Project List 2014-2015

Introduction

The projects are designed as separate stages of a video processing pipeline. Students will be able to connect their projects together in order to observe the effects of the processing implemented by their project, and the interaction with other processing steps.

FPGA Board

Each project, and therefore processing stage, is to be implemented in a Digilent Nexys 2 FPGA board, equipped with a Spartan-3E XC3S500 Board. Details about the board can be found on the Digilent website.

Video Stream

Each board will process a video stream which conforms to either the 800x600 resolution at 60Hz, corresponding to a 40 MHz pixel clock, or the 640x480 resolution at 60Hz, corresponding to a 25 MHz clock. More details about the video timing is available here. The video stream is received, processed, and forwarded by each FPGA board, such that multiple boards may be daisy-chained to form a processing pipeline.

The video stream is carried over a physical interface consisting of 8 data bits and 1 clock signal, located on the Hirose FX2 connector of the Nexys 2 board. The clock signal operates at the pixel clock rate:

Signal	Direction	Width	Location on board
SCLK_IN	Input	1	FX2-CLKIN
SDATA_IN	Input	8	FX2-IO[1-7]
SCLK_OUT	Output	1	FX2-CLKOUT
SDATA_OUT	Output	8	FX2-IO[8-15]

Each word on SDATA_IN or SDATA_OUT consists of either a valid pixel value (all values except 0 and 255) or a part of a synchronization sequence (values 0 and 255). Synchronization sequences consist of 4 consecutive words on SDATA_IN/SDATA_OUT, and are used to signal the start of the displayed portion of a video line (SAV) or the end of a line (EAV). Different SAV/EAV codes are used during vertical blanking:

Sequence Name	Sequence Value	Meaning
SAV1	0xFF 0x00 0xFF 0x00	Start of a displayed line
EAV1	0xFF 0x00 0x00 0xFF	End of a displayed line
SAV2	0xFF 0xFF 0xFF 0x00	Start of a blanked line
EAV2	0xFF 0xFF 0x00 0xFF	End of a blanked line

Pixel values are RGB-encoded, with 3 red bits (0-2), 3 green bits (3-5), and 2 blue bits (6-7).

Projects

1. Design a video test pattern generator, which may generate one of the following patterns: color bar, monochrome horizontal ramp, monochrome vertical ramp. The test pattern generator will support both resolutions (640x480 and 800x600). The test pattern and resolutions will be programmable in software, and controlled through a serial (RS232) connection to the FPGA board.
2. Design a defective pixel correction circuit, capable of automatically detecting and correcting defective pixels. In this context, a defective pixel is one whose value in luminosity differs from the average of its neighbors by more than 25% of the maximum luminosity range. If a pixel is identified as defective, its value will be replaced by the average of its neighbours (top, bottom, left, right). The user will be able to specify, through a RS232 connection with the board, a fixed value for up to 5 pixels in the stream, to be used as an "correction override" mechanism. If a user-defined value is specified for a pixel, then that value is used instead of the average of the neighbours.
3. Design a defective pixel insertion circuit, capable of replacing valid pixel values in the stream with user-programmed values. The circuit will support up to 50 generated defective pixels, in two modes: user controlled, whereby the user specifies the position and value for each of the 50 pixels, random position, whereby the value is the same for all inserted pixels but the positions are randomized at initialization, or random value, whereby the positions are user-specified and the values are randomized at initialization.
4. Design a histogram calculator, which will generate the luminance histogram of 4 user-programmed sections of the video frame. The luminance is defined as the sum of the R,G,B values. The user-programmed sections will be specified through corner coordinates, may overlap, and may be as large as the entire frame or as small as 4 pixels (the corner coordinates of the same section may not overlap).
5. Design a per-color-channel histogram calculator, which generates the histogram of each color channel (R, G, B) over the entire video frame, averaged over the last 300 frames.
6. Design an on-screen display engine, which will be capable of overlaying a user-defined image, of maximum size 100x100 pixels, on top of the existing video stream, with user-configurable transparency. The transparecy will be achieved through the blending of the user-defined image and video stream. The transparency will be defined by a number between 0 and 255, where 0 defines maximum transparency of the overlay, and 255 defines minimum transparency (completely opaque).
7. Design a video rescaler, capable of converting between a 800x600 stream and a 640x480 stream, generating the proper output timing.
8. Design a video rescaler, capable of converting between a 640x480 stream and a 800x600 stream, generating the proper output timing.
9. Design a video "tap", which inspects a passthrough video stream and outputs its contents to the VGA interface of the Nexys 2, with proper timing signals according to the VESA specifications.