This document illustrates the implementation of Still Image Capture feature in a UVC device

Content:

- A common feature of video cameras is the support of still image capture associated with a video stream. Still capture can be initiated either by programmatic software triggers or hardware triggers. In this project, still capture is implemented by hardware trigger.
- There are 3 methods of capturing still image supported by UVC Spec version 1.1. UVC device will have to specify which method it supports in the class-specific descriptors within the relevant VideoStreaming interface.
  - **Method 1**: The host software will extract the next available video frame from the active video pipe in the relevant VideoStreaming interface upon receiving the hardware trigger event. The hardware does not interrupt or alter the video stream in this case. For this method, the still image frame is always the same size as the video frames being streamed.
  - **Method 2**: If the device supports higher-quality still images, it has the option of streaming still image-specific packets across the active video pipe. In this case, the host software will temporarily suspend video streaming, select the optimal bandwidth alternate setting based on the still probe/commit negotiation (subject to bandwidth availability), send a VS_STILL_IMAGE_TRIGGER_CONTROL Set request with the "Transmit still image" option, and prepare to receive the still image data. The device transmits the still image data marked as such in the payload header. Once the complete still image is received, the host software will then revert back to the original alternate setting, and resume video streaming.
  - **Method 3**: This method enables the capture of higher-quality still images from a dedicated bulk still image pipe. By doing so, the active streams would continue uninterrupted.

- In the project attached, Method 2 of Still Image Capture is implemented

- **Modifications done in the example project associated with AN75779 to implement Still Capture:**

- ✓ In cyfxuvcdscr.c file**:**

- Within the Class-specific Video Streaming Input Header Descriptor of CyFxUSBHSConfigDscr[ ] and CyFxUSBSSConfigDscr[ ], following fields have to be modified with the values given:

wTotalLength	: 0x51,0x00
bStillCaptureMethod	: 0x02
bTriggerSupport	: 0x01

(refer 'Table 3-13: Class-specific VS Interface Input Header Descriptor' in UVC 1.1 Spec)

- Within the Class-specific Uncompressed VS Frame descriptor of CyFxUSBHSConfigDscr[ ] and CyFxUSBSSConfigDscr[ ], modify the bmCapabilities field to 0x02.
  (refer 'Table 3-2: Uncompressed Video Frame Descriptors' in UVC 1.1 spec for Uncompressed Payload)

- After the Class-specific Uncompressed VS Frame descriptor of CyFxUSBHSConfigDscr[ ] and CyFxUSBSSConfigDscr[ ], add a Still Image Frame Descriptor:

```
0x0A,                           /* Descriptor size */
0x24,                           /* Descriptor type */
0x03,                           /* VS_STILL_IMAGE_FRAME descriptor
                                   subtype */
0x00,                           /* If method 2 of still image capture
                                   is used, this field shall be set to
                                   zero */
0x01,                           /* Number of Image Size patterns of
                                   This format */
0x80, 0x02,                      /* Width in pixel - 640 */
0xE0, 0x01,                      /* Height in pixel - 480 */
0x00,                           /* Number of Compression pattern of
                                   this format */
```

(refer 'Table 3-17: Still Image Frame Descriptor' in UVC 1.1 spec)

- After the BULK-IN endpoint descriptor, add a Vendor Interface:

```
0x09,                           /* Descriptor size */
CY_U3P_USB_INTRFC_DESCR,        /* Interface Descriptor type */
0x02,                           /* Interface number */
0x00,                           /* Alternate setting number */
0x00,                           /* Number of end points */
0xFF,                           /* Interface class */
0x00,                           /* Interface sub class */
0x00,                           /* Interface protocol code */
0x00                            /* Interface descriptor string index */
```

  Update the length and number of interfaces fields of CyFxUSBHSConfigDscr[ ] to 0xE0, 0x00 and 0x03 respectively.

- Update the length and number of interfaces fields of CyFxUSBSSConfigDscr [ ] to 0xEC, 0x00 and 0x03 respectively.

