

## 引文格式:

康磊, 任旭超, 陈宇骞, 等. 基于可编程逻辑门阵列软硬件协同设计的心律失常分类系统 [J]. 集成技术, 2023, 12(3): 82-93.

Kang L, Ren XC, Chen YQ, et al. An electrocardiogram arrhythmia classification system based on software hardware co-design with field programmable gate array [J]. Journal of Integration Technology, 2023, 12(3): 82-93.

# 基于可编程逻辑门阵列软硬件协同设计的心律失常分类系统

康磊<sup>1</sup> 任旭超<sup>1,2</sup> 陈宇骞<sup>2,3</sup> 梅海红<sup>1</sup> 颜延<sup>2\*</sup>

<sup>1</sup>(西安石油大学计算机学院 西安 710065)

<sup>2</sup>(中国科学院深圳先进技术研究院 深圳 518055)

<sup>3</sup>(五邑大学智能制造学部 江门 529000)

**摘要** 近年来, 心律失常分类成为生理信号分析中的研究热点。心律失常现象在临床上十分常见, 其出现时伴随心电信号中的心拍呈现具有反常形态和节律的波形。正确及时地检测、发现心律失常, 并准确地进行心血管疾病的预警, 在临床诊断初期具有重要意义。但人工判断异常心电图的远程系统实时性较低, 可能延误病人的最佳治疗时机。将心律失常分类算法应用在可穿戴设备等边缘侧智能终端, 一方面能够对心电信号进行实时分析处理, 另一方面也提高了设备的灵活性及安全性。可编程逻辑门阵列器件作为边缘计算的一种实现形式, 在生理信号处理中已经得到了广泛的应用, 虽然可编程逻辑门阵列可进行实时流水线操作, 但其基于 Verilog 或 VHDL 硬件描述语言, 具有开发周期长、成本高、难度大及调试困难等缺点。针对这一问题, 该文采用 Xilinx 公司新推出的高层次综合工具 Vivado HLS, 以实现基于 MIT-BIH 数据集的心律失常五分类算法, 并使用 Xilinx Zynq FPGA 作为硬件平台, 在心电信号测试集上进行测试。测试结果显示, 该系统的平均分类准确率可达 99.12%, 单个心拍分类平均耗时 3.185 ms, 与纯 PS 端的单 ARM 核相比, 该系统实现了 5.64 倍以上的加速性能。

**关键词** 心电图; 心律失常分类; Vivado HLS; Zynq 嵌入式系统

**中图分类号** TP 391.4 **文献标志码** A **doi**: 10.12146/j.issn.2095-3135.20221028001

收稿日期: 2022-10-28 修回日期: 2023-02-10

基金项目: 国家重点研发计划项目(2020YFC2007203)

作者简介: 康磊, 副教授, 研究方向为计算机体系结构、嵌入式系统; 任旭超, 硕士研究生, 研究方向为嵌入式系统、FPGA 开发; 陈宇骞, 硕士研究生, 研究方向为人工智能、生物信号处理; 梅海红, 硕士研究生, 研究方向为计算机体系结构; 颜延(通讯作者), 助理研究员, 研究方向为信号模式识别及拓扑数据分析, E-mail: yan.yan@siat.ac.cn.

# An Electrocardiogram Arrhythmia Classification System Based on Software Hardware Co-design with Field Programmable Gate Array

KANG Lei<sup>1</sup> REN Xuchao<sup>1,2</sup> CHEN Yuqian<sup>2,3</sup> MEI Haihong<sup>1</sup> YAN Yan<sup>2\*</sup>

<sup>1</sup>( College of Computer Science, Xi'an Shiyou University, Xi'an 710065, China )

<sup>2</sup>( Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China )

<sup>3</sup>( Faculty of Intelligent Manufacturing, Wuyi University, Jiangmen 529000, China )

\*Corresponding Author: yan.yan@siat.ac.cn

**Abstract** Arrhythmia classification is a hot topic in physiological signal analysis. Arrhythmias are very common in clinical practice, and they are accompanied by abnormal patterns and rhythms in the heartbeat of the electrocardiogram signal. Correct and timely detection of arrhythmias and accurate early warning of cardiovascular diseases are of particular importance in the early stage of clinical diagnosis. However, the lack of real time diagnosis of electrocardiogram may delay the best time for patient treatment. Implementing heart rate disorder classification algorithms at edge-side smart terminals such as wearable devices enable real-time analysis and processing of electrocardiogram signals. In addition, they improve the flexibility and safety of the devices as well. By far, the field programmable gate array devices have been widely used in physiological signal processing as a form of edge computing due to its capability of real-time pipeline operation. Whereas, the field programmable gate array implementation needs a long development cycle, has high cost and is difficult to debug. To address these problems, the new high-level synthesis tool Vivado HLS from Xilinx is used to implement the arrhythmia five classification algorithm based on the MIT-BIH dataset. By using a Xilinx Zynq field programmable gate array, an average classification accuracy of 99.12% on the electrocardiogram signal test set is achieved. Moreover, an average of 3.185 ms required to classify a single heartbeat is realized, which leads to a speedup of more than 5.64 times compared to a single ARM core on the pure PS side.

