



PROGRAMMING GUIDE

VIP Sample Programming Guide

Version 1.0

Copyright Notice:

Copyright © 2007 VIA Technologies Incorporated. All rights reserved. No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise without the prior written permission of VIA Technologies Incorporated. The material in this document is for information only and is subject to change without notice. VIA Technologies Incorporated reserves the right to make changes in the product design without reservation and without notice to its users.

Trademark Notices:

All trademarks are the properties of their respective owners.

Disclaimer Notice:

No license is granted, implied or otherwise, under any patent or patent rights of VIA Technologies. VIA Technologies make no warranties, implied or otherwise, in regard to this document and to the products described in this document. The information provided by this document is believed to be accurate and reliable as of the publication date of this document. However, VIA Technologies assume no responsibility for any errors in this document. Furthermore, VIA Technologies and assume no responsibility for the use or misuse of the information in this document and for any patent infringements that may arise from the use of this document. The information and product specifications within this document are subject to change at any time, without notice and without obligation to notify any person of such change.

Office:**Taiwan Office:**

1st Floor, No. 531

Chung-Cheng Road, Hsin-Tien

Taipei, Taiwan ROC

Tel: 886-2-2218-5452

FAX: 886-2-2218-5453

Home page: <http://www.via.com.tw>

REVISION HISTORY

Version	Date	Revision
1.0	12/3/07	Updated format and fixed grammar.

TABLE OF CONTENTS

- Revision History i**
- Table of Contents ii**
- 1 DDraw Surface Control Interface 1**
 - 1.1 VIA Defined Structure for Video Capture 1
- 2 VIA DDraw Surface Control Definition 2**
 - 2.1 VIA_CREATEDDSURFACE 2
 - 2.2 VIA_CREATEVIDEOPORT 2
 - 2.3 VIA_UPDATEOVERLAYSURFACE..... 3
 - 2.4 VIA_STARTVIDEO 3
 - 2.5 VIA_STOPVIDEO..... 4
 - 2.6 VIA_SETCOLORCONTROLS 4
 - 2.7 VIA_DESTROYDDSURFACE 4
 - 2.8 VIA_DESTROYVIDEOPORT..... 4
- 3 Display Video from Capture Port 5**
 - 3.1 Load Library 5
 - 3.2 DDraw Surface Control Flow 5
- 4 HQV Switch..... 6**
 - 4.1 HQV Control Definition..... 6
 - 4.2 HQV Control Samples 6

1 DDRAW SURFACE CONTROL INTERFACE

The DDrawSurfaceControl function is exported from VCaptureSuf.dll and can be used like an API.

The function in VCaptureSuf.dll is **DDrawSurfaceControl(WORD wMessage, LPVOID lpParam1, LPVOID lpParam2)**

The descriptions about the input arguments are as follows:

WORD wMessage: Indicates which DDraw surface control will be performed.

LPVOID lpParam1: VIA defined structure for video capture.

LPVOID lpParam2: The input value is always NULL.

1.1 VIA Defined Structure for Video Capture

The structure for video capture is defined as follows:

```
typedef struct{
    //Client Read Section.
    DWORD DDSurfPtr1;    //Linear address / Selector to DDraw Video Surfaces 1
    DWORD DDSurfPtr2;    //Linear address / Selector to DDraw Video Surfaces 2
    DWORD DDSurfOffset1; //Physical Address of DDraw Video Surfaces 1
    DWORD DDSurfOffset2; //Physical Address of DDraw Video Surfaces 2
    //Client Write Section.
    DWORD dwSurfWidth;    //Width of Source Buffer. We use it when creating DDraw Surface
    DWORD dwSurfHeight;   //Height of Source Buffer. We use it when creating DDraw Surface
    DWORD dwVPESrcWidth;  //Width of Video Source from Video Decoder. We use it when creating Video Port
    DWORD dwVPESrcHeight; //Height of Video Source from Video Decoder. We use it when creating Video Port
    DWORD dwPreScaledWidth; //Width of Video in frame buffer.
    DWORD dwPreScaledHeight; //Height of Video in frame buffer.
    RECTL SrcRectL;       //RECT in Source Buffer
    RECTL DstRectL;       //RECT in Overlay Video Window.
    RECTL CropRectL;      //RECT for Cropping Video data from Video Decoder
    DWORD dwFlags;        //Flags
    DWORD dwInterlace;    //Interlace Mode / NonInterlace mode
    DWORD dwExtendFunction; //call to SAA7113 relation function
    DWORD dwValue;        //color control setting
    int VPortID; //Video Port ID
}VIA_VIDCAPTURE, FAR * LPVIA_VIDCAPTURE;
```

2 VIA DDRAW SURFACE CONTROL DEFINITION

Some DDraw surface control code for capture AP use has been defined as follows:

```
#define VIA_CREATEDDSURFACE      0x00000001
#define VIA_CREATEVIDEOPORT     0x00000002
#define VIA_UPDATEOVERLAYSURFACE 0x00000003
#define VIA_STARTVIDEO          0x0000000C
#define VIA_STOPVIDEO           0x0000000D
#define VIA_SETCOLORCONTROLS    0x00000010
#define VIA_DESTROYDDSURFACE    0x00000015
#define VIA_DESTROYVIDEOPORT    0x00000016
```

The following are the details of the DDraw surface control. The examples are based on open VIP port 0.

2.1 VIA_CREATEDDSURFACE

This DDraw surface control is used to create primary surface and overlay surface.

Example:

```
lpVidCapture->VPortID = iVideoPort; // 0
lpVidCapture->dwSrcWidth = SURFACE WIDTH; // 720
lpVidCapture->dwSrcHeight = SURFACE HEIGHT; // 576

dwRet = lpfnDDrawSurfaceControl( VIA_CREATEDDSURFACE, lpVidCapture, NULL);
```

2.2 VIA_CREATEVIDEOPORT

This DDraw surface control is used to create a video port.

