Encoding Functions Library

Summary Tables

The following table summaries the functions that belong to Advantech VAPI (Video Application Program Interface) library. Functions are grouped by tasks you might wish to perform.

Data Type Summary

Name	Description
EncRes	The method returned code
PSTREAMREADOPEN	The pointer of the Stream Read Open Callback function
PSTREAMREADPROC	The pointer of the Stream Read Process Callback function
PSTREAMREADCLOSE	The pointer of the Stream Read Close Callback function
PMOTIONDETECTPROC	The pointer of the Motion Detect Process Callback function
PENCFAILPROC	The pointer of the Encoding Failed Process Callback function
STREAMREAD_STRUCT	The structure stores the Stream Read callback function pointers
MDRegion	The structure represents the region of the motion detection

Method Summary

Initiation and release	
Name	**Description**
DVP2420_CreateEncSDKInstence	Creates SDK instance
DVP2420_InitSDK	Initializes the SDK
DVP2420_CloseSDK	Closes up the SDK
DVP2420_InitChips	Initializes the codec chip
DVP2420_ReleaseChips	Releases the codec chip
DVP2420_DownloadFW	Downloads the firmware into the chip

Get the system information	
Name	**Description**
DVP2420_GetChipCount	Gets the number of the chips
DVP2420_GetSDKVersion	Gets the version of the SDK

Video control	
Method	**Description**
DVP2420_StartEncode	Starts to encode the video on a specified channel
DVP2420_StopEncode	Stops to encode the video on a specified

	channel
DVP2420_StartPreview	Starts to preview the video on a specified channel
DVP2420_StopPreview	Stops to preview the video on a specified channel
DVP2420_StartMotionDetect	Starts motion detection on a specified video channel
DVP2420_StopMotionDetect	Stops motion detection on a specified video channel
DVP2420_SetFileName	Sets the storage file name to SDK to generate the corresponding Log file
DVP2420_GetCurImage	Gets the current rendered image
DVP2420_PSMPEG4ToDivx	Converts the PS MPEG4 file to the DIVX format file
DVP2420_IsPSMPEG4Type	Detects if the specified file is PS MPEG4 type

Set/Get encoding parameters	
Method	**Description**
DVP2420_SetSignalType	Sets the signal type of the video stream
DVP2420_SetMPEGType	Sets the MPEG standard of the video stream
DVP2420_SetBrightness	Sets the video brightness
DVP2420_GetBrightness	Gets the video brightness
DVP2420_SetContrast	Sets the video contrast
DVP2420_GetContrast	Gets the video contrast
DVP2420_SetSaturation	Sets the video color saturation
DVP2420_GetSaturation	Gets the video color saturation
DVP2420_SetHue	Sets the video hue value
DVP2420_GetHue	Sets the video hue value
DVP2420_SetGOPType	Sets the number of frames in a GOP and the GOP structure
DVP2420_GetGOPType	Gets the number of frames in GOP and the GOP structure
DVP2420_SetFrameRate	Sets the frame rate of the video stream
DVP2420_GetFrameRate	Gets the frame rate of the video stream
DVP2420_InsertMotionDetectMask	Sets the motion detect regions and thresholds
DVP2420_GetMotionDetectMask	Gets the motion detect regions and thresholds
DVP2420_SetVideoBitrate	Sets the bit rate of the video stream
DVP2420_GetVideoBitrate	Gets the bit rate of the video stream
DVP2420_SetAudioBitrate	Sets the bit rate of the audio stream
DVP2420_GetAudioBitrate	Gets the bit rate of the audio stream
DVP2420_GPIOSetData	Sets the value of the GPIO
DVP2420_GPIOGetData	Gets the value of the GPIO

Register callback function	
Method	**Description**
DVP2420_RegisterStreamReadCB	Registers the StreamRead callback function
DVP2420_RegisterMotionDetectCB	Registers the MotionDetect callback function
DVP2420_RegEncFailCB	Registers the EncFail callback function

Decoding Functions Library

Summary Tables

The following table summaries the functions that belong to Advantech VAPI (Video Application Program Interface) library. Functions are grouped by tasks you might wish to perform.

