文章编号:1671-4598(2025)03-0205-08 DOI:10.16526/j.cnki.11-4762/tp.2025.03.026 中图分类号:TP368 文献标识码:A

基于 GD32 的低功耗语音唤醒模块设计与实现

杨晓平，马兴录，陈　明，李瑶祺
（青岛科技大学 信息科学技术学院，山东 青岛　266061）

摘要：唤醒模块可以为机器人等长期待机运行的设备提供快速唤醒、降低功耗等功能；低功耗及智能化是评价该类模块的重要性能指标；设计了一款基于 GD32 的低功耗语音唤醒模块，该模块采用微型机器学习技术进行语音识别，将部分机器学习运算转移到单片机上进行；在电脑端对音频文件进行预处理，并将处理后的音频通过短时傅里叶变换转化为频谱图，进行模型训练；将训练好的模型由 TensorFlow Lite 转化为单片机可使用的 C 语言数组，通过微控制器开发环境将数组部署到 GD32 芯片上；以婴儿哭声检测为例，对 2 900 条音频数据进行 16 000 次训练，经实验测试，模块识别准确率约在 70%左右，模型和数据集仍有进一步优化的空间；对模块的功耗进行了测试与优化，优化后的模块工作频率 96 MHz，功耗电流为 34.2 mA；该模块的设计与实现对微控制器实现机器学习具有一定的参考意义。

关键词：TinyML；频谱图；语音识别；GD32F470ZGT6；嵌入式 AI

Design and Implementation of Low-power Voice Wake-up Module Based on GD32

YANG Xiaoping，MA Xinglu，CHEN Ming，LI Yaoqi
（College of Information and Science and Technology，Qingdao University of Science and Technology，Qingdao　266061，China）

Abstract：Wake-up module has the functions of fast wake-up and power consumption reduction，which can provide long-term standby operation for devices such as robots；It is an important performance index for low power consumption and intelligence to evaluate this module；A low-power voice wake-up module based on GD32 is designed，which adopts tiny machine learning（TinyML）technology for voice recognition. Some machine learning operations are transferred to run on a microcontroller，preprocess audio files on a computer，and transfer the processed audio into a spectrogram by short-time Fourier transform（STFT）for model training；The trained model is transformed from TensorFlow Lite to a C language array that can be used by MCU，and the array is deployed on GD32 chip through microcontroller development environment；By taking the baby crying detection as an example，2 900 pieces of voice data are processed 16 000 times of training. Experimental results show that the recognition accuracy of this module is about 70%，with a further optimization of the model and data set；The power consumption of the module is tested and optimized，with a optimized frequency of 96 MHz and a power consumption current of 34.2 mA；It is of certain reference significance for the design and implementation of this module to achieve machine learning in microcontrollers.

Keywords：TinyML；spectrogram；speech recognition；GD32F470ZGT6；embedded AI

0　引言

近几年来，随着机器学习技术的不断发展，其在音频领域的应用也逐渐增多。传统的机器学习推断需要将用户信息上传至云端，利用高算力服务器统一处理，导致用户隐私在信息传输过程中存在着被泄露的风险[1]。除此之外，在收集数据、上传数据、处理数据、返回结果的过程中对网络存在着过度依赖，可能会造成较大的延迟，降低用户体验感。随着嵌入式系统及芯片技术的发展，为了解决这些问题，人们开始考虑在微控制器上

收稿日期:2024-01-12；　**修回日期**:2024-03-01。
基金项目:国家自然基金面上项目(22374086)；山东省教育厅本科教改项目重点项目(Z2023152)。
作者简介:杨晓平(1984-)，男，在职硕士，工程师。
通讯作者:马兴录(1970-)，男，硕士，副教授。
引用格式:杨晓平，马兴录，陈　明，等. 基于 GD32 的低功耗语音唤醒模块设计与实现[J]. 计算机测量与控制，2025，33(3)：205-212.

部署机器学习算法。

微型机器学习（TinyML，tiny machine learning）是一种轻量级的机器学习技术，专注于在低功耗的微控制器设备上开发和部署机器学习模型。TinyML 使机器学习可以在安全、低延迟、低功耗的设备上运行。随着芯片新架构的出现以及硬件算力的提高，TinyML 在低于 100 KB 算法方面，网络和模型获得重大突破。例如，谷歌的 TensorFlow Lite Micro、Edge Impulse 的部署框架的建立与完善[2]。此外，机器学习优化库、函数接口的不断研发为其发展打下了坚实的基础。TinyML 可以通过特定的转换器将训练好的模型部署到微控制器上执行边缘计算，摆脱了原先的依赖远程处理[3]。TinyML 的主要优势在于其精简高效的机器学习算法带来的低资源消耗，其关键技术在机器学习和 MCU 芯片开发的结合[4]。