**Keywords** ECG; arrhythmia classification; Vivado HLS; Zynq embedded system

**Funding** This work is supported by National Key Research and Development Program of China (2020YFC2007203)

## 1 引言

小型嵌入式处理器的性能由于成本、功耗和尺寸而受到限制<sup>[1]</sup>, 但硬件加速会提升如嵌入式处理器等片上系统(system on chip, SoC)的性能, 因此, 目前的 SoC(如 Xilinx Zynqs)通常将 ARM 处理器与可编程逻辑门阵列(field programmable gate array, FPGA)结合在一个芯片上<sup>[2]</sup>。结合后的 SoC 具有 ARM 处理器与 FPGA

的优点<sup>[3]</sup>: 系统的某些部分(如操作系统)可以更好地在软件中执行, 与性能相关、并行度较高的任务也可更好地在硬件中完成。本文借助 Vivado 高层次综合等设计工具, 基于 C 和 C++ 代码进行硬件综合<sup>[4]</sup>, 尽管该方法可以快速实现硬件加速, 但为满足资源和性能要求, 须进行设计优化。

FPGA 作为一种半定制电路, 克服了专用集成电路领域中定制电路不可编程的缺点<sup>[5]</sup>, 其数

量庞大的逻辑单元,使得可编程器件灵活性更好、应用领域更广<sup>[6]</sup>。基于FPGA高效、低功耗等优点,国内外学者进行了大量研究,以提高心电信号处理算法的实时性和能效性。

2012年,Chou等<sup>[7]</sup>基于FPGA提出了一种可穿戴式无线生物医学健康监测系统,该系统具有集成度高、实时性好等优点。2016年,Ali等<sup>[8]</sup>提出了用于人体识别的互联健康系统,该系统包括对心电信号进行加解密,以及基于心电生物识别技术对个人进行识别。实验结果显示,在Xilinx Zynq ZC702开发板上,系统处理一个心拍样本仅需10.71 ms,能耗107 mW,仅使用所有片上可用资源的30%;Ma等<sup>[9]</sup>和Kuzume等<sup>[10]</sup>使用改进的小波变换算法,基于FPGA实现了对心电信号R波的实时检测。2017年,Wess等<sup>[11]</sup>提出一种基于神经网络的机器学习算法,以检测心电图信号中的异常情况,利用主成分分析算法将特征减少至12个维度,并使用多层感知器对6种心律失常进行分类,最终基于FPGA实现了该算法。2017年,Tsoutsouras等<sup>[12]</sup>利用Vivado高层次综合(high level synthesis, HLS)工具,提出基于支持向量机(support vector machine, SVM)算法的心律失常分析和检测系统。与纯软件端相比,经HLS工具优化的加速器在许多单核或双核异构平台上表现较好,达到了10~78倍的速度提升。2020年,朱培钰<sup>[13]</sup>使用基于时间序列时间相关性特征的图像化算法结合卷积神经网络的组合模型,搭建了基于FPGA的心律失常分类系统,其计算速度约为ARM计算速度的5.73倍。

鉴于FPGA在深度学习加速方面的可行性,本文以FPGA作为硬件平台,研究心律失常分类算法模型的FPGA实现。在保证低延迟的条件下,从资源利用方面进行优化,以适应给定的FPGA。此外,选择最符合条件的协议,以优化硬件和软件组件之间的通信。实验结果表明,与

纯ARM软件相比,本文系统的性能提高了5.64倍以上,并在高效利用有限硬件资源的同时,满足了实时约束。此外,本文还将硬件IP核集成到一个复杂的嵌入式系统中。

## 2 材料和方法

本文的研究工作以麻省理工学院生理信号数据库(MIT-BIH)中的心律失常数据集的单导联心电信号作为实验数据,该数据集包括48组记录文件<sup>[14]</sup>。为使结果更有说服力,本文采用10折交叉验证法验证分类效果,并选取最常见的正常心跳、房性早搏、室性早搏心跳、左束支传导阻滞、右束支传导阻滞进行五分类研究。

### 2.1 预处理

心电信号是十分微弱的信号,容易受环境影响。在心电信号的采集、放大、传输过程中,心电信号会受到多种噪声干扰,由于心电信号的各特征波频率及幅度各不相同,受噪声影响的心电信号的特征波信息会被噪声掩盖,影响心电信号分析和诊断的准确性和可靠性。通常情况下,影响心电信号的噪声主要包括基线漂移噪声、工频干扰和肌电干扰<sup>[15]</sup>。因此,在进行心律失常分类研究前,需要对原始心电信号进行相应的去噪处理。经实验对比,本文选取基于db4小波基的四层小波去噪算法,对心电信号进行降噪处理。

小波阈值去噪使用多分辨率分析的思想,并设置合适的阈值对信号中的噪声进行去除<sup>[16]</sup>。经小波分解后,信号的小波系数较大,噪声的小波系数较小,通过选取一个合适的阈值,若小波系数大于阈值,那么认为是由信号产生的,应予以保留,否则认为是由噪声产生的,将其置为零,从而达到去噪的目的。

小波阈值去噪算法过程为:首先,对含有噪声的信号进行小波分解变换,得到原始小波系数,离散序列的Mallat算法分解公式如公式(1)~(2)

所示。

$$A_{j+1}(n) = H(n) \times A_j(n) = \sum_k H(k) A_j(2n-k) \quad (1)$$

$$D_{j+1}(n) = G(n) \times A_j(n) = \sum_k G(k) A_j(2n-k) \quad (2)$$

其中,  $H(n)$  为 db4 小波基对应的低通滤波器的响应系数序列;  $G(n)$  为 db4 小波基对应的高通滤波器的响应系数序列。由算法的分解原理可知, 分解后的序列由该层原序列  $A_j$  与滤波器序列的卷积, 再进行隔点抽取而来。

然后根据原始小波系数确定阈值, 完成小波系数的阈值处理, 保留大于阈值的小波系数, 去除小于阈值的小波系数, 本文采取基于 VisuShrink 阈值的处理方法, 如公式(3)所示。

$$\lambda = \frac{MAD}{0.6745} \sqrt{2 \ln N} \quad (3)$$

其中,  $N$  为信号长度;  $\frac{MAD}{0.6745}$  为信号的标准差。

对保留的小波系数进行小波重构, 得到小波阈值去噪后的信号, 将去噪前后的心拍进行对比, 如图 1 所示。由图 1 可知, 去噪后的波形更加平滑, 且基本保留了原始心电信号的波形特征, 本文采用的去噪方法基本满足实际滤波需求。

