

# Application Note

## AM62A 上的多媒体应用



Suren Porwar and Tarkesh Pande

### 摘要

本应用手册针对使用 AM62A 处理器时的多媒体性能提供了各种基准 [1]。本文中的多媒体是指对视频流进行编码和解码以及对静态图像进行编码的能力。AM62A 中的硬件加速器能够进行 H.264/265 编码/解码和 MJPEG 编码。还介绍了示例用例，并对典型用例负载进行了系统分析。

### 内容

1 引言.....	2
2 AM62A 处理器.....	2
3 AM62A 视频处理单元 (VPU) 功能.....	3
4 AM62A JPEG 编码能力.....	3
4.1 开源 GStreamer 概述.....	3
4.2 TI 提供的用于多媒体的 V4L2 驱动程序.....	4
4.3 硬件加速 GStreamer 插件.....	4
5 软件驱动器架构.....	4
6 性能测量.....	5
7 多媒体用例.....	5
7.1 智能 AI 盒.....	5
7.2 安全监控.....	7
8 总结.....	9
9 参考资料.....	10

### 插图清单

图 2-1. AM62A 简化方框图.....	2
图 5-1. 软件驱动程序架构概述.....	4
图 7-1. 智能 AI 盒.....	6
图 7-2. 智能监控单个编码用例.....	7
图 7-3. 智能监控多重编码用例.....	8

### 表格清单

表 3-1. 编码器/解码器规格.....	3
表 4-1. JPEG 规格.....	3
表 4-2. JPEG 编码器的理论吞吐量.....	3
表 4-3. GStreamer 插件和格式支持.....	4
表 6-1. 采用 H.264/H.265 编码的单实例性能结果.....	5
表 6-2. 采用 H.264/H.265 解码的单实例性能结果.....	5
表 7-1. AM62A 在 850MHz 下使用 C7/MMA 时的深度学习性能示例.....	6
表 7-2. 智能 AI 盒应用的资源利用率表示例.....	6
表 7-3. 具有编码和分析功能的 800 万像素监控摄像头的资源利用率.....	7
表 7-4. 智能监控多重编码用例资源利用率.....	8

### 商标

所有商标均为其各自所有者的财产。

## 1 引言

在当今快速发展的数字化环境中，物联网应用的边缘有多种技术正在融合。基于摄像头和视频的应用现在可以执行智能边缘处理，例如检测人员、目标物体或事件，同时计算和功耗显著降低。然后，这些经过处理的数据可以无缝地流式传输到一个中心点，例如可视门铃、无人机和安防摄像头等监控应用。对于这些类型的应用，出色的编码/解码能力以及支持多种配置（例如多流和多分辨率编码/解码）的能力非常重要。本应用手册展示了如何使用 AM62A 处理器来应对这些用例。其 H.264 和 H.265 编码和解码功能允许对高清视频内容进行无缝压缩和解压缩，确保对实时流式传输和视频监控等应用进行高效的数据处理。要以上所有目标，解决方案需要具备低功耗和小尺寸特性，从而更大幅度地延长电池寿命。本应用手册主要介绍 AM62A 器件的多媒体功能 [2]。本文档简要介绍了 AM62A，并描述了用于访问多媒体加速器的软件框架。接下来，提供了系统设计人员关心的典型基准，例如延时和 DDR 带宽消耗。最后，两个系统示例使用案例展示了系统流程以及资源负载。这些示例应用已经在 AM62A 入门套件 [3] 上进行了测试。

## 2 AM62A 处理器

AM62A Edge AI 微处理器如图 2-1 所示。AM62A 简化方框图是一款面向分析应用而设计的异构处理器。不同的硬件加速器针对不同的任务进行了优化，从而能够针对不同的视觉和多媒体应用优化功耗和成本以及占用空间