智能语音识别技术在未来智能设备的发展中有着巨大的市场，能够吸引更多的商业巨头来加入[5]。目前，在微控制器上实现语音识别应用的案例还是比较少的，在国产化芯片上的实现就更少了。传统的语音识别模块是将微控制芯片和语音识别芯片分离开来[6]，用微控制器来对语音识别芯片进行操作。本文在 GD32 的开发平台上，使用单一主控芯片识别和控制，基于 TinyML 设计了一个在家庭生活场景中特定唤醒词的低功耗语音唤醒模块，以丰富现有家庭语音助手的功能。具体的实现流程：对有标签的音频数据进行预处理，经过深度学习训练样本，生成相关模型，并将模型转换后，部署在 GD32 微控制器上。微控制器读取数字麦克风的音频数据，处理完后送入模型中，判断婴儿哭声原因。同时，在保证识别效果的基础上，对整机的功耗进行优化。

该低功耗语音唤醒模块具有以下优点：首先，采用 TinyML 技术可以将部分机器学习运算转移到嵌入式设备上进行，减轻了对云端和网络的依赖，降低了延迟和隐私泄露的风险；其次，通过对整机的功耗进行优化，可以降低能源消耗；最后，该模块可以广泛应用于智能家居、婴幼儿监护等领域，具有一定的实用价值。

1 系统架构

本文的设计目的是在国产微控制器 GD32 上部署机器学习模型，实现语音识别。模块的总体设计包含三部分：模型的训练生成，模型在微控制器的部署实现，模块的低功耗优化。

模型的训练生成。下载有标签的婴儿啼哭数据集，使用 Python 脚本对音频数据进行预处理，将音频数据进行分割、去除掉静音、噪音等，把样本数据分成训练集、验证集、测试集 3 个部分。在电脑端使用 Anaconda 建立虚拟环境，使用 Jupyter Notebook 对训练集的样本进行短时快速傅里叶变换，提取频谱图。将生成的频谱图输入到机器学习模型中，通过卷积神经网络（CNN，convolutional neural network）对模型进行训练、验证、测试，获得本模块的机器学习模型。TensorFlow Lite 是应用于资源受限的边缘设备上的轻量级深度学习优化框架[7]。它通过量化技术将生成的模型转换成一个保持原模型性能但规模降低的文件，降低在微控制器端的内存资源占用。由于微控制器没有本地文件系统的支持，TensorFlow Lite 模型转换后的 C 源文件包含一个 char 类型的 C 数组，将其部署到微控制器端的应用程序中。

模型在微控制器的部署实现。电脑端搭建基于 Keil 的开发环境，在微控制器上移植 TensorFlow Lite Micro 代码，并将模型 C 数组加载到工程中。GD32 通过集成电路内置音频总线（I^2S，integrated interchip sound），读取数字麦克风 INMP441 的音频数据，并对其进行短时傅立叶变换及窗口滑动处理，然后将处理完的数据送入模型中，识别出婴儿哭声，对哭声原因进行判断。整个系统的框图如图 1 所示。

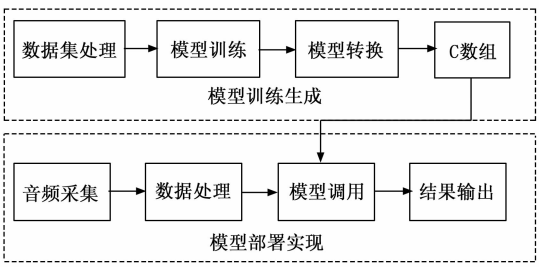

图 1 系统框图

模块的低功耗优化。通常对低功耗的设计主要是考虑硬件方面，而现在软件低功耗设计的作用也越来越重要[8]。本模块从芯片的硬件外设，工作的方式等方面，在保证识别效果的基础上，对模块的功耗进行合理优化。

2 系统硬件设计

唤醒模块主要供电单元、控制单元、通讯接口、单片机和麦克风等组成，如图 2 所示。

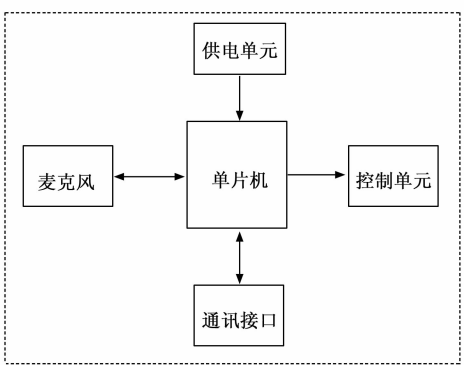

图 2 唤醒模块硬件组成图

供电单元配有两种接口，一种是电池接口，用于在移动场合下的电池供电；一种是外接直流电源接口，用于兼容外接直流电源的场景。供电单元里设计有低压差线性稳压器，可将电池电压或外接直流电源稳定在 3.3 V，供整个模块使用。

控制单元是模块用来控制大功率外设的接口。该单元设计为低压场效应管控制，可用单片机的 IO 口直接控制其开启和关闭，降低整个模块的功耗。大功率外设可根据实际情况进行配置，本模块测试外接 LED 指示灯，用于提示模块的识别状态。

通讯接口设计为 UART 串口。单片机的 TX 和 RX 引脚分别串入 100 欧姆的保护电阻，并通过 4.7 K 电阻上拉到电源，连接到接线插座。通讯接口为模块与大功率外设的补充接口，用于软件层的交互。

