



Reese Grimsley and Colin Callaghan

摘要

TI 的 AM6xA 处理器专为需要密集图像分析的视觉应用而设计。商品识别、条形码扫描和解码以及防盗检测等零售结账和扫描仪应用均受益于成像和视觉分析，可提高准确性、速度以及对新环境的通用性。本应用手册分析了一个零售结账演示应用，该应用使用原始摄像头传感器，并在 AM62A 的异构架构上运行具有用于物体检测的深度学习功能且基于 gstreamer 的应用。AM62A 处理器上的核心负载用于选择成本优化型片上系统 (SoC)，应用的功耗显示 SoC 的有功功率低于 2 瓦。

内容

1 引言	2
2 AM62A 处理器	2
3 深度学习基准测试	3
4 零售结账扫描仪应用	3
5 核心负载	4
6 器件选择	6
7 功耗	6
8 总结	7
9 参考文献	7

插图清单

图 2-1. AM62A 简化方框图	2
图 2-2. AM6xA 视觉应用数据流	3
图 4-1. 具有分辨率和像素格式的零售结账应用流程。(30fps 是可实现的最大值；在该应用中，由于应用程序代码方面的瓶颈，FPS 接近 15)	4
图 5-1. 运行零售结账演示时的 tiperfoverlay Gstreamer 插件	5
图 5-2. 运行零售结账演示时的 perf_stats 命令行内核负载	5
图 5-3. 使用“htop”Linux 实用程序时的零售结账演示负载	6

表格清单

表 3-1. AM62A 的基准测试	3
表 6-1. 基于深度学习性能和 CPU 内核数的 AM62A SoC 型号	6

商标

所有商标均为其各自所有者的财产。

1 引言

由于图像中包含丰富的信息内容，并且边缘的数据处理能力不断提高，智能摄像头在零售领域越来越受欢迎。诸如结账扫描仪、条形码图像、资产和人员跟踪、防盗检测等用例正在帮助实现客户体验自动化，并为客户带来更加简单便捷的经验。

尽管视觉对于人类来说是一个直观的概念，但计算机视觉却具有挑战性。图像包含大量信息，视觉模式能以多种形状、大小和环境形式出现。传统的计算机视觉使用滤波器、变换和专用算法，效果相当不错；但是，这些算法通常很难完全加速，并且可能需要针对环境和任务的具体细节进行调优。机器学习和深度学习图像处理方法更为通用，并且通常具有更高的精度。虽然这些基于学习的算法 (AI) 通常计算更加复杂，但由于非常依赖矩阵数学，因此也更易于加速。

利用深度学习和云端 AI 处理图像非常容易，但成本比本地处理明显要高，而且是经常性成本。由于网络延迟，结账等反应式应用对客户而言会很慢，容易引起客户不满。安全应用可能会引发隐私问题。此外，随着解决方案的扩展，相关的云成本也会同样地扩展。在捕获视频的设备上本地处理图像可以解决这些问题，但需要一个符合成本、功耗和性能要求的处理器。

2 AM62A 处理器

AM62A Edge AI 微处理器 (如图 2-1 所示) 专为对成本和功耗敏感但需要密集图像分析的单摄像头或双摄像头应用而设计。硬件加速可以帮助视觉应用提升图像质量、提高预处理速度并对深度神经网络等分析算法进行加速。

