MTLC

Capacitive Multi-Touch LCD with Camera Module

User Manual

5M Pixel

hsmc

G Sensor

Light-Sensor

terasIC
www.terasic.com

ALTERA.

CONTENTS

Chapter 1

Introduction

The Terasic Capacitive Multi-touch LCD and Camera Module (MTLC) is an all-purpose LCD multimedia color touch-screen for FPGA applications and provides multi-touch gesture and single-touch support. A 5-megapixel digital image sensor, ambient light sensor, and 3-axis accelerometer make up the rich feature-set. A HSMC cable is provided to interface with various Terasic FPGA development boards, such as Terasic DE2-115 and TR4 development boards through a HSMC interface on the MTLC. The kit contains complete reference designs and source code for camera, sensing, and painter demonstrations.

Once the MTLC is connected and preconfigured with an FPGA hardware reference design including several ready-to-run demonstration applications stored on the provided SD card, software developers can use these reference designs as their platform to quickly architect, develop and build complex embedded systems.

Developers can benefit from the use of FPGA-based embedded processing system such as mitigating design risk and obsolescence, design reuse, reducing bill of material (BOM) costs by integrating powerful graphics engines within the FPGA, and lower cost.

Figure 1-1 shows a photograph of MTLC.

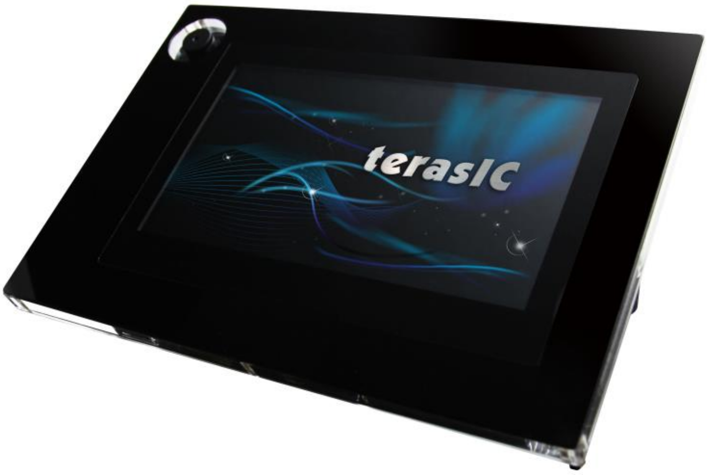


Figure 1-1 Video and Embedded Development Kit – Multi-touch

■ Capacitive LCD Touch Screen

- Equipped with an 7-inch amorphous-TFT-LCD (Thin Film Transistor Liquid Crystal Display) module
- Module composed of LED backlight
- Supports 24-bit parallel RGB interface
- Converting the X/Y touch coordinates to corresponding digital data via Touch controller.

Table 1-1 shows the general physical specifications of the touch screen (Note*).

Table 1-1 General Physical Specifications of the LCD

Item	*Specification*	*Unit*
LCD size	**7-inch (Diagonal)**	**-**
Resolution	**800 x3(RGB) x 480**	**dot**
Dot pitch	**0.1926(H) x0.1790 (V)**	**mm**
Active area	**154.08 (H) x 85.92 (V)**	**mm**
Module size	**164.9(H) x 100.0(V) x 5.7(D)**	**mm**
Surface treatment	**Glare**	**-**
Color arrangement	**RGB-stripe**	**-**
Interface	**Digital**	**-**

■ 5-Megapixel Digital Image Sensor

- Superior low-light performance
- High frame rate
- Low dark current
- Global reset release, which starts the exposure of all rows simultaneously
- Bulb exposure mode, for arbitrary exposure times
- Snapshot-mode to take frames on demand
- Horizontal and vertical mirror image
- Column and row skip modes to reduce image size without reducing field-of-view
- Column and row binning modes to improve image quality when resizing
- Simple two-wire serial interface
- Programmable controls: gain, frame rate, frame size, exposure

Table 1-2 shows the key parameters of the CMOS sensor (Note*).

Table 1-2 Key Performance Parameters of the CMOS sensor

Parameter		*Value*
Active Pixels		2592Hx1944V
Pixel size		2.2umx2.2um
Color filter array		RGB Bayer pattern
Shutter type		Global reset release(GRR)
Maximum data rate/master clock		96Mp/s at 96MHz
Frame rate	Full resolution	Programmable up to 15 fps
	VGA mode	Programmable up to 70 fps
ADC resolution		12-bit
Responsivity		1.4V/lux-sec(550nm)
Pixel dynamic range		70.1dB
SNRMAX		38.1dB
Supply Voltage	Power	3.3V
	I/O	1.7V～3.1V

■ Digital Accelerometer

- Up to 13-bit resolution at +/- 16g
- SPI (3- and 4-wire) digital interface
- Flexible interrupts modes

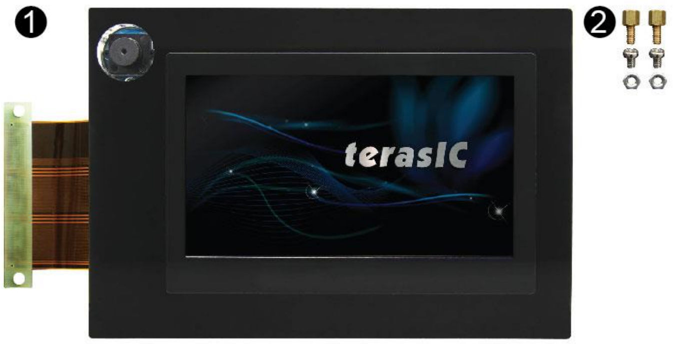
■ Ambient Light Sensor

- Approximates human-eye response
- Precise luminance measurement under diverse lighting conditions
- Programmable interrupt function with user-defined upper and lower threshold settings
- 16-bit digital output with I^2C fast-mode at 400 kHz
- Programmable analog gain and integration time
- 50/60-Hz lighting ripple rejection

Note: for more detailed information of the LCD touch panel and CMOS sensor module, please refer to their datasheets respectively.