Data Type Summary

Name	Description
DecRes	The method returned code
PDECENDOFFILEPROC	The pointer of the Decoding EOF Process Callback function
PDECFAILPROC	The pointer of the Decoding Failed Process Callback function

Method Summary

Initiation and release	
Name	**Description**
DVP2420_CreateDecSDKInstence	Creates SDK instance
DVP2420_InitSDK	Initializes the SDK
DVP2420_CloseSDK	Closes up the SDK
DVP2420_InitChips	Initializes the codec chip
DVP2420_ReleaseChips	Releases the codec chip
DVP2420_DownloadFW	Downloads the firmware into the chip

Get the system information	
Name	**Description**
DVP2420_GetChipCount	Gets the number of the chips
DVP2420_GetSDKVersihon	Gets the version of the SDK
DVP2420_GetCurrentFrameNum	Gets the current frame number
DVP2420_GetFileTotalFrames	Gets the number of the total frames in the video file
DVP2420_GetPlayedTime	Gets the current video time
DVP2420_GetFileTime	Gets the total video file time

Video control	
Method	**Description**
DVP2420_StartDecode	Starts to decode the video on a specified channel
DVP2420_StopDecode	Stops to decode the video on a specified channel
DVP2420_Pause	Pauses or continues to play the video
DVP2420_Fast	Speeds up to play the video
DVP2420_Rewind	Rewinds to play the video
DVP2420_SingleStep	Steps forward one frame of the video

DVP2420_SetPlayPos	Seeks the video to the specified video file time

Set/Get decoding parameters	
Method	**Description**
DVP2420_SetSignalType	Sets the signal type of the video stream
DVP2420_SetMPEGType	Sets the MPEG standard of the video stream

Register callback function	
Method	**Description**
DVP2420_RegDecEndOfFileCB	Registers the DecEndOfFile callback function
DVP2420_RegDecFailCB	Registers the DecFail callback function

Encoding Functions Reference

Data Type

EncRes

Syntax

```
typedef enum
{
    ENC_SUCCEEDED       = 1,
    ENC_FAILED          = 0,
    ENC_SDKINITFAILED   = -1,
    ENC_DEVINITFAILED   = -2,
    ENC_PARAMERROR      = -3,
    ENC_CHNUMERROR      = -4
} EncRes;
```

Description

The method returned code.

PSTREAMREADOPEN

Syntax
typedef void (*PSTREAMREADOPEN)(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
None.

Description
The pointer of the Stream Read Open Callback function. This callback function is called when the stream read process starts.

See Also
STREAMREAD_STRUCT

PSTREAMREADPROC

Syntax

typedef void (*PSTREAMREADPROC)(int nChNum, BYTE *pStreamBuf, UINT32 nBytesToTransfer, BOOL bIFrame, UINT32 nIFrameOffset)

Parameters

nChNum:	Specifies the channel ID number.
pStreamBuf:	A byte point of the data buffer that stores the encoded data.
nBytesToTransfer:	Specifies the size of the encoded data.
bIFrame:	Specifies if there is any I-frame in this encoded data.
nIFrameOffset:	Specifies the number of bytes from the start of the data buffer to first I-frame start code.

Return Value

None.

Description

The pointer of the Stream Read Process Callback function. This callback function is called when the encoded stream is read.

See Also

STREAMREAD_STRUCT

PSTREAMREADCLOSE

Syntax
typedef void (*PSTREAMREADCLOSE)(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
None.

Description
The pointer of the Stream Read Close Callback function. This callback function is called when the stream read process finishes.

See Also
STREAMREAD_STRUCT

PMOTIONDETECTPROC

Syntax
typedef void (*PMOTIONDETECTPROC)(int nChNum, int nRegionNum, unsigned short mbCount)

Parameters
nChNum:	Specifies the channel ID number.
nRegionNum:	Specifies the ID number of the video region.
mbCount:	Specifies the number of the macroblocks that there are motions being detected.

Return Value
None.

Description
The pointer of the Motion Detection Callback function. This callback function is called when there are the motions being detected in the specified video region.

See Also
DVP2420_RegisterMotionDetectCB

PENCFAILPROC

Syntax
typedef void (*PENCFAILPROC)(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
None.

Description
The Encoding Fail Process Callback function. This callback function is called when encoding process fails.

See Also
DVP2420_RegEncFailCB

Struct MDRegion

```
typedef struct {
    int enable;
    int left;
    int right;
    int top;
    int bottom;
}MDRegion;
```

Parameters:

enable:	Used to enable/disable motion detection.
left:	Define the left most macroblock number of the horizontal coordinates of the region.
right:	Define the right most macroblock number of the horizontal coordinates of the region.
top:	Define the top most macroblock number of the horizontal coordinates of the region.
bottom:	Define the bottom most macroblock number of the horizontal coordinates of the region.

Description

This structure stores the settings of the motion detection region. The unit of the size of the motion detect region is macroblock.

See Also

DVP2420_InsertMotionDetectMask
DVP2420_GetMotionDetectMask

Struct STREAMREAD_STRUCT

```
typedef struct
{
    void (*PSTREAMREADOPEN)(int nChNum);
    void (*PSTREAMREADPROC)(int nChNum, BYTE *StreamBuf,
    UINT32 bytes_to_transfer , int contain_iframe, UINT32
    iframe_offset);
    void (*PSTREAMREADCLOSE)(int nChNum);
}STREAMREAD_STRUCT;
```

Parameters:

PSTREAMREADOPEN:	The pointer of the Stream Read Open Callback function. This callback function is called when the stream read process starts.
PSTREAMREADPROC:	The pointer of the Stream Read Process Callback function. This callback function is called when the encoded stream is read.
PSTREAMREADCLOSE:	The pointer of the Stream Read Close Callback function. This callback function is called when the stream read process finishes.

