网络的开发周期。

使用FPGA实现卷积神经网络中卷积计算模块的过程中，通常采用循环平铺和循环展开[4]的方式实现。这种方式以扩大并行度来达到网络的时间复杂度。但是当输入和输出特征图维度增加时，扩大并行度会带来硬件设计中长广播、多扇入/扇出的数据通路，导致卷积计算模块无法在较高的主频上运行。因此，很多神经网络加速器都使用脉动阵列来优化加速器架构设计，如谷歌TPU加速器[5]、ShiDianNao加速器[6]等。而在这些加速器架构设计中大多是采用im2col[7]的方式，即将参与卷积计算的输入特征图和权重展开为两个矩阵，然后进行矩阵乘法运算。这种实现方式因为卷积步长的存在而产生大量的数据重叠，不利于在FPGA的片上块存储器(Block RAM，BRAM)内进行存储。

为了解决上述存在的问题，本文提出一种基于脉动阵列的卷积计算模块设计，将由并行展开所带来的长数据通路变为每个处理单元的短数据通路；并按照存储矩阵的坐标向卷积计算模块中输入特征图数据，以解决im2col方式存在的数据重叠，不利于BRAM存储的问题。整体设计使用Vivado HLS开发环境进行实现与优化。

1 本文工作

1.1 脉动阵列实现卷积计算模块

脉动阵列(Systiloc Array)[8]是1970年KUNG H T[9]提出的一种应用在片上多处理器的体系结构,由多个相同的、结构简单的计算单元(Processing Element,PE)以网格状形式连接而成,具有并行性、规律性和局部通信的特征。信号处理算法如卡尔曼滤波[10]和数值线性代数算法都可以用脉动阵列来实现。本文卷积计算模块中采用的脉动阵列实现方式如图1所示。

在图1中,I表示输入特征图,W表示权重参数,O表示输出特征图,r、c分别表示特征图的长和宽,m、n分别表示输入特征图与输出特征图的层数。在开始进行卷积运算之前,将特征图数据输入到BRAM中进行缓存,将权重输入到每个PE中进行缓存。开始计算之后,输入缓存中的数据沿脉动阵列的列方向进行传输,PE计算的结果沿脉动阵列行方向进行传输。非第0列的PE按照公式(1)进行计算:

$$PE_{out}(cho,chi)=PE_{out}(cho,chi-1)+PE_{in}(cho,chi)\times W(cho,chi) \quad (1)$$

式中PE_{in}表示从输入缓存或者上一个PE读取输入数据,W表示PE缓存的权重数据,PE_{out}表示每个PE储存的计算结果,cho与chi分别表示当前PE的行列坐标。第0列的PE只进行乘法运算。在每一行PE的末尾处连接一个result缓存,用来累加和存储每行最后一个PE输出的结果,并在完成一个卷积核运算之后将存储结果输出到输出缓存中。本文脉动阵列结构设置为矩形脉动阵列,长和宽分别以输入特征图和输出特征图的层数来部署,并且每个PE在一个时钟周期内进行一个乘法和加法运算,根据这个条件可以得到所有PE完成计算所需要的时间计算公式:

$$T_{sum}=(R\times C\times K\times K+(C_{out}-1)+(C_{in}-1))\times T_{prc} \quad (2)$$

式中,T_{sum}表示完成所有输出特征图的计算所需要的时间,R、C分别表示输出特征图的长和宽,C_{in}、C_{out}分别表示输入特征图层数和输出特征图层数,K表示卷积核的边长,T_{prc}表示每个时钟周期所需要的时间。如果R和C的大小设置为5,C_{in}和C_{out}分别设置为3和8,K设置为3,T_{prc}为10 ns,根据公式(2)可以得到理论上需要的时间为2 340 ns。

1.2 卷积计算模块硬件设计

根据1.1节中脉动阵列在卷积计算模块中的实现方式,在Vivado HLS开发环境上对卷积计算模块进行设计。卷积计算模块分为三个部分:输入阶段、计算阶段和累加输出阶段。

输入阶段的流程图如图2所示,图中in(chi,ir,ic)表示BRAM存储的输入特征图,ir、ic分别表示特征图的长和宽。mid_in(cho,chi)和mid_out(cho,chi)分别表示每一个PE中的输入和输出缓存,CHI表示输入特征图的层数,CHO表示输出特征图层数,COUNT表示一个PE需要进行卷积运算的次数。chi、cho、loc表示三个变量,分别表示脉动阵列的列坐标、行坐标以及运行计数。根据图2中的运行方式,每经过一个时钟周期,位于脉动阵列非0列的mid_in会从上一个相同行的mid_in中读取输入数据。位于脉动阵列的第0列的mid_in,会根据运行状态判断是否需要从BRAM中读取输入特征图数据。