1.1 About the Package

The kit includes everything users need to run the demonstrations and develop custom designs, as shown in **Figure 1-2**.



❶ Capacitive Multi-Touch LCD and Camera Module

❷ Screw and copper pillar package

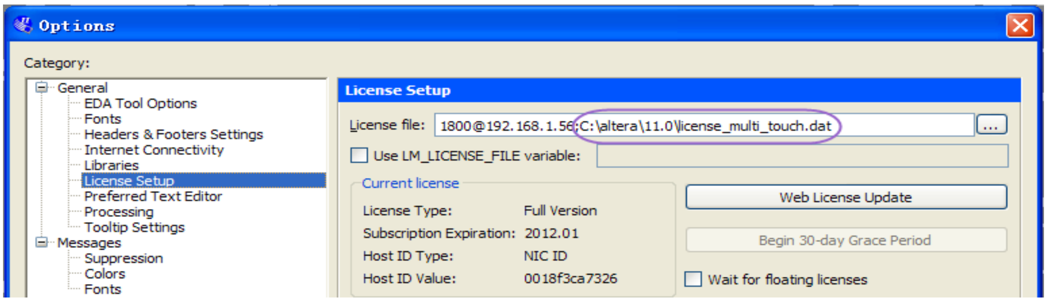
Figure 1-2 MTLC kit package contents

1.2 Setup License for Terasic Multi-Touch IP

To utilize the multi-touch panel in a Quartus II project, the Terasic Multi-Touch IP is required for operation. Error messages will be displayed if the license file for the Multi-Touch IP is not added before compiling projects. The license file is located at:

MTLC System CD\License\license_multi_touch.dat

There are two ways to install the license. The first one is to add the license file (license_multi_touch.dat) to the "License file" listed in Quartus II, as shown in Figure 1-3. In order to reach this window, please navigate through to Quartus II → Tools → License Setup.


Options

Category:
- General
  - EDA Tool Options
  - Fonts
  - Headers & Footers Settings
  - Internet Connectivity
  - Libraries
  - License Setup
  - Preferred Text Editor
  - Processing
  - Tooltip Settings
- Messages
  - Suppression
  - Colors
  - Fonts

License Setup

License file: 1800@192.168.1.56;C:\altera\11.0\license_multi_touch.dat ...

□ Use LM_LICENSE_FILE variable:

Current license

License Type: Full Version
Subscription Expiration: 2012.01
Host ID Type: NIC ID
Host ID Value: 0018f3ca7326

Web License Update

Begin 30-day Grace Period

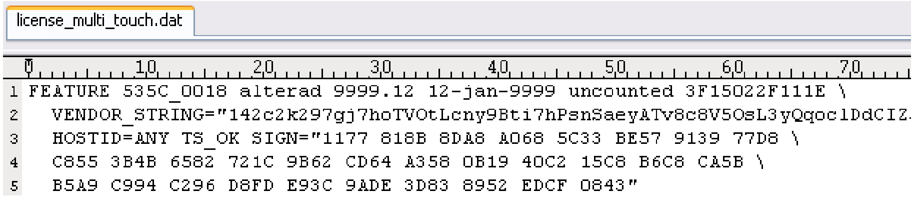
□ Wait for floating licenses

Figure 1-3 License Setup

The second way is to add license content to the existing license file. The procedures are listed below:

Use Notepad or other text editing software to open the file license_multi_touch.dat.

1. The license contains the FEATURE lines required to license the IP Cores as shown in Figure 1-4.



```
FEATURE 535C_0018 alterad 9999.12 12-jan-9999 uncounted 3F15022F111E \
  VENDOR_STRING="142c2k297gj7hoTVOtLcny9Bti7hPsnSaeyATv8c8V5OsL3yQqoclDdCIZ
  HOSTID=ANY TS_OK SIGN="1177 818B 8DA8 A068 5C33 BE57 9139 77D8 \
  C855 3B4B 6582 721C 9B62 CD64 A358 0B19 40C2 15C8 B6C8 CA5B \
  B5A9 C994 C296 D8FD E93C 9ADE 3D83 8952 EDCF 0843"
```

Figure 1-4 Content of license_multi_touch.dat

2. Open your Quartus II license.dat file in a text editor.
3. Copy everything under license_multi_touch.dat and paste it at the end of your Quartus II license file. (Note: Do not delete any FEATURE lines from the Quartus II license file. Doing so will result in an unusable license file.) .
4. Save the Quartus II license file.

1.3 Assembly of MTLC onto Boards with HSMC Connectors

In this section, we would like to introduce how to successfully install the MTLC daughter card to FPGA boards that are equipped with HSMC connectors on top:

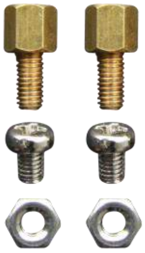


Figure 1-5 Fixed components in a MTLC kit

Inside every MTLC kit package, there should be 2 sets of copper pillars, screws, and nuts as shown in **Figure 1-5**

These parts are used to install the MTLC on the FPGA board through the mounting holes shown in **Figure 1-6.**

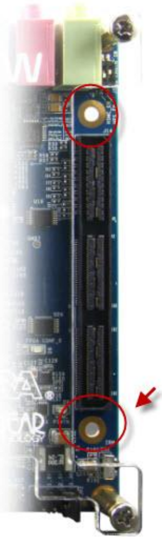


Figure 1-6 Mounting holes next to the HSMC connector

By doing so, this will ensure a rigid connection between the host board and the HSMC cable. Install the copper pillars and nuts on the mounting holes as shown in **Figure 1-7.**