Description

This structure stores the pointer of the Stream Read Callback function.

See Also

PSTREAMREADOPEN
PSTREAMREADPROC
PSTREAMREADCLOSE
DVP2420_RegisterStreamReadCB

Method

DVP2420_CreateEncSDKInstence

Syntax

int DVP2420_CreateEncSDKInstence (void **pp)

Parameters

pp:	A pointer to the SDK instance.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_PARAMERROR:	Parameter error.

Description

This function creates SDK instance.

See Also

DVP2420_InitSDK

Syntax
int DVP2420_InitSDK ()

Parameters
None.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.

Description
This function initializes the SDK.

See Also
DVP2420_CloseSDK

DVP2420_CloseSDK

Syntax
int DVP2420_CloseSDK ()

Parameters
None.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.

Description
This function closes up the SDK.

See Also
DVP2420_InitSDK

DVP2420_InitChips

Syntax
int DVP2420_InitChips(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function initializes the codec chip on a specified channel.

See Also
DVP2420_ReleaseChips

DVP2420_ReleaseChips

Syntax
int DVP2420_ReleaseChips(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function releases the codec chip on a specified channel.

See Also
DVP2420_InitChips

DVP2420_DownloadFW

Syntax
int DVP2420_DownloadFW(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function downloads the firmware into the chip on a specified channel. Two firmware files boot.sre and pscodec.sre will be used by the library. Notes: Don't download the firmware into the chip when the decoding process is running in the chip.

See Also

DVP2420_GetChipCount

Syntax
int DVP2420_GetChipCount(int *pChipCnt)

Parameters

pChipCnt:	An integer pointer to store the returned number of the chips.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the number of the codec chips.

See Also

DVP2420_GetSDKVersion

Syntax
float DVP2420_GetSDKVersion()

Parameters
None.

Return Value
If function succeeded, it returns the version of the SDK. Otherwise, it returns -1.

Description
This function gets the version of the SDK.

See Also

DVP2420_StartEncode

Syntax
int DVP2420_StartEncode(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function starts to encode the video on a specified channel.

See Also
DVP2420_StopEncode

DVP2420_StopEncode

Syntax
int DVP2420_StopEncode(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function stops encoding on a specified channel.

See Also
DVP2420_StartEncode

DVP2420_StartPreview

Syntax

int DVP2420_StartPreview(int nChNum,
HWND hWnd, int nFrameRate, BOOL bDisableAudio = FALSE)

Parameters

nChNum:	Specifies the channel ID number.
hWnd	Specifies a windows handle for display area.
nFrameRate	A value to set display frame rate of specified channel.
bDisableAudio	Specifies if to preview without the audio.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function starts to preview the video on a specified channel. The resolution of the video depends on the size of the display window.

See Also

DVP2420_StopPreview

DVP2420_StopPreview

Syntax
int DVP2420_StopPreview(int nChNum)

Parameters
nChNum: Specifies the video channel ID number.

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function stops to preview the video on a specified channel.

See Also
DVP2420_StartPreview

DVP2420_StartMotionDetect

Syntax
int DVP2420_StartMotionDetect(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function stops the motion detection on a specified channel.

See Also
DVP2420_StopMotionDetect

DVP2420_StopMotionDetect

Syntax
int DVP2420_StopMotionDetect(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function stops the motion detection on a specified channel.

See Also
DVP2420_StartMotionDetect

DVP2420_SetSignalType

Syntax
int DVP2420_SetSignalType(int nChNum, int nSignalType)

Parameters
nChNum:	Specifies the channel ID number.
nSignalType:	A value to set the video signal type. (1: PAL, 2: NTSC) (Default value is 2)

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function sets the video signal type for encoding.

See Also

DVP2420_SetMPEGType

Syntax

int DVP2420_SetMPEGType(int nChNum, int nMPEGType, int nVideoSize)

Parameters

nChNum: Specifies the channel ID number.

nMPEGType: A value to set the video MPEG standard. (1: MPEG1, 2: MPEG2, 4: MPEG4) (Default value is 4)

nVideoSize: A value to set the video resolution. MPEG1 video only supports SIF resolution. (Default resolution is SIF for MPEG1 and D1 for MPEG2/ MPEG4.)

nVideoSize	Video Resolution	Resolution (Horiz. x Vert.)		Video Format		
		NTSC	PAL	MPEG1	MPEG2	MPEG4
0	D1	720x480	720x576		X	X
1	VGA	640x480	640x576		X	X
2	SIF	352x240	352x288	X	X	X
3	CIF	352x288	352x288		X	X
4	QCIF	176x144	176x144		X	X

Return Value

ENC_SUCCEEDED: Function succeeded.

