

Magewell Pro Capture 系列采集卡

软件开发指南 for Linux

版权所有 2011–2016 [南京美乐威电子科技有限公司](#) 保留所有权利。

最后更新时间 2016–02–17

1 开发简介

本文主要介绍如何在 Linux 平台下开发基于 Pro Capture 系列采集卡的应用软件。

1.1 编程接口

一般来说，开发人员主要使用以下两种编程接口进行开发：

- V4L2 和 ALSA 标准编程接口
 - 使用标准接口实现音视频数据的采集；
 - 支持标准采集控制，包括采集大小、色彩空间等各项采集参数设置；
- I/O Control 编程接口
 - 基于 V4L2 驱动的 I/O Control 扩展功能实现；
 - 能够实现更加灵活、更加丰富的采集特性支持；

以上两种方法可以独立使用，也可以组合使用，开发人员可以根据实际需要进行选择。

1.2 I/O Control 功能特性

基于 I/O Control 编程接口开发应用软件，可以实现以下功能特性：

- 低延时图像数据采集；

把图像数据分行传输，信号延时仅为输入 64 行图像信号所需的时间。

- 多码流图像数据输出；

对于同一个采集设备能够创建多个采集实例，每个实例拥有不同的采集大小和色彩空间，每个采集设备保留 400MB–700MB 的独享传输带宽，用户在使用上能够更加灵活。

- 支持 10 位色彩格式的图像采集；

支持 AYUV、V408、Y410、V410、RGB10、BGR10 等色彩格式的图像采集。

- 高精度板载时钟调度；

精度为 100 纳秒的板载时钟为采集到的所有音视频帧提供的时间信息，并且支持网络上多个采集设备之间的时钟同步处理。

- OSD 图像数据硬件合成；

支持 RGBA 格式的文本或图片通过硬件实时合成到采集画面上，并且可以实现多个采集画面合成叠加，形成画中画（PIP）或者分屏（PAP）效果。

- HDMI InfoFrame 数据包获取;

支持获取 HDMI 原始 InfoFrame 数据包，包括 AVI、Audio、SPD、MS、VS、ACP 等等，可以从中获取时间码等的各类扩展信息。

- 8 声道音频采集;

支持 IEC60958/IEC61937 格式的音频数据采集，包括 5.1 声道、7.1 声道、DTS、THX、SRS 等原始音频格式和 AAC、MP3 等各种压缩音频格式。

- 板载固件升级;

支持板载固件通过 API 调用方式进行升级，便于用户进行深度集成，以及实现远程升级等功能。

2 示例介绍

本节主要介绍开发包内提供的各项示例代码

2.1 I/O Control 示例

- 库文件位于 lib 目录下
- 示例位于 src 目录下;
- 在 sdk 根目录下直接运行 make，就可以编译所有示例，生成的可执行文件在 bin 目录下;
- 包含多种示例程序，涵盖 I/O Control 开发的各个方面;
 - mwcap-upgrade : 读取 MWF 固件升级文件，并将其写入对应的采集设备，以完成固件升级操作;
 - mwcap-info : 读取 Pro Capture 系列采集卡各种属性信息，包括输入源信息、设备硬件属性信息、HDMI 信息帧信息;
 - mwcap-video-capture : 获取采集设备信息，并采集的视频数据保存为 BMP 文件;
 - mwcap-audio-capture : 获取采集设备信息，并采集的音频数据保存为 WAV 文件;
 - mwcap-video-set : 获取采集设备视频通道的输入端口信息，并对端口进行选择设置;
 - mwcap-audio-set : 获取采集设备音频通道的输入端口信息，并对端口进行选择设置;
 - mwcap-lowlatency : 对于输入逐行视频信号，使用低延时模式进行视频分块采集;

3 I/O Control 开发接口

本节主要介绍 I/O Control 编程接口。其主要开发按以下步骤进行：

3.1 开发步骤

I/O Control 开发主要按以下步骤进行：

- 使用 open() 打开 /dev/video* 视频设备 (* 为从0开始的数字);
- 使用 ioctl() 函数调用 I/O Control 各种指令接口;
- 使用 close() 关闭视频设备句柄;

3.2 I/O Control 控制指令

I/O Control 控制指令定义在 inc 目录的 mw-linux.h 中，它包含以下几类指令：

3.2.1 设备流信息获取指令

Pro Capture 系列采集设备在驱动程序安装后，会在 /dev 目录下生成若干个 video* 设备(* 为从0开始的数字)。每个 video* 设备对应一个采集通道，每个采集通道可以同时被打开多次，设备流定义为对某个采集通道的一次打开操作。以下控制指令用于获取当前设备的设备流信息。

名称	注释
MWCAP_IOCTL_GET_STREAMS_COUNT	获取当前设备的设备流数目
MWCAP_IOCTL_GET_STREAMS_INFO	获取当前设备的设备流详细信息，参见 3.3.2 MWCAP_STREAMS_INFO
MWCAP_IOCTL_GET_CTRL_STREAM_ID	获取当前控制的设备流 ID
MWCAP_IOCTL_SET_CTRL_STREAM_ID	设置当前控制的设备流 ID
MWCAP_IOCTL_GET_SELF_STREAM_ID	获取当前设备的设备流 ID

3.2.2 设备信息获取指令；

名称	注释
MWCAP_IOCTL_GET_CHANNEL_INFO	获取当前采集通道的信息，参见 3.3.3 MWCAP_CHANNEL_INFO
MWCAP_IOCTL_GET_FAMILY_INFO	获取当前采集通道的设备类型信息，参见 3.3.4 MWCAP_PRO_CAPTURE_INFO
MWCAP_IOCTL_GET_VIDEO_CAPS	获取当前视频通道的支持特性，参见 3.3.5 MWCAP_VIDEO_CAPS
MWCAP_IOCTL_GET_AUDIO_CAPS	获取当前音频通道的支持特性，参见 3.3.6 MWCAP_AUDIO_CAPS

3.2.3 固件升级指令；

名称	注释
MWCAP_IOCTL_GET_FIRMWARE_STORAGE	获取当前通道的固件存储信息，参见 3.3.7 MWCAP_FIRMWARE_STORAGE
MWCAP_IOCTL_SET_FIRMWARE_ERASE	插除当前通道的固件存储区域，参见 3.3.8 MWCAP_FIRMWARE_ERASE
MWCAP_IOCTL_SET_FIRMWARE_DATA	向当前通道的固件存储区域写入数据，参见 3.3.9 MWCAP_FIRMWARE_DATA
MWCAP_IOCTL_GET_FIRMWARE_DATA	从当前通道的固件存储区域读取数据，参见3.3.9 MWCAP_FIRMWARE_DATA

3.2.4 硬件时钟指令；

名称	注释
MWCAP_IOCTL_GET_TIME	获取当前通道的硬件时钟
MWCAP_IOCTL_SET_TIME	设置当前通道的硬件时钟
MWCAP_IOCTL_TIME_REGULATION	校准当前通道的硬件时钟
MWCAP_IOCTL_TIMER_REGISTRATION	注册时钟调度对象
MWCAP_IOCTL_TIMER_DEREGISTRATION	

ON	注销时钟调度对象
MWCAP_IOCTL_TIMER_EXPIRE_TIME	设置时钟调度对象失效时间，参见 3.3.10 MWCAP_TIMER_EXPIRE_TIME
MWCAP_IOCTL_TIMER_WAIT	等待时钟调度对象失效，参见 3.3.11 MWCAP_TIMER_WAIT
MWCAP_IOCTL_TIMER_CANCEL	取消最后一次等待时钟调度对象失效

3.2.5 异步通知指令；

名称	注释
MWCAP_IOCTL_NOTIFY_REGISTRATION	注册异步事件通知对象，参见 3.3.12 MWCAP_NOTIFY_REGISTRATION_S
MWCAP_IOCTL_NOTIFY_DEREGISTRATION	注销异步事件通知对象
MWCAP_IOCTL_NOTIFY_STATUS	获取事件通知的状态，参见 3.3.13 MWCAP_NOTIFY_STATUS
MWCAP_IOCTL_NOTIFY_WAIT	等待异步通知事件返回，参见3.3.14 MWCAP_NOTIFY_WAIT
MWCAP_IOCTL_NOTIFY_CANCEL	取消最后一次等待异步通知事件返回
MWCAP_IOCTL_NOTIFY_ENABLE	使异步事件类型值生效，参见 3.3.15 MWCAP_NOTIFY_ENABLE

3.2.6 音视频采集指令；

名称	注释
MWCAP_IOCTL_GET_VIDEO_BUFFER_INFO	获取视频缓冲信息，参见 3.3.16 MWCAP_VIDEO_BUFFER_INFO
MWCAP_IOCTL_GET_VIDEO_FRAME_INFO	获取视频帧信息，参见 3.3.17 MWCAP_VIDEO_FRAME_INFO
MWCAP_IOCTL_VIDEO_CAPTURE_OPEN	启动通道的视频采集
MWCAP_IOCTL_VIDEO_CAPTURE_FRAME	采集一帧视频数据，参见 3.3.18 MWCAP_VIDEO_CAPTURE_FRAME
MWCAP_IOCTL_VIDEO_CAPTURE_STATUS	获取视频采集状态，参见 3.3.19 MWCAP_VIDEO_CAPTURE_STATUS
MWCAP_IOCTL_VIDEO_CAPTURE_CLOSE	关闭通道的视频采集
MWCAP_IOCTL_AUDIO_CAPTURE_OPEN	启动通道的音频采集
MWCAP_IOCTL_AUDIO_CAPTURE_FRAME	采集一帧音频数据，参见 3.3.20 MWCAP_AUDIO_CAPTURE_FRAME
MWCAP_IOCTL_AUDIO_CAPTURE_CLOSE	关闭通道的音频采集