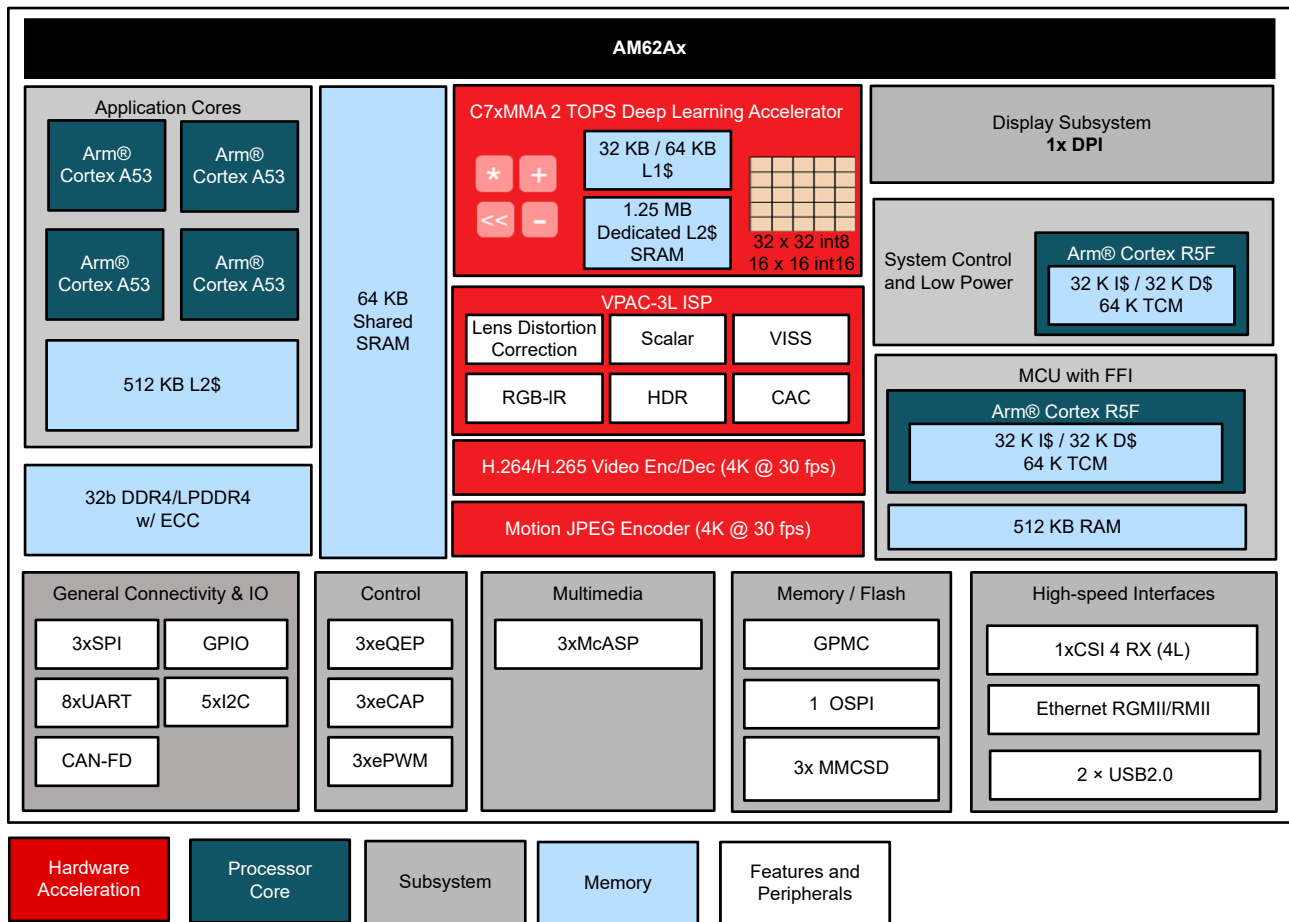


图 2-1. AM62A 简化方框图

AM62A 中来自多媒体环境的主要处理和计算子系统如下所示：

- 四核 A53 ARM 内核：这些内核的运行频率最高可达 1.4GHz，并提供高达 16.8k Dhrystone 每秒百万条指令 (DMIPS) 的性能。
- C7 数字信号处理器 (DSP) 和矩阵乘法加速器 (MMA)：TI AM62A 上的深度学习加速器在时钟频率为 1GHz 时支持 2TOPS 运算。
- H.264/265 编码器和解码器：其测得的总编码/解码能力为 240MP/s，并可同时支持最多 32 个通道。

- MJPEG 编码器：其测得的编码能力为 416MP/s，可轻松支持 4K30 应用。
- 视觉处理加速器 (VPAC3L)：使用 TI 的全新一代 ISP 技术执行图像操作，其中一些示例包括颜色转换、色差校正、金字塔缩放和镜头失真校正。其总吞吐量高达 300MP/s，并支持 RGB-IR 摄像头。