## 2.2 卷积神经网络模型

卷积神经网络作为一种典型的深度学习神经网络结构, 由自然视觉认知机制启发而来, 在视

频处理<sup>[17]</sup>、人脸识别<sup>[18-20]</sup>、语音识别<sup>[21]</sup>和自然语言处理<sup>[22]</sup>等方面均有突破, 本文采用的卷积神经网络由 5 个卷积层、2 个池化层以及 1 个全连接层组成, 具体模型结构和网络参数设置如图 2 所示。其中, 模型采用的激活函数为 ReLU 函数, 优化算法使用的是 Adam 优化算法, 学习率设置为 0.01, 迭代次数为 30 次, batch\_size 设置为 32。

经小波阈值去噪后, 在数据集注释文件的 R 波峰值点左取 100 个采样点, 右取 200 个采样点组成心拍数据分割, 经计算, MIT-BIH 数据库中关于此研究的心电信号共有 92 192 条心拍数据; 将所有心拍样本按照 10 折交叉验证策略放入卷积神经网络中进行训练, 实验结果显示, 该模型分类的平均准确率为 99.29%。最后对 Python 环境下训练得到的卷积神经网络模型进行处理, 将其结构中卷积层和池化层的权值参数(w)和偏置参数(b)保存为 12 个不同的文件。

在算法的性能评估方面, 除平均准确率外, 本文还选择了查全率(Recall)、查准率(Precision)、F1 值(F1-score)3 个指标, 评估结果如表 1 所示。由表 1 可知, 房性早搏查全率较低, 部分房性早搏异常心拍被检测为正常心拍, 而正常或室性早搏心跳、左束支传导阻滞、右束支传导阻滞的分类指标均较高, 分类效果良好。

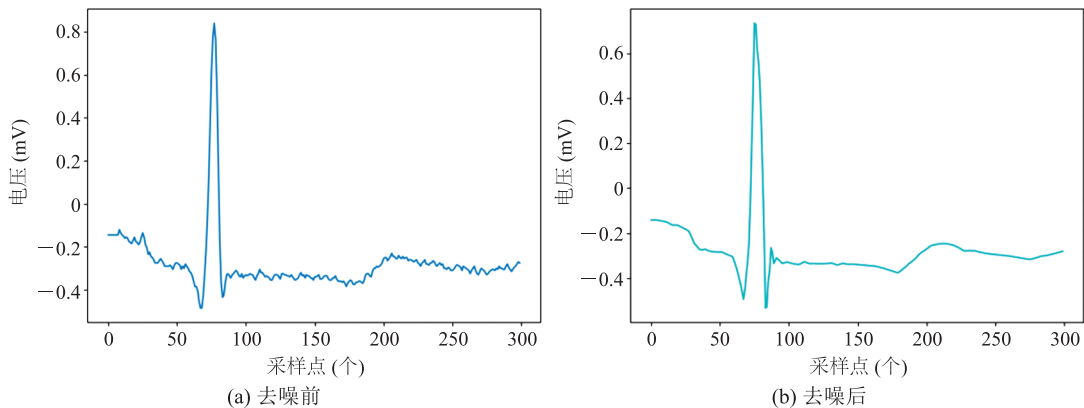


图 1 去噪前后效果对比图

Fig. 1 Comparison of effects before and after denoising

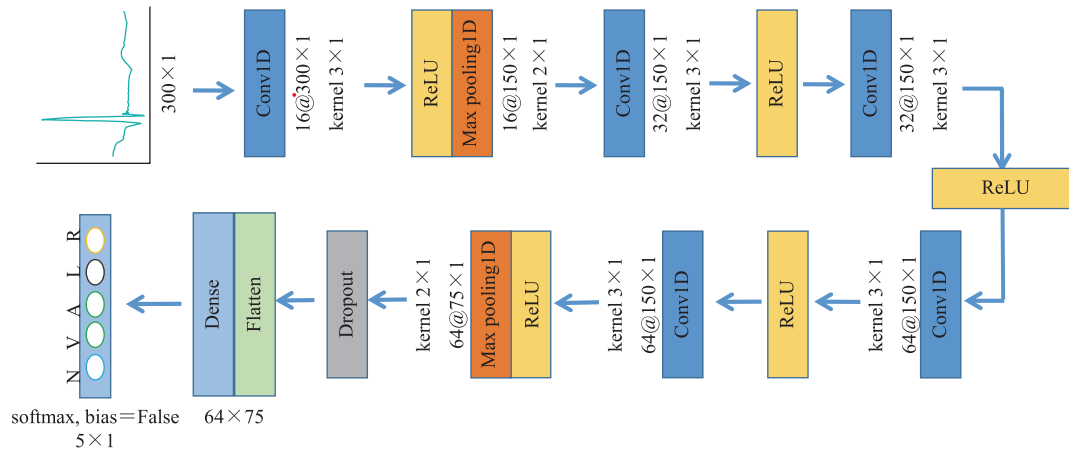


图2 卷积神经网络模型结构

Fig. 2 Convolutional neural network model structure

表1 心律失常分类模型性能评估

Table 1 Arrhythmia classification model performance

evaluation			
心律失常类型	Recall (%)	Precision (%)	F1-score (%)
正常心跳	99.36	99.86	99.61
房性早搏	85.84	88.83	91.90
室性早搏心跳	99.11	96.24	97.65
左束支传导阻滞	99.70	99.69	99.69
右束支传导阻滞	98.39	99.69	99.19

### 2.3 基于 Zynq FPGA 的嵌入式系统开发

本小节将对心律失常分类算法的嵌入式系统硬件实现进行详细介绍。本系统利用 Zynq-7000 平台 ARM+FPGA 异型架构的优势<sup>[23-24]</sup>，采用软硬件协同设计方法，以实现心电信号的五分类算法。可编程逻辑部分 (programmable logic, PL) 通过 Vivado HLS 2019.2 高层次综合工具，设计实现卷积神经网络卷积池化层及全连接层 IP 核；PS (processing system) 部分则通过 vitis 平台，采取软硬件协同设计方法在 FPGA 开发板上实现心律失常分类系统；将算法在 FPGA 上运行的结果以及各项性能指标进行测试和分析。