3.2.7 采集通道输入指令；

名称	注释
MWCAP_IOCTL_VIDEO_INPUT_SOURC	

E_COUNT	获取视频通道的输入接口数目
MWCAP_IOCTL_VIDEO_INPUT_SOURCE_ARRAY	获取视频通道的输入接口，参见 3.3.21 MWCAP_INPUT_SOURCE_ARRAY
MWCAP_IOCTL_AUDIO_INPUT_SOURCE_COUNT	获取音频通道的输入接口数目
MWCAP_IOCTL_AUDIO_INPUT_SOURCE_ARRAY	获取音频通道的输入接口，参见 3.3.21 MWCAP_INPUT_SOURCE_ARRAY
MWCAP_IOCTL_SET_INPUT_SOURCE_SCAN	设置视频输入是否自动扫描
MWCAP_IOCTL_GET_INPUT_SOURCE_SCAN	获取视频输入是否自动扫描
MWCAP_IOCTL_GET_INPUT_SOURCE_SCAN_STATE	获取采集通道的输入接口扫描状态
MWCAP_IOCTL_SET_AV_INPUT_SOURCE_LINK	设置音频是否自动连接到视频输入
MWCAP_IOCTL_GET_AV_INPUT_SOURCE_LINK	获取音频是否自动连接到视频输入
MWCAP_IOCTL_SET_VIDEO_INPUT_SOURCE	设置该通道当前视频输入源，参见 3.4.16 MWCAP_VIDEO_INPUT_TYPE
MWCAP_IOCTL_GET_VIDEO_INPUT_SOURCE	获取该通道当前视频输入源，参见 3.4.16 MWCAP_VIDEO_INPUT_TYPE
MWCAP_IOCTL_SET_AUDIO_INPUT_SOURCE	设置该通道当前音频输入源，参见 3.4.17 MWCAP_AUDIO_INPUT_TYPE
MWCAP_IOCTL_GET_AUDIO_INPUT_SOURCE	获取该通道当前音频输入源，参见 3.4.17 MWCAP_AUDIO_INPUT_TYPE

3.2.8 信号状态指令；

名称	注释
MWCAP_IOCTL_GET_INPUT_SPECIFIC_STATUS	获取该通道的输入信号状态，参见 3.3.28 MWCAP_INPUT_SPECIFIC_STATUS
MWCAP_IOCTL_GET_VIDEO_SIGNAL_STATUS	获取视频输入信号的状态，参见 3.3.29 MWCAP_VIDEO_SIGNAL_STATUS
MWCAP_IOCTL_GET_AUDIO_SIGNAL_STATUS	获取音频输入信号的状态，参见 3.3.30 MWCAP_AUDIO_SIGNAL_STATUS
MWCAP_IOCTL_GET_HDMI_INFOFRAME_VALID	获取 HDMI InfoFrame 有效状态，参见 3.4.13 MWCAP_HDMI_INFOFRAME_ID
MWCAP_IOCTL_GET_HDMI_INFOFRAME_PACKET	获取 HDMI 信息帧的数据，参见 3.3.32 HDMI_INFOFRAME_PACKET

3.2.9 视频处理指令

名称	注释
MWCAP_IOCTL_GET_VIDEO_INPUT_ASPECT_RATIO	获取视频输入信号的宽高比，参见 3.3.33 MWCAP_VIDEO_ASPECT_RATIO
MWCAP_IOCTL_SET_VIDEO_INPUT_ASPECT_RATIO	设置视频输入信号的宽高比，参见 3.3.33 MWCAP_VIDEO_ASPECT_RATIO
MWCAP_IOCTL_GET_VIDEO_INPUT_COLORSPACE	获取视频输入信号的色彩空间格式，参见 3.4.4 MWCAP_VIDEO_COLORSPACE

OLOR_FORMAT	DEO_COLOR_FORMAT
MWCAP_IOCTL_SET_VIDEO_INPUT_COLOR_FORMAT	设置视频输入信号的色彩空间格式，参见 3.4.4 MWCAP_VIDEO_COLOR_FORMAT
MWCAP_IOCTL_GET_VIDEO_INPUT_QUANTIZATION_RANGE	获取视频输入的量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
MWCAP_IOCTL_SET_VIDEO_INPUT_QUANTIZATION_RANGE	设置视频输入的量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE

3.2.10 OSD 指令

名称	注释
MWCAP_IOCTL_VIDEO_CREATE_IMAGE	创建 OSD 图像，参见 3.3.34 MWCAP_VIDEO_CREATE_IMAGE
MWCAP_IOCTL_VIDEO_OPEN_IMAGE	打开 OSD 图像，参见 3.3.35 MWCAP_VIDEO_IMAGE_REF
MWCAP_IOCTL_VIDEO_CLOSE_IMAGE	关闭 OSD 图像，参见 3.3.35 MWCAP_VIDEO_IMAGE_REF
MWCAP_IOCTL_VIDEO_UPLOAD_IMAGE	上载图像到采集设备，参见 3.3.36 MWCAP_VIDEO_UPLOAD_IMAGE

3.2.11 时序指令

名称	注释
MWCAP_IOCTL_GET_VIDEO_AUTO_HALIGN	获取视频水平方向是否自动调整
MWCAP_IOCTL_SET_VIDEO_AUTO_HALIGN	设置视频水平方向是否自动调整
MWCAP_IOCTL_GET_VIDEO_SAMPLING_PHASE	获取视频采样相位大小
MWCAP_IOCTL_SET_VIDEO_SAMPLING_PHASE	设置视频采样相位大小
MWCAP_IOCTL_GET_VIDEO_SAMPLING_PHASE_AUTO	获取视频采样相位是否自动调整
MWCAP_IOCTL_SET_VIDEO_SAMPLING_PHASE_AUTO	设置视频采样相位是否自动调整
MWCAP_IOCTL_SET_VIDEO_TIMING	获取视频通道时序信息，参见 3.3.37 MWCAP_VIDEO_TIMING
MWCAP_IOCTL_GET_VIDEO_PREFERRED_TIMING_ARRAY	获取预设置视频时序参数，参见 3.3.37 MWCAP_VIDEO_TIMING
MWCAP_IOCTL_SET_VIDEO_CUSTOM_TIMING	设置自定义视频时序参数，参见 3.3.38 MWCAP_VIDEO_CUSTOM_TIMING
MWCAP_IOCTL_GET_VIDEO_CUSTOM_TIMING_COUNT	获取自定义视频时序个数
MWCAP_IOCTL_GET_VIDEO_CUSTOM_TIMING_ARRAY	获取自定义视频时序，参见 3.3.39 MWCAP_VIDEO_CUSTOM_TIMING_PAR
MWCAP_IOCTL_SET_VIDEO_CUSTOM_TIMING_ARRAY	设置自定义视频时序，参见 3.3.39 MWCAP_VIDEO_CUSTOM_TIMING_PAR
MWCAP_IOCTL_GET_VIDEO_CUSTOM_TIMING_ARRAY	

RESOLUTION_COUNT	获取自定义视频分辨率个数
MWCAP_IOCTL_GET_VIDEO_CUSTOM_RESOLUTION_ARRAY	获取自定义视频分辨率，参见 3.3.40 MWCAP_VIDEO_CUSTOM_RESOLUTION_PAR
MWCAP_IOCTL_SET_VIDEO_CUSTOM_RESOLUTION_ARRAY	设置自定义视频分辨率，参见 3.3.40 MWCAP_VIDEO_CUSTOM_RESOLUTION_PAR

3.2.12 其他指令；

名称	注释
MWCAP_IOCTL_SET_LED_MODE	设置 LED 模式，参见 3.4.15 MWCAP_LED_MODE
MWCAP_IOCTL_SET_POST_RECONFIG	发送重读配置的指令，延迟执行
MWCAP_IOCTL_GET_CORE_TEMPERATURE	获取采集设备芯片温度
MWCAP_IOCTL_GET_EDID_DATA	获取 EDID 信息，参见 3.3.41 MWCAP_EDID_DATA
MWCAP_IOCTL_SET_EDID_DATA	设置 EDID 信息，参见 3.3.41 MWCAP_EDID_DATA

3.2.13 V4L2 控制指令

名称	注释
MWCAP_IOCTL_GET_VIDEO_CONNECTION_FORMAT	获取视频采集格式信息，参见 3.3.42 MWCAP_VIDEO_CONNECTION_FORMAT
MWCAP_IOCTL_GET_VIDEO_PROCESS_SETTINGS	获取视频采集处理配置参数，参见 3.3.43 MWCAP_VIDEO_PROCESS_SETTINGS
MWCAP_IOCTL_SET_VIDEO_PROCESS_SETTINGS	设置视频采集处理配置参数，参见 3.3.43 MWCAP_VIDEO_PROCESS_SETTINGS
MWCAP_IOCTL_GET_VIDEO_OSD_SETTINGS	获取视频采集 OSD 配置参数，参见 3.3.44 MWCAP_VIDEO_OSD_SETTINGS
MWCAP_IOCTL_SET_VIDEO_OSD_SETTINGS	设置视频采集 OSD 配置参数，参见 3.3.44 MWCAP_VIDEO_OSD_SETTINGS
MWCAP_IOCTL_GET_VIDEO_OSD_IMAGE	获取视频采集 OSD 图像区域，参见 3.3.45 MWCAP_VIDEO_OSD_IMAGE
MWCAP_IOCTL_SET_VIDEO_OSD_IMAGE	设置视频采集 OSD 图像区域，参见 3.3.45 MWCAP_VIDEO_OSD_IMAGE
MWCAP_IOCTL_GET_BRIGHTNESS	获取视频采集亮度
MWCAP_IOCTL_SET_BRIGHTNESS	设置视频采集亮度
MWCAP_IOCTL_GET_CONTRAST	获取视频采集对比度
MWCAP_IOCTL_SET_CONTRAST	设置视频采集对比度
MWCAP_IOCTL_GET_HUE	获取视频采集色度
MWCAP_IOCTL_SET_HUE	设置视频采集色度
MWCAP_IOCTL_GET_SATURATION	获取视频采集饱和度
MWCAP_IOCTL_SET_SATURATION	设置视频采集饱和度