ENC_FAILED: Function failed.

ENC_SDKINITFAILED: SDK does not be initialized successfully.

ENC_DEVINITFAILED Chip does not be initialized successfully.

ENC_CHNUMERROR: Invalid channel ID number.

ENC_PARAMERROR: Parameter error.

Description

This function sets the video MPEG standard and the video resolution for encoding.

See Also

DVP2420_SetBrightness

Syntax
int DVP2420_SetBrightness(int nChNum, int nBrightness)

Parameters

nChNum:	Specifies the channel ID number.
nBrightness:	A value to set the video brightness of a specified channel. The range is 0~255. (Default value is 128)

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function sets the video brightness on a specified channel.

See Also
DVP2420_GetBrightness

DVP2420_GetBrightness

Syntax
int DVP2420_GetBrightness(int nChNum, int *pBrightness)

Parameters

nChNum:	Specifies the channel ID number.
pBrightness:	An integer pointer to store the returned video brightness of a specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the video brightness on a specified channel.

See Also
DVP2420_SetBrightness

DVP2420_SetContrast

Syntax
int DVP2420_SetContrast(int nChNum, int nContrast)

Parameters

nChNum:	Specifies the channel ID number.
nContrast:	A value to set the video contrast of a specified channel. The range is 0~127. (Default value is 68)

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function sets the video contrast on a specified channel.

See Also
DVP2420_GetContrast

DVP2420_GetContrast

Syntax
int DVP2420_GetContrast(int nChNum, int *pContrast)

Parameters

nChNum:	Specifies the channel ID number.
pContrast:	An integer pointer to store the returned video contrast of a specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the video contrast on a specified channel.

See Also
DVP2420_SetContrast

DVP2420_SetSaturation

Syntax
int DVP2420_SetSaturation(int nChNum, int nSaturation)

Parameters

nChNum:	Specifies the channel ID number.
nSaturation:	A value to set the video color saturation of a specified channel. The range is 0~127. (Default value is 64)

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function gets the video color saturation on a specified channel.

See Also
DVP2420_GetSaturation

DVP2420_GetSaturation

Syntax
int DVP2420_GetSaturation(int nChNum, int *pSaturation)

Parameters

nChNum:	Specifies the channel ID number.
pSaturation:	An integer pointer to store the returned video color saturation of a specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the video contrast on a specified channel.

See Also
DVP2420_SetContrast

DVP2420_SetHue

Syntax
int DVP2420_SetHue(int nChNum, int nHue)

Parameters
nChNum:	Specifies the channel ID number.
nHue:	A value to set the video hue value of a specified channel. The range is -128~127. (Default value is 0)

Return Value
ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function sets the video hue value on a specified channel.

See Also
DVP2420_GetHue

DVP2420_GetHue

Syntax
int DVP2420_GetHue(int nChNum, int *pHue)

Parameters

nChNum:	Specifies the channel ID number.
pHue:	An integer pointer to store the returned video hue value of a specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the video hue value on a specified channel.

See Also
DVP2420_SetHue

DVP2420_SetGOPType

Syntax

int DVP2420_SetGOPType (int nChNum, int nKeyFrameIntervals, int nRefFramesDistance);

Parameters

nChNum:	Specifies the channel ID number.
nKeyFrameIntervals:	A value to set the number of frames in a GOP. The range is 1~256. (Default value is 15)
nRefFramesDistance:	A value to set the frame distance between the reference frames. The range is 0~3. (Default value is 3 for MPEG1/MPEG2 and 1 for MPEG4) (MPEG4 does not support B-Frame) 0 -- The GOP structure is I. (Encoder will generate I frames only) 1 -- The GOP structure is IP. 2 -- The GOP structure is IBP. 3 -- The GOP structure is IBBP.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function sets the number of frames in a GOP and the frame distance between the reference frames on a specified channel.

See Also

DVP2420_GetGOPType

DVP2420_GetGOPType

Syntax

int DVP2420_GetGOPType (int nChNum, int *pKeyFrameIntervals, int *pRefFramesDistance)

Parameters

nChNum:	Specifies the channel ID number.
pKeyFrameIntervals:	An integer pointer to store the returned number of frames in a GOP.
pRefFramesDistance:	An integer pointer to store the returned frame distance. 0 -- The GOP structure is I. (Encoder will generate I frames only) 1 -- The GOP structure is IP. 2 -- The GOP structure is IBP. 3 -- The GOP structure is IBBP.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function gets the number of frames in GOP and the frame distance between the reference frames on a specified channel.