#### 2.3.1 系统设计方案的确定

本系统总体分为 PC 端和 FPGA 硬件平台 (Zynq-7000 核心板的 Zynq 芯片型号为 XC7Z020CLG400-2, PL 逻辑单元达 85K,

BRAM 存储资源为 4.9 Mbit) 两个部分，PC 端一方面负责与 FPGA 平台进行串口数据传输，另一方面负责将 PC 端训练好的卷积神经网络权值参数导入 FPGA 平台；硬件平台部分则通过建立 Vivado 工程，在块设计中添加设计好的用户 IP 核，配置处理器系统核的各个 IP 接口、时钟，实现各个模块的自动连接，通过综合实现后生成比特流文件，将其加载到 FPGA 平台，运行 vitis 进行软件开发，最终将计算结果通过串口返回 PC 端进行显示。在此过程中，FPGA 平台的 PS 部分主要负责控制驱动各个接口，PL 部分则负责卷积神经网络模型的运算加速。设计方案的总体框图如图 3 所示。

该系统卷积神经网络中大量计算过程由 PL 部分加速实现；PS 部分主要起到控制器的作用，负责从 DDR 存储器内存取数据，传到 FPGA 中进行处理。PL 和 PS 两部分通过 AXI4 高速总线接口相连，实现了高速的数据传输。该方案既发挥了 FPGA 逻辑控制对大量数据进行高速处理的优势，又结合了 ARM 软件编程灵活的特点。

## 3 基于 HLS 的 IP 核设计以及优化

近年来，高层次综合工具被不断改进，基于

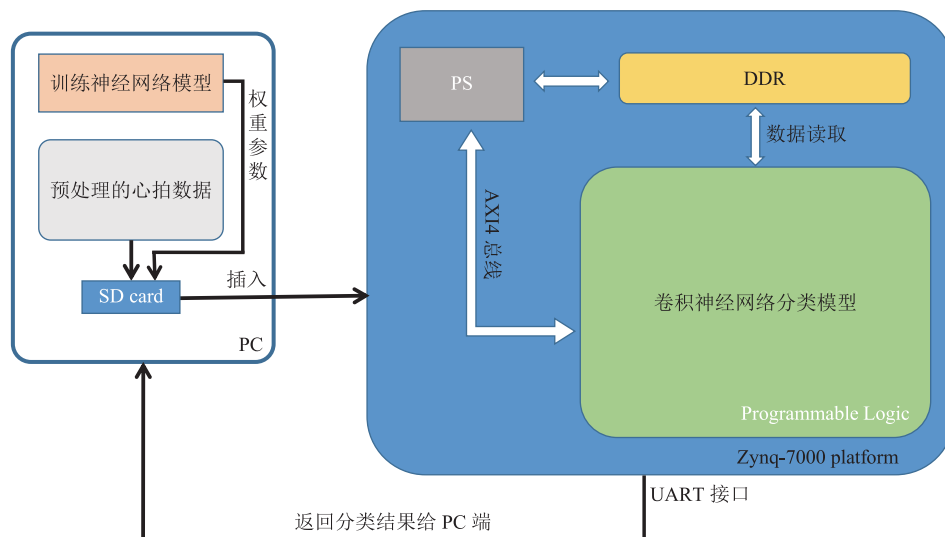


图 3 设计方案总体框图

Fig. 3 Overall block diagram of the design

该工具的硬件加速器<sup>[25]</sup>应用取得了可喜的成果。基于 HLS IP 核的设计框架如图 4 所示, Vivado HLS 的设计流程为编译、执行(模拟)和调试 C 算法; 选择性地使用用户优化指令, 将 C 算法映射到寄存器转换级(register transfer level, RTL)电路实现中; 综合生成全面的报告, 并对设计进行分析; 联合仿真验证 RTL 实现; 将 RTL 实施打包成多种 IP 格式。与传统的 FPGA 开发方式相比,

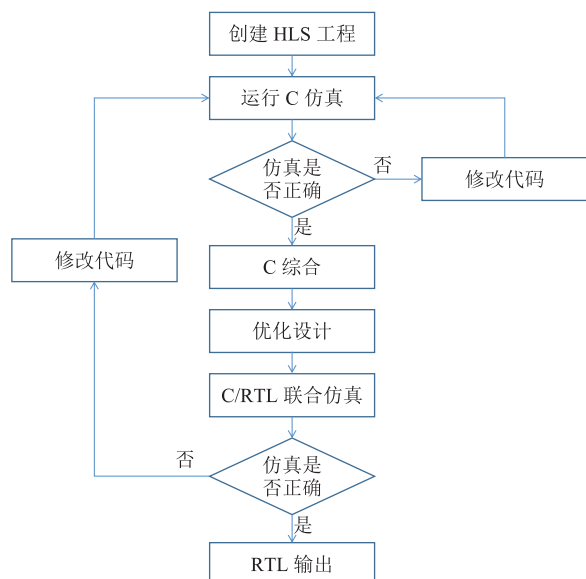


图 4 Vivado HLS 设计流程图

Fig. 4 Vivado HLS design flowchart

Vivado HLS 具有以下优点: (1) 由于 HLS 采用 C/C++ 高级语言进行开发, 可以灵活快速实现函数功能, 大大缩短了开发周期。(2) 测试激励除用于 C 仿真以验证算法功能正确性外, 还可用于 RTL 代码的功能验证, 相比于传统的仿真验证方法, Vivado HLS 功能验证的效率更高。(3) 采用高级语言开发有利于更新和维护, 一旦功能不满足要求, 可以灵活编程从而改变函数的功能, 直至满足设计需求。