计算阶段的流程图如图3所示,其中weight(cho,chi,kr,kc)表示缓存到PE中权重参数,kr、kc分别表示卷积

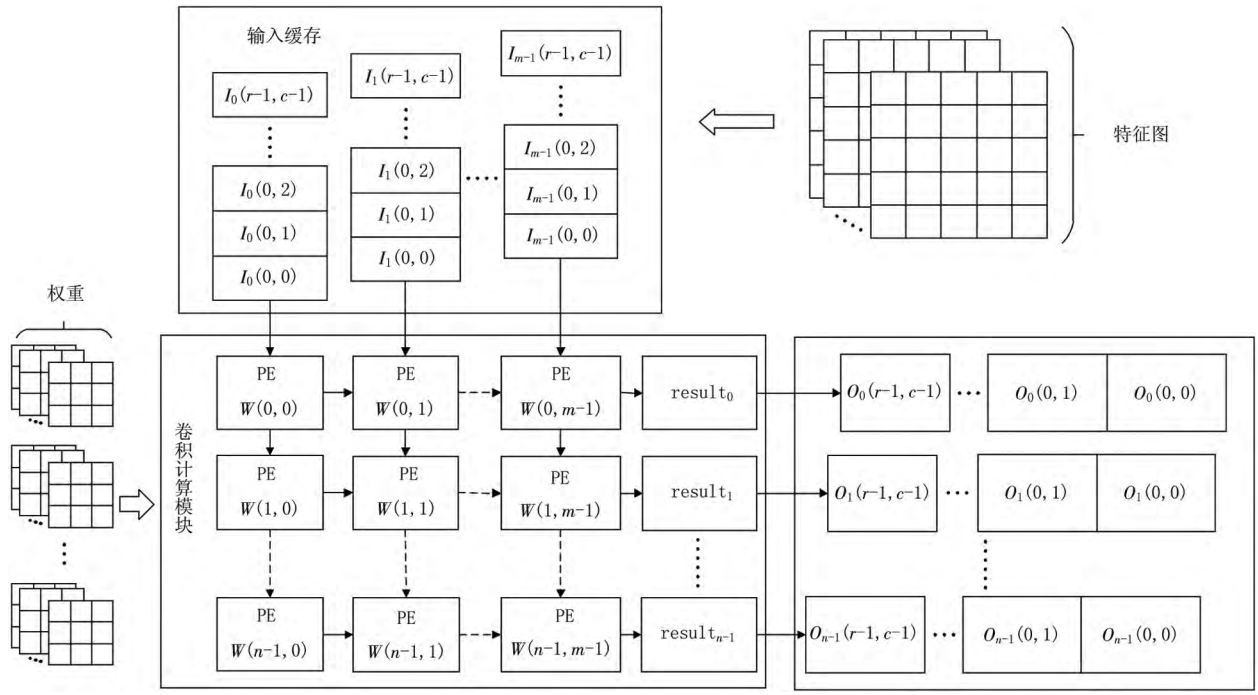

图1 采用脉动阵列实现卷积计算模块

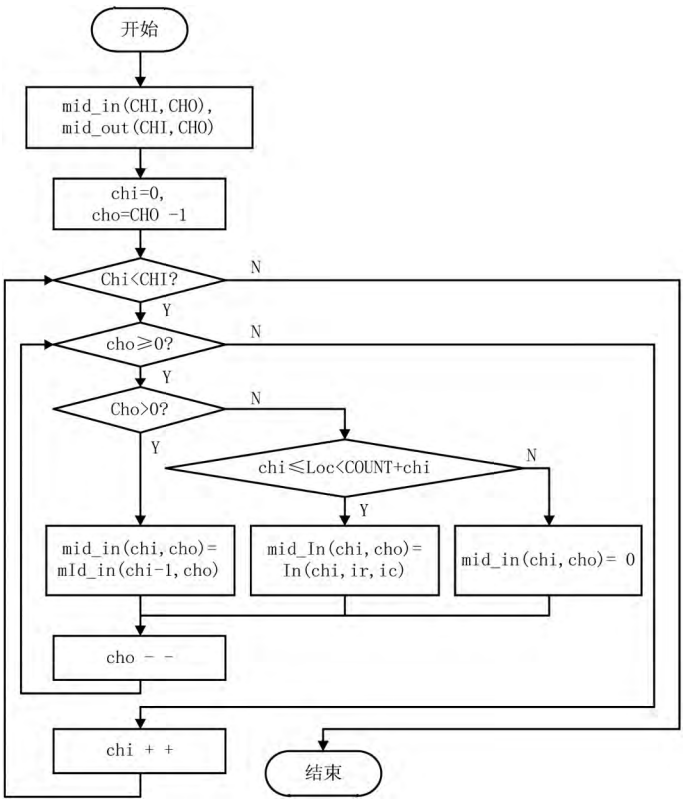

图 2 输入阶段流程图

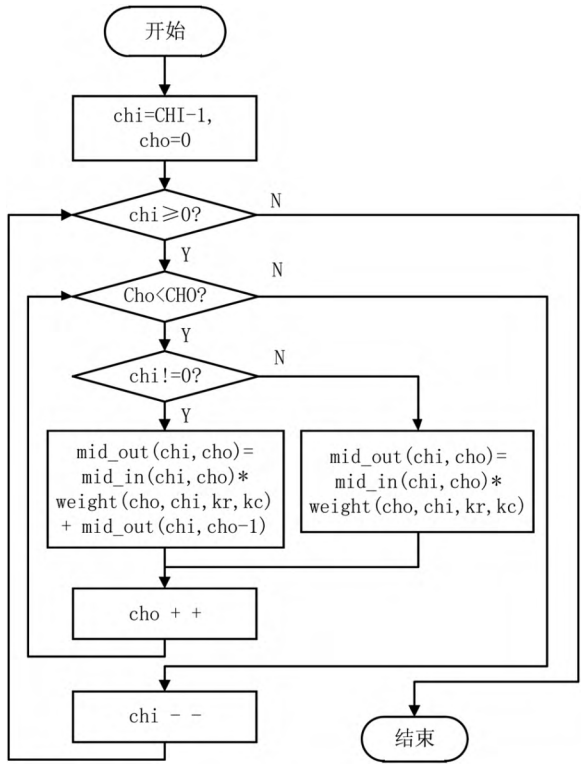

图 3 计算阶段流程图

核的列坐标与行坐标。当输入阶段结束之后，进入计算阶段。位于脉动阵列非 0 排的 mid_out 会将 mid_in 中的数据与 weight 中的数据做乘法，再和上一个相同列的 mid_out 中的输出数据做加法。位于脉动阵列第 0 排的 mid_out 则只做一次乘法。PE 计算结果保存在当前 mid_out 中。

累加输出阶段的流程图如图 4 所示，图中 out(cho，r，c)表示 BRAM 中输出特征图缓存，r、c 分别表示输出特征图的列坐标与行坐标。result 表示输出缓存到 BRAM 之间的累加寄存器，用来将脉动阵列末尾行 PE 的结果进行累加，k 表示寄存器运行的累加次数。当一个卷积核运算完成之后，将结果按照当前的行列坐标输出到 BRAM 中的 out 缓存中，之后 result 寄存器清零，继续下一次累加计算。