3.3 I/O Control 结构定义

3.3.1 MWCAP_STREAM_INFO

- 结构原型

```
typedef struct MWCAPSTREAM_INFO {
    int stream_id;
    pid_t pid;
    har comm[TS_COMM_LEN];
    bool streaming;
} MWCAP_STREAM_INFO;
```

• 参数说明

名称	注释
stream_id	设备流ID
pid	占用此设备流的进程ID
comm	占用此设备流的进程名称
streaming	是否为v4l2设备流

3.3.2 MWCAP_STREAMS_INFO

• 结构原型

```
typedef struct MWCAPSTREAMS_INFO {
    int count;
    MWCAP_STREAM_INFO *infos;
} MWCAP_STREAMS_INFO;
```

• 参数说明

名称	注释
count	设备流的数量
infos	设备流的数据，参见 3.3.1 MWCAP_STREAM_INFO

3.3.3 MWCAP_CHANNEL_INFO

• 结构原型

```
typedef struct MWCAPCHANNEL_INFO {
    WORD wFamilyID;
    WORD wProductID;
    CHAR chHardwareVersion;
    BYTE byFirmwareID;
    DWORD dwFirmwareVersion;
    DWORD dwDriverVersion;
    CHAR szFamilyName[MW_FAMILY_NAME_LEN];
    CHAR szProductName[MW_PRODUCT_NAME_LEN];
    CHAR szFirmwareName[MW_FIRMWARE_NAME_LEN];
    CHAR szBoardSerialNo[MW_SERIAL_NO_LEN];
    BYTE byBoardIndex;
    BYTE byChannelIndex;
} MWCAP_CHANNEL_INFO;
```

• 参数说明

名称	注释
wFamilyID	通道所属采集设备的产品系列，参见 3.4.1 MW_FAMILY_ID
wProductID	通道所属采集设备的设备类型 ID, 参见 3.4.2 MW_PRODUCT_ID
chHardwareVersion	通道所属采集设备的硬件版本号
byFirmwareID	通道所属采集设备的固件 ID

dwFirmwareVersion	通道所属采集设备的固件版本
dwDriverVersion	通道所属采集设备的驱动版本
szFamilyName	通道所属采集设备的产品系列名称
szProductName	通道所属采集设备的产品类型名称
szFirmwareName	通道所属采集设备的固件名称
szBoardSerialNo	通道所属采集设备的硬件序列号
byBoardIndex	设备开关序号,采集设备上旋钮开关指示的序号,序号从 0 开始
byChannelIndex	通道序号,一个采集设备上有多个通道时,序号从 0 开始

3.3.4 MWCAP_PRO_CAPTURE_INFO

- 结构原型

```
typedef struct MWCAPPRO_CAPTURE_INFO {
    BYTE byPCIBusID;
    BYTE byPCIDevID;
    BYTE byLinkType;
    BYTE byLinkWidth;
    BYTE byBoardIndex;
    WORD wMaxPayloadSize;
    WORD wMaxReadRequestSize;
    DWORD cbTotalMemorySize;
    DWORD cbFreeMemorySize;
} MWCAP_PRO_CAPTURE_INFO;
```

- 参数说明

名称	注释
byPCIBusID	采集设备连接到计算机上的系统总线
byPCIDevID	采集设备连接到计算机上的 PCI 设备
byLinkType	采集设备和计算机 PCI-e 接口协商一致后的数据传输标准
byLinkWidth	采集设备连接计算机的 PCI-e 插槽 Lane 数量
byBoardIndex	采集设备上拨码开关的数值
wMaxPayloadSize	采集设备上的最大负载大小
wMaxReadRequestSize	采集设备上的最大读请求大小
cbTotalMemorySize	采集设备上的总的内存大小
cbFreeMemorySize	采集设备上的空闲的内存大小

3.3.5 MWCAP_VIDEO_CAPS

- 结构原型

```
typedef struct MWCAPVIDEO_CAPS {
    DWORD dwCaps;
    WORD wMaxInputWidth;
    WORD wMaxInputHeight;
    WORD wMaxOutputWidth;
    WORD wMaxOutputHeight;
} MWCAP_VIDEO_CAPS;
```

- 参数说明

名称	注释
----	----

dwCaps	视频采集能力
wMaxInputWidth	输入视频的最大宽度
wMaxInputHeight	输入视频的最大高度
wMaxOutputWidth	输出视频的最大宽度
wMaxOutputHeight	输出视频的最大高度

3.3.6 MWCAP_AUDIO_CAPS

- 结构原型

```
typedef struct MWCAPAUDIO_CAPS {
    DWORD dwCaps;
} MWCAP_AUDIO_CAPS;
```

- 参数说明

名称	注释
dwCaps	音频采集能力

3.3.7 MWCAP_FIRMWARE_STORAGE

- 结构原型

```
typedef struct MWCAPFIRMWARE_STORAGE {
    DWORD cbStorage;
    DWORD cbEraseBlock;
    DWORD cbProgramBlock;
    DWORD cbHeaderOffset;
} MWCAP_FIRMWARE_STORAGE;
```

- 参数说明

名称	注释
cbStorage	固件存储区域的长度
cbEraseBlock	擦除区域的长度
cbProgramBlock	程序块的存储区域长度
cbHeaderOffset	固件头部区域的存储偏移量

3.3.8 MWCAP_FIRMWARE_ERASE

- 结构原型

```
typedef struct MWCAPFIRMWARE_ERASE {
    DWORD cbOffset;
    DWORD cbErase;
} MWCAP_FIRMWARE_ERASE;
```

- 参数说明

名称	注释
cbOffset	要擦除的存储区域的偏移地址
cbErase	擦除区域的长度

3.3.9 MWCAP_FIRMWARE_DATA

- 结构原型

```
typedef struct MWCAPFIRMWARE_DATA {
    unsigned int offset;
    unsigned int size;
    unsigned char *data;
} MWCAP_FIRMWARE_DATA;
```

• 参数说明

名称	注释
cbOffset	要写入的数据的起始位置
size	要写入的数据长度
data	要写入的数据

3.3.10 MWCAP_TIMER_EXPIRE_TIME

• 结构原型

```
typedef struct MWCAPTIMER_EXPIRE_TIME {
    MWCAP_PTR pvTimer;
    LONGLONG llExpireTime;
} MWCAP_TIMER_EXPIRE_TIME;
```

• 参数说明

名称	注释
pvTimer	时钟调度对象
llExpireTime	时钟调度对象失效时间

3.3.11 MWCAP_TIMER_WAIT

• 结构原型

```
typedef struct MWCAPTIMER_WAIT {
    MWCAP_PTR pvTimer;
    LONGLONG llTimeout;
} MWCAP_TIMER_WAIT;
```

• 参数说明

名称	注释
pvTimer	时钟调度对象
llTimeout	时钟调度对象等待超时时间

3.3.12 MWCAP_NOTIFY_REGISTRATION_S

• 结构原型

```
typedef struct MWCAPNOTIFY_REGISTRATION_S {
    MWCAP_PTR pvNotify;

    ULONGLONG ullEnableBits;
} MWCAP_NOTIFY_REGISTRATION_S;
```

• 参数说明

名称	注释
pvNotify	异步事件通知对象
ullEnableBits	事件响应类型值

3.3.13 MWCAP_NOTIFY_STATUS

• 结构原型

```
typedef struct MWCAPNOTIFY_STATUS {  
    MWCAP_PTR pvNotify;  
  
    ULONGLONG ullStatusBits;  
} MWCAP_NOTIFY_STATUS;
```

• 参数说明

名称	注释
pvNotify	异步事件通知对象
ullEnableBits	事件响应类型值

3.3.14 MWCAP_NOTIFY_WAIT

• 结构原型

```
typedef struct MWCAPNOTIFY_WAIT {  
    MWCAP_PTR pvNotify;  
  
    MWCAP_BOOL bGetStatus;  
    LONGLONG llTimeout;  
  
    ULONGLONG ullStatusBits;  
} MWCAP_NOTIFY_WAIT;
```

• 参数说明

名称	注释
pvNotify	异步事件通知对象
bGetStatus	是否获取事件响应类型值
llTimeout	等待异步事件超时时间
ullEnableBits	事件响应类型值

3.3.15 MWCAP_NOTIFY_ENABLE

• 结构原型

```
typedef struct MWCAPNOTIFY_ENABLE {  
    MWCAP_PTR pvNotify;  
  
    ULONGLONG ullEnableBits;  
} MWCAP_NOTIFY_ENABLE;
```

• 参数说明

名称	注释
pvNotify	异步事件通知对象
ullEnableBits	事件响应类型值

3.3.16 MWCAP_VIDEO_BUFFER_INFO

• 结构原型

```
typedef struct MWCAPVIDEO_BUFFER_INFO {
```

```

DWORD cMaxFrames;
BYTE iNewestBuffering;
BYTE iBufferingFieldIndex;

BYTE iNewestBuffered;

BYTE iBufferedFieldIndex;

BYTE iNewestBufferedFullFrame;
DWORD cBufferedFullFrames;
} MWCAP_VIDEO_BUFFER_INFO;

```

• 参数说明

名称	注释
cMaxFrames	板载缓存中的最大帧数
iNewestBuffering	正在缓存的片序号，一帧视频数据可能包含多个片
iBufferingFieldIndex	正在缓存的场的序号
iNewestBuffered	最新缓冲好的片序号
iBufferedFieldIndex	最新缓冲好的场序号
iNewestBufferedFullFrame	最新缓冲好的帧序号
cBufferedFullFrames	缓冲好的整帧数量

3.3.17 MWCAP_VIDEO_FRAME_INFO

• 结构原型

```

typedef struct MWCAPVIDEO_FRAME_INFO {
    MWCAP_VIDEO_FRAME_STATE state;

    BOOLEAN bInterlaced;
    BOOLEAN bSegmentedFrame;
    BOOLEAN bTopFieldFirst;
    BOOLEAN bTopFieldInverted;

    int cx;
    int cy;
    int nAspectX;
    int nAspectY;

    LONGLONG allFieldStartTimes[2];
    LONGLONG allFieldBufferedTimes[2];
    MWCAP_SMPTE_TIMECODE aSMPTETimeCodes[2];
} MWCAP_VIDEO_FRAME_INFO;