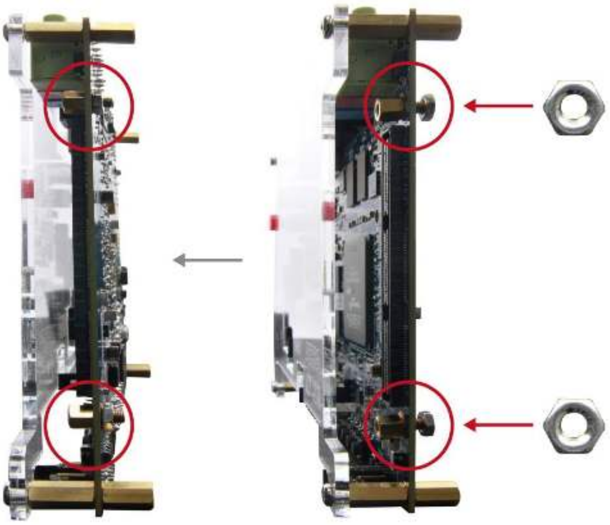


Figure 1-7 Install copper pillars and nuts on the mounting hole

The HSMC cable should be already connected to the MTLC right out of the box. User only needs to connect the HSMC cable to the HSMC connector on the host board as shown in **Figure 1-8.**

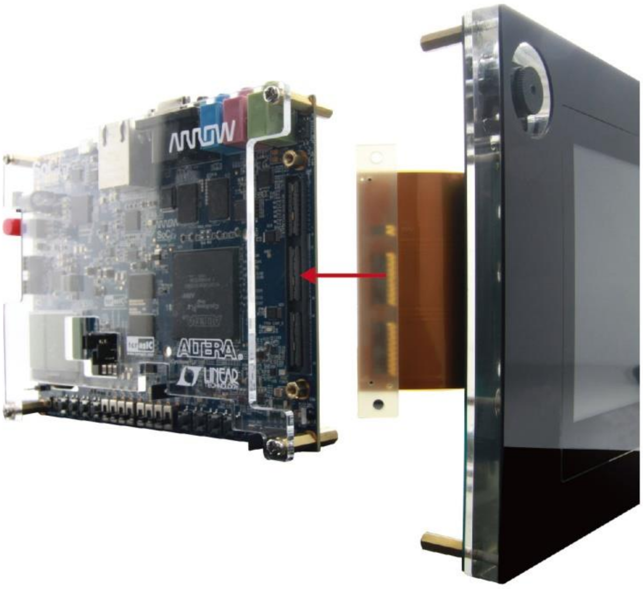


Figure 1-8 Connect the HSMC cable to the HSMC connector on the host board

The final step would be to fasten the screws through the HSMC cable and the copper pillar as shown in **Figure 1-9.**

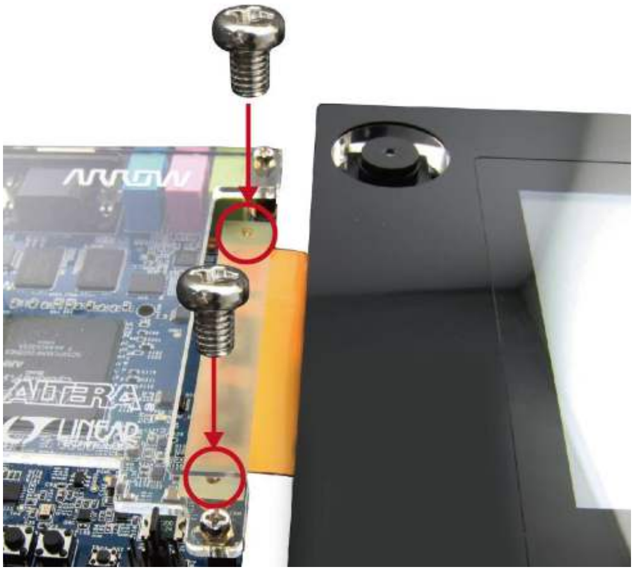


Figure 1-9 Fasten screws through the HSMC cable to the copper pillars

1.4 Connectivity

Here we provide examples of MTLC being connected to different FPGA development boards: Arrow's SoCKit, TR4, DE2-115, and Altera Cyclone V SoC FPGA development board (C5SoC).