### 3 AM62A 视频处理单元 (VPU) 功能

AM62A 中的编解码器硬件 IP 块支持 H.264 和 H.265 编码/解码功能。表 3-1 说明了编码器/解码器块的一些高级规格。

表 3-1. 编码器/解码器规格

最大分辨率	8192x 8192
最小分辨率	256x128
时钟速率	400MHz
约束条件	图片宽度和高度都是 8 的倍数
并发编码/解码流	取决于分辨率和帧速率 4 个 1920x1080 @ 30fps 编码流 + 4 个解码流
H.264 编码	H.264 Baseline/Constrained Baseline/ Main/High Profiles Level @ L5.2
H.265 编码	H.265 Main 和 Main Still Picture Profile @ L5.1 High Tier
H.264 解码	H.264 Baseline/Constrained Baseline/ Main/High Profiles Level @ L5.2
H.265 解码	H.265 Main 和 Main Still Picture Profile @ L5.1 High Tier

### 4 AM62A JPEG 编码能力

AM62A 中的 JPEG 硬件 IP 块是一种有状态且可扩展的高性能静态图像编码器。它能够对 YUV420/YUV422 原始图片数据进行实时编码，使其转换为完全压缩图像 (jpeg) 或压缩图像序列 (MJPEG)。

表 4-1 展示了 MJPEG 编码器的一些高级规格。

表 4-1. JPEG 规格

最大分辨率	8k x 8k
时钟速率	250MHz
编码率	4 字节/周期 (YUV422 中为 2 个像素/周期, YUV420 中为 2.66 个像素/周期)
特性支持	JPEG 基线配置文件 配置压缩比 两个量化表

表 4-2 列出了理论吞吐量。测得的吞吐量通常在 80% 左右，对于 YUV 4:2:2 格式，测得的吞吐量为 416MP/s。

表 4-2. JPEG 编码器的理论吞吐量

颜色二次采样	吞吐量
YUV 4:2:0	250x2.66= 666MP/s
YUV 4:2:2	250x2 = 500MP/s

JPEG 编码单个流 1080p@30FPS 的硬件和软件延时 (考虑生成标头、比特流) 测量结果约为 8.04ms。

有关 JPEG 编码器的详细信息可在 [6] 中找到。

#### 4.1 开源 GStreamer 概述

GStreamer 是一个开源框架，它简化了媒体播放器和捕获编码器等多媒体应用的开发。它通过使用标准接口封装了现有的多媒体软件元件 (例如编解码器和硬件加速器)，并为各种应用提供了统一的框架。值得注意的是，新的 GStreamer 功能正在不断添加，核心库也得到了 GStreamer 社区成员的积极支持。有关 GStreamer 框架的附加信息，请访问 GStreamer 工程网站 [4]。

GStreamer 的模块化性质有利于添加新功能，并能够实现应用开发和测试的灵活性。处理节点通过具有多个接收端和/或源端焊盘的 Gstreamer 插件来实现。许多插件通常作为 ARM 软件实现运行，但对于更复杂的 SOC，某些函数在硬件加速 IP 上的执行效果更好。用于 AM62A 视觉处理的 GStreamer 插件可在 [5] 中找到。

## 4.2 TI 提供的用于多媒体的 V4L2 驱动程序

Video4Linux 版本 2 (V4L2) 是一种开源框架，它提供了一个媒体接口来连接所有基于 Linux 的应用。V4L2 是一组器件驱动程序和 API，用于支持 Linux 系统上的实时视频捕捉和视频存储器到存储器操作。

分别使用编码器和解码器硬件的视频编码和解码作为 V4L2 驱动程序启用。V4L2 通过一个实现 V4L2 节点 ioctl 的薄层与编码器和解码器驱动器集成，并将 V4L2 数据结构转换为编码器/解码器可理解的数据结构。

## 4.3 硬件加速 GStreamer 插件

使用 GStreamer 作为多媒体框架的一个好处是内核库已在 ARM Linux 上构建并运行。Gstreamer 插件是在具有 ARM 和多媒体硬件加速器的 TI 嵌入式处理器上启用额外硬件功能所需的唯一元件。开源 GStreamer 插件为 GStreamer 流水线提供了一些元素，可以通过 V4L2 GStreamer 插件使用硬件加速视频解码。表 4-3 中提供了 AM62A Processor SDK 中的 GStreamer 插件和比特流格式支持。

表 4-3. GStreamer 插件和格式支持

	GStreamer 插件支持	支持的比特流格式
编码器	v4l2h264enc v4l2h265enc	V4L2_PIX_FMT_H264 V4L2_PIX_FMT_HEVC
解码器	v4l2h264dec v4l2h265dec	V4L2_PIX_FMT_H264 V4L2_PIX_FMT_HEVC