```

• 参数说明

名称	注释
state	视频帧的状态，参见 3.4.3 MWCAP_VIDEO_FRAME_STATE
bInterlaced	是否是隔行的
bSegmentedFrame	是否是分段的帧
bTopFieldFirst	是否是顶场在前
bTopFieldInverted	顶场是否反转
cx	视频帧的宽度
cy	视频帧的高度

nAspectX	宽高比中的宽度
nAspectY	宽高比中的高度
allFieldStartTimes	顶场和底场分别开始采集的时间
allFieldBufferedTimes	顶场和底场分别缓冲好的时间
aSMPTETimeCodes	顶场和底场各自的时间码

3.3.18 MWCAP_VIDEO_CAPTURE_FRAME

- 结构原型

```
typedef struct MWCAPVIDEO_CAPTURE_FRAME {
    DWORD dwFOURCC;
    WORD cx;
    WORD cy;
    int nAspectX;
    int nAspectY;
    MWCAP_VIDEO_COLOR_FORMAT colorFormat;
    MWCAP_VIDEO_QUANTIZATION_RANGE quantRange;
    MWCAP_VIDEO_SATURATION_RANGE satRange;

    SHORT sContrast;
    SHORT sBrightness;
    SHORT sSaturation;
    SHORT sHue;

    RECT rectSource;
    RECT rectTarget;
    MWCAP_VIDEO_DEINTERLACE_MODE deinterlaceMode;
    MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE aspectRatioConvertMode;
    char iSrcFrame;

    MWCAP_PTR pOSDImage;
    RECT aOSDRects[MWCAP_VIDEO_MAX_NUM_OSD_RECTS];
    int cOSDRects;

    BOOLEAN bPhysicalAddress;
    union {
        MWCAP_PTR pvFrame;
        LARGE_INTEGER liPhysicalAddress;
    };

    DWORD cbFrame;
    DWORD cbStride;

    BOOLEAN bBottomUp;

    WORD cyPartialNotify;

    DWORD dwProcessSwitchs;

    MWCAP_PTR pvContext;
} MWCAP_VIDEO_CAPTURE_FRAME;
```

- 参数说明

名称	注释
dwFOURCC	色彩格式，参考 MWFOURCC.h
cx	采集宽度
cy	采集高度

nAspectX	宽高比中的宽度
nAspectY	宽高比中的高度
colorFormat	颜色格式标准，参见 3.4.3 MWCAP_VIDEO_FRAME_STATE
quantRange	量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
satRange	饱和度范围，参见 3.4.11 MWCAP_VIDEO_SATURATION_RANGE
sContrast	对比度
sBrightness	亮度
sSaturation	饱和度
sHue	色度
rectSource	采集的源区域
rectTarget	采集的目标区域
deinterlaceMode	去隔行模式，参见 3.4.6 MWCAP_VIDEO_DEINTERLACE_MODE
aspectRatioConvertMode	宽高比转换模式，参见 3.4.7 MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE
iSrcFrame	采集的帧序号
pOSDImage	用于 OSD 合成的图片的句柄, 通过 MWCAP_IOCTL_VIDEO_CREATE_IMAGE 接口获得
aOSDRects	OSD 图片的合成的目标区域
cOSDRects	图片的合成的目标区域个数
bPhysicalAddress	是否将采集到的数据存放在指定的物理地址中
pvFrame	采集到的帧数据被存放的内存地址
liPhysicalAddress	采集到的帧数据被存放的物理地址
cbFrame	这一帧数据的长度
cbStride	这一帧数据所使用的步长
bBottomUp	是否上下颠倒
cyPartialNotify	部分通知的个数
dwProcessSwitchs	视频处理的掩码值，参考 MWCAP_VIDEO_PROCESS_XX
pvContext	采集该帧视频的上下文环境

3.3.19 MWCAP_VIDEO_CAPTURE_STATUS

- 结构原型

```
typedef struct MWCAPVIDEO_CAPTURE_STATUS {
    MWCAP_PTR pvContext;

    BOOLEAN bPhysicalAddress;
    union {
        MWCAP_PTR pvFrame;
        LARGE_INTEGER liPhysicalAddress;
    };

    BOOLEAN bFrameCompleted;
    WORD cyCompleted;
};
```

```
WORD cyCompletedPrev;
} MWCAP_VIDEO_CAPTURE_STATUS;
```

- 参数说明

名称	注释
pvContext	视频采集的上下文环境
bPhysicalAddress	是否使用物理地址作存放采集到的数据
pvFrame	存放采集数据的内存地址
liPhysicalAddress	存放采集数据的物理地址
bFrameCompleted	一帧是否采集完成
cyCompleted	采集完成的帧数量
cyCompletedPrev	先前采集完成的帧数量

3.3.20 MWCAP_AUDIO_CAPTURE_FRAME

- 结构原型

```
typedef struct MWCAPAUDIO_CAPTURE_FRAME {
    DWORD cFrameCount;
    DWORD iFrame;
    DWORD dwSyncCode;
    DWORD dwReserved;
    LONGLONG liTimestamp;
    DWORD adwSamples[MWCAP_AUDIO_SAMPLES_PER_FRAME * MWCAP_AUDIO_MAX_NUM_CHANNELS];
} MWCAP_AUDIO_CAPTURE_FRAME;
```

- 参数说明

名称	注释
cFrameCount	缓存的帧数量
iFrame	当前这一帧的索引号
dwSyncCode	音频帧数据的同步码
dwReserved	保留字段
liTimestamp	该音频帧的时间戳
adwSamples	音频采样数据，每个采样 32 比特宽，高位有效，通道顺序为，左 0，左 1，左 2，左 3，右 0，右 1，右 2，右 3

3.3.21 MWCAP_INPUT_SOURCE_ARRAY

- 结构原型

```
typedef struct MWCAPINPUT_SOURCE_ARRAY {
    unsigned int count;
    unsigned int *data;
} MWCAP_INPUT_SOURCE_ARRAY;
```

- 参数说明

名称	注释
count	视频输入源类型个数
data	视频输入源索引值

3.3.22 MWCAP_HDMI_VIDEO_TIMING

• 结构原型

```
typedef struct MWCAPIHDMI_VIDEO_TIMING {
    BOOLEAN bInterlaced;
    DWORD dwFrameDuration;
    WORD wHSyncWidth;
    WORD wHFrontPorch;
    WORD wHBackPorch;
    WORD wHActive;
    WORD wHTotalWidth;
    WORD wField0VSyncWidth;
    WORD wField0VFrontPorch;
    WORD wField0VBackPorch;
    WORD wField0VActive;
    WORD wField0VTotatHeight;
    WORD wField1VSyncWidth;
    WORD wField1VFrontPorch;
    WORD wField1VBackPorch;
    WORD wField1VActive;
    WORD wField1VTotatHeight;
} MWCAP_HDMI_VIDEO_TIMING;
```

• 参数说明

名称	注释
bInterlaced	是否交错
dwFrameDuration	帧时间间隔
wHSyncWidth	水平方向同步宽度
wHFrontPorch	水平方向的前沿宽度
wHBackPorch	水平方向的后沿宽度
wHActive	水平方向的有效宽度
wHTotalWidth	水平方向的总宽度
wField0VSyncWidth	第 0 场的垂直同步宽度
wField0VFrontPorch	第 0 场的垂直方向前沿宽度
wField0VBackPorch	第 0 场的垂直方向后沿宽度
wField0VActive	第 0 场的垂直方向有效宽度
wField0VTotatHeight	第 0 场的垂直方向的总高度
wField1VSyncWidth	第 1 场的垂直同步宽度
wField1VFrontPorch	第 1 场的垂直方向前沿宽度
wField1VBackPorch	第 1 场的垂直方向后沿宽度
wField1VActive	第 1 场的垂直方向有效宽度
wField1VTotatHeight	第 1 场的垂直方向的总高度

3.3.23 MWCAP_VIDEO_TIMING_SETTINGS

• 结构原型

```
typedef struct MWCAPVIDEO_TIMING_SETTINGS {
    WORD wAspectX;
    WORD wAspectY;
    WORD x;
    WORD y;
    WORD cx;
    WORD cy;
    WORD cxTotal;
```

```

    BYTE byClampPos;
} MWCAP_VIDEO_TIMING_SETTINGS;

```

- 参数说明

名称	注释
wAspectX	视频宽高比中的宽度
wAspectY	视频宽高比中的高度
x	水平方向的起始位置
y	竖直方向的起始位置
cx	宽度
cy	高度
cxTotal	水平方向总的宽度
byClampPos	钳制位置

3.3.24 MWCAP_SDI_SPECIFIC_STATUS

- 结构原型

```

typedef struct MWCAPSDI_SPECIFIC_STATUS {
    SDI_TYPE sdiType;
    SDI_SCANNING_FORMAT sdiScanningFormat;
    SDI_BIT_DEPTH sdiBitDepth;
    SDI_SAMPLING_STRUCT sdiSamplingStruct;
    BOOLEAN bST352DataValid;
    DWORD dwST352Data;
} MWCAP_SDI_SPECIFIC_STATUS;

```

- 参数说明

名称	注释
sdiType	SDI 信号类型
sdiScanningFormat	SDI 扫描格式
sdiBitDepth	SDI 位深
sdiSamplingStruct	SDI 采样的数据结构
bST352DataValid	ST352 数据是否有效
dwST352Data	ST352 数据