单片机采用兆易创新的 GD32F470ZGT6。该芯片为 Arm Cortex-M4 内核，处理器主频高达 240 MHz，Flash 1 MB，SRAM 512 KB。作为国产品牌的高端芯片，可支持算法复杂度较高的嵌入式应用，具备更快速的实时处理能力，并拥有业界领先的大容量存储优势。GD32F4 系列具有丰富的外设资源特性，及友好的 SWD 调试接口。该芯片所具有的存储及内存资源可满足该模块的软件运行。

麦克风采用 INMP441。该型号麦克风是一款高性能，低功耗，带数字输出和底部端口的全向 MEMS 麦克风。声音发散是全方位的，声音在空气中是向四周发散的，传播的距离多远，取决于很多因素。全向式麦克风对于来自不同角度的声音，其灵敏度是相同的[9]。一般拾音半径很大，比较适合本模块的应用场景。INMP441 由一个 MEMS 传感器，模数转换器，信号组成调节器，电源管理，抗混叠滤波器和行业标准的 24 位 I2S 接口，电流消耗低至 1.4 mA，具有高信噪比、高灵敏度的特点。I2S 是飞利浦在 1986 年定义的数字音频传输标准，用于数字音频数据在系统内部器件之间传输，例如编解码器 CODEC、数字输入/输出接口、数字滤波器等。它允许 INMP441 直接连接到数字处理器，无需使用用于系统中的音频编解码器。I2S 总线主要有 3 个信号线：

1）串行时钟 SCK，也叫位时钟 BCLK，对应数字音频的每一位数据，SCK 都有 1 个脉冲。

2）字段选择信号 WS，也叫 LRCLK，用于左右声道的数据切换。

3）串行数据 SD，就是用二进制补码表示的音频数据。I^2S 串行数据在传输的时候，由高位到低位依次进行传输。

模块总体使用的外围器件较少，只需简单的电路拼接就可以组建系统。单片机的 PB12、PB13、PB15 分别与麦克风的 WS、SCK、SD 引脚相连，并使用电阻上拉至电源，配置硬件 I2S，单片机为 master，麦克风为 slave，对麦克风进行读写操作。

3　短时傅里叶变换

现实生活中的声音是通过一定介质传播的连续的波，它由周期和振幅两个重要指标描述。语音信号本质上是一种携带特定信息的模拟信号，这对声音保存和长距离传输造成很大的困难，一般的做法是把模拟量转成对应的数字量保存。要尽可能完整的提取这些信息，需要运用数字信号处理技术对其进行处理。傅里叶变换是分析线性系统和平稳信号的有力手段，但语音信号是不平稳的，所以在实际应用中，经常用到的是短时傅里叶变换（STFT，short-time fourier transform）[10]。短时傅里叶变换其实就是在傅里叶变换的基础上，采用时频局域化的窗函数，并假设非平稳信号在分析窗的短时间隔内是平稳的，用窗口函数进行滑动计算。

将窗函数用 $g(u)$ 表示，信号源用 $h(u)$ 表示，那么 STFT 的计算公式为：

$$SWFT(t,f)=\int_{-\infty}^{\infty}[h(u)g(u-t)]e^{-j2\pi fu}\,\mathrm{d}u \quad (1)$$

在数字信号处理过程中，傅里叶变换只能作用于有限长度的时域数据变换，因此，需要对时域信号进行信号截断处理，以满足需求。当截断的时间长度并不是周期的整数倍，截取的信号将存在泄漏风险。为了将这个泄漏降低到最小程度，可以采用加权函数，即窗函数。

矩形窗的优点是主瓣比较集中，频率分辨率最高，不足之处是旁瓣也较高[11]。为了改善频率泄露的情况，加非矩形窗，一般都是加汉明窗，因为汉明窗的幅频特性是旁瓣衰减较大。窗函数其实很简单。可以使用一个公式来表示：

$$w(n)=a_0-\overbrace{(1-a_0)}^{a_1}\cdot\cos\left(\frac{2\pi n}{N-1}\right),0\leqslant n\leqslant N-1 \quad (2)$$

$a_0=0.538\ 36$，称作 Hamming 窗；

$a_0=0.5$，则称作 Hann 窗。

Hann 窗是升余弦窗，是应用较广的一种函数窗[12]。Hann 窗可以看作是 3 个矩形时间窗的频谱之和，或 3 个 sinc（t）型函数之和，括号中的两项相对于第一个谱窗向左、右各移动了 π/T，使旁瓣互相抵消，消去高频干扰和漏能。

4　语音识别模型训练

语音识别分为在线语音识别和离线语音识别。在线语音识别为常见的设计思路，即将语音数据上传至云端，使用云端引擎，有很强的算力，识别率和鲁棒性更高，缺点是必须依赖于网络，时延较高，隐私性差。离

线语音识别，使用本地的计算引擎，算力和模型大小有限，通常都基于特定使用场景定制，语料覆盖和泛化能力都是受到限制，鲁棒性也差一些，优点是不依赖于网络，通常响应速度很快，隐私性好。

目前较为流行的离线语音识别算法较多，如深度学习算法、卷积神经网络算法以及动循环神经网络算法等。本模块设计为离线语音识别，采用 Tensorflow Lite 的卷积神经网络算法[13]。