See Also

DVP2420_SetGOPType

DVP2420_SetFrameRate

Syntax

int DVP2420_SetFrameRate (int nChNum, int nFrameRate, int nSkipFrameNum);

Parameters

nChNum:	Specifies the video channel ID number.
nFrameRate:	A value to set the frame rate field in the sequence header. The range is 1~8. (Default value is 4) 1 -- 24,000/1001 23.976 fps 2 -- 24 Film 3 -- 25 PAL 4 -- 30,000/1001 29.97 fps NTSC 5 -- 30 NTSC drop frame 6 -- 50 Double frame rate PAL 7 -- 60,000/1001 Double frame rate NTSC 8 -- 60 Double frame rate drop frame NTSC
nSkipFrameNum:	A value to set the minimum number of frames to skip. The range is 0~15. (Default value is 0)

If the frame rate is 30 and the minimum to skip is 1 (one frame is displayed, the next one is not), then the effective frame rate becomes 15 (only half of the frames are displayed). Similarly,
If skip = 2, the effective frame rate = 30/3 = 10 fps (one out of three frames is displayed)
If skip = 3, the effective frame rate = 30/4 = 7.5 fps (one out of four frames is displayed)
etc.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function sets video frame rate on a specified channel.

See Also

DVP2420_GetFrameRate

DVP2420_GetFrameRate

Syntax

int DVP2420_GetFrameRate (int nChNum,
int *pFrameRate, int *pSkipFrameNum)

Parameters

nChNum:	Specifies the channel ID number.
pFrameRate:	An integer pointer to store the returned frame rate field in the sequence header. 1 -- 24,000/1001 23.976 fps 2 -- 24 Film 3 -- 25 PAL 4 -- 30,000/1001 29.97 fps NTSC 5 -- 30 NTSC drop frame 6 -- 50 Double frame rate PAL 7 -- 60,000/1001 Double frame rate NTSC 8 -- 60 Double frame rate drop frame NTSC
pSkipFrameNum:	An value to set the minimum number of frames to skip.

If the frame rate is 30 and the minimum to skip is 1 (one frame is displayed, the next one is not), then the effective frame rate becomes 15 (only half of the frames are displayed). Similarly,
If skip = 2, the effective frame rate = 30/3 = 10 fps (one out of three frames is displayed)
If skip = 3, the effective frame rate = 30/4 = 7.5 fps (one out of four frames is displayed)
etc.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function gets the video frame rate on a specified channel.

See Also

DVP2420_SetFrameRate

DVP2420_InsertMotionDetectMask

Syntax

int DVP2420_InsertMotionDetectMask(int nChNum, MDRegion md_regions[], int nThreshold[])

Parameters

nChNum:	Specifies the channel ID number.
md_regions:	A structure array to set the regions of the motion detection of a specified channel. The number of the elements of the array must be larger than nine.
nThreshold:	A integer array to set the motion vector thresholds of corresponding regions. The number of the elements of the array must be larger than nine. (The default value is 80 for every region)

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function sets the motion detect regions and thresholds on a specified video channel. The setting of the default regions are shown as follows:

Region1:	left=0,	right=15,	top=0,	bottom=10
Region2:	left=15,	right=30,	top=0,	bottom=10
Region3:	left=30,	right=45,	top=0,	bottom=10
Region4:	left=0,	right=15,	top=10,	bottom=20
Region5:	left=15,	right=30,	top=10,	bottom=20
Region6:	left=30,	right=45,	top=10,	bottom=20
Region7:	left=0,	right=15,	top=20,	bottom=30
Region8:	left=15,	right=30,	top=20,	bottom=30
Region9:	left=30,	right=45,	top=20,	bottom=30

See Also

Struct MDRegion
DVP2420_GetMotionDetectMask

DVP2420_GetMotionDetectMask

Syntax

int DVP2420_GetMotionDetectMask(int nChNum, MDRegion md_regions[], int nThreshold[])

Parameters

nChNum:	Specifies the channel ID number.
md_regions:	A structure array to store the returned regions of the motion detection of a specified channel. The number of the elements of the array must be larger than nine.
nThreshold:	A integer array to store the returned motion vector thresholds of corresponding regions of a specified channel. The number of the elements of the array must be larger than nine.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function gets the motion detect regions and thresholds on a specified channel.

See Also

Struct MDRegion
DVP2420_InsertMotionDetectMask

DVP2420_SetVideoBitrate

Syntax

int DVP2420_SetVideoBitrate(int nChNum, int nBitRate, int nAvgBitRate)

Parameters

nChNum:	Specifies the channel ID number.
nBitrate:	A value to set maximum bit rate of specified video channel. The range is 128 kbps~15 Mbps. (Default value is 4 Mbps)

The suggested video bit rate is from 1.5 Mbps to 15 Mbps for 1/2 D1 and above.
The suggested video bit rate is from 512 kbps to 15 Mbps for SIF.
The suggested video bit rate is from 128 kbps to 15 Mbps for QSIF.

nAvgBitRate:	A value to set average bit rate of specified channel. The range is 128 kbps~9 Mbps. (Default value is 3.5 Mbps for PS, 3Mbps for TS)

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function sets the bit rate of the video stream on a specified video channel. The constant bit rate is used when nBitRate equals nAvgBitRate.