3.3.25 MWCAP_HDMI_SPECIFIC_STATUS

- 结构原型

```

typedef struct MWCAPHDMI_SPECIFIC_STATUS {
    BOOLEAN bHDMIMode;
    BOOLEAN bHDCP;
    BYTE byBitDepth;
    HDMI_PIXEL_ENCODING pixelEncoding;
    BYTE byVIC;
    BOOLEAN bITContent;
    BOOLEAN b3DFormat;
    BYTE by3DStructure;
    BYTE bySideBySideHalfSubSampling;
    MWCAP_HDMI_VIDEO_TIMING videoTiming;
} MWCAP_HDMI_SPECIFIC_STATUS;

```

- 参数说明

--	--

名称	注释
bHDMIMode	是否是 HDMI 模式
bHDCP	是否是 HDCP 加密信号
byBitDepth	位深
pixelEncoding	像素格式
byVIC	EDID 中的视频识别码，用于指定标准的分辨率和时序
bITContent	IT Content 标志位
b3DFormat	是否是 3D 格式
by3DStructure	3D 结构
bySideBySideHalfSubSampling	并排半采样
videoTiming	视频时序，参见 3.3.22 MWCAP_HDMI_VIDEO_TIMING

3.3.26 MWCAP_COMPONENT_SPECIFIC_STATUS

- 结构原型

```
typedef struct MWCAPCOMPONENT_SPECIFIC_STATUS {
    MWCAP_VIDEO_SYNC_INFO syncInfo;
    BOOLEAN bTriLevelSync;
    MWCAP_VIDEO_TIMING videoTiming;
    MWCAP_VIDEO_TIMING_SETTINGS videoTimingSettings;
} MWCAP_COMPONENT_SPECIFIC_STATUS;
```

- 参数说明

名称	注释
syncInfo	视频同步信息
bTriLevelSync	是否是三级同步
videoTiming	视频时序信息
videoTimingSettings	视频时序设置，参见 3.3.23 MWCAP_VIDEO_TIMING_SETTINGS

3.3.27 MWCAP_CVBS_YC_SPECIFIC_STATUS

- 结构原型

```
typedef struct MWCAPCVBS_YC_SPECIFIC_STATUS {
    MWCAP_SD_VIDEO_STANDARD standard;
    BOOLEAN b50Hz;
} MWCAP_CVBS_YC_SPECIFIC_STATUS;
```

- 参数说明

名称	注释
standard	定义使用的视频标准，参见 3.4.8 MWCAP_SD_VIDEO_STANDARD
b50Hz	扫描频率是否是 50Hz

3.3.28 MWCAP_INPUT_SPECIFIC_STATUS

- 结构原型

```
typedef struct MWCAPINPUT_SPECIFIC_STATUS {
    BOOLEAN bValid;
```

```

DWORD dwVideoInputType;
union {
    MWCAP_SDI_SPECIFIC_STATUS sdiStatus;
    MWCAP_HDMI_SPECIFIC_STATUS hdmiStatus;
    MWCAP_COMPONENT_SPECIFIC_STATUS vgaComponentStatus;
    MWCAP_CVBS_YC_SPECIFIC_STATUS cvbsYcStatus;
};
} MWCAP_INPUT_SPECIFIC_STATUS;

```

- 参数说明

名称	注释
bValid	定义该输入信号状态是否有效
dwVideoInputType	输入视频信号的类型
sdiStatus	SDI 信号状态，参见 3.3.24 MWCAP_SDI_SPECIFIC_STATUS
hdmiStatus	HDMI 信号状态，参见 3.3.25 MWCAP_HDMI_SPECIFIC_STATUS
vgaComponentStatus	VGA component 信号状态，参见 3.3.26 MWCAP_COMPONENT_SPECIFIC_STATUS
cvbsYcStatus	CVBS-YC 信号状态，参见 3.3.27 MWCAP_CVBS_YC_SPECIFIC_STATUS

3.3.29 MWCAP_VIDEO_SIGNAL_STATUS

- 结构原型

```

typedef struct MWCAPVIDEO_SIGNAL_STATUS {
    MWCAP_VIDEO_SIGNAL_STATE state;
    int x;
    int y;
    int cx;
    int cy;
    int cxTotal;
    int cyTotal;
    BOOLEAN bInterlaced;
    DWORD dwFrameDuration;
    int nAspectX;
    int nAspectY;
    BOOLEAN bSegmentedFrame;
    MWCAP_VIDEO_FRAME_TYPE frameType;
    MWCAP_VIDEO_COLOR_FORMAT colorFormat;
    MWCAP_VIDEO_QUANTIZATION_RANGE quantRange;
    MWCAP_VIDEO_SATURATION_RANGE satRange;
} MWCAP_VIDEO_SIGNAL_STATUS;

```

- 参数说明

名称	注释
state	定义该视频信号的可访问情况，参见 3.4.9 MWCAP_VIDEO_SIGNAL_STATE
x	水平方向的起始位置
y	垂直方向的起始位置
cx	视频画面的宽度
cy	视频画面的高度
cxTotal	总宽度

cyTotal	总高度
bInterlaced	是否是交错信号
dwFrameDuration	视频帧的间隔时间
nAspectX	视频宽高比的宽度
nAspectY	视频宽高比的高度
bSegmentedFrame	是否是分段的帧
frameType	视频帧的类型，参见 3.4.10 MWCAP_VIDEO_FRAME_TYPE
colorFormat	视频颜色格式，参见 3.4.4 MWCAP_VIDEO_COLOR_FORMAT
quantRange	量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
satRange	饱和度范围，参见 3.4.11 MWCAP_VIDEO_SATURATION_RANGE

3.3.30 MWCAP_AUDIO_SIGNAL_STATUS

- 结构原型

```
typedef struct MWCAPAUDIO_SIGNAL_STATUS {
    WORD ChannelValid;
    BOOLEAN bLPCM;
    BYTE cBitsPerSample;
    DWORD dwSampleRate;
    BOOLEAN bChannelStatusValid;
    IEC60958_CHANNEL_STATUS channelStatus;
} MWCAP_AUDIO_SIGNAL_STATUS;
```

- 参数说明

名称	注释
ChannelValid	有效声道的 mask 值，最低位表示第 1 和第 2 声道是否有效，第二位表示 3 和 4 声道是否有效，第三位表示 5 和 6 声道是否有效，第四位表示 7 和 8 声道是否有效
bLPCM	表示是否是 LPCM 格式
cBitsPerSample	每个音频采样的位深
dwSampleRate	采样率
bChannelStatusValid	表示以下的通道状态是否有效
channelStatus	该音频通道的状态

3.3.31 HDMI_INFOFRAME_HEADER

- 结构原型

```
typedef struct HDMIINFOFRAME_HEADER {
    BYTE byPacketType;
    BYTE byVersion;
    BYTE byLength : 5;
    BYTE byReservedZero : 3;
} HDMI_INFOFRAME_HEADER;
```

- 参数说明

名称	注释

byPacketType	返回 HDMI 信息帧的类型
byVersion	返回 HDMI 信息帧的版本
byLength	返回 HDMI 信息帧的大小（单位:字节）
byReservedZero	保留的字节大小

3.3.32 HDMI_INFOFRAME_PACKET

- 结构原型

```
typedef struct HDMIINFOFRAME_PACKET {
    HDMI_INFOFRAME_HEADER header;
    BYTE byChecksum;

    union {
        BYTE abyPayload[27];
        HDMI_AVI_INFOFRAME_PAYLOAD aviInfoFramePayload;
        HDMI_AUDIO_INFOFRAME_PAYLOAD audioInfoFramePayload;
        HDMI_SPD_INFOFRAME_PAYLOAD spdInfoFramePayload;
        HDMI_VS_INFOFRAME_PAYLOAD vsInfoFramePayload;
    };
} HDMI_INFOFRAME_PACKET;
```

- 参数说明

名称	注释
header	HDMI 信息帧头信息，参见 3.3.31 HDMI_INFOFRAME_HEADER
byChecksum	HDMI 信息帧总和检验码
abyPayload	HDMI 信息帧数据
aviInfoFramePayload	HDMI AVI 信息帧数据
audioInfoFramePayload	HDMI AUDIO 信息帧数据
spdInfoFramePayload	HDMI SPD 信息帧数据
vsInfoFramePayload	HDMI VS 信息帧数据

3.3.33 MWCAP_VIDEO_ASPECT_RATIO

- 结构原型

```
typedef struct MWCAPVIDEO_ASPECT_RATIO {
    int nAspectX;
    int nAspectY;
} MWCAP_VIDEO_ASPECT_RATIO;
```

- 参数说明

名称	注释
nAspectX	输入视频宽高比的宽
nAspectY	输入视频宽高比的高

3.3.34 MWCAP_VIDEO_CREATE_IMAGE

- 结构原型

```
typedef struct MWCAPVIDEO_CREATE_IMAGE {
    WORD cx;
    WORD cy;
```

```

    MWCAP_PTR pvlImage;
} MWCAP_VIDEO_CREATE_IMAGE;

```

- 参数说明

名称	注释
cx	创建 OSD 图像的宽度
cy	创建 OSD 图像的高度
pvlImage	返回创建的 OSD 图像的句柄

3.3.35 MWCAP_VIDEO_IMAGE_REF

- 结构原型

```

typedef struct MWCAPVIDEO_IMAGE_REF {
    MWCAP_PTR pvlImage;
    long nRefCount;
} MWCAP_VIDEO_IMAGE_REF;

```

- 参数说明

名称	注释
pvlImage	OSD 图像的句柄
nRefCount	返回 OSD 图像的引用次数

3.3.36 MWCAP_VIDEO_UPLOAD_IMAGE

- 结构原型

```

typedef struct MWCAPVIDEO_UPLOAD_IMAGE {
    MWCAP_PTR pvDestImage;
    MWCAP_VIDEO_COLOR_FORMAT cfDest;
    WORD xDest;
    WORD yDest;
    WORD cxDest;
    WORD cyDest;
    MWCAP_VIDEO_QUANTIZATION_RANGE quantRangeDest;
    MWCAP_VIDEO_SATURATION_RANGE satRangeDest;

    BOOLEAN bSrcPhysicalAddress;
    union {
        MWCAP_PTR pvSrcFrame;
        LARGE_INTEGER liSrcPhysicalAddress;
    };

    DWORD cbSrcFrame;
    DWORD cbSrcStride;

    WORD cxSrc;
    WORD cySrc;
    BOOLEAN bSrcBottomUp;
    BOOLEAN bSrcPixelAlpha;
    BOOLEAN bSrcPixelXBGR;
} MWCAP_VIDEO_UPLOAD_IMAGE;