### 3.1 卷积神经网络模型 IP 核设计

卷积神经网络模型如图 2 所示, HLS 工具则需要构建 3 个主要的 IP 核: 卷积 IP 核、池化 IP 核以及全连接 IP 核, 本文利用 HLS 工具实现了卷积神经网络中的卷积层(包含可选择的池化层结构)和全连接层两个 IP, 下面将对其进行详细阐述。

#### 3.1.1 Conv1D 卷积池化 IP 核

卷积层作为卷积神经网络(convolutional neural network, CNN)中最重要的一层, 通过卷积运算对图像的隐藏特征信息进行提取<sup>[26]</sup>。卷积神经网络具有稀疏交互和权值共享两大特点, 使其可以减少部分参数, 但在复杂的训练任务中,

中间结果数量庞大的问题仍难以解决，因此，通常在卷积层之后使用池化层对卷积层的输出特征进行降维。

本文的卷积神经网络结构中共有 5 个卷积层和 2 个最大池化层，实验通过设计一个 Conv1D 卷积池化 IP 核来完成此部分功能，卷积运算的实质是两个矩阵的点乘运算，图 5 为 Conv1D 中的函数层次关系及采取的 HLS 软件优化策略。设计时定义如下参数：输入卷积层的一维特征图尺寸 `size`；输入卷积层的特征图个数 `ch_in`；卷积层(包含可选择的池化层结构)特征图输入 `in`；卷积层输出的特征图个数 `ch_out`；定义数组指针 `Weight` 和 `bias` 用于保存权重参数，参数 `pool` 表示是否进行池化，使得该 IP 核可以应用于不同的中间层过程，图 5 右侧为 Conv1D IP 核中所采取的优化指令。

### 3.1.2 全连接层 IP 核

连接层可将卷积得到的各项“特征”进行整

合分类，为减少硬件开销，本实验卷积神经网络模型仅采用一层全连接，用于将输出分为指定个类。全连接层主要定义的参数为：输入中间层结果 `in`，全连接层参数矩阵 `weight`，偏置 `bias`。此外，定义数组 `out` 以保存全连接层输出结果，全连接(full connected, FC)层中进行全连接操作的顶层函数声明为 `void fc (data_t* in,data_t* weight,data_t* bias,data_t* out)`，主要是输入神经元和权重矩阵进行乘累加运算，输出分类结果，因此可将 PIPELINE 优化指令加入函数内循环中进行并行运算。

将上述两个 IP 核算法编写成 C 文件，经仿真综合后，在主要循环结构及数组中添加优化指令(详见第 3.2 小节)，再添加 `Interface` 指令设置 IP 核接口，从而完成用户 IP 核设计。由于本文的硬件实现只涉及神经网络的前向传播，Dropout 层仅在训练中使用，因此，训练中用于优化模型参数的层并未在 HLS 中集成，最后通

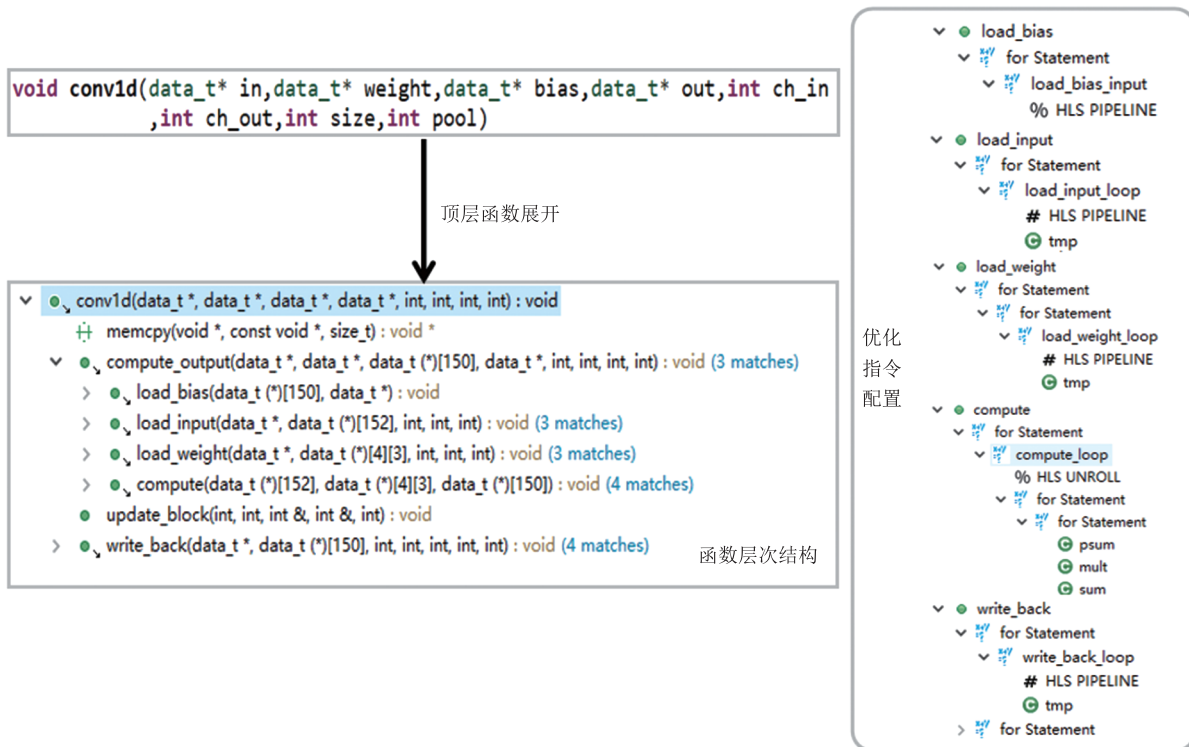


图 5 函数层次关系以及优化策略

Fig. 5 Function hierarchy and optimization strategies

过心律失常五分类算法的分析和验证结果, 评估该方法的有效性。

### 3.2 优化指令配置

在第 3.1 小节中, 主要通过修改 HLS 源代码结构创建有效的 RTL 架构, 本节将通过仔细调整可用的 HLS 指令, 进一步提高性能。在 Vivado HLS 综合报告中, 衡量算法时效性的评价指标为启动间隔、等待时间。启动间隔指两次数据块输入的时间间隔, 启动间隔越小, 表示模块的吞吐量越大; 等待时间指数据从输入到输出所经过的时间间隔, 代表了算法运行的时间, 其越小表示算法模块执行的时间越短。通过在 HLS 高层次综合工具中添加不同的编译优化指令, 产生不同的 RTL 结构, 合适的优化指令可以增加算法模块的并行性与吞吐量, 使得计算时延更低。