4.1 数据集的获取

本文目的是设计一个在家庭生活场景中特定唤醒词的低功耗语音唤醒模块。家庭生活场景中的语音场景比较丰富，如咳嗽声、脚步声、婴儿哭声等。以婴儿哭声作为语音唤醒词，是对低功耗语音唤醒的探索性研究。该模块在家庭生活场景中实现低功耗的、非实时连接网络的集婴儿哭声检测，能够辅助分析婴儿哭闹产生的原因，为监护人照顾婴儿提供参考。选用婴儿哭声作为唤醒词的另一个原因是，在百度深度学习平台中，有现成的带有标签的婴儿啼哭数据集，大大缩短了开发周期。虽然它并不是标准的数据集，数据量也很小，在判断网络模型优劣方面并不具有权威性，所以本文将使用婴儿哭声测试集作为评估模型的一个参考。

婴儿在各个不同生理状态下哭声所呈现的声音频谱图是不同的，具有组间差异大、组内差异小的特点。婴儿当前生理状态与当前哭声具有一定相关性，不同生理状态下婴儿哭声有所差异。但在实际情况下，婴儿哭声可能是两种或两种以上情绪的混合，识别特征不是特别明显。本文暂且不讨论这个问题，可认为在婴儿不同的生理状态下，只对应一种情绪，对婴儿哭声数据进行分类划分提取特征值具有可行性。本文在微型机器学习模型构建过程中，将按照该分类方法进行探究。

数据集的音频格式为 wav，数据时间长度 5～30 s 不等。这个数据集每个文件都比较大，不能直接使用，需要对其进行预处理。为了更直观地阐述婴儿不同生理状态所对应的婴儿哭声频谱，使用 Adobe Audition 软件打开各标签下的音频文件，查看对应的频谱图。标签名称和频谱图的对应关系如表 1 所示，标签对应的频谱图如图 3 所示。

表 1 数据集频谱图对应表

标签名称	含义	样本数	频谱图
Awake	醒来	160	a
Diaper	换尿布	134	b
Hug	想被抱	160	c
Hungry	饥饿	160	d
Sleepy	困倦	144	e
Uncomfortable	不舒服	160	f

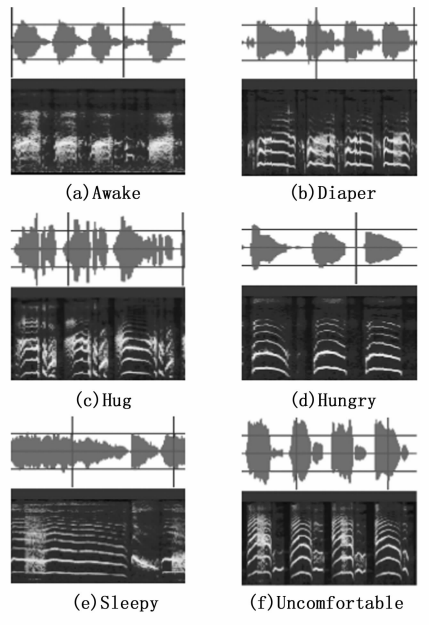

图 3 数据集对应频谱图

4.2 数据集的预处理

由于数据集是时长不等的音频文件，还包含许多异常值和噪声数据，且婴儿哭声的周期和规律性并不总是明显，在一种情感状态的哭声中可能会夹杂其他声音。这些含有不准确特征值的数据会大大降低模型训练的准确率和效率。因此，在构建模型前需对数据进行分析和预处理，为模型训练提供可靠的数据源。为了提高训练的准确率和效率，本文采用以下措施对数据进行处理，首先去除无声音、时长为零秒或接近零秒的数据。其次，不断改变分割时长，来选择一个合适的时长作为哭声周期的平均值，对数据音频进行分割。

最后，可以采用提取周期内哭声音频频谱特征值的办法来生成数据样本。处理完后，每种情感状态下的样本集数量约为 2 900 条，大大增强了数据样本，也为识别准确率的提高打下了基础。

4.3 特征生成

本文所使用的数据集音频采样率为 16 KHz，即每秒要采样 16 000 次。要对这些数字信号的关键特征点进行提取，这个过程在机器学习领域中称为“特征生成”，其目标是将数据的输入格式转换成容易让机器学习模型理解的格式。从上文婴儿哭声频谱图上可以看出不同生理状态下的哭声其频谱图之间存在差异。相比原始波形图，模型更容易区分它们。另一方面，生成的频谱图样本数据会更小。频谱图是音频数据的概要，可减少神经网络必须完成的工作量。由于本文所提到的设计最终需要运行在微控制器上，要考虑到资源受限的环境。因此，选择频谱图作为模型的样本输入。