```

- 参数说明

名称	注释
pvDestImage	目标图像句柄

cfDest	色彩格式标准，参见 3.4.4 MWCAP_VIDEO_COLOR_FORMAT
xDest	目标区域的水平方向起始位置
yDest	目标区域的垂直方向起始位置
cxDest	目标区域的宽度
cyDest	目标区域的高度
quantRangeDest	目标图像的量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
satRangeDest	目标图像的饱和度范围，参见 3.4.11 MWCAP_VIDEO_SATURATION_RANGE
bSrcPhysicalAddress	原视频是否使用的物理地址
pvSrcFrame	原视频数据的内存地址
liSrcPhysicalAddress	原视频数据的物理地址
cbSrcFrame	原视频数据的总长度
cbSrcStride	原视频数据的步长
cxSrc	原视频的宽度
cySrc	原视频的高度
bSrcBottomUp	原视频是否上下颠倒
bSrcPixelAlpha	原视频像素是否具有 Alpha 分量
bSrcPixelXBGR	原视频像素是否是 XBGR 格式

3.3.37 MWCAP_VIDEO_TIMING

- 结构原型

```
typedef struct MWCAPVIDEO_TIMING {
    DWORD dwType;
    DWORD dwPixelClock;
    BOOLEAN bInterlaced;
    BYTE bySyncType;
    BOOLEAN bHSPolarity;
    BOOLEAN bVSPolarity;
    WORD wHActive;
    WORD wHFrontPorch;
    WORD wHSyncWidth;
    WORD wHBackPorch;
    WORD wVActive;
    WORD wVFrontPorch;
    WORD wVSyncWidth;
    WORD wVBackPorch;
} MWCAP_VIDEO_TIMING;;
```

- 参数说明

名称	注释
dwType	视频通道时序类型，参见 3.4.14 MWCAP_VIDEO_TIMING_TYPE
dwPixelClock	视频通道时序的像素时钟
bInterlaced	视频时序是否交错
bySyncType	视频时序的同步类型
bHSPolarity	视频通道时序的水平方向同步极性是否为正
bVSPolarity	视频通道时序的垂直方向同步极性是否为正

wHActive	视频通道时序的水平方向有效时间
wHFrontPorch	视频通道时序的水平方向前沿
wHSyncWidth	视频通道时序的水平方向同步宽度
wHBackPorch	视频通道时序的水平方向后沿
wVActive	视频通道时序的垂直方向有效时间
wVFrontPorch	视频通道时序的垂直方向前沿
wVSyncWidth	视频通道时序的垂直方向同步宽度
wVBackPorch	视频通道时序的垂直方向后沿

3.3.38 MWCAP_VIDEO_CUSTOM_TIMING

- 结构原型

```
typedef struct MWCAPVIDEO_CUSTOM_TIMING {
    MWCAP_VIDEO_SYNC_INFO syncInfo;
    MWCAP_VIDEO_TIMING_SETTINGS videoTimingSettings;
} MWCAP_VIDEO_CUSTOM_TIMING;
```

- 参数说明

名称	注释
syncInfo	视频同步信息
videoTimingSettings	视频时序设置，参见 3.3.23 MWCAP_VIDEO_TIMING_SETTINGS

3.3.39 MWCAP_VIDEO_CUSTOM_TIMING_PAR

- 结构原型

```
typedef struct MWCAPVIDEO_CUSTOM_TIMING_PAR {
    long count;
    MWCAP_VIDEO_CUSTOM_TIMING timings[];
} MWCAP_VIDEO_CUSTOM_TIMING_PAR;
```

- 参数说明

名称	注释
count	自定义视频时序个数
timings	自定义视频时钟接口

3.3.40 MWCAP_VIDEO_CUSTOM_RESOLUTION_PAR

- 结构原型

```
typedef struct MWCAPVIDEO_CUSTOM_RESOLUTION_PAR {
    long count;
    MWCAP_SIZE resolutions[];
} MWCAP_VIDEO_CUSTOM_RESOLUTION_PAR;
```

- 参数说明

名称	注释
count	自定义视频时序个数
resolutions	自定义分辨率接口

3.3.41 MWCAP_EDID_DATA

• 结构原型

```
typedef struct MWCAPEDID_DATA {
    int size;
    unsigned char *data;
} MWCAP_EDID_DATA;
```

• 参数说明

名称	注释
size	EDID 数据大小
data	EDID 数据

3.3.42 MWCAP_VIDEO_CONNECTION_FORMAT

• 结构原型

```
typedef struct MWCAPVIDEO_CONNECTION_FORMAT {
    BOOLEAN bConnected;
    LONG cx;
    LONG cy;
    DWORD dwFrameDuration;
    DWORD dwFOURCC;

    int nAspectX;
    int nAspectY;
    MWCAP_VIDEO_COLOR_FORMAT colorFormat;
    MWCAP_VIDEO_QUANTIZATION_RANGE quantRange;
    MWCAP_VIDEO_SATURATION_RANGE satRange;
} MWCAP_VIDEO_CONNECTION_FORMAT;
```

• 参数说明

名称	注释
bConnected	表示是否连接
cx	视频画面的宽度
cy	视频画面的高度
dwFrameDuration	视频帧的间隔时间
dwFOURCC	色彩空间
nAspectX	视频宽高比的宽度
nAspectY	视频宽高比的高度
colorFormat	视频颜色格式标准，参见 3.4.4 MWCAP_VIDEO_COLOR_FORMAT
quantRange	量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
satRange	饱和度范围，参见 3.4.11 MWCAP_VIDEO_SATURATION_RANGE

3.3.43 MWCAP_VIDEO_PROCESS_SETTINGS

• 结构原型

```
typedef struct MWCAPVIDEO_PROCESS_SETTINGS {
    DWORD dwProcessSwitchs;
```

```

RECT rectSource;
int nAspectX;
int nAspectY;
BOOLEAN bLowLatency;
MWCAP_VIDEO_COLOR_FORMAT colorFormat;
MWCAP_VIDEO_QUANTIZATION_RANGE quantRange;
MWCAP_VIDEO_SATURATION_RANGE satRange;
MWCAP_VIDEO_DEINTERLACE_MODE deinterlaceMode;
MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE aspectRatioConvertMode;
} MWCAP_VIDEO_PROCESS_SETTINGS;

```

- 参数说明

名称	注释
dwProcessSwitchs	视频处理的掩码值
rectSource	要处理的源区域
nAspectX	视频宽高比的宽度
nAspectY	视频宽高比的高度
bLowLatency	是否启用低延迟模式
colorFormat	视频颜色格式标准，参见 3.4.4 MWCAP_VIDEO_COLOR_FORMAT
quantRange	量化范围，参见 3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE
satRange	饱和度范围，参见 3.4.11 MWCAP_VIDEO_SATURATION_RANGE
deinterlaceMode	去隔行模式，参见 3.4.6 MWCAP_VIDEO_DEINTERLACE_MODE
aspectRatioConvertMode	宽高比转换模式，参见 3.4.7 MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE

3.3.44 MWCAP_VIDEO_OSD_SETTINGS

- 结构原型

```

typedef struct MWCAPVIDEO_OSD_SETTINGS {
    BOOLEAN bEnable;
    char szPNGFilePath[MAXPATH];
} MWCAP_VIDEO_OSD_SETTINGS;

```

- 参数说明

名称	注释
bEnable	是否使用 OSD 图像
szPNGFilePath	上载的 PNG 图片的地址

3.3.45 MWCAP_VIDEO_OSD_IMAGE

- 结构原型

```

typedef struct MWCAPVIDEO_OSD_IMAGE {
    MWCAP_PTR pvOSDImage;
    RECT aOSDRects[MWCAP_VIDEO_MAX_NUM_OSD_RECTS];
    int cOSDRects;
} MWCAP_VIDEO_OSD_IMAGE;

```

- 参数说明

--	--

名称	注释
pvOSDImage	OSD 图片的句柄
aOSDRects	需要合成该图片的目标区域
cOSDRects	需要合成该图片的目标区域数量

3.4 I/O Control 枚举定义

3.4.1 MW_FAMILY_ID

- 枚举原型

```
typedef enum MWFAMILY_ID {
    MW_FAMILY_ID_PRO_CAPTURE = 0x00,
    MW_FAMILY_ID_VALUE_CAPTURE = 0x01,
    MW_FAMILY_ID_USB_CAPTURE = 0x02
} MW_FAMILY_ID;
```

- 常量定义

名称	注释
MW_FAMILY_ID_PRO_CAPTURE	OSD 使用 PCI-e 接口和计算机相连接的采集设备
MW_FAMILY_ID_VALUE_CAPTURE	使用 PCI-e 接口和计算机相连接的采集设备,和 Pro Capture 系列相比,在软硬件上做了一定的简化, 包括不支持多流输出、不使用板载缓存等
MW_FAMILY_ID_USB_CAPTURE	使用 USB 接口和计算机相连接的采集设备

3.4.2 MWCAP_PRODUCT_ID

- 枚举原型

```
typedef enum MWCAPPRODUCT_ID {
    MWCAP_PRODUCT_ID_PRO_CAPTURE_AIO = 0x00000102,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_DVI = 0x00000103,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_HDMI = 0x00000104,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_SDI = 0x00000105,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_SDI = 0x00000106,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_DVI = 0x00000107,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_HDMI = 0x00000108,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_QUAD_SDI = 0x00000109,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_QUAD_HDMI = 0x00000110,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_MINI_HDMI = 0x00000111,
    MWCAP_PRODUCT_ID_PRO_CAPTURE_HDMI_4K = 0x00000112
} MWCAP_PRODUCT_ID;
```