Example:

```
lpVidCapture->dwInterlace = INTERLACE; // 1 (TRUE)
lpVidCapture->dwVPESrcWidth = (DWORD) DEFAULT_X_NTSC; // 720
lpVidCapture->dwVPESrcHeight = (DWORD) DEFAULT_Y_NTSC; // 480

dwRet = lpfnDDrawSurfaceControl( VIA_CREATEVIDEOPORT, lpVidCapture , NULL );
```

2.3 VIA_UPDATEOVERLAYSURFACE

This DDraw surface control is used to update video overlay.

Example:

```
lpVidCapture->SrcRectL.top      = 0;
lpVidCapture->SrcRectL.bottom  = dwSrcHeight;
lpVidCapture->SrcRectL.left    = 0;
lpVidCapture->SrcRectL.right   = dwSrcWidth;

lpVidCapture->DstRectL.top      = iYPos + nTitleHeight + nBorderHeight;
lpVidCapture->DstRectL.bottom  = iYPos + nTitleHeight + nBorderHeight + iHeight;
lpVidCapture->DstRectL.left    = iXPos + nBorderWidth;
lpVidCapture->DstRectL.right   = iXPos + nBorderWidth + iWidth;
dwRet = lpfnDDrawSurfaceControl( VIA_UPDATEOVERLAYSURFACE, lpVidCapture, NULL);
```

2.4 VIA_STARTVIDEO

This DDraw surface control is used to enable the hardware video port and starts the flow of video data.

Example:

```
WORD wVcapmode = DEINTERLAY;

lpVidCapture->CropRectL.left    = 0;
lpVidCapture->CropRectL.top     = 0;
lpVidCapture->CropRectL.right   = (DWORD) DEFAULT_X_NTSC;
lpVidCapture->CropRectL.bottom  = (DWORD) DEFAULT_Y_NTSC;
lpVidCapture->dwPreScaledWidth  = (DWORD) DEFAULT_X_NTSC;
lpVidCapture->dwPreScaledHeight = (DWORD) DEFAULT_Y_NTSC;
lpVidCapture->dwFlags           = (DWORD)wVcapmode;

dwRet = lpfnDDrawSurfaceControl( VIA_STARTVIDEO, lpVidCapture, NULL);
```

2.5 VIA_STOPVIDEO

This DDraw surface control is used to stop the flow of video port data into the frame buffer.

Example:

```
lpfnDDrawSurfaceControl(VIA_STOPVIDEO, lpVidCapture, NULL);
```

2.6 VIA_SETCOLORCONTROLS

This DDraw surface control sets the color control settings associated with the video port.

Example:

```
lpVidCapture->dwExtendFunction = V1OnTop;  
lpfnDDrawSurfaceControl(VIA_SETCOLORCONTROLS, lpVidCapture, NULL);
```

2.7 VIA_DESTROYDDSURFACE

This DDraw surface control release all created surfaces (primary, overlay).

Example:

```
dwRet = lpfnDDrawSurfaceControl(VIA_DESTROYDDSURFACE, NULL, NULL);
```

2.8 VIA_DESTROYVIDEOPORT

This DDraw surface control release video port and video port container.

Example:

```
dwRet = lpfnDDrawSurfaceControl(VIA_DESTROYVIDEOPORT, NULL, NULL);
```


3 DISPLAY VIDEO FROM CAPTURE PORT

In this section, the basic display video steps will be described.

3.1 Load Library

First of all, the application should load VCaptureSuf.dll. Then get the DDraw control function pointer.

Example:

```
hVPE = LoadLibrary(_T("VCaptureSuf.dll"));  
(FARPROC&) lpfnDDrawSurfaceControl = GetProcAddress(hVPE, TEXT("DDrawSurfaceControl"));
```

3.2 DDraw Surface Control Flow

After getting the function pointer, the application can use the function to control the capture port. The steps are the following:

Create Surface → Create Video Port → Update Video Port → Update Overlay Surface

The DDraw control interface calling sequence is like the above. Before calling the function, the data in lpVidCapture should be filled correctly. The following is the sequence:

```
lpfnDDrawSurfaceControl( VIA CREATEDDSURFACE, lpVidCapture, NULL);  
lpfnDDrawSurfaceControl( VIA CREATEVIDEOPORT, lpVidCapture, NULL );  
lpfnDDrawSurfaceControl( VIA STARTVIDEO, lpVidCapture, NULL);  
lpfnDDrawSurfaceControl( VIA UPDATEOVERLAYSURFACE, lpVidCapture, NULL);
```

4 HQV SWITCH

This section describes the HQV switch interface. The application can use this interface to enable / disable HQV function.

4.1 HQV Control Definition

The following are the definitions of HQV control:

```
#define V1_HQV_ON      0x000000B0  
#define V1_HQV_OFF    0x000000C0  
#define V3_HQV_ON      0x000000D0  
#define V3_HQV_OFF    0x000000E0
```

4.2 HQV Control Samples

If an application needs to enable the HQV function, follow the example below.

Example:

```
lpVidCapture->dwExtendFunction = V1_HQV_ON;  
lpfnDDrawSurfaceControl(VIA_SETCOLORCONTROLS, lpVidCapture, NULL);
```

If an application needs to disable the HQV function, follow the example below.

Example:

```
lpVidCapture->dwExtendFunction = V1_HQV_OFF;  
lpfnDDrawSurfaceControl(VIA_SETCOLORCONTROLS, lpVidCapture, NULL);
```