音频信号在宏观上和微观上显示出极大的差异，表

现为宏观上的不平稳性和微观上的平稳性[14]。微观上是指“短时间”范围内可以认为是稳态的，这个短时间一般指 10～30 ms。本文的处理方法是设定时间片为 30 ms，给定的时间片的音频数据乘以 Hann 窗口，降低窗口两端对采样所造成的影响。以当前的音频采样频率计算，30 ms 需要采样 480 次，每次的采样为 16 位数据的话，需要 960 字节来存储，这个样本量还是有些偏大，需要对这些样本进行降采样处理。这种降采样不是线性的，而是基于人类感知的梅尔频率倒谱系数（MFCC，mel frequency cepstrum coefficient）标度来进行的[15]。MFCC 其实是在语音处理领域中为了对语音进行预处理而开发出的一套特征提取算法，该方法将语音在符合人耳听觉的非线性梅尔频率下进行余弦变换，最后得到一条可以表示语音能量的短时功率谱[16]。特征生成示意图如图 4 所示。

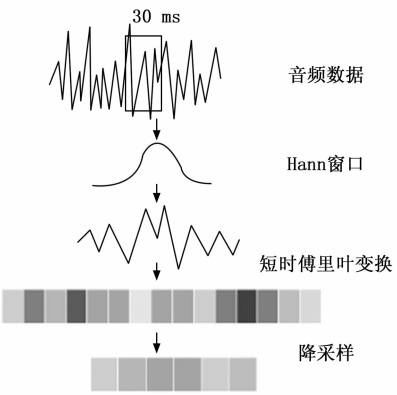

图 4　特征生成示意图

4.4　模型架构

卷积神经网络是对采用卷积核配合层叠网格结构构成的流水线，来进行特征提取的一类神经网络的统称。该类模型擅长在抽象图片或超二维信息类型的高维特征。一般应用于二维像素网格的图像，也可以用于多维向量输入[17]。本文采用 TensorFlow Lite 的卷积神经网络来处理音频频谱图数据，该模型包括一个输入层，一个卷积层，一个全连接层和一个 Softmax 层。TensorFlow Lite 是基于 TensorFlow 框架的轻量级版本，它专注于在小型设备上进行高效的机器学习预测[18]。

输入层用接收预处理的音频数据。

卷积层用于识别输入图像中的二维数据，每个卷积核都是一个矩形阵列，该阵列作为滑动窗口在图像中移动，输出数据表示输入数据和卷积核在每个点上的匹配程度。

全连接层的输入张量值都有一个权重，表示的是通过比较每个值之后，输入与当前权重的所存在的匹配程度。可以使用期望的结果作为输入，而实际输出其与实际输入的接近程度，模型中的每个类别都有自己的权重。

Softmax 层用来优化得分最高和其最接近的竞争者之间存在的差异，得分最高的即为模型的预测结果。

模型网络结构如图 5 所示。

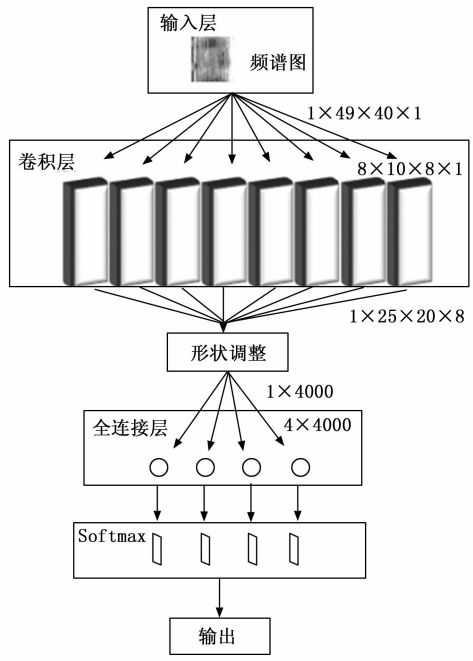

图 5　模型网络结构图

本设计在卷积层和全连接层输出的结果上增加了偏差调整函数，用来帮助调整其输出。此外，在每层之后增加了 ReLU 激励函数，用来确保没有输出小于零的情况发生。激励函数可以使得深度学习更加有效，可以让训练过程以更快的速度收敛[19]。

4.5　训练环境搭建及结果输出

在本地 Anaconda 下，建立虚拟环境，使用 Jupyter Notebook 设置学习率，训练步长及其他训练参数等。由于不是所有的 TensorFlow 模型都能在 TensorFlow Lite 中运行，因为解释器只支持部分 TensorFlow 运算符，为了实现更好的兼容性，本模型的训练使用 TensorFlow 1.15 版本。

训练结束后，生成图 6 和图 7 两张图。图 6 展示了模型的准确率，大概在 70％左右，表示该模型在当前正确识别婴儿哭声的能力，但能力比较弱，还需要进一步优化。图 7 展示了模型的损失值，量化了模型的预测值与正确值之间的差距。

模型训练结束后，训练好的模型不能直接被部署到单片机上，因为文件太大。一般情况下，模型中的权重和偏差被存储为 32 位浮点数，以便在训练过程中可以进行高精度计算。由于这个模型将部署在一个微控制器上，需要对模型文件进行量化。量化是一种减小模型尺