- 常量定义

名称	注释
MWCAP_PRODUCT_ID_PRO_CAPTURE_AIO	使用 PCI-e 接口和计算机相连接的采集设备
MWCAP_PRODUCT_ID_PRO_CAPTURE_DVI	具有 PCI-e 主机接口和 DVI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_HDMI	具有 PCI-e 主机接口和 HDMI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE	

MWCAP_PRODUCT_ID_PRO_CAPTURE_SDI	具有 PCI-e 主机接口和 SDI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_SDI	具有 PCI-e 主机接口和两路 SDI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_DVI	具有 PCI-e 主机接口和两路 DVI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_DUAL_HDMI	具有 PCI-e 主机接口和两路 HDMI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_QUAD_SDI	具有 PCI-e 主机接口和四路 SDI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_QUAD_HDMI	具有 PCI-e 主机接口和四路 HDMI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_MINI_HDMI	具有 PCI-e 主机接口和迷你 HDMI 输入接口的采集卡
MWCAP_PRODUCT_ID_PRO_CAPTURE_HDMI_4K	具有 PCI-e 主机接口和 4K HDMI 输入接口的采集卡

3.4.3 MWCAP_VIDEO_FRAME_STATE

- 枚举原型

```
typedef enum MWCAP_VIDEO_FRAME_STATE {
    MWCAP_VIDEO_FRAME_STATE_INITIAL,
    MWCAP_VIDEO_FRAME_STATE_F0_BUFFERING,
    MWCAP_VIDEO_FRAME_STATE_F1_BUFFERING,
    MWCAP_VIDEO_FRAME_STATE_BUFFERED
} MWCAP_VIDEO_FRAME_STATE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_FRAME_STATE_INITIAL	初始状态
MWCAP_VIDEO_FRAME_STATE_F0_BUFFERING	正在缓冲第 0 场
MWCAP_VIDEO_FRAME_STATE_F1_BUFFERING	正在缓冲第 1 场
MWCAP_VIDEO_FRAME_STATE_BUFFERED	视频帧已经缓冲好

3.4.4 MWCAP_VIDEO_COLOR_FORMAT

- 枚举原型

```
typedef enum MWCAP_VIDEO_COLOR_FORMAT {
    MWCAP_VIDEO_COLOR_FORMAT_UNKNOWN = 0x00,
    MWCAP_VIDEO_COLOR_FORMAT_RGB = 0x01,
    MWCAP_VIDEO_COLOR_FORMAT_YUV601 = 0x02,
    MWCAP_VIDEO_COLOR_FORMAT_YUV709 = 0x03,
    MWCAP_VIDEO_COLOR_FORMAT_YUV2020 = 0x04,
    MWCAP_VIDEO_COLOR_FORMAT_YUV2020C = 0x05
} MWCAP_VIDEO_COLOR_FORMAT;
```

- 常量定义

名称	注释
MWCAP_VIDEO_COLOR_FORMAT_UNKNOWN	无法识别的色彩格式
MWCAP_VIDEO_COLOR_FORMAT_RGB	RGB 色彩格式
MWCAP_VIDEO_COLOR_FORMAT_YUV601	YUV601 色彩格式
MWCAP_VIDEO_COLOR_FORMAT_YUV709	YUV709 色彩格式
MWCAP_VIDEO_COLOR_FORMAT_YUV2020	YUV2020 色彩格式
MWCAP_VIDEO_COLOR_FORMAT_YUV2020C	YUV2020C 色彩格式

3.4.5 MWCAP_VIDEO_QUANTIZATION_RANGE

- 枚举原型

```
typedef enum MWCAPVIDEO_QUANTIZATION_RANGE {
    MWCAP_VIDEO_QUANTIZATION_UNKNOWN = 0x00,
    MWCAP_VIDEO_QUANTIZATION_FULL = 0x01,
    MWCAP_VIDEO_QUANTIZATION_LIMITED = 0x02
} MWCAP_VIDEO_QUANTIZATION_RANGE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_QUANTIZATION_UNKNOWN	无法识别的视频量化范围
MWCAP_VIDEO_QUANTIZATION_FULL	全域视频量化范围，在该范围下，8 位黑白颜色范围为 0-255
MWCAP_VIDEO_QUANTIZATION_LIMITED	局域视频量化范围，在该范围下，8 位黑白颜色范围为 16-235

3.4.6 MWCAP_VIDEO_DEINTERLACE_MODE

- 枚举原型

```
typedef enum MWCAPVIDEO_DEINTERLACE_MODE {
    MWCAP_VIDEO_DEINTERLACE_WEAVE = 0x00,
    MWCAP_VIDEO_DEINTERLACE_BLEND = 0x01,
    MWCAP_VIDEO_DEINTERLACE_TOP_FIELD = 0x02,
    MWCAP_VIDEO_DEINTERLACE_BOTTOM_FIELD = 0x03
} MWCAP_VIDEO_DEINTERLACE_MODE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_DEINTERLACE_WEAVE	不进行去隔行
MWCAP_VIDEO_DEINTERLACE_BLEND	使用混合模式去隔行，默认去隔行方式
MWCAP_VIDEO_DEINTERLACE_TOP	

_FIELD	通过只取顶场数据去隔行
MWCAP_VIDEO_DEINTERLACE_BOTTOM_FIELD	通过只取底场数据去隔行

3.4.7 MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE

- 枚举原型

```
typedef enum MWCAPVIDEO_ASPECT_RATIO_CONVERT_MODE {
    MWCAP_VIDEO_ASPECT_RATIO_IGNORE = 0x00,
    MWCAP_VIDEO_ASPECT_RATIO_CROPPING = 0x01,
    MWCAP_VIDEO_ASPECT_RATIO_PADDING = 0x02
} MWCAP_VIDEO_ASPECT_RATIO_CONVERT_MODE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_ASPECT_RATIO_IGNORE	忽略宽高比
MWCAP_VIDEO_ASPECT_RATIO_CROPPING	通过裁剪转换宽高比
MWCAP_VIDEO_ASPECT_RATIO_PADDING	通过填补转换宽高比

3.4.8 MWCAP_SD_VIDEO_STANDARD

- 枚举原型

```
typedef enum MWCAPSD_VIDEO_STANDARD {
    MWCAP_SD_VIDEO_NONE,
    MWCAP_SD_VIDEO_NTSC_M,
    MWCAP_SD_VIDEO_NTSC_433,
    MWCAP_SD_VIDEO_PAL_M,
    MWCAP_SD_VIDEO_PAL_60,
    MWCAP_SD_VIDEO_PAL_COMBN,
    MWCAP_SD_VIDEO_PAL_BGHID,
    MWCAP_SD_VIDEO_SECAM,
    MWCAP_SD_VIDEO_SECAM_60
} MWCAP_SD_VIDEO_STANDARD;
```

- 常量定义

名称	注释
MWCAP_SD_VIDEO_NONE	无
MWCAP_SD_VIDEO_NTSC_M	使用 NTSC_M 标准
MWCAP_SD_VIDEO_NTSC_433	使用 NTSC_433 标准
MWCAP_SD_VIDEO_PAL_M	使用 PAL_M 标准
MWCAP_SD_VIDEO_PAL_60	使用 PAL_60 标准
MWCAP_SD_VIDEO_PAL_COMBN	使用 PAL_COMBN 标准
MWCAP_SD_VIDEO_PAL_BGHID	使用 PAL_BGHID 标准
MWCAP_SD_VIDEO_SECAM	使用 SECAM 标准
MWCAP_SD_VIDEO_SECAM_60	使用 SECAM_60 标准

3.4.9 MWCAP_VIDEO_SIGNAL_STATE

- 枚举原型

```
typedef enum MWCAPVIDEO_SIGNAL_STATE {  
    MWCAP_VIDEO_SIGNAL_NONE,  
    MWCAP_VIDEO_SIGNAL_UNSUPPORTED,  
    MWCAP_VIDEO_SIGNAL_LOCKING,  
    MWCAP_VIDEO_SIGNAL_LOCKED  
} MWCAP_VIDEO_SIGNAL_STATE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_SIGNAL_NONE	无视频信号
MWCAP_VIDEO_SIGNAL_UNSUPPORTED	无效的视频信号，采集设备检测到有输入信号，但无法对该信号锁定识别
MWCAP_VIDEO_SIGNAL_LOCKING	正在锁定视频信号，视频信号的状态有效，但还没有锁定
MWCAP_VIDEO_SIGNAL_LOCKED	已锁定的视频信号，采集设备能够采集当前输入信号

3.4.10 MWCAP_VIDEO_FRAME_TYPE

- 枚举原型

```
typedef enum MWCAPVIDEO_FRAME_TYPE {  
    MWCAP_VIDEO_FRAME_2D = 0x00,  
    MWCAP_VIDEO_FRAME_3D_TOP_AND_BOTTOM_FULL = 0x01,  
    MWCAP_VIDEO_FRAME_3D_TOP_AND_BOTTOM_HALF = 0x02,  
    MWCAP_VIDEO_FRAME_3D_SIDE_BY_SIDE_FULL = 0x03,  
    MWCAP_VIDEO_FRAME_3D_SIDE_BY_SIDE_HALF = 0x04  
} MWCAP_VIDEO_FRAME_TYPE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_FRAME_2D	2 维视频帧
MWCAP_VIDEO_FRAME_3D_TOP_AND_BOTTOM_FULL	分顶场和底场的全景 3 维视频帧
MWCAP_VIDEO_FRAME_3D_TOP_AND_BOTTOM_HALF	分顶场和底场的半景 3 维视频帧
MWCAP_VIDEO_FRAME_3D_SIDE_BY_SIDE_FULL	并排全景的 3 维视频帧
MWCAP_VIDEO_FRAME_3D_SIDE_BY_SIDE_HALF	并排半景的 3 维视频帧