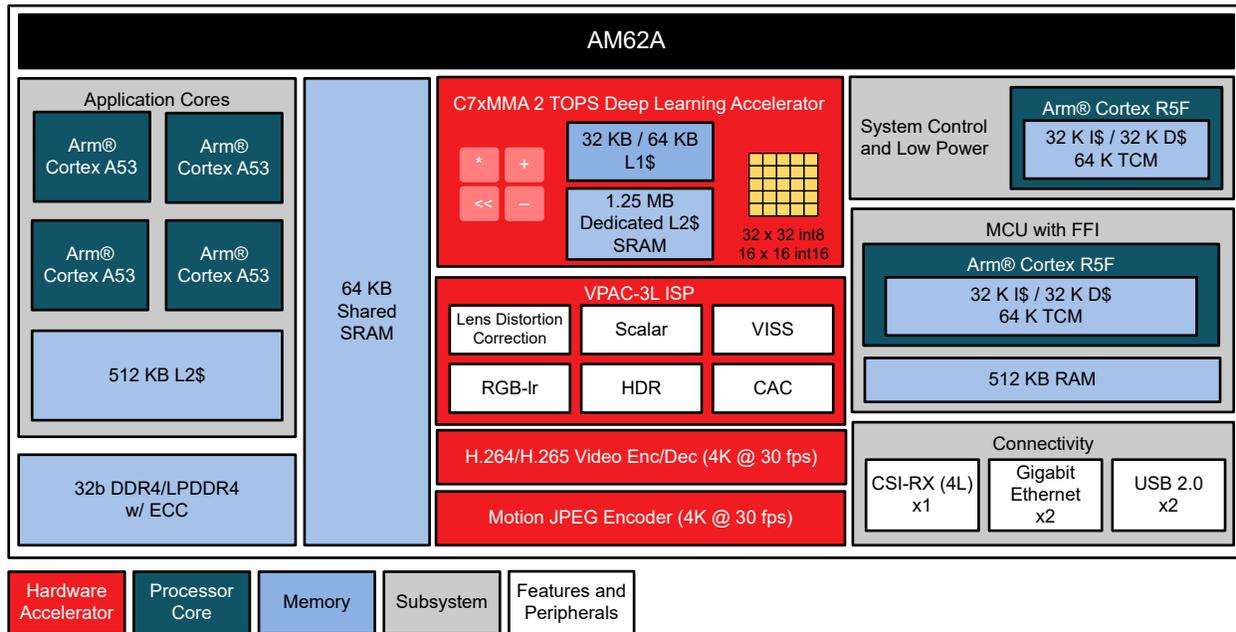


图 2-1. AM62A 简化方框图

图 2-2 演示了视觉分析应用的一般数据流。图像由低成本原始图像传感器 (即摄像头) 生成。原始图像数据通过 4 通道 MIPI-CSI2 端口进入处理器, 该端口可以拆分为多个虚拟通道, 以便连接更多摄像头。图像由 ISP 进行增强, 以减少噪点、调整白平衡和增益、滤波和内插颜色信息以及处理高动态范围 (HDR) 信息。对于具有广角镜头的应用, 镜头失真校正 (LDC) 加速器会减少镜头的扭曲效应。在图像经过预处理以满足 AI 模型的输入规格后, 硬件加速器会以 50 至 100 倍的 CPU 性能运行模型; 请参阅表 3-1 以了解几个模型基准。AI 模型可以完成以下任务: 识别食品、在包装上找到条形码、确定客户在何处花费时间最多或检测盗窃模式。