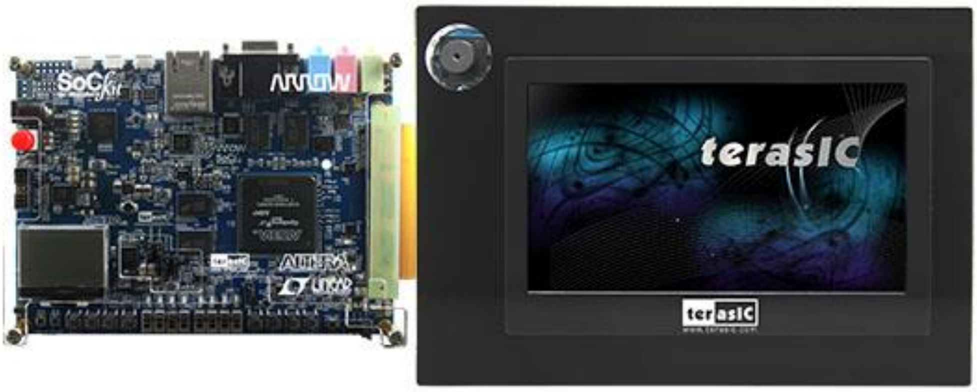


Figure 1-10 MTLC Connect C5S

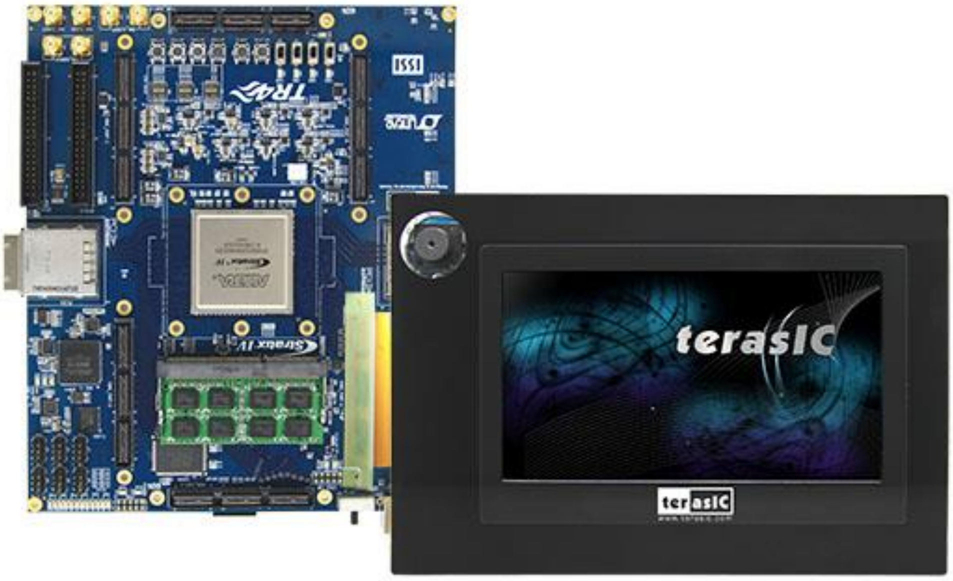


Figure 1-11 MTLC Connect TR4

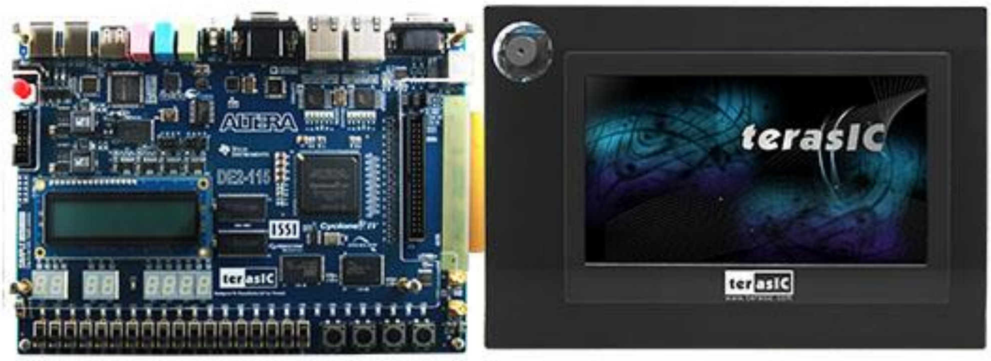


Figure 1-12 MTLC Connect DE2-115

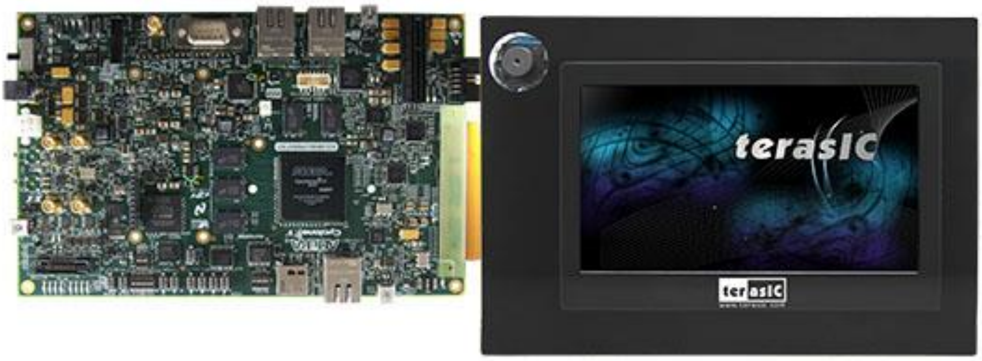


Figure 1-13 MTLC Connect C5SOC

1.5 Getting Help

Here is the contact information if you encounter any problem:

Terasic Technologies
Tel: +886-3-575-0880
Email: support@terasic.com

Chapter 2

Architecture of MTLC

This chapter provides information regarding features and architecture of the Terasic Capacitive Multi-touch LCD and Camera Module.

2.1 Features

The key features of this module are listed as follows:

- 800x480 pixel resolution LCD with 24-bit color depth
- Single touch and two-point multi-gesture support
- 5-Megapixel Digital Image Sensor
- Digital Accelerometer
- Ambient Light Sensor

2.2 Layout and Components

The picture of the MTLC is shown in **Figure 2-1** and **Figure 2-2**. It depicts the layout of the board and indicates the locations of the connectors and key components.

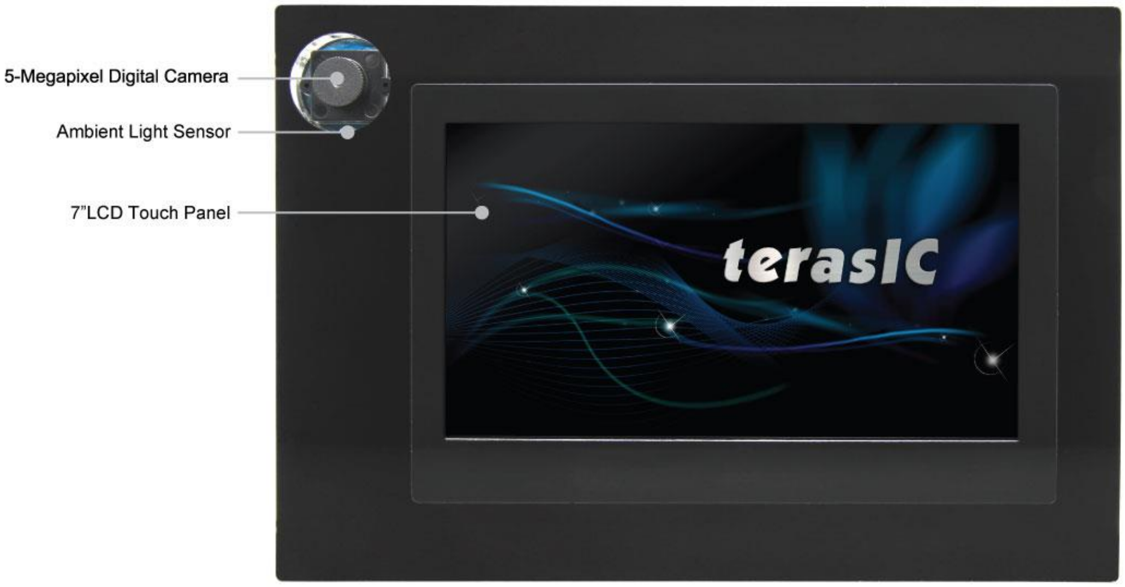


Figure 2-1 MTLC PCB and Component Diagram (Top)