See Also

DVP2420_GetVideoBitrate

DVP2420_GetVideoBitrate

Syntax

int DVP2420_GetVideoBitrate(int nChNum, int *pBitRate, int *pAvgBitRate)

Parameters

nChNum:	Specifies the channel ID number.
pBitrate:	An integer pointer to store the returned maximum bit rate of specified channel.
pAvgBitRate:	An integer pointer to store the returned average bit rate of specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function gets the bit rate of the video stream on a specified channel.

See Also

DVP2420_SetVideoBitrate

DVP2420_SetAudioBitrate

Syntax

int DVP2420_SetAudioBitrate(int nChNum, int nBitrate)

Parameters

nChNum:	Specifies the channel ID number.
nBitrate:	A value to set bit rate of the audio stream on a specified channel. The range is 0 ~ 13. (Default value is 7 (128 kbps))

0 -- 32 kbps	1 -- 48 kbps	2 -- 56 kbps
3 -- 64 kbps	4 -- 80 kbps	5 -- 96 kbps
6 -- 112 kbps	7 -- 128 kbps	8 -- 160 kbps
9 -- 192 kbps	10 -- 224 kbps	11 -- 256 kbps
12 -- 320 kbps	13 -- 384 kbps	

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function sets the bit rate of the audio stream on a specified channel.

See Also

DVP2420_GetAudioBitrate

DVP2420_GetAudioBitrate

Syntax

int DVP2420_GetAudioBitrate(int nChNum, int *pBitrate)

Parameters

nChNum: Specifies the channel ID number.

pBitrate: An integer pointer to store the returned bit rate of the audio stream on a specified channel. The range is 0 ~ 13.

0 -- 32 kbps	1 -- 48 kbps	2 -- 56 kbps
3 -- 64 kbps	4 -- 80 kbps	5 -- 96 kbps
6 -- 112 kbps	7 -- 128 kbps	8 -- 160 kbps
9 -- 192 kbps	10 -- 224 kbps	11 -- 256 kbps
12 -- 320 kbps	13 -- 384 kbps	

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function gets the bit rate of the audio stream on a specified channel.

See Also

DVP2420_SetAudioBitrate

DVP2420_GPIOSetData

Syntax
int DVP2420_GPIOSetData(int nDONum, BOOL bValue)

Parameters

nDONum:	Specifies the digital output number. The range is 0~3.
bValue:	A value to the value of the specified digital output.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description
This function sets the value of the specified digital output. The first chip controls the GPIO, thus must to initialize the chip and download the firmware into the chip before controlling GIPO.

See Also
DVP2420_GPIOGetData

DVP2420_GPIOGetData

Syntax
int DVP2420_GPIOGetData(int nDINum, BOOL* pValue)

Parameters

nDINum:	Specifies the digital input number. The range is 0~3.
pValue:	A point to get the value of the specified digital input.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_DEVINITFAILED	Chip does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description
This function gets the value of the specified digital input. The first chip controls the GPIO, thus must to initialize the chip and download the firmware into the chip before controlling GIPO.

See Also
DVP2420_GPIOSetData

DVP2420_RegisterStreamReadCB

Syntax

int DVP2420_RegisterStreamReadCallback(
PSTREAMREADPROC pStreamRead)

Parameters

pStreamRead:	A STREAMREAD_STRUCT pointer for the StreamRead callback function.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.

Description

This function registers the StreamRead callback functions.

See Also

Struct PSTREAMREADPROC

DVP2420_RegisterMotionDetectCB

Syntax

int DVP2420_RegisterMotionDetectCallback(
PMOTIONDETECTPROC pMotionDetect)

Parameters

pMotionDetect:	A function pointer of the MotionDetect callback function.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.

Description

This function registers the MotionDetect callback function.

See Also

Type PMOTIONDETECTPROC

DVP2420_RegEncFailCB

Syntax
int DVP2420_RegEncFailCB(PENCFAILPROC pEncFail)

Parameters
pEncFail: A function pointer of the EncFail callback function.

Return Value
ENC_SUCCEEDED: Function succeeded.
ENC_SDKINITFAILED: SDK does not be initialized successfully.

Description
This function registers the EncFail callback function.

See Also
Type PENCFAILPROC

DVP2420_SetFileName

Syntax

int DVP2420_SetFileName(int nChNum, char *pFileName)

Parameters

nChNum:	Specifies the video channel ID number.
pFileName:	A NULL-terminated string for the storage video file name.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.
ENC_PARAMERROR:	Parameter error.