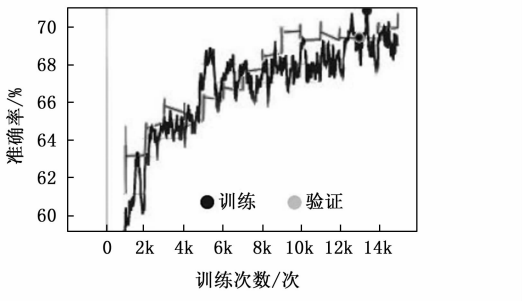

图 6 准确率

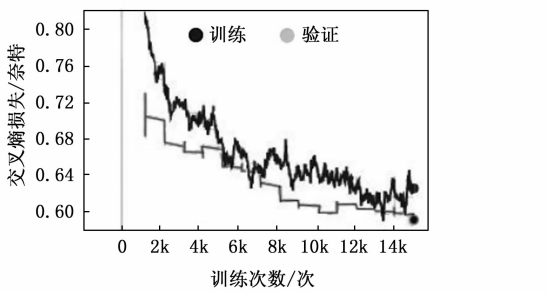

图 7 交叉熵损失

寸的技术，它允许降低这些权重和偏差的精度，使其可以存储为 8 位整数大小会减小为原来的 1/4，节省了内存。由于微控制器使用整数执行数学运算要比使用浮点数更容易，所以量化后模型的运行速度会更快。最后，将模型文件转换为 C 语言数组。

5 模型在微控制器的部署实现

目前嵌入式推理框架主要面临两大挑战：一个是嵌入式系统没有一个统一的模型推理框架，另一个是针对不同的芯片，很难用相同的评价指标以及优化。本文所要用到的 TensorFlow Lite Micro 框架就是为了解决这些问题，大大提高了模型部署的便捷性和灵活 性，能方便地进行剪裁和配置，使得模型可以很方便地部署到不同的硬件平台[20]。TensorFlow Lite Micro 是相对于 TensorFlow Lite 更加简洁的轻量化框架，专门针对边缘的 MCU，能够使其自主进行机器学习运算。

在微控制器开发环境 Keil 下，建立基于单片机 GD32F470ZGT6 的基本工程，配置 IO 口、串口通讯。定时器及 I²S 等外设。考虑到实际工程需要 C 语言来做硬件驱动，而 C++来做算法处理，项目编译器采用交叉编译环境 GCC。在开发环境下，设置编译器为外部 GCC。将 TensorFlow Lite Micro 核心处理代码 tensorflow 和 third _ party 两个文件下的文件添加到前面建立的基本工程中，并将训练的 C 语言模型数组也加入工程，调整工程中的 .c、.cc、.h 等文件包含关系，直至编译无错误。

微控制器程序在运行推断之前，首先要进行初始化，大致包含以下几个方面：

1）加载模型。将之前生成的 C 模型数组模型加载进来，返回一个模型指针。同时检查模型版本，以确保它的架构版本与正在使用的版本兼容，确保后续的运算正常。

2）实例化算子。TensorFlow Lite Micro 中有多种算子，这些运算会占用大量内存。给定的模型仅会用到这些运算中的一部分，该步骤仅加载所需的算子，以节省内存资源。

3）分配内存并实例化解释器。为了保证模型的正常运行，需要预先为输入、输出以及中间变量分配一定的内存。内存的大小需要通过 C 语言模型数组的大小，按一定的倍数来调整，并尽可能的小些。内存分配后，创建一个解释器，并传入之前创建的变量。

4）分配并检查张量。在创建解释器后，将前面定义的内存分配给张量，检查该张量，以确认其形状和类型是否符合预期。

5）初始化特征程序参数。唤醒模块的音频来自于麦克风，在运行推断时需获取固定周期的音频频谱图并提取特征，该步骤对特征程序的参数进行初始化。

微控制器程序进入推断后，通过 I²S 读取麦克风采集的音频数据。在模型训练时，训练集的音频分割成长度为 1 S 的音频文件，而微控制器端 1S 的音频文件会占用大量的内存。为了降低内存占用率，对音频数据格式转换，将 24 位数据的低 8 位舍弃，转换为 16 位数据。为了能快速的处理采集的语音数据，I2S 的时钟设置为 1 Mb/s。通过 I2S 传送过来的数据通过直接内存访问（DMA，direct memory access）存储到预先定义的数据缓冲区内。特征提供程序采用音频切片的形式，将原始音频数据转换为模型所需要的频谱图，切片时间设置为 30 ms。在处理语音数据时，为了使特征参数平滑，两个相邻切片之间保留 10 ms 的重叠区域。这些切片拼接成 1 S 的音频，生成频谱图与模型的特征数据进行匹配。语音识别的结果可以通过模块的串口打印到电脑上或通过控制单元控制 LED 进行指示。

对模块的识别性能进行测试，测试主要包括：婴儿哭声唤醒词准确度测试、系统响应时间测试、数据安全性测试、硬件满足性测试如表 2 所示。

表 2 性能测试表

测试名称	测试过程	预测结果	测试结果
准确度	播放标签为 awake，hug 的哭声音频	准确度约在 70%	符合预期
响应时间	播放标签为 awake，hug 的哭声音频、计算响应时长	时长在三秒以下	符合预期
数据安全性	家庭沉默场景下使用	不存在数据泄露	符合预期
硬件	计算模型下载至硬件	硬件满足要求	符合预期