## 5 软件驱动器架构

如前一节所述，在用户空间级别，TI 使用 GStreamer 这种多媒体应用开源框架。Gstreamer 库又与 Video for Linux (V4L2 插件) 连接，后者处理特定于底层硬件加速器的所有详细信息。

具体来说，对于 VPU，基于 GStreamer 的 V4L2 插件与 V4L2 编解码器内核驱动器接口连接。而 V4L2 驱动器则通过处理器间通信 (IPC) 与编码器/解码器上运行的固件进行通信。同样，JPEG 驱动程序也基于 Video4Linux 2 (V4L2) API，负责配置 JPEG 编码器硬件，并在生成压缩图像后生成 JPEG 图像标头。驱动程序是使用 V4L2 的 M2M 框架实现的，该框架是存储器到存储器器件的框架，用于从存储器中获取数据、执行一些处理并将处理后的数据写回存储器。用于利用编码/解码和 JPEG 编码功能的总体软件驱动程序架构如图 5-1 所示。

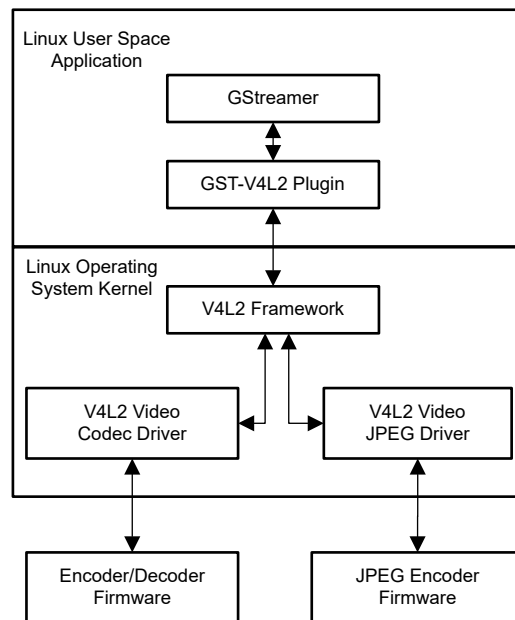


图 5-1. 软件驱动程序架构概述

## 6 性能测量

本节提供了 AM62A 上 H.264 和 H.265 编码/解码的存储器要求、延时和 DDR 带宽测量，适用于 480p 至 8MP 的典型配置。测量是使用 Linux SDK 9.0 在 AM62Ax 入门套件 [3] 上进行的。用于测量 DDR 带宽和延时的工具是 perf\_stats 工具。有关复制性能基准和相关 gstreamer 命令的详细信息，请访问 [9]。

**表 6-1. 采用 H.264/H.265 编码的单实例性能结果**

	8MP@15 FPS	5MP@30FPS	1080P@30FPS	480P@30FPS
	IMX219	OV5647	IMX219	IMX219
存储器要求 (MB)	257.7MB	76.16MB	43.2MB	23.6MB
延时 (ms)	33.3ms	29ms	8.3ms	4ms
DDR 带宽 (MB/s)	1271MB/s	1200MB/s	547MB/s	137MB/s

**表 6-2. 采用 H.264/H.265 解码的单实例性能结果**

	1080P@30FPS	480P@30FPS
	IMX219	IMX219
存储器要求 (MB)	62.8MB	29.5MB
延时 (ms)	77ms	36ms
DDR 带宽 (MB/s)	796MB/s	75MB/s

## 7 多媒体用例

### 7.1 智能 AI 盒

使用智能 AI 盒向零售店、交通道路、工厂和楼宇中的现有非分析型摄像头增加智能功能，是一种具有成本效益的方法。此类系统接收来自多个摄像头的实时视频流，对其进行解码并在边缘进行智能视频分析，从而减轻了将大型视频流传输回云进行分析的负担。AI 盒的应用包括具有异常或事件检测功能的安全监控系统、用于验证工人在进入危险区域之前是否穿戴了护目镜、安全背心和安全帽等个人防护装备 (PPE) 的工作场所安全系统。

图 7-1 展示了系统的数据流，其中有两个摄像头输入通过以太网进入，需要拆包并解码。通过深度学习进行的智能分析在解码的图像流上运行。TI 提供了一个模型分析器工具 [7]，AI 分析设计人员可以根据精度、推理时间和 DDR 带宽消耗选择所需的深度学习模型。表 7-1 中为 SSDLite-MobDet-EdgeTPU-coco 网络提供了一个此类利用的示例。