Description

This function sends the storage video file name to SDK to generate corresponding log files filename.log and filename_pts.log. The two log files are necessary for hardware playback.

See Also

DVP2420_GetCurImage

Syntax

int DVP2420_GetCurImage(int nChNum, long *lpImageBuf, long *lpBufSize)

Parameters

nChNum:	Specifies the video channel ID number.
lpImageBuf:	A long pointer to store the returned image data on a specified channel.
lpBufSize:	A long pointer to store the returned size of the image on a specified channel.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.
ENC_SDKINITFAILED:	SDK does not be initialized successfully.
ENC_CHNUMERROR:	Invalid channel ID number.

Description

This function gets current rendered image on a specified channel.

See Also

DVP2420_PSMPEG4ToDivx

Syntax

int DVP2420_PSMPEG4ToDivx(char *src_filename,
char * des_filename)

Parameters

src_filename:	A string for the PS MPEG4 video file name.
des_filename:	A string for the DIVX format file name.

Return Value

ENC_SUCCEEDED:	Function succeeded.
ENC_FAILED:	Function failed.

Description

This function converts the MPEG4 video file to the DIVX format video file.

See Also

DVP2420_IsPSMPEG4Type

Syntax
BOOL DVP2420_IsPSMPEG4Type(char * mpegfilename)

Parameters
mpeg4filename: A string for the PS MPEG video file name.

Return Value
TRUE: The specified video file is PS MPEG4 video type.
FALSE: The specified video file is not PS MPEG4 video type.

Description
This function detects if the specified video file is PS MPEG4 video type.

See Also

Decoding Functions Reference

Data Type

DecRes

Syntax

```
typedef enum
{
    DEC_SUCCEEDED       = 1,
    DEC_FAILED          = 0,
    DEC_SDKINITFAILED   = -1,
    DEC_DEVINITFAILED   = -2,
    DEC_PARAMERROR      = -3,
    DEC_CHNUMERROR      = -4,
    DEC_MISSLOGFILE     = -5
} DecRes;
```

Description

The method returned code.

PDECENDOFFILEPROC

Syntax
typedef void (*PDECENDOFFILEPROC)(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
None.

Description
The Decoding EOF Process Callback function. This callback function is called when decoding process reaches END_OF_FILE.

See Also
DVP2420_RegDecEndOfFileCB

PDECFAILPROC

Syntax
typedef void (*PDECFAILPROC)(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
None.

Description
The Decoding Failed Process Callback function. This callback function is called when decoding process fails.

See Also
DVP2420_RegDecFailCB

Method

DVP2420_CreateDecSDKInstence

Syntax

int DVP2420_CreateDecSDKInstence (void **pp)

Parameters

pp:	A pointer to the SDK instance.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_PARAMERROR:	Parameter error.

Description

This function creates SDK instance.

See Also

DVP2420_InitSDK

Syntax
int DVP2420_InitSDK ()

Parameters
None.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.

Description
This function initializes the SDK.

See Also
DVP2420_CloseSDK

DVP2420_CloseSDK

Syntax
int DVP2420_CloseSDK ()

Parameters
None.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.

Description
This function closes up the SDK.

See Also
DVP2420_InitSDK

DVP2420_InitChips

Syntax
int DVP2420_InitChips(int nChNum)

Parameters

nChNum:	Specifies the channel ID number.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description
This function initializes the codec chip on a specified channel.

See Also
DVP2420_ReleaseChips

DVP2420_ReleaseChips

Syntax
int DVP2420_ReleaseChips(int nChNum)

Parameters

nChNum:	Specifies the channel ID number.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function releases the codec chip on a specified channel.

See Also
DVP2420_InitChips

DVP2420_DownloadFW

Syntax
int DVP2420_DownloadFW(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_DEVINITFAILED	Chip does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function downloads the firmware into the chip on a specified channel. Two firmware files boot.sre and pscodec.sre will be used by the library. Notes: Don't download the firmware into the chip when the encoding process is running in the chip.

See Also

DVP2420_GetChipCount

Syntax

int DVP2420_GetChipCount(int *pChipCnt)

Parameters

pChipCnt:	An integer pointer to store the returned number of the chips.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_PARAMERROR:	Parameter error.

Description

This function gets the number of the codec chips.

See Also

DVP2420_GetSDKVersion

Syntax

float DVP2420_GetSDKVersion()

Parameters

None.

Return Value

If function succeeded, it returns the version of the SDK. Otherwise, it returns -1.

Description

This function gets the version of the SDK.