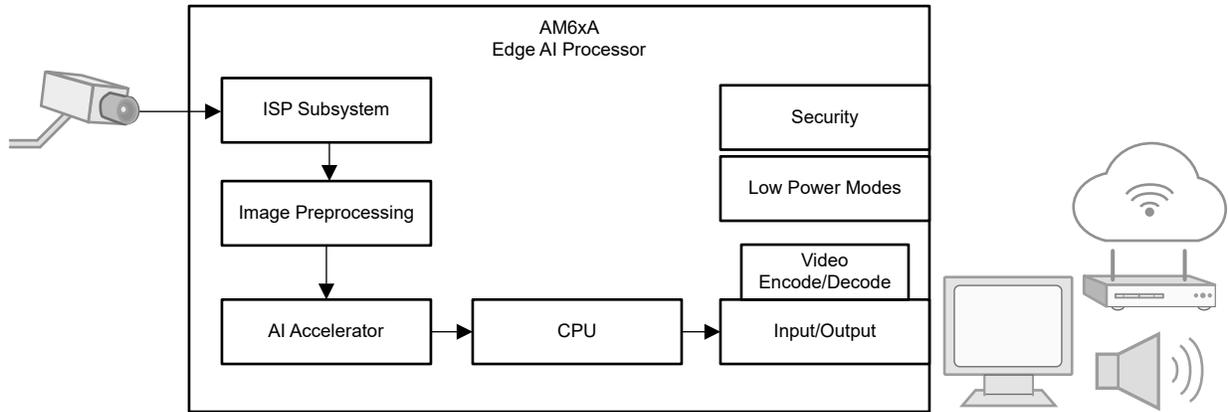


图 2-2. AM6xA 视觉应用数据流

AI 模型运行后, 特定应用可以决定如何处理信息, 例如通过网络进行通信、在显示屏上显示信息或播放警报声。处于非活动状态时, 低功耗模式会显著降低功耗; 在 100% 负载下运行时, SoC 在高达 85°C 的温度下功耗不到 3W, 从而减少了对主动冷却的需求。器件上的安全功能可防止篡改, 从而保护数据和固件 IP。

AM62A 具有内部设计的 2TOPS 深度学习加速器。该加速器包含一个与矩阵乘法加速器 (MMA) 紧密耦合的 256 位 C7x DSP。这种紧密耦合使数据能够快速高效地移动到加速器, 从而确保加速器的高利用率。2TOPS 指标是指每秒对 8 位量化矩阵执行的最大运算次数。但是, TOPS 并不是深度学习加速性能的理想指标, 因为根据加速器架构, 甚至是模型/神经网络架构, 1TOPS 的推理时间和功耗可能会有很大不同。因此, 查看显示推理速率 (每秒帧数) 的基准会更有用。

3 深度学习基准测试

表 3-1 展示了在入门套件 EVM 版本 E2 上运行 Linux Edge AI SDK 版本 8.6 的 AM62A 的基准测试。请注意, 由于 E2 EVM 的 PMIC 提供 0.75V VDD 内核电压, 因此这些数字反映出最高性能为 1.7TOPS。要实现完整的 2TOPS, 需要 0.85V VDD。E2 之后的 EVM 将使用更新的 PMIC 来获得完整的性能授权。

表 3-1. AM62A 的基准测试

模型名称	精度 (C7xMMA)	每秒帧数 (C7xMMA)	每秒帧数 (CPU)	分辨率
分类网络 - 精度为首要指标				
TFL-CL-0010-mobileNetV2	74.51	251	5.5	224 x 224
物体检测网络 - 精度为 mAP50-95				
TFL-OD-2000-ssd-mobV1-coco-mlperf-300x300	26.14	152	2.4	300 x 300

TI 的 Model Zoo 拥有更多在 C7xMMA 上经过预先训练和完全加速的模型。最新的基准测试 (相对于软件和 EVM 版本) 可在 [TI 的模型分析器工具 \[1\]](#) 中找到。

4 零售结账扫描仪应用

我们为 AM62A 开发了一个参考应用, 以展示其在使用物体检测神经网络的自动结账系统中的功能。我们训练了一个自定义模型, 以识别十几种不同的食品, 并使用 TI 的 gstreamer 插件围绕该模型编写了一个 Linux Python3 应用程序, 以尽可能利用硬件加速。Github 上提供了 [应用程序源代码 \[2\]](#) 和 [该演示工作原理的详细说明](#)。有关构建

此类应用程序的更多指导，请参阅相关的应用程序手册 [3]。图 4-1 中的方框图描述了 gstreamer 中的应用流程以及各种插件如何在 SoC 中的远程内核上执行。

