ECE 332 – Embedded Systems Lab

Lab 2: Interfacing a camera and implementing image processing algorithms with the DE1-SoC board

Objectives

- Understand the process of capturing image through a camera
- Learn to develop C programs for the ARM hard core processor using Altera monitor program
- Understand basic image processing concepts
- Understand how images are displayed on a monitor

Overview

In this lab, you shall do the following:

- Interface the Terasic D5M camera module to the DE1-SoC
- Capture an image using the camera module
- Implement certain basic image processing algorithms in C and execute them on the ARM hard core processor
- Display the processed image on a monitor

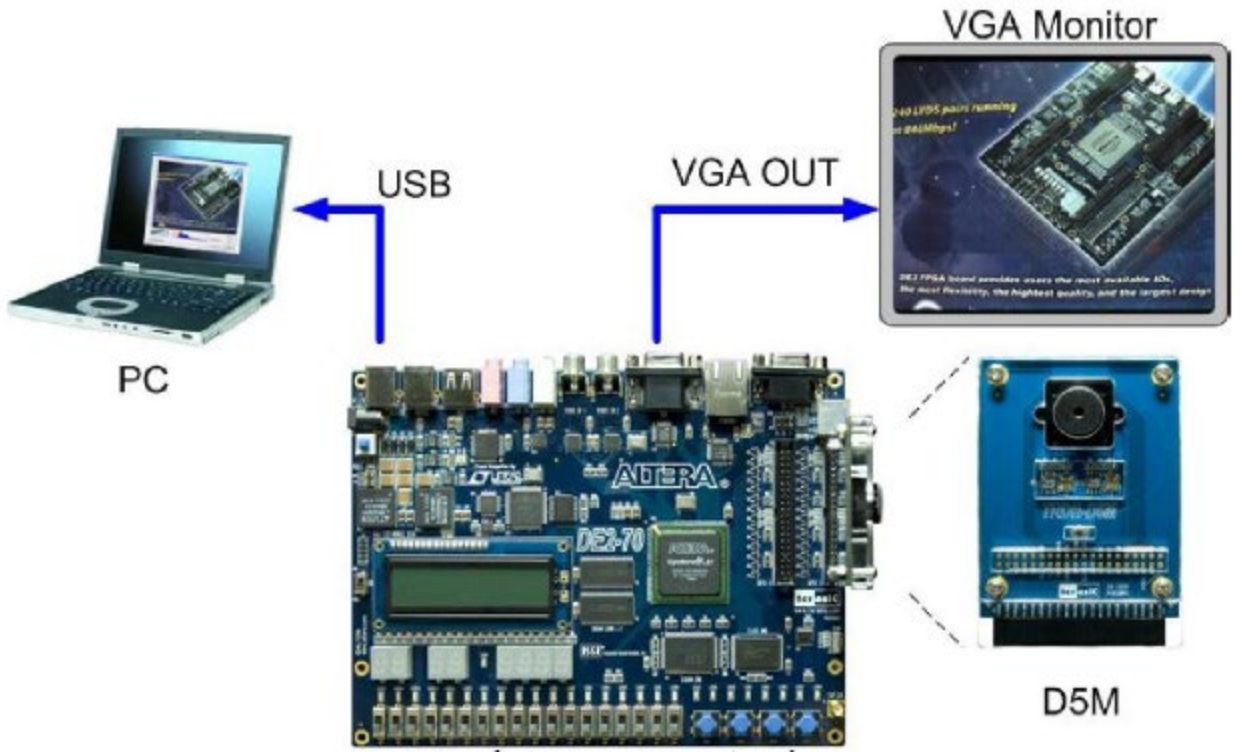
Required reading

- Go through the following sections from the Altera Monitor tutorial provided on the website.
  - Project creation
  - Compiling and loading programs onto the board
  - Executing the programs on the ARM hard core processor

- Understand the provided QSYS system (documentation for the video IP cores)

- Go through the ‘DE1-SoC Computer System with ARM Cortex-A9’ document

Setup



[1] TRDB_D5M_Userguide

http://www.terasic.com.tw/attachment/archive/281/TRDB_D5M_UserGuide.pdf

Design Flow

- Camera – streams high-res video frames
- Video-In Clipper – down-samples the high-res video frames from the camera (provided in the QSYS system)
- Video-In DMA controller – writes the video frames to the FPGA on-chip memory (provided in the QSYS system)
- ARM hardcore - C code (written by you) to
  - Capture a single frame of the video stream
  - Perform the image processing tasks
  - Write the processed image to the FPGA on-chip memory
- Pixel buffer DMA controller - sends the image stored on the FPGA on-chip memory to the VGA controller (provided in the QSYS system)
- VGA controller - displays the processed image on the monitor (provided in the QSYS system)

Video IP cores for image capture and display

- QSYS components
  - Computer system
  - D5M subsystem
  - VGA subsystem
- D5M Subsystem – Clipper, RGB resampler, Video-In DMA controller
- VGA subsystem – Pixel buffer DMA controller, VGA Pixel FIFO, VGA controller
- The image stored on the FPGA on-chip memory is of the resolution 320x240 with 16 bits representing the color (RGB) of each pixel

To get started with the lab

- Download the project content files available on the lab 2 page
- Open the QSYS system and identify the functionality of each component present
- Import the capture_image.amps present in the capture_image folder given to you into the Altera monitor Program
- Software components are present under the capture_image folder
  - capture_image.c

Getting started with C code

- Open the capture_image.amps file in Altera monitor program, compile and run it
- Starter C code in the project performs the following operations
  - Captures one frame of video when any key is pressed
  - Displays the captured image on the monitor upon another key press
- Go through the ‘DE1-SoC Computer System with ARM Cortex-A9’ document
- Understand how to access the peripherals like LEDs, switches, push buttons etc. using C
- Please go through the examples given in the document which might be useful for successful completion of the lab

Image Processing tasks

- Convert the captured color image into black and white
- Implement all of the following tasks
  - Add timestamp onto the image
  - Counter to keep track of number of pictures taken
  - Rotate, mirror, invert image
- You can perform the following challenging tasks for extra credit
  - Edge detection
  - Detect changes between two images

Questions