从实际测试的测试结果来看，识别准确度并不太高，影响微控制器识别准确率原因可能有以下几个方面：

1）数据集的预处理。由于婴儿啼哭声具有较大随机性，现有数据集里的音频数据，可能掺杂婴儿其他情绪。在预处理时，音频的分割，也可能破坏原有的特征。

2）模型的生成。在模型的训练及量化过程中，由于参数的调整，模型的准确率受到了影响。

3）麦克风音频采集。全向麦克风拾音半径较大，会将更多环境噪音拾取进来，可能原因导致的音频识别率低的问题。

4）微控制器音频处理。由于微控制器的资源限制，在对音频数据进行数据截断，及进行窗口处理时损失部分真实特征。

6　功耗优化

本模块所采用的嵌入式低功耗设备开发板，其与桌面系统或者移动系统相比最大的优势是功耗非常小。相比服务器 CPU 的功率可能会达到几十瓦或者上百瓦，微控制器的运行功率为毫瓦级。

GD32F470ZGT6 微处理器集成电源管理单元（PMU，power management unit）。PMU 提供了 3 种省电模式，包括睡眠模式，深度睡眠模式和待机模式。这些模式能减少电源能耗，且使得应用程序可以在 CPU 运行时间要求、速度和功耗的相互冲突中获得最佳折衷。模块将通过以下方式，对功耗进行优化：

1）降低系统的工作时钟。依据微控制器的规格书，频率越高功耗越大。处理器功耗同时钟频率存在密切联系，多数的处理器有正常模式、休眠模式、空闲模式和关机模式，不同模式功耗有所不同[21]。采用降频的方式来降低功耗，但需考虑模块的运行效果；

2）关闭不使用的模块，如定时器，UART 等，不使用的 IO 引脚设置为输入，以降低整体的功耗；

3）睡眠模式与工作模式交替进行。嵌入式设备处理器都可以进入睡眠模式。在睡眠模式下，它将不会执行任何计算，这样会有很低的能耗，但不能检测到外部的音频。在这种模式下，可以启用定时器，在固定的时间唤醒，但需保证不能影响到检测效果。

以上几种方式，可以结合到一起使用，从而达到降低综合功耗的目的。但是实际中，如果存在运行延迟的现象，则需要逐步调整，在运行效果和低功耗之间平衡。

将万用表打到直流电流档位，串入模块的供电端。模块正常的工作频率为 240 MHz，测量出的消耗电流为 59.1 mA。调整模块的工作频率值，汇总功耗数据如表 3 所示。

表 3　各频率下的功耗及运行效果

序号	频率值/MHz	功耗/mA	运行效果
1	240	59.1	良好
2	120	40.2	良好
3	96	36.5	良好
4	72	32.5	较差

根据表 3 的数据，系统可选择工作在 96 MHz 的频率下。

关闭不用的外设，整个系统的功耗降低的不多，效果不明显，如表 4 所示。

表 4　优化不用的模块

项目	频率值/MHz	功耗/mA	运行效果
优化前	96	36.5	良好
优化后	96	36.3	良好

GD32 在低功耗下还能工作的定时器有 RTC 和独立看门狗。但 RTC 的最小单位是秒，不能满足系统的需求，所以采用独立看门狗。独立看门狗定时器（FWDGT）有独立时钟源（IRC32K），即使主时钟失效，FWDGT 依然能保持正常工作状态，适用于需要独立环境且对计时精度要求不高的场合。

模型在微控制器上通过滑动窗口，拼接 1 s 音频的频谱图，所以暂且设置单个休眠和工作的时间周期为 1 s。如图 8 所示，t_3 和 t_1 的间隔为 1 s。通过调节 t_1、t_2 的休眠时间的长度来优化功耗和运行效果。实验表明，休眠时间为 100 ms，工作时间为 900 ms 时，系统能正常工作，此时的功耗为 34.2 mA。

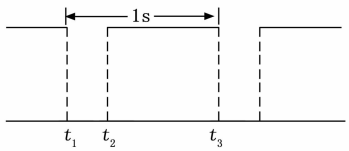

图 8　睡眠、工作交替模式

该模块经过优化后，工作频率为 96 MHz，休眠时间为 100 ms，工作时间为 900 ms，此时功耗最优。

7　结束语

本文设计了一款基于 GD32 的低功耗语音唤醒模块，硬件使用兆易创新的 GD32F470ZGT6 和 INMP441 麦克风，采用 TensorFlow Lite 框架，利用音频的频谱图实现语音识别功能，其识别结果判断婴儿哭声的原因。从目前模型的准确度和识别效果来看，准确度不高，只有 70％左右。数据集的处理及模型参数的优化的有待于进一步提高。模块本身的音频处理和特征识别也需要进一步改进。该模块的设计实现，对国产化芯片

部署机器学习，实现 TinyML 全流程，具有参考意义。

参考文献：

[1] 王 浩. 基于车联网的 ESP32 模组在线语音识别控制系统设计 [J]. 办公自动化，2022，27 (20)：22 - 26.

[2] 吴建邦，邱 天，张 昕，等. TinyML 的研究现状及展望 [J]. 单片机与嵌入式系统应用，2023，23 (2)：7 - 11.