检测到的目标对象会在其周围绘制边界框，然后重新对图像进行编码。这种类型的设置用于为没有连接分析处理器的传统 FHD 摄像机提供分析功能。

表 7-2 提供系统工程师可以构建的最终资源利用率表。通过 [8] 中提供的 PET 工具估算功耗。

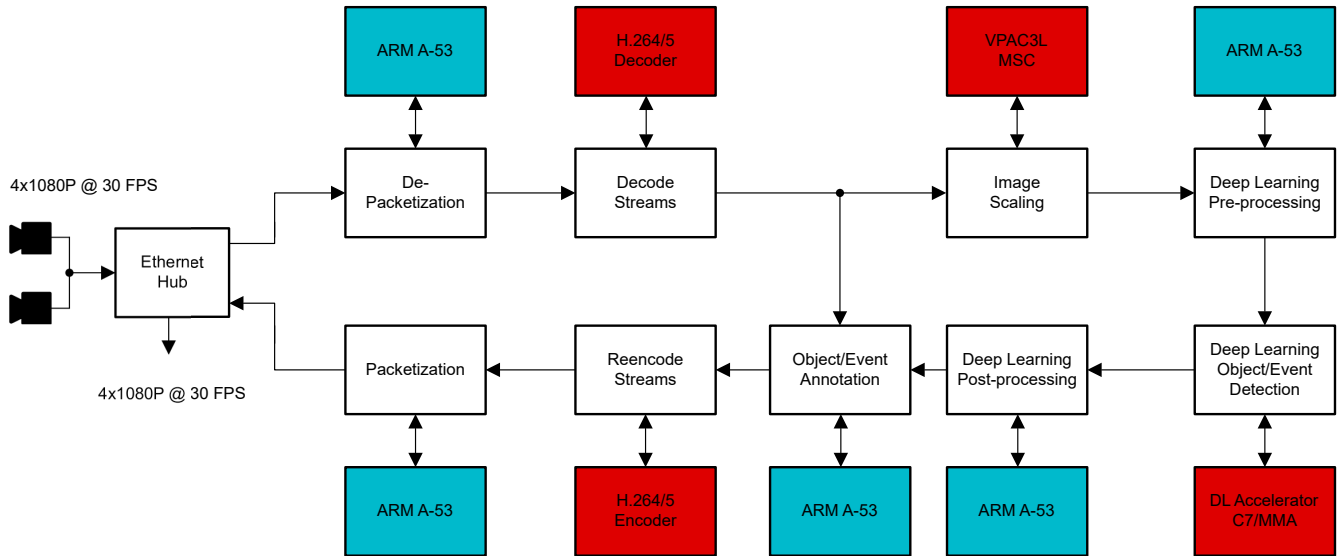


图 7-1. 智能 AI 盒

表 7-1. AM62A 在 850MHz 下使用 C7/MMA 时的深度学习性能示例

型号	分辨率	目标 FPS	MAP 精度 使用 CoCo 数据集	延迟 (ms)	深度学习 利用率	DDR 带宽 利用率
SSDLite-MobDet-EdgeTPU-coco	320x320	30	29.7	8.35	25%	543MB/s

表 7-2. 智能 AI 盒应用的资源利用率表示例

主 IP	负载
解码器+ 编码器	4 x2MP@ 30fps = 240MP/s
VPAC3L (ISP)	4x2MP@ 30fps = 240MP/s
ARM 加载 @ 1.25GHz	11.4%
深度学习 C7/MMA @850MHz	50%
DDR 带宽	2138MB/s ( 2522- 384 ( 显示 ) )
使用 PET 时的功耗估计值 (85c)	约为 1.8W

为了测量 DDR 带宽，构建了一个基于 gstreamer 的流水线，用于模拟图 7-1 中的数据流。在服务器端将 4 个已编码的数据流放到网络上，客户端用于对这些数据流进行解码，查看数据流并重新编码

