Project Description ECE 484W

The goal of this semester's ECE 484W project is to design, build, test and evaluate an integrated system that performs real-time video processing based on field-programmable gating arrays (FPGAs) and is controlled via a graphical user interface (GUI) on a PC. The data flow in the system is illustrated in Fig. 1.

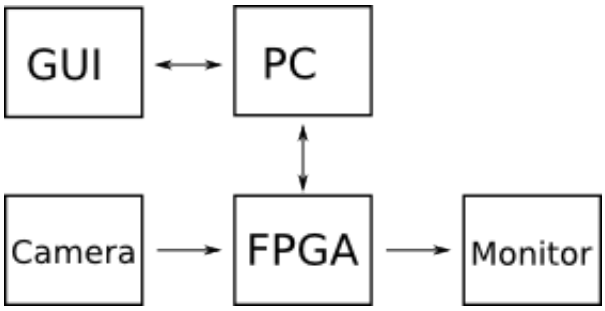

Figure 1. Data flow in the real-time video processing system.

Two types of video processing will be implemented: 1) Overlaying each frame of the video with a user-selected image, such as a logo or subtitles; 2) Histogram equalization (in some modules replaced by adjusting brightness and contrast).

Real-time video processing is challenging because it requires significant data processing in tight time constraints. FPGAs are an excellent choice for such an application because the can be programmed to perform the processing of the video frames in a massively parallel way.

Overlaying user-selected image

Figure 2 shows an example of overlaying a user-selected image. This technique can be useful in video processing for a variety of reasons, including adding information, such as logos or subtitles, or adding visual context such as a map.

Figure 2: Overlay example (overlaid picture contains only text).

Histogram equalization

Histogram equalization (HE) is an image processing technique that increases the contrast of images. The word “histogram” in HE refers to the histogram of pixel brightnesses. Figure 3 illustrates the effect of HE on a low-contrast image. The basic idea is to change the brightness of the individual pixels in a way that a) preserves the brightness order (i.e. if pixel A was brighter than pixel B before HE, it will

still be brighter after HE); b) makes the histogram close to uniform, i.e. all pixel brightnesses occur approximately equally often. Figure 3 illustrates HE in an example.

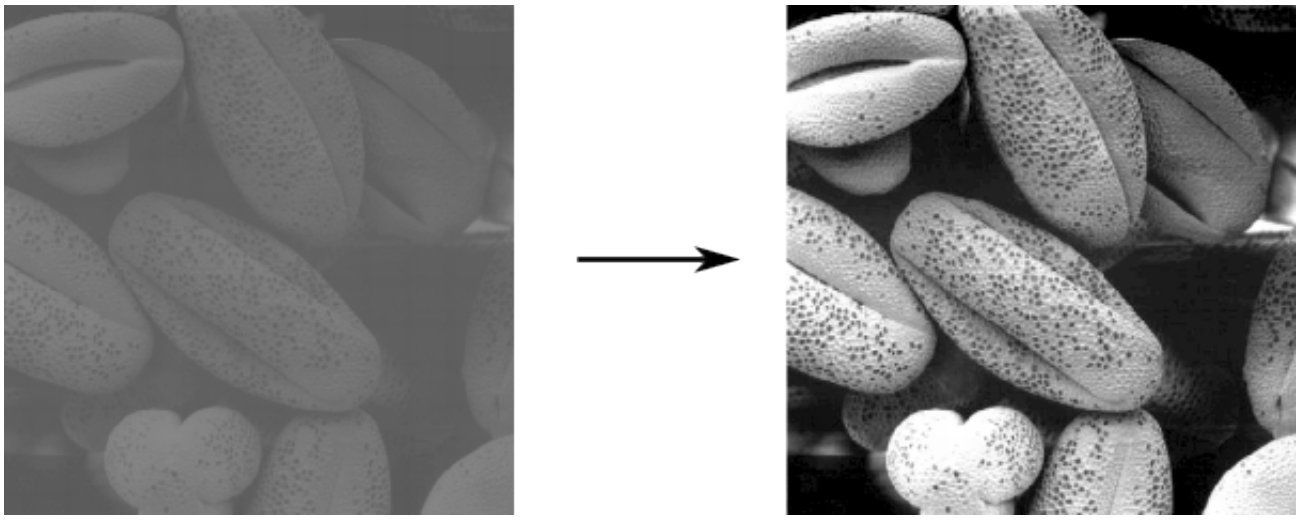

Figure 3: Example of histogram equalization in a low-contrast image.

The project will be divided into a series of milestones. Each milestone will require design, simulation, or verification for one or more components of the overall system. For each milestone, each group submits a report, which has to follow the format of the sample report that is posted on Blackboard and will be discussed in class.

Milestone definitions

Milestone 1: Develop two command line programs, one that perform an image overlay and one that performs HE. Both programs must be developed in C++ (due 9/17).

Milestone 2: Develop a GUI that allows the user to read/write images and to perform image overlay, histogram equalization, and brightness/contrast adjustments on them (due 10/1).

Milestone 3: Program Altera's DE2-115 board (which includes a "Cyclone IV" FPGA) via the Quartus software to perform several simple tasks (due 10/22).

Milestone 4: Establish communication between your PC and the DE2-115 via the RS 232 port (due 11/5).

Milestone 5: Establish complex communication between your PC and the DE2-115, including image transfer, via the RS 232 port (due 11/19).

Full project: Video streaming from a external camera through the DE2-115 to an external monitor, with processing performed by the DE2-115 and controlled by a PC (due 12/11).

This project is intended to introduce and/or reinforce the following concepts:

- Top-down digital system design
- GUI development
- Computer-aided digital circuit design and simulation
- Digital circuit prototyping and debugging
- Serial communication

All the associated digital circuitry, which makes up the system, will be implemented using the Altera-DE2 board, programmed with Altera's Quartus software. The GUI will be developed with Qt Each team will be assigned a lab kit, which includes an Altera-DE2 experimental board with power supply and a ByteBlaster cable. Purchase of some additional parts (chips, ribbon cables, connectors, sockets, replacements for blown chips, etc.) should be requested and will be done using departmental funds.

Specific goals for the project are as follows:

- Understand and implement overlaying of images and histogram equalization
- Develop a GUI interface using Qt, for reading images, overlaying images, performing HE, and saving images.
- Develop communication between PC and Altera-DE2 board
- Design and implement a communication protocol for the system
- Integrate all components (PC, Altera-DE2, Camera, Monitor) to form a working system
- Record and display results (graphical form)
- Write a formal final report to document the project.