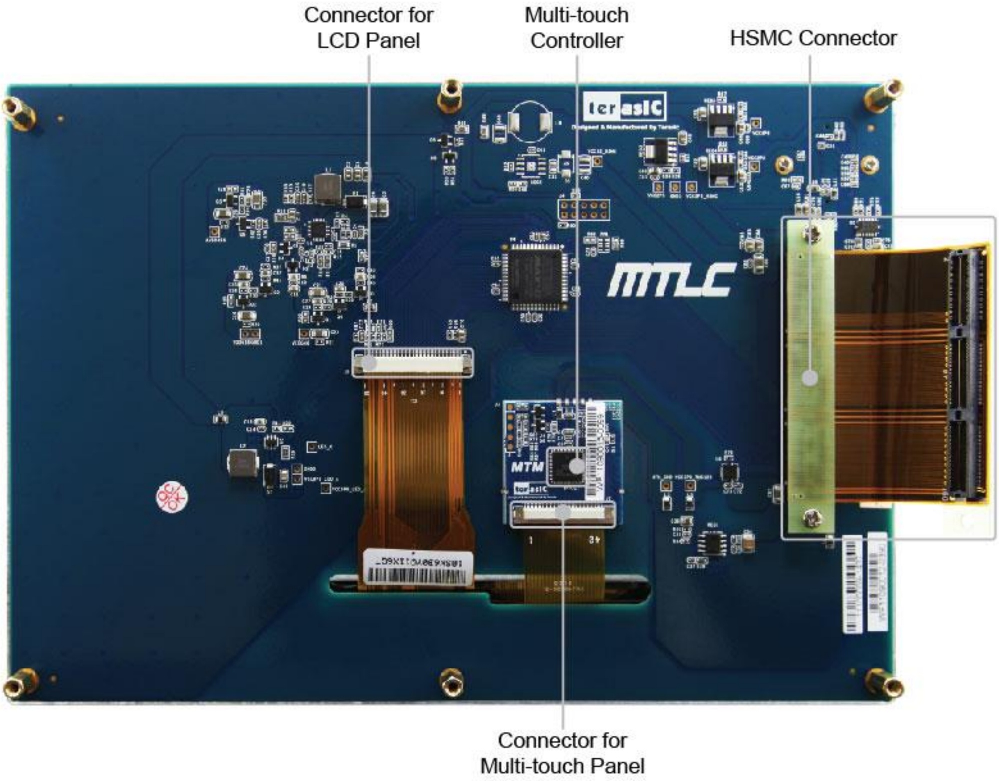


Figure 2-2 MTLC PCB and Component Diagram (Bottom)

2.3 Block Diagram of the MTLC

Figure 2-3 gives the block diagram of the MTLC board. The HSMC connector houses all the wires from peripheral interfaces, connecting to the FPGA of a development kit through the HSMC cable. Thus, the user can configure the FPGA to implement any system design. Figure 2-4 illustrates the connection for MTLC to the Terasic FPGA boards.

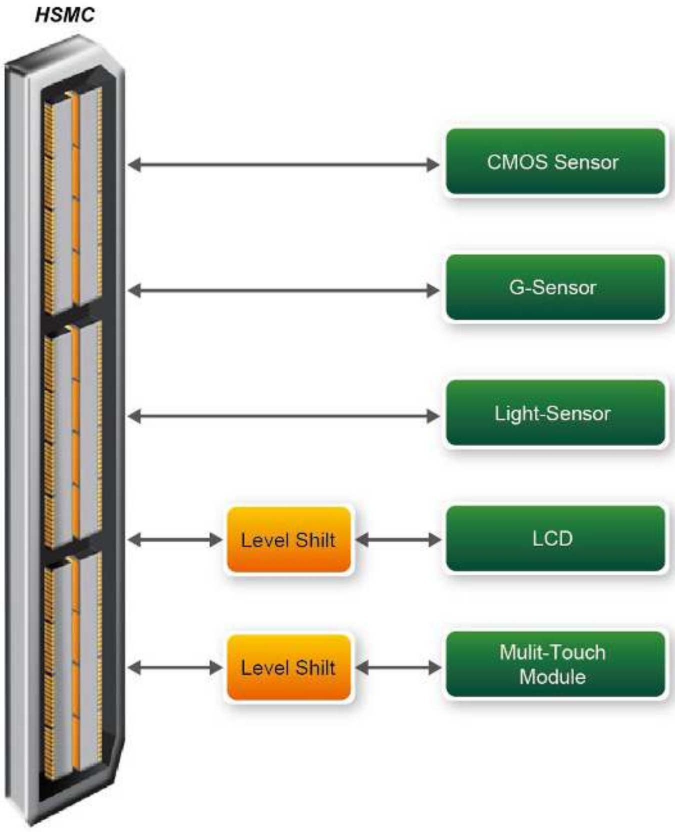


Figure 2-3 Block Diagram of MTLC

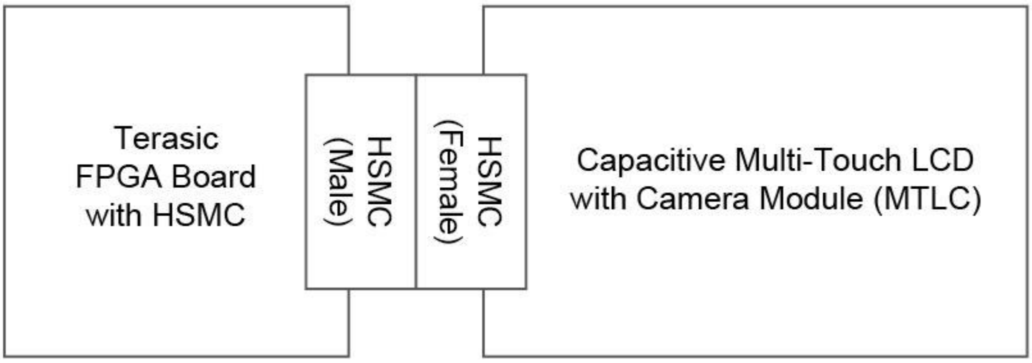

Terasic FPGA Board with HSMC
HSMC (Male)
HSMC (Female)
Capacitive Multi-Touch LCD with Camera Module (MTLC)