• 服务器端 Gstreamer 流水线：

```
gst-launch-1.0 -v v4l2src device=/dev/video-rpi-cam0 io-mode=dmauf-import ! \
video/x-bayer, width=1920, height=1080, framerate=60/1, format=rggb10 ! \
tiovxisp sink_0::device=/dev/v4l-rpi-subdev0 sensor-name="SENSOR_SONY_IMX219_RPI" dcc-isp- \
file=/opt/imaging/imx219/linear/dcc_viss_10b.bin sink_0::dcc-2a-file=/opt/imaging/imx219/linear/ \
dcc_2a_10b.bin format-msb=9 ! \
video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! \
tiovxmultiscaler name=msc \
msc ! queue ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! v4l2h264enc ! \
rtph264pay ! udpsink host=192.168.1.134 port=6001 \
msc ! queue ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! v4l2h264enc ! \
rtph264pay ! udpsink host=192.168.1.134 port=6002 \
msc ! queue ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! v4l2h264enc ! \
rtph264pay ! udpsink host=192.168.1.134 port=6003 \
msc ! queue ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! v4l2h264enc ! \
rtph264pay ! udpsink host=192.168.1.134 port=6004
```

• 客户端 Gstreamer 流水线：

```
gst-launch-1.0 -v udpsrc port=6001 ! 'application/x-rtp, media=(string)video, clock-rate=(int)90000, encoding-name=(string)H264, payload=(int)96' ! \
rtpH264depay ! h264parse ! v4l2h264dec capture-io-mode=4 ! tiovxmultiscaler name=msc \
msc ! queue ! video/x-raw, width=640, height=480, format=NV12 ! kmssink driver-name=tidss \
msc ! queue ! video/x-raw, format=NV12, width=1920, height=1080 ! v4l2h264enc output-io-mode=5 \
extra-controls="controls,h264_i_frame_period=60,video_gop_size=60" ! rtpH264pay ! udpsink \
host=128.247.75.190 port=6001 &
gst-launch-1.0 -v udpsrc port=6002 ! 'application/x-rtp, media=(string)video, clock-rate=(int)90000, encoding-name=(string)H264, payload=(int)96' ! \
rtpH264depay ! h264parse ! v4l2h264dec capture-io-mode=4 ! queue ! v4l2h264enc output-io-mode=dmabuf-import extra-controls="controls,h264_i_frame_period=60,video_gop_size=60" ! \
rtpH264pay ! udpsink host=128.247.75.190 port=6002 &
gst-launch-1.0 -v udpsrc port=6003 ! 'application/x-rtp, media=(string)video, clock-rate=(int)90000, encoding-name=(string)H264, payload=(int)96' ! \
rtpH264depay ! h264parse ! v4l2h264dec capture-io-mode=4 ! queue ! v4l2h264enc output-io-mode=dmabuf-import extra-controls="controls,h264_i_frame_period=60,video_gop_size=60" ! \
rtpH264pay ! udpsink host=128.247.75.190 port=6003 &
gst-launch-1.0 -v udpsrc port=6004 ! 'application/x-rtp, media=(string)video, clock-rate=(int)90000, encoding-name=(string)H264, payload=(int)96' ! \
rtpH264depay ! h264parse ! v4l2h264dec capture-io-mode=4 ! queue ! v4l2h264enc output-io-mode=dmabuf-import extra-controls="controls,h264_i_frame_period=60,video_gop_size=60" ! \
rtpH264pay ! udpsink host=128.247.75.190 port=6004 &
```

7.2 安全监控

在交通监控、事件识别系统和无人机等许多监控应用中，需要以 800 万像素处理单个摄像头流。AM62A 的 ISP 具有 4k 线宽，因此能够支持高分辨率摄像头传感器。内部 ISP 随后会将图像比例缩小至较低的分辨率，并在由深度学习加速器处理之前校正摄像头镜头失真效应。如图 7-2 所示。

