

CC Pilot™ XS

Video interface description



Table of Contents

| | |
|-------------------------------|----------|
| Introduction..... | 3 |
| Purpose..... | 3 |
| References..... | 3 |
| History..... | 3 |
| Introduction..... | 4 |
| Starting CCVideoXS | 4 |
| Controlling CCVideoXS | 5 |
| Windows messages | 5 |

Introduction

Purpose

This document describes the software interface to the video application.

References

-

History

| Rev | Date | Author | Remarks |
|------------|-------------|------------------|----------------|
| 1.0 | 08-07-28 | Tobias Andersson | First version. |

Introduction

A video image can be displayed via the video application which can be started without frames and consequently show only the actual image. This document describes how to control the video application from a custom C# application.

Starting CCVideoXS

Description

CCVideoXS can be started using the `Process.Start()` method.

Flags

CCVideo takes the following flags:

-noframe Starts CCVideo with no menu, buttons or pop-up menu, which means that it can only be controlled via Windows messages. Also, it's always on top.

-startminimized starts CCVideo with the window initially being minimized if used together with the `-noframe` flag. The `-startminimized` flag being used alone does nothing.

-croptosize <width> <height> crop the sides, the top and the bottom of the video input image leaving only the center part of the video image. This yields a much better picture quality if the size of the video window is set slightly smaller than 720px by 480px or 360px by 240px for NTSC and 720px by 576px or 360px by 288px for PAL, and a better picture with any other resolution. The picture is more or less smaller but this may be solved in other ways. This does not set the size of the window. It has to be set separately via Windows messages. To get an optimal video image use the same width and height for the `-croptosize <width> <height>`-flag as you do when you set the size of the window using windows messages (or exactly half of it or exactly a quarter of it (see elaboration)). If you want to achieve an optimal video image without using the `-croptosize` flag you have to set the window size to 1/1 or 1/2 of the input signal, which is 720px by 480px and 360px by 240px for NTSC or 720px by 576px and 360px by 288px for PAL. (Elaboration: The driver which we are using is a closed source DirectX driver from Microsoft. This driver first looks at the output window to see if it's larger or smaller than the input video signal format. If it's smaller than it (720px by 480px for NTSC), it shrinks the input signal to half of it's original size cutting out every other column of pixels and every other row of pixels (This is what's destroying the picture quality). After that it stretches out the image to fit the size of the output window, duplicating lines and rows. If the size of the output window is smaller than half the size of the input signal, 360px by 240px for NTSC, it shrinks the input images to 1/4 by 1/4 of the original size and then stretches the image to fit the output window. If it's smaller than that no output image is shown. The video window goes blue.)

Syntax

```
public bool System.Diagnostics.Process.Start(@"CCVideoXS.exe",  
"<optional flags>")
```

Return value

true if a process resource is started; **false** if no new process resource is started.

Example

```
System.Diagnostics.Process.Start(@"CCVideoXS.exe", "-noframe -  
croptosize 560 370");
```

Controlling CCVideoXS

Description

CCVideoXS is controlled with Windows messages. A window message can be sent with the Windows function `SendMessage()`. Before calling `SendMessage()` use `FindWindow()` to get the handle to the video window.

Syntax

```
[DllImport("coredll.dll", CharSet = CharSet.Auto)]
    private static extern int SendMessage(IntPtr hWnd, int wMsg,
    IntPtr wParam, IntPtr lParam);

[DllImport("coredll.dll", EntryPoint = "FindWindow")]
    private static extern IntPtr FindWindow(string lpClassName, string
    lpWindowName);
```

Example

```
IntPtr hWnd = FindWindow(null, "CCVideo XS");
```

Windows messages

Minimize window - 0x400 + 277

Example

```
SendMessage(hWnd, 0x400 + 277, IntPtr.Zero, IntPtr.Zero);
```

Restore window - 0x400 + 278 – restore the window after being minimized

Example

```
SendMessage(hWnd, 0x400 + 278, IntPtr.Zero, IntPtr.Zero);
```

Set window size - 0x400 + 279

Example

```
SendMessage(hWnd, 0x400 + 279, new IntPtr(560), new IntPtr(370));
```

Set channel - 0x400 + 280

Example

```
SendMessage(hWnd, 0x400 + 280, IntPtr.Zero, new IntPtr(0)); //channel 1
SendMessage(hWnd, 0x400 + 280, IntPtr.Zero, new IntPtr(1)); //channel 2
```

Set window place - 0x400 + 281

Example

```
SendMessage(hWnd, 0x400 + 281, new IntPtr(2), new IntPtr(2));
```

Set full screen - 0x400 + 282

Example

```
SendMessage(hWnd, 0x400 + 282, IntPtr.Zero, new IntPtr(1));
```

The lParam should always be 1 in this case.

Close video application - 0x10

Example

```
SendMessage(hWnd, 0x10, IntPtr.Zero, IntPtr.Zero);
```