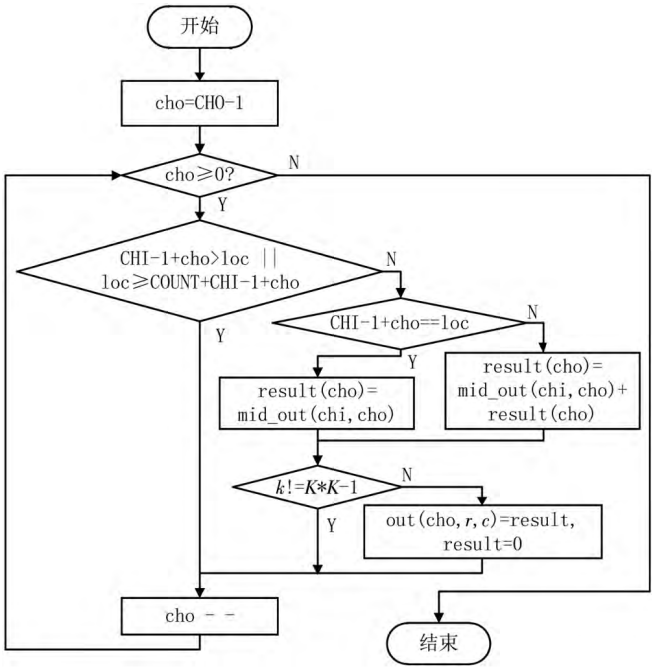

图 4 累加输出阶段流程图

在这三个阶段运行过程中，希望在每一个时钟周期内，当前 PE 可以从外部或者相邻的 PE 中读取一个数据进行乘加计算。因此使用 #pragma HLS PIPELINE 管道操作优化指令，并设置流水为 1 级流水，保证每个时钟内都能够开启一次新的计算。在使用流水化操作之后，为了保证在输入阶段和计算阶段的数据不会被覆盖，采用移位寄存器的运行方式，由末尾的 PE 开始依次从相邻的上一个 PE 读取数据，新数据最后再进行输入。

同时，因为采用将权重固定到 PE 中的计算模式，所以将 In 和 Out 以数组的形式存放到 BRAM 中。因为每个 BRAM 最大可配置的输入输出端口数为 2，所以当同时从 BRAM 中输入输出数量大于 2 时，就需要等待上一个操作结束之后，再执行下一个操作，这样就无法达到 1 级流水所需要的时间间隔。因此使用 #pragma HLS ARRAY_PARTITION variable =<variable> complete dim =X 指令，其中<variable>表示需要展开的数组名，X 表示需要展开的数组维度，将 in(chi，ir，ic)和 out(cho，r，c)按照第一个维度展开成多个数组，来匹配脉动阵列的输入与

输出。

2 实验结果与分析

本文在 Vivado HLS 18.3 开发环境上进行综合与仿真，使用的 FPGA 芯片型号为赛灵思公司的xc7z020clg484-1，运行时钟为 100 MHz。使用输入特征图 7×7 大小的 3 层矩阵，得到的输出特征图为 5×5 的 8 层矩阵。卷积核采用 8×3 组大小为 3×3 的矩阵。单个 PE 的功能仿真波形如图 5 所示。

其中，din0 和 din1 分别表示输入特征图和权重数据，din2 表示从上一个相邻 PE 中读取的计算结果。dout 表示乘加计算之后的结果，d0 表示 result 寄存器中的输入数据，q0 表示 result 寄存器的输出数据。由于本文采用的时钟频率为100 MHz，每个时钟周期为 10 ns。从图 5 中可以看出，当 ce0 信号置 1 时，表示开始进行运算。之后在每一个时钟周期内，都进行一次乘加运算，并将结果通过 dout 进行输出。因为采用 1 级流水优化指令，dout 输出的结果将会在下一个时钟周期输出到 d0。然后进行累加操作。当完成 3×3 次加法运算，输出标志位 we0 信号置 1，然后 result 寄存器将当前的 d0 结果按照 address0 的地址通过 q0 输出到指定 BRAM 上的输出缓存中。然后 d0 清零，等待下一次计算开始。

卷积计算模块的输出波形如图 6 所示。从图中可以看出，在 ap_start 使能信号置 1 之后，经过 17 个时钟周期得到输出结果。从开始输出结果，到输出完成，一共经过 234 个时钟周期，也就是 2 340 ns，与公式(2)中计算结果相同。

卷积计算模块的总体时延如表 1 所示，从表中可以看出，每次计算 PE 需要 17 个时钟周期才能够得到结果。在使用了流水化操作，并且达到了 1 级流水的目标之后，每个时钟周期都会开启一个新的循环，PE 无需等