See Also

DVP2420_StartDecode

Syntax

int DVP2420_StartDecode(int nChNum, LPCSTR lpcsFileName, HWND hWnd)

Parameters

nChNum:	Specifies the channel ID number.
lpcsFileName:	A NULL-terminated string for the decoded file name.
hWnd:	Specifies the window handle of the display area.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_DEVINITFAILED	Chip does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.
DEC_MISSLOGFILE:	Load the playback log failed.

Description

This function starts to decode the video on a specified channel.

See Also

DVP2420_StopDecode

DVP2420_StopDecode

Syntax
int DVP2420_StopDecode(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function stops decoding on a specified channel.

See Also
DVP2420_StartDecode

DVP2420_SetSignalType

Syntax
int DVP2420_SetSignalType(int nChNum, int nSignalType)

Parameters
nChNum:	Specifies the channel ID number.
nSignalType:	A value to set the video signal type. (1: PAL, 2: NTSC) (Default value is 2)

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_DEVINITFAILED	Chip does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description
This function sets the video signal type for decoding.

See Also

DVP2420_SetMPEGType

Syntax
int DVP2420_SetMPEGType(int nChNum, int nMPEGType)

Parameters

nChNum:	Specifies the channel ID number.
nMPEGType:	A value to set the video MPEG standard. (1: MPEG1, 2: MPEG2, 4: MPEG4) (Default value is 4)

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_FAILED:	Function failed.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_DEVINITFAILED	Chip does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description
This function sets the video MPEG standard decoding.

See Also

DVP2420_Pause

Syntax
int DVP2420_Pause(int nChNum)

Parameters
nChNum:	Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function pauses or continues to play the video.

See Also

DVP2420_Fast

Syntax
int DVP2420_Fast(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function speeds up to play the video. The function doubles the speed by one time, 3 times at most.

See Also

DVP2420_Rewind

Syntax
int DVP2420_Rewind(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function rewinds to play the video. The function doubles the speed by one time, 3 times at most.

See Also

DVP2420_SingleStep

Syntax
int DVP2420_SingleStep(int nChNum)

Parameters
nChNum: Specifies the channel ID number.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description
This function steps forward one frame of the video.

See Also

DVP2420_SetPlayPos

Syntax

int DVP2420_SetPlayPos(int nChNum, ULONG64 ulRefTime)

Parameters

nChNum:	Specifies the channel ID number.
ulRefTime:	A value of the video time to seek video position.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.

Description

This function seeks the video to the specified video file time.

See Also

DVP2420_GetCurrentFrameNum

Syntax

int DVP2420_GetCurrentFrameNum(int nChNum, int *pCurFrameNum)

Parameters

nChNum:	Specifies the channel ID number.
pCurFrameNum:	An integer pointer to store the returned frame number.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description

This function gets the current decoded frame number.

See Also

DVP2420_GetFileTotalFrames

DVP2420_GetFileTotalFrames

Syntax
int DVP2420_GetFileTotalFrames(int nChNum, int *pTotalFrame)

Parameters
nChNum:	Specifies the channel ID number.
pTotalFrame:	An integer pointer to store the returned number of the total frames.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description
This function gets the number of the total frames in the video file.

See Also
DVP2420_GetCurrentFrameNum

DVP2420_GetPlayedTime

Syntax

int DVP2420_GetPlayedTime(int nChNum, ULONG64 *lpPlayedTime)

Parameters

nChNum:	Specifies the channel ID number.
lpPlayedTime:	A pointer to store the returned current video time.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description

This function gets the current played time of the video.

See Also

DVP2420_GetFileTime

DVP2420_GetFileTime

Syntax
int DVP2420_GetFileTime(int nChNum, ULONG64 *lpFileTime)

Parameters
nChNum:	Specifies the channel ID number.
lpFileTime:	A pointer to store the returned video file time.

Return Value
DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.
DEC_CHNUMERROR:	Invalid channel ID number.
DEC_PARAMERROR:	Parameter error.

Description
This function gets the video file time.

See Also
DVP2420_GetPlayedTime

DVP2420_RegDecEndOfFileCB

Syntax

int DVP2420_RegDecEndOfFileCB(PDECENDOFFILEPROC pDecEndOfFile)

Parameters

pDecEndOfFile: A function pointer of the DecEndOfFile callback function.

Return Value

DEC_SUCCEEDED:	Function succeeded.
DEC_SDKINITFAILED:	SDK does not be initialized successfully.

Description

This function registers the DecEndOfFile callback function.

See Also

Type PDECENDOFFILEPROC

DVP2420_RegDecFailCB

Syntax
int DVP2420_RegDecFailCB(PDECFAILPROC pDecFail)

Parameters
pDecFail: A function pointer of the DecFail callback function.

Return Value
DEC_SUCCEEDED: Function succeeded.
DEC_SDKINITFAILED: SDK does not be initialized successfully.

Description
This function registers the DecFail callback function.

See Also
Type PDECFAILPROC