3.4.11 MWCAP_VIDEO_SATURATION_RANGE

- 枚举原型

```
typedef enum MWCAPVIDEO_SATURATION_RANGE {  
    MWCAP_VIDEO_SATURATION_UNKNOWN = 0x00,  
    MWCAP_VIDEO_SATURATION_FULL = 0x01,  
    MWCAP_VIDEO_SATURATION_LIMITED = 0x02,  
    MWCAP_VIDEO_SATURATION_EXTENDED_GAMUT = 0x03  
} MWCAP_VIDEO_SATURATION_RANGE;
```


- 常量定义

名称	注释
MWCAP_VIDEO_SATURATION_UNKNOWN	未知的视频饱和范围
MWCAP_VIDEO_SATURATION_FULL	全域视频饱和范围，在该范围下，8 位黑白颜色范围为 0–255
MWCAP_VIDEO_SATURATION_LIMITED	局域视频饱和范围，在该范围下，8 位黑白颜色范围为 16–235
MWCAP_VIDEO_SATURATION_EXTENDED_GAMUT	扩展局域视频饱和范围，在该范围下，8 位黑白颜色范围为 1–254

3.4.12 MWCAP_HDMI_INFOFRAME_ID

- 枚举原型

```
typedef enum MWCAPHDMI_INFOFRAME_ID {
    MWCAP_HDMI_INFOFRAME_ID_AVI,
    MWCAP_HDMI_INFOFRAME_ID_AUDIO,
    MWCAP_HDMI_INFOFRAME_ID_SPD,
    MWCAP_HDMI_INFOFRAME_ID_MS,
    MWCAP_HDMI_INFOFRAME_ID_VS,
    MWCAP_HDMI_INFOFRAME_ID_ACP,
    MWCAP_HDMI_INFOFRAME_ID_ISRC1,
    MWCAP_HDMI_INFOFRAME_ID_ISRC2,
    MWCAP_HDMI_INFOFRAME_ID_GAMUT,
    MWCAP_HDMI_INFOFRAME_COUNT
} MWCAP_HDMI_INFOFRAME_ID;
```

- 常量定义

名称	注释
MWCAP_HDMI_INFOFRAME_ID_AVI	AVI 信息帧
MWCAP_HDMI_INFOFRAME_ID_AUDIO	Audio 信息帧
MWCAP_HDMI_INFOFRAME_ID_SPD	SPD 信息帧
MWCAP_HDMI_INFOFRAME_ID_MS	MS 信息帧
MWCAP_HDMI_INFOFRAME_ID_VS	VS 信息帧
MWCAP_HDMI_INFOFRAME_ID_ACP	ACP 信息帧
MWCAP_HDMI_INFOFRAME_ID_ISRC1	ISRC1 信息帧
MWCAP_HDMI_INFOFRAME_ID_ISRC2	ISRC2 信息帧
MWCAP_HDMI_INFOFRAME_ID_GAMUT	GAMUT 信息帧
MWCAP_HDMI_INFOFRAME_COUNT	信息帧类型总数

3.4.13 MWCAP_HDMI_INFOFRAME_ID

- 枚举原型

```
typedef enum MWCAPHDMI_INFOFRAME_MASK {
    MWCAP_HDMI_INFOFRAME_MASK_AVI =(1<< MWCAP_HDMI_INFOFRAME_ID_AVI),
```

```

MWCAP_HDMI_INFOFRAME_MASK_AUDIO =(1<< MWCAP_HDMI_INFOFRAME_ID_AUDIO),
MWCAP_HDMI_INFOFRAME_MASK_SPD =(1<< MWCAP_HDMI_INFOFRAME_ID_SPD),
MWCAP_HDMI_INFOFRAME_MASK_MS =(1<< MWCAP_HDMI_INFOFRAME_ID_MS),
MWCAP_HDMI_INFOFRAME_MASK_VS =(1<< MWCAP_HDMI_INFOFRAME_ID_VS),
MWCAP_HDMI_INFOFRAME_MASK_ACP =(1<< MWCAP_HDMI_INFOFRAME_ID_ACP),
MWCAP_HDMI_INFOFRAME_MASK_ISRC1 =(1<< MWCAP_HDMI_INFOFRAME_ID_ISRC1),
MWCAP_HDMI_INFOFRAME_MASK_ISRC2 =(1<< MWCAP_HDMI_INFOFRAME_ID_ISRC2),
MWCAP_HDMI_INFOFRAME_MASK_GAMUT =(1<< MWCAP_HDMI_INFOFRAME_ID_GAMUT)
} MWCAP_HDMI_INFOFRAME_MASK;

```

- 常量定义

名称	注释
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_AVI	AVI 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_AUDIO	Audio 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_SPD	SPD 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_MS	MS 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_VS	VS 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_ACP	ACP 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_ISRC1	ISRC1 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_ISRC2	ISRC2 信息帧
MWCAP_HDMI_INFOFRAME_INFOFRAME_MASK_GAMUT	GAMUT 信息帧

3.4.14 MWCAP_VIDEO_TIMING_TYPE

- 枚举原型

```

typedef enum MWCAP_VIDEO_TIMING_TYPE {
    MWCAP_VIDEO_TIMING_NONE = 0x00000000,
    MWCAP_VIDEO_TIMING_LEGACY = 0x00000001,
    MWCAP_VIDEO_TIMING_DMT = 0x00000002,
    MWCAP_VIDEO_TIMING_CEA = 0x00000004,
    MWCAP_VIDEO_TIMING_GTF = 0x00000008,
    MWCAP_VIDEO_TIMING_CVT = 0x00000010,
    MWCAP_VIDEO_TIMING_CVT_RB = 0x00000020,
    MWCAP_VIDEO_TIMING_FAILSAFE = 0x00002000
} MWCAP_VIDEO_TIMING_TYPE;

```

- 常量定义

名称	注释
MWCAP_VIDEO_TIMING_NONE	视频通道无时序
MWCAP_VIDEO_TIMING_LEGACY	视频通道使用 LEGACY 时序
MWCAP_VIDEO_TIMING_DMT	视频通道使用 DMT 时序

MWCAP_VIDEO_TIMING_CEA	视频通道使用 CEA 时序
MWCAP_VIDEO_TIMING_GTF	视频通道使用 GTF 时序
MWCAP_VIDEO_TIMING_CVT	视频通道使用 CVT 时序
MWCAP_VIDEO_TIMING_CVT_RB	视频通道使用 CVT_RB 时序
MWCAP_VIDEO_TIMING_FAILSAFE	视频通道使用 FAILSAFE 时序

3.4.15 MWCAP_LED_MODE

- 枚举原型

```
typedef enum MWCAPLED_MODE {
    MWCAP_LED_AUTO = 0x00000000,
    MWCAP_LED_OFF = 0x80000000,
    MWCAP_LED_ON = 0x80000001,
    MWCAP_LED_BLINK = 0x80000002,
    MWCAP_LED_DBL_BLINK = 0x80000003,
    MWCAP_LED_BREATH = 0x80000004
} MWCAP_LED_MODE;
```

- 常量定义

名称	注释
MWCAP_LED_AUTO	自动模式
MWCAP_LED_OFF	LED 保持熄灭状态
MWCAP_LED_ON	LED 保持常亮
MWCAP_LED_BLINK	LED 保持闪烁
MWCAP_LED_DBL_BLINK	LED 每闪烁两次停一次
MWCAP_LED_BREATH	呼吸模式

3.4.16 MWCAP_VIDEO_INPUT_TYPE

- 枚举原型

```
typedef enum MWCAPVIDEO_INPUT_TYPE {
    MWCAP_VIDEO_INPUT_TYPE_NONE = 0x00,
    MWCAP_VIDEO_INPUT_TYPE_HDMI = 0x01,
    MWCAP_VIDEO_INPUT_TYPE_VGA = 0x02,
    MWCAP_VIDEO_INPUT_TYPE_SDI = 0x04,
    MWCAP_VIDEO_INPUT_TYPE_COMPONENT = 0x08,
    MWCAP_VIDEO_INPUT_TYPE_CVBS = 0x10,
    MWCAP_VIDEO_INPUT_TYPE_YC = 0x20
} MWCAP_VIDEO_INPUT_TYPE;
```

- 常量定义

名称	注释
MWCAP_VIDEO_INPUT_TYPE_NONE	无信号输入接口
MWCAP_VIDEO_INPUT_TYPE_HDMI	HDMI 信号输入接口
MWCAP_VIDEO_INPUT_TYPE_VGA	VGA 信号输入接口
MWCAP_VIDEO_INPUT_TYPE_SDI	SDI 信号输入接口
MWCAP_VIDEO_INPUT_TYPE_COMPONENT	COMPONENT 信号输入接口

MWCAP_VIDEO_INPUT_TYPE_CVBS	CVBS 信号输入接口
MWCAP_VIDEO_INPUT_TYPE_YC	YC 信号输入接口

3.4.17 MWCAP_AUDIO_INPUT_TYPE

• 枚举原型

```
typedef enum MWCAPAUDIO_INPUT_TYPE {
    MWCAP_AUDIO_INPUT_TYPE_NONE = 0x00,
    MWCAP_AUDIO_INPUT_TYPE_HDMI = 0x01,
    MWCAP_AUDIO_INPUT_TYPE_SDI = 0x02,
    MWCAP_AUDIO_INPUT_TYPE_LINE_IN = 0x04,
    MWCAP_AUDIO_INPUT_TYPE_MIC_IN = 0x08
} MWCAP_AUDIO_INPUT_TYPE;
```

• 常量定义

名称	注释
MWCAP_AUDIO_INPUT_TYPE_NONE	无信号输入接口
MWCAP_AUDIO_INPUT_TYPE_HDMI	HDMI 信号输入接口
MWCAP_AUDIO_INPUT_TYPE_SDI	SDI 信号输入接口
MWCAP_AUDIO_INPUT_TYPE_LINE_IN	线路输入信号输入接口
MWCAP_AUDIO_INPUT_TYPE_MIC_IN	麦克风信号输入接口