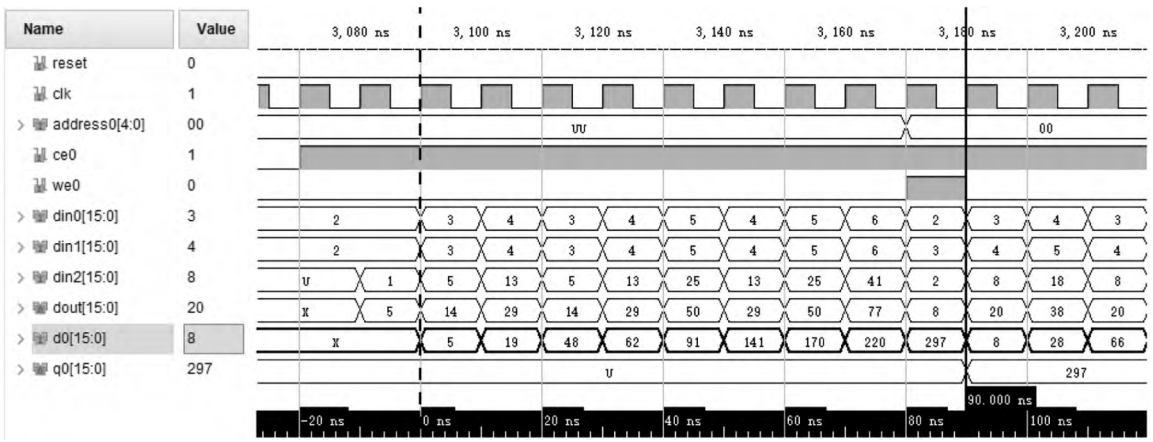

图 5 单个 PE 功能波形图

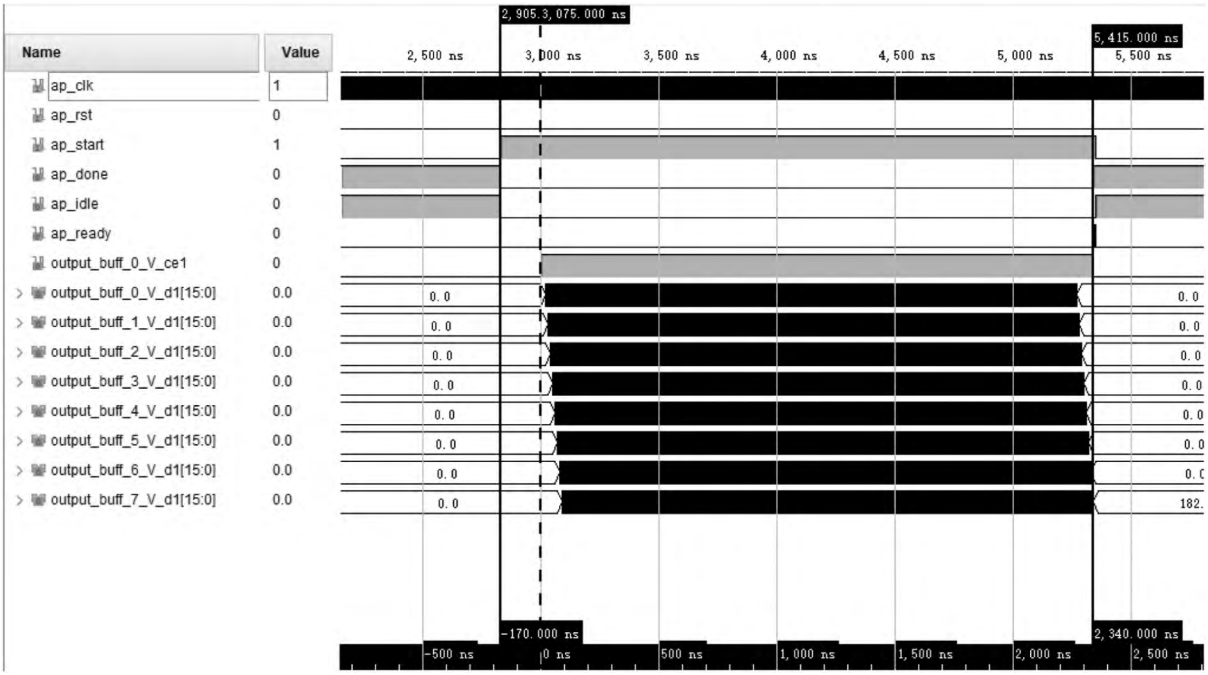

图 6 卷积计算模块输出波形图

表 1 卷积计算模块的总体时延

名称	时延		单次时延	启动间隔		循环次数	是否并行
	最小值	最大值		实现	目标		
卷积模块	251	251	17	1	1	234	是

待到计算出结果就可以进行下一个循环。与图 6 中的波形结果相同。

表 2 所示为卷积计算模块的总体资源消耗，卷积计算的核心就是乘法运算。赛灵思 xc7z020clg484-1 型FPGA芯片内置了 DSP48 核，可同时进行一组乘法和加法运算。根据卷积模块的设计，每一个 PE 使用一个 DPS48 核进行乘加计算。在 CHI 为 3，CHO 为 8 的情况下，一共需要 24 个 DPS48 核，与表中使用资源情况相符。触发器和查找表资源使用也较少，为之后扩大卷积计算模块和设计卷积神经网络其余模块预留了充足的资源。