Figure 2-4 Connection Diagram of MTLC Kit with Terasic FPGA boards

Chapter 3

Using MTLC

This section describes the detailed information of the components, connectors, and pin assignments of the MTLC.

3.1 Using the 7” LCD Capacitive Touch Screen

The MTLC features a 7-inch capacitive amorphous TFT-LCD panel. The LCD touch screen offers resolution of (800x480) to provide users the best display quality for developing applications. The LCD panel supports 24-bit parallel RGB data interface.

The MTLC is also equipped with a Touch controller, which can read the coordinates of the touch points through a serial port interface.

To display images on the LCD panel correctly, the RGB color data along with the data enable and clock signals must act according to the timing specification of the LCD touch panel as shown in Table 3-1.

Table 3-2 gives the pin assignment information of the LCD touch panel.

Table 3-1 LCD timing specifications

ITEM		SYMBOL	MIN.	TYP.	MAX.	UNIT	NOTE
DCLK	Dot Clock	1/tCLK		33		MHZ	
	DCLK pulse duty	Tcwh	40	50	60	%	
DE	Setup time	Tesu	8			ns	
	Hold time	Tehd	8			ns	
	Horizontal period	tH		1056		tCLK	
	Horizontal Valid	tHA	800			tCLK	
	Horizontal	tHB		2		tCLK	

	Blank			56			
	Vertical Period	tV		525		tH	
	Vertical Valid	tVA	480			tH	
	Vertical Blank	tVB		45		tH	
SYNC	HSYNC setup time	Thst	8			ns	
	HSYNC hold time	Thhd	8			ns	
	VSYNC Setup Time	Tvst	8			ns	
	VSYNC Hold Time	Tvhd	8			ns	
	Horizontal Period	th		1056		tCLK	
	Horizontal Pulse Width	thpw		30		tCLK	thb+thpw=46DCLK is fixed
	Horizontal Back Porch	thb		16		tCLK	
	Horizontal Front Porch	thfp		210		tCLK	
	Horizontal Valid	thd		800		tCLK	
	Vertical Period	tv		525		th	
	Vertical Pulse Width	tvpw		13		th	tvpw + tvb = 23th is fixed
	Vertical Back Porch	tvb		10		th	
	Vertical Front Porch	tvfp		22		th	
	Vertical Valid	tvd	480			th	
DATA	Setup time	Tdsu	8			ns	
	Hold time	Tdsu	8			ns	

Table 3-2 Pin assignment of the LCD touch panel

Signal Name	FPGA Pin No.	Description	I/O Standard
LCD_B0	P28	LCD blue data bus bit 0	2.5V
LCD_B1	P27	LCD blue data bus bit 1	2.5V
LCD_B2	J24	LCD blue data bus bit 2	2.5V
LCD_B3	J23	LCD blue data bus bit 3	2.5V
LCD_B4	T26	LCD blue data bus bit 4	2.5V
LCD_B5	T25	LCD blue data bus bit 5	2.5V
LCD_B6	R26	LCD blue data bus bit 6	2.5V
LCD_B7	R25	LCD blue data bus bit 7	2.5V
LCD_DCLK	V24	LCD Clock	2.5V
LCD_DE	H23	Data Enable signal	2.5V
LCD_DIM	P21	LCD backlight enable	2.5V
LCD_DITH	L23	Dithering setting	2.5V
LCD_G0	P26	LCD green data bus bit 0	2.5V
LCD_G1	P25	LCD green data bus bit 1	2.5V
LCD_G2	N26	LCD green data bus bit 2	2.5V
LCD_G3	N25	LCD green data bus bit 3	2.5V
LCD_G4	L22	LCD green data bus bit 4	2.5V
LCD_G5	L21	LCD green data bus bit 5	2.5V
LCD_G6	U26	LCD green data bus bit 6	2.5V
LCD_G7	U25	LCD green data bus bit 7	2.5V
LCD_HSD	U22	Horizontal sync input.	2.5V
LCD_MODE	L24	DE/SYNC mode select	2.5V
LCD_POWER_CTL	M25	LCD power control	2.5V
LCD_R0	V28	LCD red data bus bit 0	2.5V
LCD_R1	V27	LCD red data bus bit 1	2.5V
LCD_R2	U28	LCD red data bus bit 2	2.5V
LCD_R3	U27	LCD red data bus bit 3	2.5V
LCD_R4	R28	LCD red data bus bit 4	2.5V
LCD_R5	R27	LCD red data bus bit 5	2.5V
LCD_R6	V26	LCD red data bus bit 6	2.5V
LCD_R7	V25	LCD red data bus bit 7	2.5V
LCD_RSTB	K22	Global reset pin	2.5V
LCD_SHLR	H24	Left or Right Display Control	2.5V
LCD_UPDN	K21	Up / Down Display Control	2.5V
LCD_VSD	V22	Vertical sync input.	2.5V
TOUCH _I2C_SCL	T22	touch I2C clock	2.5V
TOUCH _I2C_SDA	T21	touch I2C data	2.5V
TOUCH _INT_n	R23	touch interrupt	2.5V

3.2 Using 5 Megapixel Digital Image Sensor

The MTLC is equipped with a 5 megapixel digital image sensor that provides an active imaging array of 2,592H x 1,944V. It features low-noise CMOS imaging technology that achieves CCD image quality. In addition, it incorporates sophisticated camera functions on-chip such as windowing, column and row skip mode, and snapshot mode.