[3] 刘旭瑶，曾天凯，戴嘉庆，等. 基于 Tiny ML 轻量化边缘侧处理的智能分类垃圾桶 [J]. 装备制造技术，2022 (6)：110 - 113.

[4] 张 增. TinyML 边缘计算在高校人工智能专业实验教学中的应用研究 [J]. 中国新通信，2023，25 (6)：74 - 76.

[5] 郝欧亚，吴 璇，刘荣凯. 智能语音识别技术的发展现状与应用前景 [J]. 电声技术，2020，44 (03)：24 - 26.

[6] 卫静婷，陈利伟，黎 斌，等. 基于 STM32 的语音控制和自动避障智能小车的设计 [J]. 电子测试，2019，(15)：24 - 25.

[7] 胡深奇，李云鹏. 基于 TensorFlow Lite 的智能辅助行进系统的设计实现 [J]. 技术与市场，2022，29 (11)：15 - 18.

[8] 张武斌. 硬件系统的低功耗设计分析 [J]. 集成电路应用，2023，40 (1)：14 - 17.

[9] 黄帮伟. 麦克风技术探讨——如何选择合适的麦克风 [J]. 智能建筑，2019，(10)：69 - 72.

[10] 陈 曦. 基于 MOOC 数据集的语音合成研究 [D]. 南京：东南大学，2022.

[11] 苏自清，李国欣. ESP32 上 TinyML 语音识别应用软硬件协同设计 [J]. 物联网技术，2023，13 (5)：47 - 51.

[12] 李 媛，王海云，王维庆，等. 基于 Hanning 和 Nuttall 的混合卷积窗谐波分析方法 [J]. 太阳能学报，2018，39 (12)：3363 - 3370.

[13] 潘 刚，伍世云，孙林平，等. 基于语音识别技术的智能小车控制系统研究 [J]. 电子设计工程，2019，27 (7)：118 - 123.

[14] 屈晓渊，崔 青. 基于梅尔频率倒谱系数的音频分类研究 [J]. 电子设计工程，2022 (9)：30.

[15] 陈 东，黄智鹏. 基于梅尔频率倒谱系数和支持向量机的汽车鸣喇叭声识别 [J]. 科学技术与工程，2021，21 (11)：4486 - 4491.

[16] 赵嘉文. 基于小样本的敏感词自动语音识别 [D]. 东南大学，2022.

[17] 刘 建. 基于深度学习的小样本声纹识别 [D]. 荆州：长江大学，2020.

[18] 陈治成，孙浩杰，谭展鸿. 基于 Tensorflow Lite 的中草药嵌入式识别器 [J]. 电子制作，2023，31 (22)：66 - 68.

[19] WARDEN P，SITUNAYAKE D. 基于 TensorFlow Lite 在 Arduino 和超低功耗微控制器上部署机器学习 [M]. 魏 兰，卜 杰译. 北京：机械工业出版社，2020.

[20] 许 鹏，宋 岩. TFLite－micro 内存管理与分配策略的优化 [J]. 单片机与嵌入式系统应用，2022，22 (10)：11 - 15.

[21] 徐良伟. 嵌入式系统低功耗设计研究 [J]. 中国新通信，2021，23 (8)：67 - 68.

(上接第 204 页)

[21] LIN W A，LIAO H F，PENG C，et al. DuDoNet：Dual domain network for CT metal artifact reduction [C] // 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition：CVPR 2019，2019：10504 - 10513.

[22] LIAO H F，LIN W A，ZHOU S K，et al. ADN：Artifact disentanglement network for unsupervised metal artifact reduction [J]. IEEE Transactions on Medical Imaging，2020，39 (3)：634 - 643.

[23] LEE J，GU J，YE J C. Unsupervised CT metal artifact learning using attention-guided β-CycleGAN [J]. IEEE Transactions on Medical Imaging，2021，40 (12)：3932 - 3944.

[24] WANG T，YU H，WANG Z W，et al. SemiMAR：semi-supervised learning for CT metal artifact reduction [J]. IEEE Journal of Biomedical and Health Informatics，2023，27 (11)：5369 - 5380.

[25] VASWANI A，SHAZEERN，PARMARN，et al. Attention is all you need [J]. ArXiv，ArXiv：1706.03762.

[26] DENG S，WEI MQ，WANG J，et al. Detail-recovery image deraining via context aggregation networks [C] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition：CVPR 2020，2020：14548 - 14557.

[27] 庄天戈. CT 原理与算法 [M]. 上海：上海交通大学出版社，1992.

[28] WANG H，LIi Y X，ZHANG H M，et al. InDuDoNet：an interpretable dual domain network for CT metal artifact reduction [C] //Medical Image Computing and Computer Assisted Intervention-MICCAI 2021：24th International Conference，2021：107 - 118.

[29] WANG H，LI Y X，HE N J，et al. DICDNet：deep interpretable convolutional dictionary network for metal artifact reduction in CT images [J]. IEEE Transactions on Medical Imaging，2022，41 (4)：869 - 880.

[30] WANG H，LI Y X，MENG D Y，et al. Adaptive convolutional dictionary network for CT metal artifact reduction [J]. ArXiv，ArXiv：2205.07471.