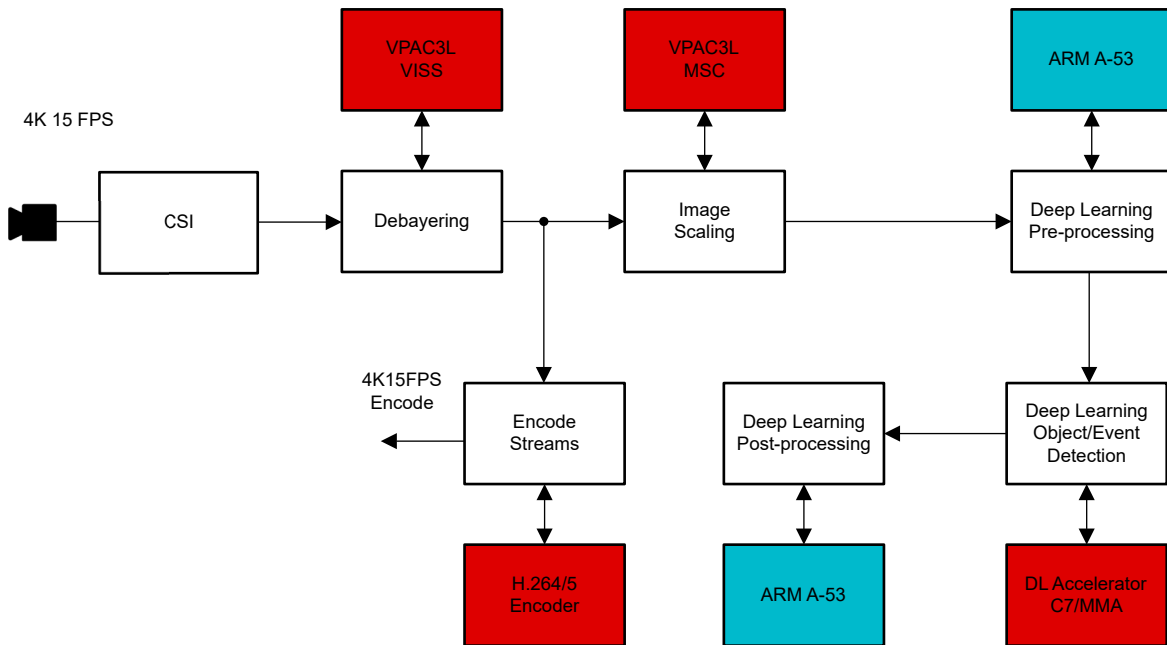


图 7-2. 智能监控单个编码用例

表 7-3. 具有编码和分析功能的 800 万像素监控摄像头的资源利用率

主 IP	负载
编码器	8MP@ 15fps = 120MP/s
VPAC3L (ISP)	8MP@ 15fps = 120MP/s
ARM 加载 @ 1.25GHz	16%
深度学习 C7/MMA @850MHz	25%
DDR 带宽	1860MB/s ( 2410 - 550 ( 显示 ) ) MB/s
使用 PET 时的功耗估计值 (85c)	约为 1.5W

• 使用的 Gstreamer 流水线：

```
gst-launch-1.0 -v v4l2src device=/dev/video3 io-mode=dmauf-import ! \
video/x-bayer, width=3280, height=2464, framerate=15/1, format=rggb ! \
tiovxisp sink_0::device=/dev/v4l-subdev2 sensor-name="SENSOR_SONY_IMX219_RPI" dcc-isp-file=/opt/ \
imaging/imx219/linear/dcc_viss.bin sink_0::dcc-2a-file=/opt/imaging/imx219/linear/dcc_2a.bin \
format-msb=7 ! \
video/x-raw, format=NV12, width=3280, height=2464, framerate=15/1 ! \
tiovxmultiscaler name=msc \
msc. ! queue ! video/x-raw, format=NV12, width=1920, height=1080 ! kmssink driver-name=tidss \
msc. ! queue ! video/x-raw, format=NV12, width=3280, height=2464, framerate=15/1 ! v4l2h264enc ! \
rtph264pay ! udpsink host=192.168.65.187 port=5002
```

在许多示例中，监控摄像头没有像以太网这样的物理网络连接，视频流需要进行编码并通过 Wi-Fi 发送。在本例中，一些应用需要以不同的分辨率同时对流进行编码，并根据网络流量能够支持的功能选择用于 Wi-Fi 传输的编码流。如图 7-3 所示。