本文选择 PIPELINE、UNROLL、ARRAY\_PARTITION 和 ARRAY\_RESHAPE(表 2)指令应用于神经网络模型, 以增加卷积神经网络各层内循环中乘加运算的并行性, 通过循环展开, 指令对访问的数组进行分区, 从而分别增加数组的端口数或字宽, 并允许同时访问它们的元素。此外, 这些指令也可应用于外部循环及其迭代器引用的数组, 不仅增加了数据读/写端口的个数, 还提高了数据吞吐率。将优化指令添加到对应的 C 代码中, 通过综合报告来评估性能, 从而不断迭代出多项解决方案, 定制出满足需要的 RTL 输出。

上段介绍了两个工程 HLS 源代码中每个循

环或数组所添加的优化指令, 这些优化指令的参数配置如表 3 所示。其中, “Factor”列括号内为每条指令设置的不同值, 括号外为最终所选择的优化指令参数配置。表 4 则反映了分别使用这些优化指令配置前后 IP 核对应的延迟以及资源利用情况。

本文通过优化设计极大降低了各层的运算时延, 利用 HLS 综合工具, 将卷积神经网络 3 个主要计算单元的卷积、池化、全连接设计成用户 IP, 并为其配置了 AXI4 高速总线接口, 方便 Vivado 块设计中的连接。

## 4 结果与分析

本文嵌入式系统 block design 最终设计如图 6 所示, 模块设计中主要有 PS 端、设计的两个卷积神经网络中间层 IP 和连接自动生成的基础架构 IP。在资源利用方面, 卷积运算中较多的乘加运算会消耗主要的硬件资源, 占用较多的分别有 LUT、BRAM 和 FF。

本实验与已有研究的对比结果如表 5 所示, Tsoutsouras 等<sup>[12]</sup>基于 Zynq-7000 平台, 提出了心律失常分类系统, 为支持向量机加速器的设计提供了利用 Vivado-HLS 的优化解决方案, 并以一个心电分析和心律失常检测系统为例, 验证了采用优化的支持向量机加速器设计的 Zynq 可编程 SoC 的优异性能。

表 2 HLS 优化指令

Table 2 HLS optimization directives

指令	描述
PIPELINE	通过在循环或函数中并发执行操作来减少启动间隔
DATAFLOW	任务级流水线, 函数和循环同时执行, 用于最小化间隔
INLINE	内联函数, 删除所有函数层次结构, 用于启用跨函数边界的逻辑优化, 并通过减少函数调用开销来改善延迟间隔
UNROLL	展开 for 循环以创建多个独立操作, 而不是串行执行的操作
ARRAY_PARTITION	将大型阵列划分为多个较小的阵列或单独的寄存器, 以改善对数据的访问并消除块 RAM 瓶颈
ARRAY_MAP	将多个较小的阵列组合成一个大型阵列, 以帮助减少块 RAM 资源
ARRAY_RESHAPE	将数组从一个包含许多元素的数组重塑为一个字宽更大的数组, 有助于在不使用更多块 RAM 的情况下改进块 RAM 访问



表3 所应用的指令及其参数配置

Table 3 Applied directives and their parameter configuration

工程名	优化指令	Variable	Factor	Dimension
Conv1D	Pipeline	load_input_loop	on (on,off)	
		load_weight_loop	off (on,off)	
		load_bias_loop	on (on,off)	
		write_back_loop	on (on,off)	
	Unroll	compute_loop	on (on,off)	
	Partition	buff_in 1	4 (complete,2,4)	rows (dim=1)
		buff_in 2	4 (complete,2,4)	rows (dim=1)
		wt_buff 1	4 (complete,2,4)	columns (dim=2)
		wt_buff 1	4 (complete,2,4)	rows (dim=1)
		wt_buff 2	4 (complete,2,4)	columns (dim=2)
wt_buff 2		4 (complete,2,4)	rows (dim=1)	
FC	Pipeline	compute_loop	on (on,off)	

表4 延迟以及资源利用情况

Table 4 Latency and resource utilization

工程名	选择项	Latency/us		Utilization (个)			
		min	max	BRAM	DSP	FF	LUT
Conv1D	添加优化指令后	67	67	28	70	11 871	18 930
	添加优化指令前	1 418	1 562	21	13	4 388	5 667
FC	添加优化指令后	241	241	8	1	2 825	3 392
	添加优化指令前	721	721	8	1	2 611	3 208

Railis 等<sup>[27]</sup>重点分析了心律失常检测软件流水线在 Zynq SoC 上运行时的功耗、性能和能量分布,评估了从纯软件实施到面向软硬件协同的大量设计备选方案,并检验了设计方案的模型准确率(94%)、延迟以及资源利用情况。

Zairi 等<sup>[28]</sup>描述了一种基于人工神经网络心律失常识别系统的可编程逻辑器件实现方案,其基于 Nexys4 Artix7 评估套件设计了优化方法,平均识别准确率达 95%,可为患者提供一个连续的、可移动的心律失常监测系统。