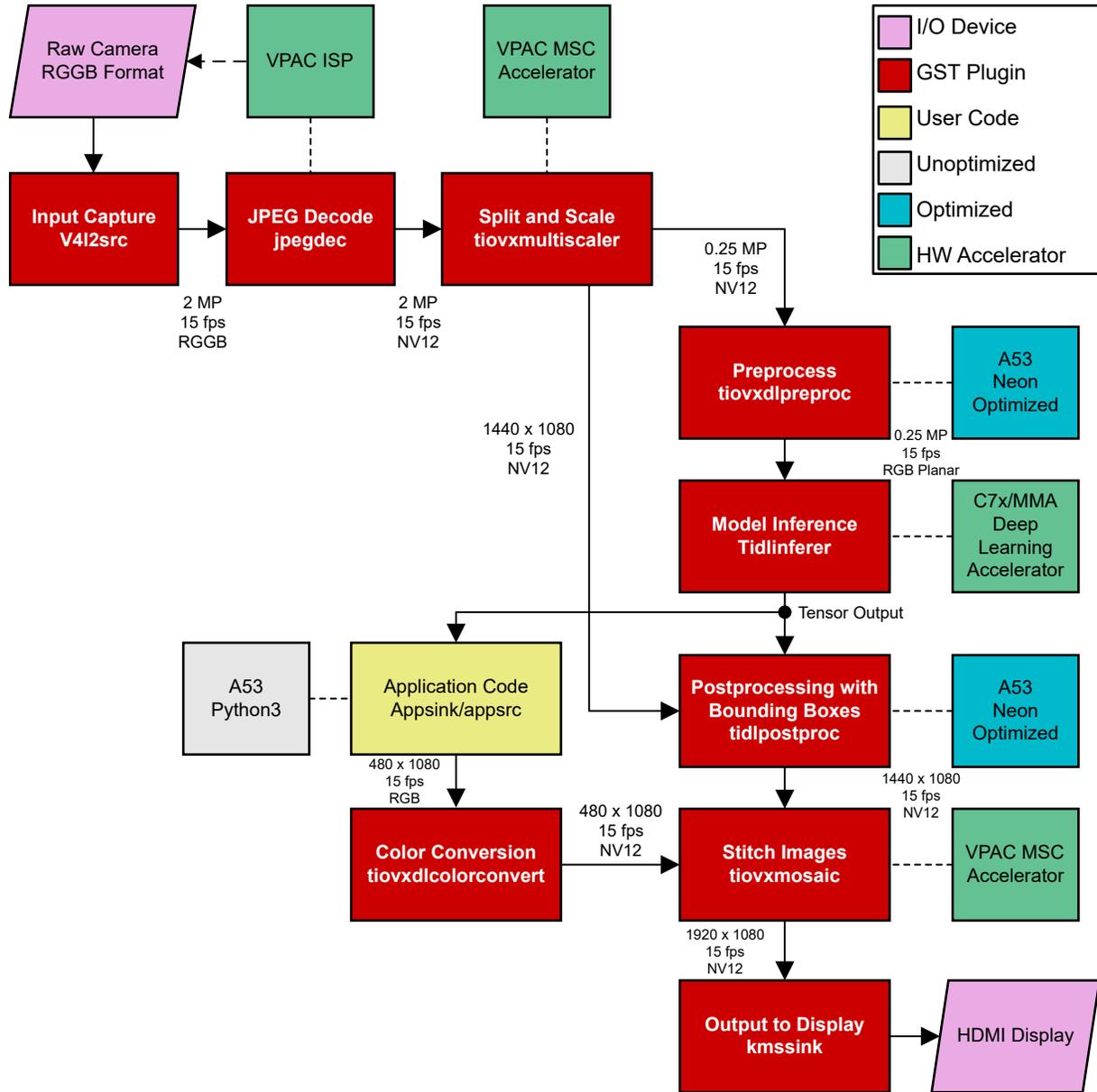


图 4-1. 具有分辨率和像素格式的零售结账应用流程。(30fps 是可实现的最大值；在该应用中，由于应用程序代码方面的瓶颈，FPS 接近 15)

本文档分析了此应用并使用内核负载来指导选择合适的 AM62A 型号，并使用功耗估算工具 [4] 提供了功耗估算。对于在使用超集型号 AM62A74 (2TOPS 加速度，4 个 Arm®Cortex A53 内核) 的入门套件 EVM 上设计和基准测试的其他应用，可以遵循此分析。

5 核心负载

AM6xA 系列中的处理器采用异构架构，具有各种 Arm® 内核和硬件加速器。典型的负载工具 (例如 Linux 中的“Top”) 不会显示集成微控制器或硬件加速器上的负载。

被测应用是零售结账演示，其运行速率约为 15fps，每帧延迟约为 200ms。这个的瓶颈主要由使用 Python3 为 CPU 编写的应用程序代码而造成。摄像头和物体检测模型 mobilenetv2SSD 可以处理更高的帧速率 (高达 60)。对于结账扫描仪，15fps 就足够了，并且允许开发人员选择 SoC 的低成本型号。