✓ In uvc.c file:

- Include a ‘Video Still Probe Control’ structure for USB 3.0 as well as USB 2.0:

```
uint8_t glStillProbeCtrl30[CY_FX_MAX_STILL_PROBE_SETTING] ={
0x01,                               /*bFormatIndex: Use 1st Still
                                      Format Index*/
0x01,                               /*bFrameIndex: Use 1st Still Frame
                                      Index*/
0x00,                               /*bCompressionIndex: No Compression
                                      supported*/
0x00, 0x20, 0x1C, 0x00,             /*dwMaxVideoFrameSize: 1280*720*2
                                      bytes*/
0x00, 0x40, 0x00, 0x00              /*dwMaxPayloadTransferSize:16KB DMA
                                      buffer size*/

};


uint8_t glStillProbeCtrl20[CY_FX_MAX_STILL_PROBE_SETTING] ={
0x01,                               /*bFormatIndex: Use 1st Still
                                      Format Index*/
0x01,                               /*bFrameIndex: Use 1st Still Frame
                                      Index*/
0x00,                               /*bCompressionIndex: No Compression
                                      supported*/
0x00, 0x60, 0x09, 0x00,             /*dwMaxVideoFrameSize: 640*480*2
                                      bytes*/
0x00, 0x40, 0x00, 0x00            /*dwMaxPayloadTransferSize:16KB DMA
                                    buffer size*/

};
```

(refer 'Table 4-50: Video Still Probe Control and Still Commit Control' in UVC 1.1 spec)

❖ Handle VS_STILL_PROBE_CONTROL, VS_STILL_COMMIT_CONTROL and VS_STILL_IMAGE_TRIGGER_CONTROL inside UVCHandleVideoStreamingRqts() function.

(refer 'Table 4-51: VS_STILL_PROBE_CONTROL Requests' , 'Table 4-52 VS_STILL_COMMIT_CONTROL Requests' , 'Table 4-54 Still Image Trigger Control' in UVC 1.1 spec)

>> In this project I am sending a vendor command-0x88, to imitate a button press. Hence, I have added this vendor interface.

>> In addition to vendor command, I have configured GPIO 45 as an input pin and register for a callback function in CyU3PGpioInit() API. In SuperSpeed explorer kit GPIO 45 is provided as Switch (SW2). If you press this switch, it will trigger an interrupt callback.

>> If you use e-cam view application for video streaming, 'Enter' key can be pressed to initiate Still Capture.

>> The captured still image is saved as a .bmp file in the Still Capture path which can be provided in the e-CamView Application.

➢ **Logical Flow of Still Capture implementation:**

❖ Once the video streaming has started, in the explorer Kit press GPIO 45 switch OR send a vendor command 0x88 from the Control Center to emulate a button press.

- When this vendor command is received, set a flag 'Button_Press_Event'. OR If you press the GPIO 45 in explorer kit, set `CY_FX_GPIOAPP_GPIO_NEG_EVENT` event
- In the infinite for loop of UVCAppThread_Entry (), monitor if the flag 'Button_Press_Event' is set. Once the flag is set, clear the flag and call a function: Send_Status_to_Host (). OR If GPIO 45 is pressed, CY_FX_GPIOAPP_GPIO_NEG_EVENT event will be checked, and call a function: Send_Status_to_Host ().

- Inside Send_Status_to_Host () function, fill the 4 byte Status Interrupt data into the DMA buffer and send it over the CPU->Status Interrupt EP DMA channel to the Host. (refer 'Table 2-1: Status Packet Format' and 'Table 2-3: Status Packet Format (VideoStreaming Interface as the Originator)).
- This Status Packet informs the Host Application about a button press event. Since, in the UVC descriptor we have informed the Host Application to initiate Still Capture on receiving a hardware trigger interrupt event from the device (bTriggerUsage:0x00), the Host App will send a VS_STILL_IMAGE_TRIGGER_CONTROL with bTrigger set to 1.
- When the device receives a VS_STILL_IMAGE_TRIGGER_CONTROL with bTrigger field set to 1, a flag 'Still_Flag' will be set.
- Since, button press event is an asynchronous one, at the time FX3 receives a VS_STILL_IMAGE_TRIGGER_CONTROL from the Host, it might be half-way through the streaming of a particular frame. So, FX3 has to send a frame with Still Image Header (refer 'Table 2-5: Format of the Payload Header'), once the current frame is completely transferred. CyFxUVCAddHeader () function has been modified to implement this procedure.