The sensor can be operated in its default mode or programmed by the user through a simple two-wire serial interface for frame size, exposure, gain settings, and other parameters. Table 3-3 contains the pin names and descriptions of the image sensor module.

Table 3-3 Pin Assignment of the CMOS Sensor

Signal Name	FPGA Pin No.	Description	I/O Standard
CAMERA_PIXCLK	J27	Pixel clock	2.5V
CAMERA_D0	F24	Pixel data bit 0	2.5V
CAMERA_D1	F25	Pixel data bit 1	2.5V
CAMERA_D2	D26	Pixel data bit 2	2.5V
CAMERA_D3	C27	Pixel data bit 3	2.5V
CAMERA_D4	F26	Pixel data bit 4	2.5V
CAMERA_D5	E26	Pixel data bit 5	2.5V
CAMERA_D6	G25	Pixel data bit 6	2.5V
CAMERA_D7	G26	Pixel data bit 7	2.5V
CAMERA_D8	H25	Pixel data bit 8	2.5V
CAMERA_D9	H26	Pixel data bit 9	2.5V
CAMERA_D10	K25	Pixel data bit 10	2.5V
CAMERA_D11	K26	Pixel data bit 11	2.5V
CAMERA_STROBE	E27	Snapshot strobe	2.5V
CAMERA_LVAL	D28	Line valid	2.5V
CAMERA_FVAL	D27	Frame valid	2.5V
CAMERA_RESET_n	F27	Image sensor reset	2.5V
CAMERA_SCLK	AE26	Serial clock	2.5V
CAMERA_TRIGGER	E28	Snapshot trigger	2.5V
CAMERA_SDATA	AE27	Serial data	2.5V
CAMERA_XCLKIN	G23	External input clock	2.5V

3.3 Using the Digital Accelerometer

The MTLC is equipped with a digital accelerometer sensor module. The ADXL345 is a small, thin, ultralow power assumption 3-axis accelerometer with high resolution measurement. Digitalized output is formatted as 16-bit twos complement and can be accessed either using SPI interface or I2C interface. This chip uses the 3.3V CMOS signaling standard. Main applications include medical instrumentation, industrial instrumentation, personal electronic aid and hard disk drive protection etc. Some of the key features of this device are listed below. For more detailed information of better using this chip, please refer to its datasheet which is available on manufacturer's website or under the /datasheet folder of the system CD.

Table 3-4 Pin Names and Descriptions of the GSENSOR Module

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
GSENSOR_INT1	G27	Interrupt 1 output	2.5V
GSENSOR_INT2	G28	Interrupt 2 output	2.5V
GSENSOR_CS_n	F28	Chip Select	2.5V
GSENSOR_ALT_ADDR	K27	I2C Address Select	2.5V
GSENSOR_SDA_SDI_SDIO	K28	Serial Data	2.5V
GSENSOR_SCL_SCLK	M27	Serial Communications Clock	2.5V

3.4 Using the Ambient Light Sensor

The APDS-9300 is a low-voltage digital ambient light sensor that converts light intensity to digital signal output capable of direct I2C communication. Each device consists of one broadband photodiode (visible plus infrared) and one infrared photodiode. Two integrating ADCs convert the photodiode currents to a digital output that represents the irradiance measured on each channel. This digital output can be input to a microprocessor where illuminance (ambient light level) in lux is derived using an empirical formula to approximate the human-eye response. For more detailed information of better using this chip, please refer to its datasheet which is available on manufacturer's website or under the /datasheet folder of the system CD.

Table 3-5 Pin names and Descriptions of Ambient Light Sensor Module

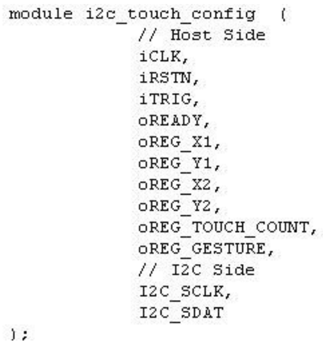
Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
LSENSOR_ADDR_SEL	J25	Chip select	2.5V
LSENSOR_INT	L28	Interrupt output	2.5V
LSENSOR_SCL	J26	Serial Communications Clock	2.5V
LSENSOR_SDA	L27	Serial Data	2.5V

3.5 Using Terasic Multi-Touch IP

Terasic Multi-Touch IP is provided for developers to retrieve user inputs, including multi-touch gestures and single-touch. The file name of this IP is **i2c_touch_config** and it is encrypted. To compile projects with the IP, users need to install the IP license first. For license installation, please refer to section **1.2 Setup License for Terasic Multi-Touch IP** in this document. The license file is located at:

MTLC System CD\License\license_multi_touch.dat

The IP decodes I2C information and outputs coordinate and gesture information. The IP interface is shown below:



```
module i2c_touch_config  (
                    // Host Side
                    iCLK,
                    iRSTN,
                    iTRIG,
                    oREADY,
                    oREG_X1,
                    oREG_Y1,
                    oREG_X2,
                    oREG_Y2,
                    oREG_TOUCH_COUNT,
                    oREG_GESTURE,
                    // I2C Side
                    I2C_SCLK,
                    I2C_SDAT
);
```

The signal purpose of the IP is described in **Table 3-6.** The IP requires a 50MHz signal as a reference clock to the **iCLK** pin and system reset signal to **iRSTN**. **iTRIG**, **I2C_SCLK**, and **IC2_SDAT** pins should be connected of the TOUCH_INT_n, TOUCH_I2C_SCL, and TOUCH_I2C_SDA signals in the 2x20 GPIO header respectively. When **oREADY** rises, it means there is touch activity, and associated information is given in the oREG_X1, oREG_Y1, oREG_X2, oREG_Y2, oREG_TOUCH_COUNT, and oREG_GESTURE pins.