本文提出的基于 Zynq-7000 平台心律失常分类系统,设计了硬件友好的卷积神经网络结构,通过高层次综合工具探索多种解决方案,以优化算法设计实现,最终采取软硬件协同设计的方法,在 FPGA 平台上取得了 99.12% 的分类准确

率;在能耗方面,本系统的动态功耗为 1.893 W;在运算速度方面,本系统完成一个心拍分类平均时间约为 3.185 ms,而在相同条件下,纯 PS 端 ARM 核(ARM Cortex-A9)则需要 17.948 ms,本系统的运算速度为 ARM 处理器的 5.64 倍以上。

综上所述,本文使用的 IP 核在兼顾效率和并行性的同时,减少了硬件资源的损耗,利用高性能运算模块 IP 核和软硬件协同设计的方法,最终在 FPGA 开发平台上实现了心律失常分类系统,并验证了其具有良好的分类性能以及更低的功耗。

## 5 结论

本文针对嵌入式系统提出了心律失常分类算

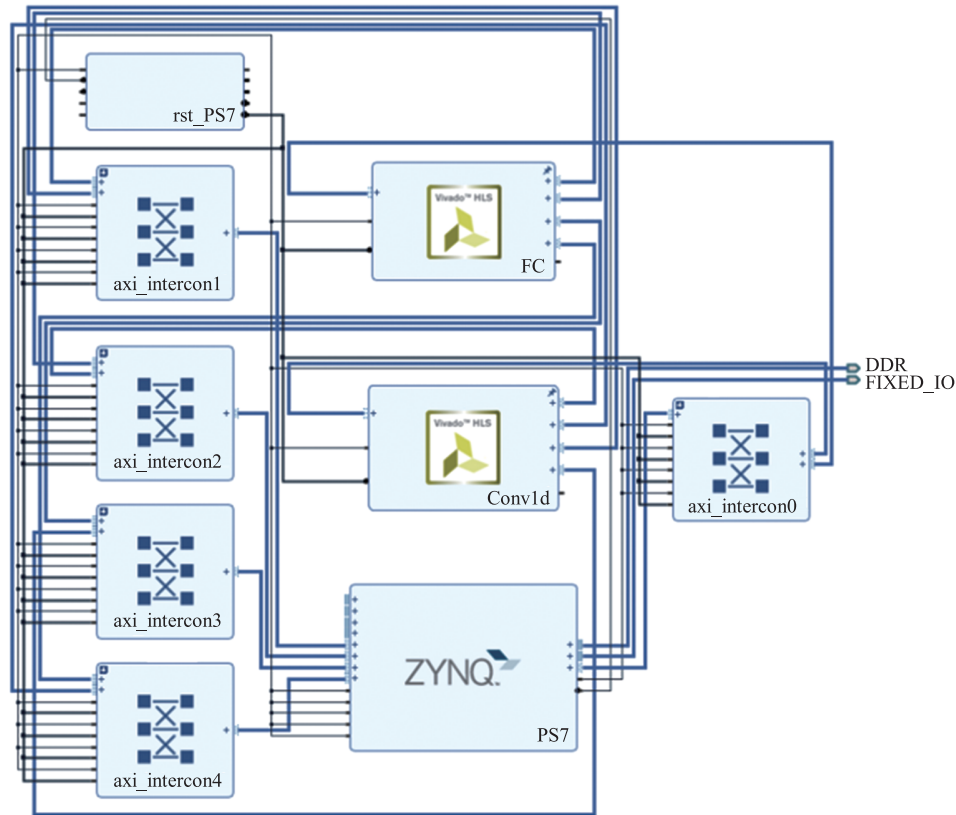


图 6 系统 block design 搭建

Fig. 6 System block design construction

表 5 心律失常分类的 FPGA 实现与已有研究的比较分析

Table 5 Comparative analysis of the arrhythmia classification's FPGA implementation with existing research

方法	分类准确率 (%)	资源利用率 (%)			
		BRAM	DSP	FF	LUT
SVM_based classifiers accelerators+Zynq-7000 <sup>[12]</sup>	—	27	50	8	29
DWT and SVM+Zynq-7000 <sup>[27]</sup>	94.00	62	51	8	32
DWT and MLP and SVM+Artix-7 FPGA board <sup>[28]</sup>	95.00	16	89	2	11
My CNN+Zynq-7000 (proposed work)	99.12	10	37	14	26

法, 进行软硬件协同设计开发, 优化设计的探索更侧重于性能端, 通过 Vivado HLS 对算法进行硬件实现, 采取不同的优化策略进行比较, 以提升算法性能。此外, 还根据资源利用率及延迟, 对实验结果进行逐步优化, 以适应给定的 Xilinx Zynq-7000 器件, 本实验最终将硬件加速器集成到一个小型嵌入式系统中, 实现了硬件的加速。未来, 拟采用 verilog HDL 硬件描述语言实现分类算法及其控制逻辑, 以期通过更少的冗余和更低的

资源利用率获得更好的性能, 并增加真实的心电异常数据, 验证该系统性能, 同时也可置入前端采集模块, 采集更多种类的生理信号进行分析, 实现一个多传感器融合的便携式健康监测站。

## 参考文献

- [1] Joseph JM, Mey M, Ehlers K, et al. Design space exploration for a hardware-accelerated embedded real-time pose estimation using Vivado HLS [C] //