8.6 SDK 中支持两种查看核心负载的方法：

- `tiperoverlay gstreamer` 插件，用于沿屏幕底部绘制条形图，如图 5-1 所示
- 图 5-2 中的 `perf_stats` 命令行工具，用于将核心负载打印到终端窗口上。

每一个都具有相同的默认更新速率，即 2 秒。`tiperoverlay` 插件会增加 DDR 和 CPU 的开销，用于将信息绘制到输出帧。开箱即用演示默认使用 `tiperoverlay`。请注意，“MPU1_0”指标是指 A53 CPU 内核上的平均 CPU 负载。

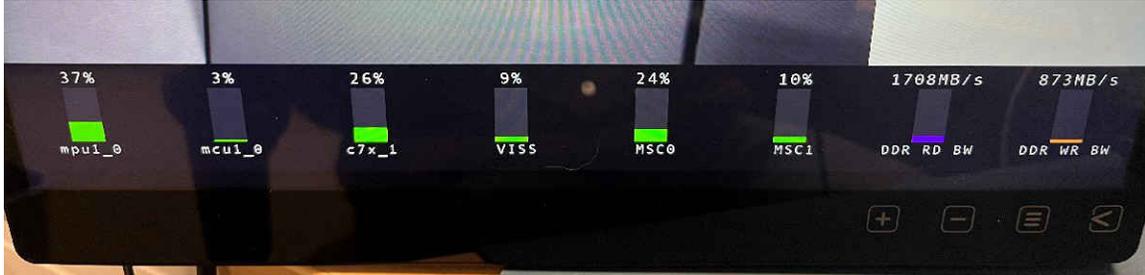


图 5-1. 运行零售结账演示时的 `tiperoverlay Gstreamer` 插件

```
Summary of CPU load,
=====
CPU: mpul_0: TOTAL LOAD = 36.25 % ( HWI = 0.50 %, SWI = 0.25 % )
CPU: mcul_0: TOTAL LOAD = 3.0 % ( HWI = 0.0 %, SWI = 0.0 % )
CPU: c7x_1: TOTAL LOAD = 25.0 % ( HWI = 0.0 %, SWI = 0.0 % )

HWA performance statistics,
=====
HWA: VISS: LOAD = 8.95 % ( 31 MP/s )
HWA: MSC0: LOAD = 23.67 % ( 96 MP/s )
HWA: MSC1: LOAD = 10.42 % ( 16 MP/s )

DDR performance statistics,
=====
DDR: READ BW: AVG = 1662 MB/s
DDR: WRITE BW: AVG = 823 MB/s
DDR: TOTAL BW: AVG = 2485 MB/s
```

图 5-2. 运行零售结账演示时的 `perf_stats` 命令行内核负载

这两个图之间存在细微差异。为了保持一致性，假设 A53 平均负载为 35%，C7x 内核负载为 25%（对于 E2 EVM，最大运行速度为 1.7TOPS），则 ISP (VISS) 为 10%，且多标量引擎 (MSC) 平均运行速度为 17%。请注意，在图 5-1 中，tiperoverlay 增加了在屏幕上绘制性能信息的开销，这对 DDR 的使用有很小的影响。

对于 MSC，总体使用量超过了 VISS，因为视觉流水线具有拆分和合并；性能影响是输入和输出的总和，在这个以大约 15fps 运行的 gstreamer 流水线中，预计为 115MP/s 的几个百分点以内。VPAC 加速器在内部包含 MSC 和 ISP (VISS)；对于此分析，请考虑 VPAC 的总体使用率为 10%。

DDR 为 3200MT/s 和 32 位，因此总共 2470MB/s 相当于大约 20% 的使用率，其中 30% 是写入传输。要查看用户空间中当前使用的内存大小，“htop” Linux 实用程序（请参阅图 5-3）非常有用。在本例中，操作系统中仅使用大约 315MB 的内存。请注意，最大值为 3.24GB。这是针对 Linux 内核和硬件加速器分割后 EVM 上 4GB LPDDR 模块的剩余容量。例如，在 8.6 处理器 SDK Linux 中，默认会为 C7xMMA 分配 256MB。考虑到这种内存使用情况，成本优化型系统可以使用 1.5GB 的 DDR，如果对 Linux 映像进行更多优化以删除不使用的服务和软件包，则可以低至 1GB。

