

LAB 5: DE1-SOC CAMERA SYSTEM AND VGA DISPLAY

I. Introduction

The DE1-SoC and camera system captures images from the D8M-GPIO 8 Megapixel digital camera. This lab will show you how to capture raw data from the digital camera, convert the raw image into RGB format and display the image onto a VGA display.

Terasic D8M-GPIO is an 8-megapixel camera kit with a 2x20 pin GPIO connector interface. It includes a MIPI camera module and a MIPI decoder. The MIPI camera module captures images and sends them out in a MIPI video signal package. This packet is converted into a 10-bit parallel Bayer pattern through a MIPI decoder. The data is transmitted through a 2x20 pin GPIO connector. The MIPI camera module has a built-in Voice Coil Motor (VCM) to control its focal length. Users can adjust the focus via the I2C interface.

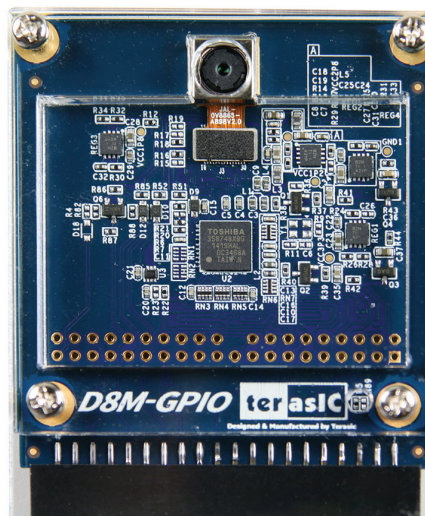


Figure 1. Camera

II. Setup

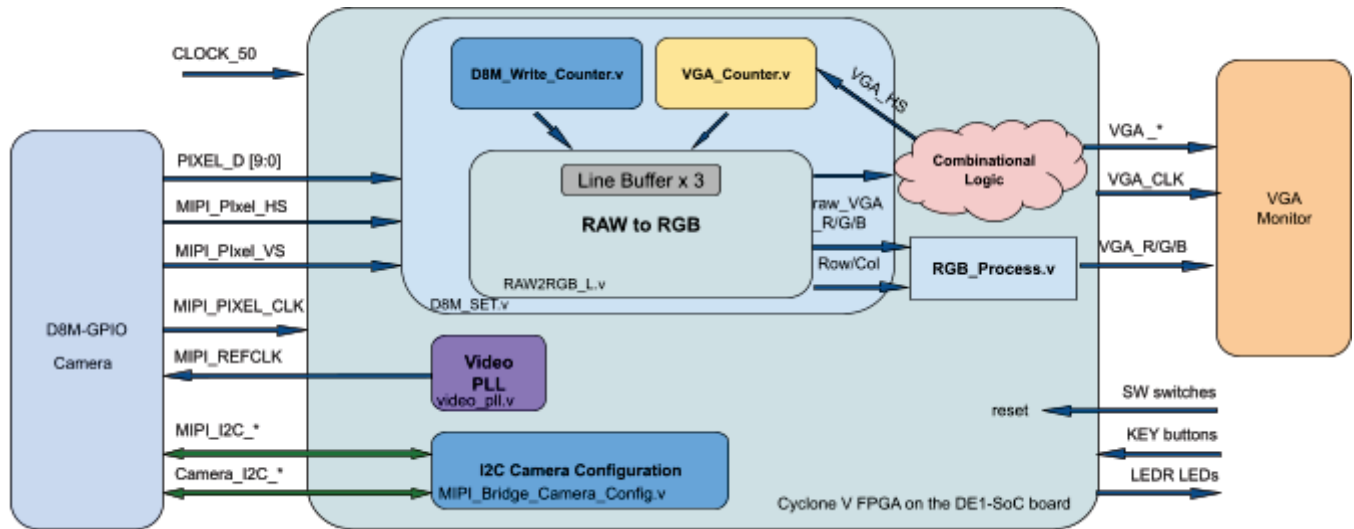
Connect an external display to the DE1-SoC board with a standard VGA cable.

Set the monitor's ratio to 4:3 so the image will appear properly proportioned.

Connect the D8M camera to the DE1-SoC board into the outermost GPIO expansion port (GPIO 1) and the camera face toward the outside of the board. Carefully make sure all the pins are secure and inserted fully into the connectors. Make sure the camera is fully pushed into the GPIO expansion slot. It should be a snug fit, be careful not to break the boards by pressing too hard.

Program the DE1-SoC board with the sample project, you should see the camera video on the display.

III. The Top-Level Module



D8M_SET.v: This module is used to process D8M camera 640×480 60 Hz raw data. It includes three line buffers that are used to store raw data. The line buffers are also used to adjust the frequency difference between the D8M camera and the VGA display.

Port Name (Verilog)	Direction	Width	Description
Clock_50	Input	1	Clock @50 MHz
RESET_SYS_N	Input	1	System reset
CCD_DATA	Input	10	Camera input RAW data
READ_EN	Input	1	Enable the Camera
sCCD_R/G/B	Output	8/8/8	Converted RGB data
col	Output	13	The col range of camera output (0 - 616)
row	Output	13	The row range of camera output (0 - 477)

Table 1. Ports of Verilog Module D8M_SET.v

MIPI_BRIDGE_CAMERA_Config.v: The D8MI2C setting controller, such as set D8M to output 640×480 @60Hz timing. It mainly writes I2C corresponding parameters to D8M MIPI decoder IC register and Camera Sensor IC register respectively through their own I2C buses. MIPI_I2C bus is used to write MIPI decoder IC (I2C Slave Address = 0x1c.), CAMERA_I2C bus is used to write Camera Sensor (IC Slave Address = 0x6c).

IV. Real-Time Image Processing

Implement the following video processing functions, or filters, on the stream of pixels coming from the camera and displayed the VGA monitor in real-time. SW switches switch the functions between the various filters.

Many details of the designs are not specified (e.g., which SW switches to use and how to encode their settings to select filters, and how to switch between various filter sub-functions such as the different levels of RGB color reduction); these details must be chosen by you and described fully in your submission.

Limited by the number of SW switches, try to implement your hardware so that multiple filters can operate at the same time where it makes sense.

Use the SW0 switch for your design's *reset* signal.

1. [30 pts] RGB color filters

- a) Using SW switches, modify the red intensity in the following steps: normal, $\frac{1}{2}$, $\frac{1}{4}$, off
- b) Using SW switches, modify the green intensity in the following steps: normal, $\frac{1}{2}$, $\frac{1}{4}$, off
- c) Using SW switches, modify the blue intensity in the following steps: normal, $\frac{1}{2}$, $\frac{1}{4}$, off

2. [30 pts] Brightness

Using SW switches and appropriate saturation logic, enable the four brightness levels: darker, normal, brighter, and much brighter.

3. [40 pts] Cursor

Draw the following two cursors on the display in purple pixels. Use the four KEY buttons to move the cursor around the screen up/down left/right. The cursor should wrap around the screen when it comes to an edge. Use a counter like the one given in the handout *Variable Frequency Clocking Hardware* to slow down the cursor movement so a user can control its movement easily and precisely.

- a) 4×4 pixel block
- b) A circle 8 pixels across