- Proceedings of the 2017 International Conference on ReConfigurable Computing and FPGAs, 2017: 1-8.
- [2] Crockett LH, Elliot RA, Enderwitz MA, et al. The Zynq book: embedded processing with the ARM Cortex-A9 on the Xilinx Zynq-7000 all programmable SoC [M]. Glasgow: Strathclyde Academic Media, 2014.
- [3] Karataş F, Koyuncu I, Tuna M, et al. Design and implementation of arrhythmic ECG signals for biomedical engineering applications on FPGA [J]. The European Physical Journal Special Topics, 2021, 231(5): 869-884.
- [4] Cornu A, Derrien S, Lavenier D. HLS tools for FPGA: faster development with better performance [C] // Proceedings of the International Symposium on Applied Reconfigurable Computing, 2011: 67-78.
- [5] Kowsalya T. Area and power efficient pipelined hybrid merged adders for customized deep learning framework for FPGA implementation [J]. Microprocessors and Microsystems, 2020, 72: 102906.
- [6] Pirog A, Bornat Y, Perrier R, et al. Multimed: an integrated, multi-application platform for the real-time recording and sub-millisecond processing of biosignals [J]. Sensors, 2018, 18(7): 2099.
- [7] Chou CC, Fang WC, Huang HC. A novel wireless biomedical monitoring system with dedicated FPGA-based ECG processor [C] // Proceedings of the 2012 IEEE 16th International Symposium on Consumer Electronics, 2012: 1-4.
- [8] Ali AAS, Zhai XJ, Amira A, et al. Heterogeneous implementation of ECG encryption and identification on the Zynq SoC [C] // Proceedings of the Field Programmable Custom Computing Machines, 2016: 139.
- [9] Ma YR, Li TQ, Ma YD, et al. Novel real-time FPGA-based R-wave detection using lifting wavelet [J]. Circuits, Systems, and Signal Processing, 2016, 35(1): 281-299.
- [10] Kuzume K, Nijijima K, Takano S. FPGA-based lifting wavelet processor for real-time signal detection [J]. Signal Processing, 2004, 84(10): 1931-1940.
- [11] Wess M, Manoj PDS, Jantsch A. Neural network based ECG anomaly detection on FPGA and trade-off analysis [C] // Proceedings of the 2017 IEEE International Symposium on Circuits and Systems, 2017: 1-4.
- [12] Tsoutsouras V, Koliogeorgi K, Xydis S, et al. An exploration framework for efficient high-level synthesis of support vector machines: case study on ECG arrhythmia detection for Xilinx Zynq SoC [J]. Journal of Signal Processing Systems, 2017, 88(2): 127-147.
- [13] 朱培钰. 基于 CNN 的心律失常分类系统的 FPGA 设计与实现 [D]. 长春: 吉林大学, 2020.
- Zhu PY. FPGA design and implementation of arrhythmia classification system based on CNN [D]. Changchun: Jilin University, 2020.
- [14] 宋喜国, 邓亲恺. MIT-BIH 心率失常数据库的识读及应用 [J]. 中国医学杂志, 2004, 21(4): 230-232.
- Song XG, Deng QK. On the format of MIT-BIH arrhythmia database [J]. Chinese Journal of Medical Physics, 2004, 21(4): 230-232.
- [15] 王笑梅, 王辉. 基于小波的 ECG 信号噪声消除 [J]. 上海师范大学学报(自然科学版), 2002, 31(2): 50-54.
- Wang XM, Wang H. Noise removal of ECG signal based on wavelet [J]. Journal of Shanghai Teachers University (Natural Sciences), 2002, 31(2): 50-54.
- [16] Li HQ, Yuan DY, Wang YX, et al. Arrhythmia classification based on multi-domain feature extraction for an ECG recognition system [J]. Sensors, 2016, 16(10): 1744.
- [17] Solovyev R, Kustov A, Telpukhov D, et al. Fixed-point convolutional neural network for real-time video processing in FPGA [C] // Proceedings of the 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, 2019: 1605-1611.
- [18] Ding CX, Tao DC. Trunk-branch ensemble convolutional neural networks for video-based face recognition [J]. IEEE Transactions on Pattern

- Analysis and Machine Intelligence, 2017, 40(4): 1002-1014.
- [19] Schroff F, Kalenichenko D, Philbin J. FaceNet: a unified embedding for face recognition and clustering [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015: 815-823.
- [20] Wen YD, Zhang KP, Li ZF, et al. A discriminative feature learning approach for deep face recognition [C] // Proceedings of the European Conference on Computer Vision, 2016: 499-515.
- [21] Abdel-Hamid O, Mohamed AR, Jiang H, et al. Convolutional neural networks for speech recognition [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2014, 22(10): 1533-1545.
- [22] Wang W, Gang JX. Application of convolutional neural network in natural language processing [C] // Proceedings of the 2018 International Conference on Information Systems and Computer Aided Education, 2018: 64-70.
- [23] 江培海, 黄启俊, 常胜, 等. 基于 SoC FPGA 的心电信号检测系统设计 [J]. 传感器与微系统, 2016, 35(2): 74-77.
- Jiang PH, Huang QJ, Chang S, et al. Design of ECG detecting system based on SoC FPGA [J]. Transducer and Microsystem Technologies, 2016, 35(2): 74-77.
- [24] Ma HQ, Wang P, Fan H, et al. Smart home monitoring system based on SoC [C] // Proceedings of the International Conference on Industrial IoT Technologies and Applications, 2017: 196-204.
- [25] Georgopoulos K, Chrysos G, Malakonakis P, et al. An evaluation of Vivado HLS for efficient system design [C] // Proceedings of the 2016 International Symposium ELMAR, 2016: 195-199.
- [26] Zhang XY, Zhou XY, Lin MX, et al. Shufflenet: an extremely efficient convolutional neural network for mobile devices [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 6848-6856.
- [27] Railis K, Tsoutsouras V, Xydis S, et al. Energy profile analysis of Zynq-7000 programmable SoC for embedded medical processing: study on ECG arrhythmia detection [C] // Proceedings of the 2016 26th International Workshop on Power and Timing Modeling, Optimization and Simulation, 2016: 275-282.
- [28] Zairi H, Kedir Talha M, Meddah K, et al. FPGA-based system for artificial neural network arrhythmia classification [J]. Neural Computing and Applications, 2020, 32(8): 4105-4120.