

## 6.111 Final Project Abstract

Ben Gelb and Venkat Chandar

For our final project, we will design and implement a real-time video effects processor on the FPGA in the 6.111 lab kit. This device will be capable of receiving an NTSC video signal, modifying the content of the video, and then outputting the modified signal to a video monitor. First, we will describe the functionality of the effects processor. Then, we will describe the user interface and give a high-level system design.

The motivation for our project is to design an effects processor that might be useful for a television station. For example, if a station was broadcasting a sports game, it might want to overlay some statistics or a company logo, and in many situations the commentators might draw on the screen to break down a key play.

Given this motivation, our effects processor will be able to perform several modifications to the video signal. The processor will be able to overlay text, still images, and freehand drawings on top of the video. Additionally, the processor will implement a digital zoom feature, which can re-center and enlarge a user-selected portion of the video image. This could be useful if we wanted to get a better look at a close call. Another obvious feature to include is instant replay, but our preliminary calculations suggest that this will require a large amount of memory. We feel that implementing the functionality already described will be quite challenging. Furthermore, we plan on designing the system so that if we have extra time/resources, we can easily add instant replay at the end.

The effects processor will be controlled by a graphical user interface displayed on a VGA monitor. The interface will allow the user to select stored still images, or enter text using a keyboard. A mouse will be used to select the position where the text or graphics should be superimposed on the video image. The interface will also allow the user to use the mouse to draw lines on top of the video. Finally, the graphical interface will allow control of the zoom functionality.

The system design can be broken into a few major sub-components – an input module, a zoom filter module, an output rendering module, and a control module. The input module is responsible for digitizing the video. The zoom module will re-center the image around a selected center point and enlarge the image by a selectable scale factor. The output rendering module will apply any overlay text or graphics and provide an NTSC output. The control module will take user input from the keyboard and mouse, and store the text overlay messages and graphics. Also, this module will store the position of overlay objects on-screen, and provide a VGA output for the control monitor.

Venkat will focus primarily on the zoom and output rendering modules, while Ben will focus on the control and input modules.