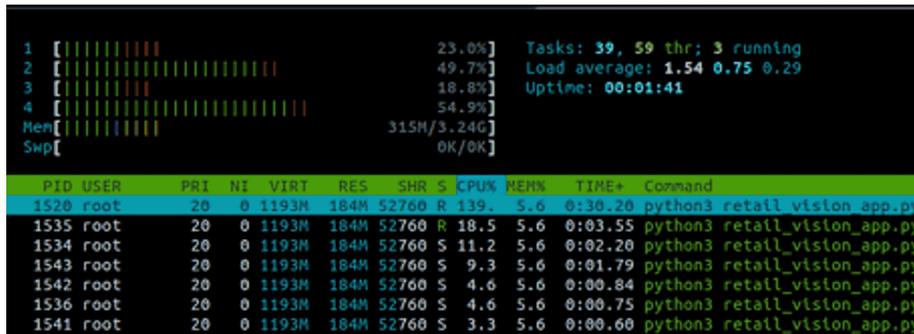


图 5-3. 使用“htop”Linux 实用程序时的零售结账演示负载

6 器件选择

AM62A SoC 包含多个成本优化型号。要做的两个主要选择是关于表 6-1 中的深度学习和 CPU 性能。

表 6-1. 基于深度学习性能和 CPU 内核数的 AM62A SoC 型号

器件	C7xMMA (TOPS)	A53 (#)
AM62A74	2	4
AM62A72	2	2
AM62A71	2	1
AM62A34	1	4
AM62A32	1	2
AM62A31	1	1

结合表格和负载信息，AM62A32 足以满足此零售结账应用的需求。此 SoC 型号将加速器从 1GHz 降频到 500MHz，具有 1TOPS 的性能，其中这个以 15fps 运行的应用仅需要 50% 的性能。考虑到四个 CPU 内核的平均负载为 35%，双核型号也是可以接受的。

7 功耗

AM62A 在构建时充分考虑了低功耗应用。AM62A 产品页面上提供了一个 [AM62Ax 功耗估算工具 \(PET\)](#) [4]。内核负载和时钟频率用作根据从裸机测试收集的台式测量值估算功率的参数，这意味着没有 Linux 等操作系统。要在此功耗估算工具中仿真 AM62A32，C7x 频率应小于或等于 500MHz，并且只有两个 A53 内核的利用率应高于 0%。要获得最准确的功耗估算，请参阅最新版本的 PET。

针对零售扫描仪应用功耗收集的台式测量结果显示，在 42°C 的结温下，处理器平均功耗为 1810mW。该处理器配置为 C7xMMA 频率使用 500MHz，CPU 频率使用 1250MHz。

此功率测量值至少比优化的处理器配置高 300mW。8.6 Linux SDK 会启用处理器上的所有元件并为这些元件提供时钟信号，而不管这些元件是否使用。因此，Linux SDK 在产品发布时的功耗高于针对低功耗进行优化的器件配置。

8 总结

AM62A 非常适合需要边缘进行高级图像处理的视觉零售系统。本文档分析了自动零售结账系统的演示应用，以确定异构 AM62A 处理器架构上的内核负载。然后使用此信息来选择合适的处理器成本优化型号和进行功耗估算。

9 参考文献

1. “模型分析器”，[在线]。网址：<https://dev.ti.com/edgeaisession/>。[2023 年 4 月 26 日访问]。
2. “Edge AI 零售结账演示”，2023 年 4 月 15 日。[在线]。网址：<https://github.com/TexasInstruments/edgeai-gst-apps-retail-checkout/blob/main/retail-shopping>。[2023 年 4 月 26 日访问]。
3. 德州仪器 (TI)：在 *AM6xA MPU* 上为自动零售扫描仪构建边缘 AI 应用
4. 德州仪器 (TI)：*AM62Ax 功耗估算工具 (PET)*

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2023，德州仪器 (TI) 公司