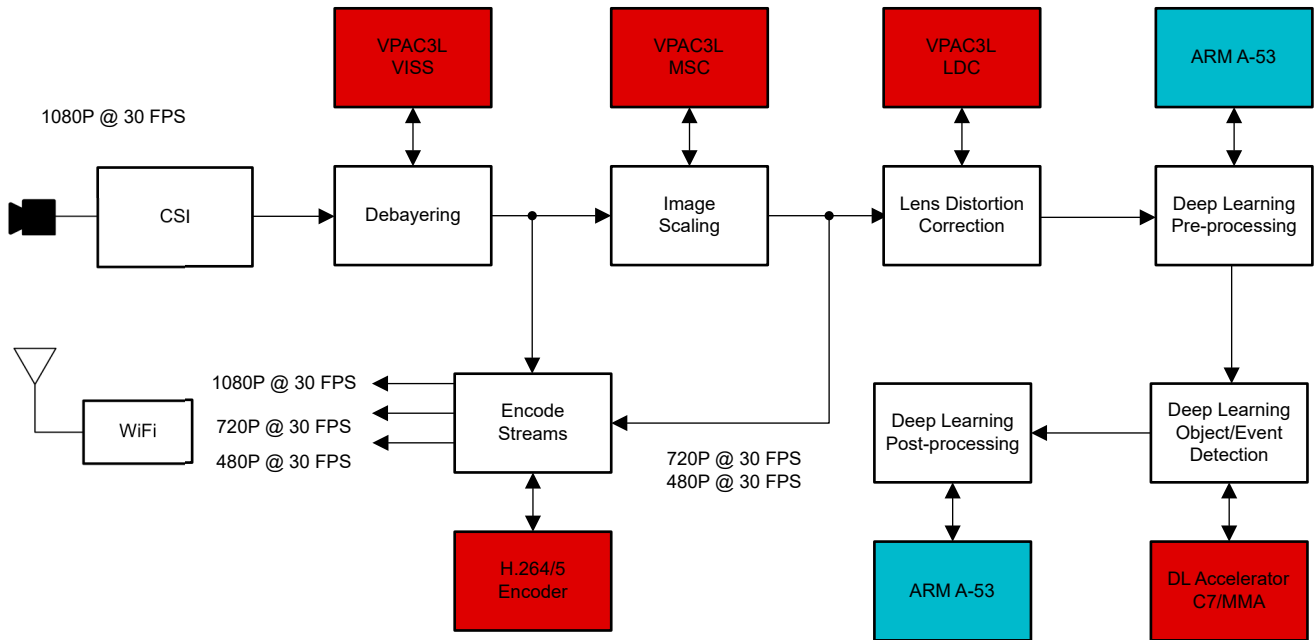


图 7-3. 智能监控多重编码用例

表 7-4. 智能监控多重编码用例资源利用率

主 IP	负载
解码器+ 编码器	(2MP+0.9M+0.3MP)@ 30fps = 96MP/s
VPAC3L (ISP)	2MP@ 30fps = 60MP/s
ARM 加载 @ 1.25GHz	34.5%
深度学习 C7/MMA @850MHz	25%
DDR 带宽	1310MB/s ( 1860-550 ( 显示 ) )
使用 PET 时的功耗估计值 (85c)	

• 使用的 Gstreamer 流水线：

```
gst-launch-1.0 -v v4l2src device=/dev/video-rpi-cam0 io-mode=dmauf-import ! \
video/x-bayer, width=1920, height=1080, framerate=30/1, format=rggb10 ! \
tiovxisp sink_0::device=/dev/v4l-rpi-subdev0 sensor-name="SENSOR_SONY_IMX219_RPI" \
dcc-isp-file=/opt/imaging/imx219/linear/dcc_viss_10b.bin \
sink_0::dcc-2a-file=/opt/imaging/imx219/linear/dcc_2a_10b.bin format-msb=9 ! \
video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! tiovxmultiscaler name=msc \
msc. ! queue ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! \
v4l2h264enc output-io-mode=dmauf-import extra- \
controls="controls,h264_i_frame_period=60,video_gop_size=60" ! \
```



```
rtph264pay ! udpsink host=192.168.65.187 port=5001 \  
msc. ! queue ! video/x-raw, format=NV12, width=1280, height=720, framerate=30/1 ! \  
v4l2h264enc output-io-mode=dmabuf-import extra-  
controls="controls,h264_i_frame_period=60,video_gop_size=60"! \  
rtph264pay ! udpsink host=192.168.65.187 port=5002 \  
msc. ! queue ! video/x-raw, format=NV12, width=640, height=480, framerate=30/1 ! tee name=t1 \  
t1. ! queue ! v4l2h264enc output-io-mode=dmabuf-import extra-  
controls="controls,h264_i_frame_period=60,video_gop_size=60"! \  
rtph264pay ! udpsink host=192.168.65.187 port=5003 \  
t1. ! queue ! kmssink driver-name=tidss
```

## 8 总结

本应用手册介绍了 AM62A 器件的多媒体功能。提供了有关如何使用多媒体加速器的典型基准和两个不同的示例。

## 9 参考资料

1. [AM62A 产品](#)
2. [AM62A 多媒体学院](#)
3. [AM62A 入门套件](#)
4. [Gstreamer 工程网站](#)
5. [适用于 Vision 的 Gstreamer 插件](#)
6. [JPEG 编码器](#)
7. [TI 的模型分析器](#)
8. 德州仪器 (TI) : [AAM62A 功耗估算工具](#)
9. [Gstreamer 流水线示例](#)

## 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 © 2024，德州仪器 (TI) 公司