表 2 卷积计算模块总体资源消耗表

名称	使用资源	总体资源	利用率/%
BRAN_18K	7	280	2
DSP48E	24	220	10
FF	11 166	106 400	10
LUT	9 399	53 200	17

3 结论

卷积神经网络中存在着大量的卷积计算，本文在资源使用情况较少的情况下，基于脉动阵列的运行方式，对并行展开的卷积计算模块进行改进，然后通过 Vivado HLS 在赛灵思 xc7z020clg484-1 型 FPGA 芯片上进行实现。在后续的研究中，可以将脉动阵列与循环展开和循环平铺等并行展开方式相结合，从提高运行速度与降低使用资源上进一步提升卷积计算模块的性能。

参考文献

[1] 张杰，隋阳，李强，等.基于卷积神经网络的火灾视频图像检测[J].电子技术应用，2019，45(4)：34-38，44.

[2] 陈辰，严伟，夏珺，等.基于 FPGA 的深度学习目标检测系统的设计与实现[J].电子技术应用，2019，45(8)：40-43，47.

[3] Zhang Xiaofan，Wang Junson，Zhu Chao，et al.DNN-Builder：an automated tool for building high-performance DNN hardware accelerators for FPGAs[C]. Proceedings of the International Conference on Computer-Aided Design.ACM，2018.

[4] Zhang Chen，Li Peng，Sun Guangyu，et al.Optimizing fpga-based accelerator design for deep convolutional neural networks[C]. Proceedings of the 2015 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM，2015.

[5] JOUPPI N P，YOUNG C，PATIL N，et al.In-datacenter performance analysis of a tensor processing unit[C].2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture(ISCA).IEEE，2017.

[6] Chen Tianshi，Du Zidong，Sun Ninghui，et al.Diannao：a small-footprint high-throughput accelerator for ubiquitous machine-learning[C].ACM Sigplan Notices.ACM，2014：269-284.

[7] HU Y H，KUNG S Y.Systolic arrays.in：handbook of signal processing systems[M].Springer，Cham，2019：939-977.

[8] SANAULLAH A，HERBORDT M C.Unlocking performance-programmability by penetrating the Intel FPGA OpenCL Toolflow[C]. 2018 IEEE High Performance Extreme Computing Conference(HPEC).IEEE，2018.

[9] KUNG H T，LEISERSON C E.Systolic arrays(for VLSI)[C]. Sparse Matrix Proceedings 1978，Society for Industrial and Applied Mathematics，1979.

[10] 王阳，陶华敏，肖山竹，等.基于脉动阵列的矩阵乘法器硬件加速技术研究[J].微电子学与计算机，2015，32(11)：120-124.

(收稿日期：2019-10-12)

作者简介：

王春林(1995-)，男，硕士研究生，主要研究方向：FPGA、卷积神经网络。

谭克俊(1963-)，通信作者，男，教授，主要研究方向：FPGA、嵌入式系统。

(上接第 56 页)

[7] Chen Chunhong，Li Zheng.A low-power CMOS analog multiplier[J].IEEE Transactions on Circuits and Systems II-Express Briefs，2006，53(2)：100-104.

[8] 吴湘锋，李志军，张黎黎.高精度电流模式四象限乘法器的设计与应用[J].微电子学，2015，45(4)：488-491.

[9] CRUZ-ALEJO J，OLIVA-MORENO L N.Low voltage FGMOS four quadrants analog multiplier[J].Advanced Materials Research，2014，918(2014)：313-318.

[10] 陆晓俊，李富华.一种低压高线性 CMOS 模拟乘法器设计[J].现代电子技术，2011，34(2)：139-144.

[11] 李志军，曾以成.多功能 AB 类四象限模拟乘法器[J].电子学报，2011，39(11)：2697-2700.

[12] 王鹏，汪涛，丁坤，等.一种高增益三级运算放大器[J].微电子学，2018，48(5)：579-584.

(收稿日期：2019-08-26)

作者简介：

丁坤(1991-)，男，硕士研究生，主要研究方向：集成电路工程。

田睿智(1998-)，男，本科，主要研究方向：集成电路。

汪涛(1981-)，通信作者，男，博士，副教授，主要研究方向：集成电路。