For the control application, when touch activity occurs, it should check whether the value of oREG_GESTURE matched a pre-defined gesture ID defined in **Table 3-7.** If it is not a gesture, it means a single-touch has occurred and the relative X/Y coordinates can be derived from oREG_X1 and oREG_Y1.

Table 3-6 Interface Definitions of Terasic Multi-touch IP

Pin Name	***Direction***	***Description***
iCLK	**Input**	**Connect to 50MHz Clock**
iRSTN	**Input**	**Connect to system reset signal**
iTRIG	**Input**	**Connect to Interrupt Pin of Touch IC**
oREADY	**Output**	**Rising Trigger when following six output data is valid**
oREG_X1	**Output**	**10-bits X coordinate of first touch point**
oREG_Y1	**Output**	**9-bits Y coordinate of first touch point**

oREG_X2	Output	10-bits X coordinate of second touch point
oREG_Y2	Output	9-bits Y coordinate of second touch point
oREG_TOUCH_COUNT	Output	2-bits touch count. Valid value is 0, 1, or 2.
oREG_GESTURE	Output	8-bits gesture ID (See Table 3-7)
I2C_SCLK	Output	Connect to I2C Clock Pin of Touch IC
I2C_SDAT	Inout	Connect to I2C Data Pin of Touch IC

The supported gestures and IDs are shown in Table 3-7.

Table 3-7 Gestures

Gesture	*ID (hex)*
One Point Gesture	
North	0x10
North-East	0x12
East	0x14
South-East	0x16
South	0x18
South-West	0x1A
West	0x1C
North-West	0x1E
Rotate Clockwise	0x28
Rotate Anti-clockwise	0x29
Click	0x20
Double Click	0x22
Two Point Gesture	
North	0x30
North-East	0x32
East	0x34
South-East	0x36
South	0x38
South-West	0x3A
West	0x3C
North-West	0x3E
Click	0x40
Zoom In	0x48
Zoom Out	0x49

Note: The Terasic Multi-Touch IP can also be found under the \IP folder in the system CD as well as the \IP folder in the reference designs.

Chapter 4

MTLC Demonstrations

This chapter gives detailed description of the provided bundles of exclusive demonstrations implemented on the DE2-115 development board with MTLC. These demonstrations are particularly designed (or ported) for MTLC, with the goal of showing the potential capabilities of the kit and showcase the unique benefits of FPGA-based SOPC systems such as reducing BOM costs by integrating powerful graphics and video processing circuits within the FPGA.

Please notice that all the demonstrations in this chapter are using the DE2-115 development board with MTLC. For the demonstrations with other FPGA board can be found on MTLC system CD.

4.1 System Requirements

To run and recompile the demonstrations, you should:

- Install Altera Quartus II 13.1 and NIOS II EDS 13.1 or later edition on the host computer
- Install the USB-Blaster driver software. You can find instructions in the tutorial "Getting Started with Altera's DE2-115 Board" (tut_initialDE2-115.pdf) which is available on the DE2-115 system CD
- Copy the entire \Demonstration\DE2-115 folder from the MTLC system CD to your host computer

4.2 Painter Demonstration

This chapter shows how to control LCD and touch controller to establish a paint demo based on SOPC Builder and Altera VIP Suite. The demonstration shows how multi-touch gestures and single-touch coordinates operate.

Figure 4-1 shows the hardware system block diagram of this demonstration. For LCD display processing, the reference design is developed based on the Altera Video and Image Processing Suite (VIP). The Frame Reader VIP is used for reading display content from the associated video memory, and VIP Video Out is used to display the display content. The display content is filled by NIOS II processor according to users' input.

For multi-touch processing, a Terasic Memory-Mapped IP is used to retrieve the user input, including multi-touch gesture and single-touch coordinates. Note, the IP is encrypted, so the license should be installed before compiling the Quartus II project. For IP--usage details please refer to the section **3.5 Using Terasic Multi-Touch IP** in this document.

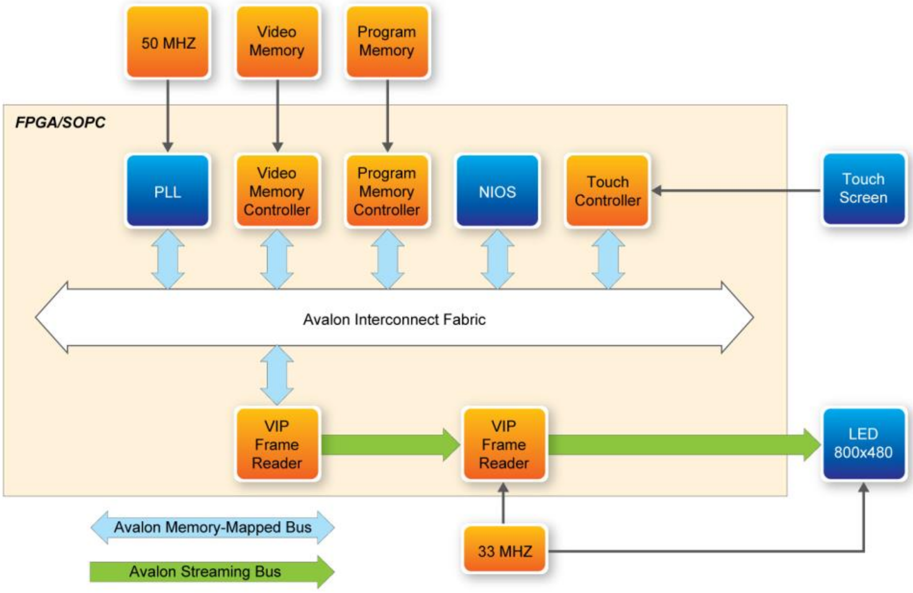


Figure 4-1 Block Diagram of the Painter Demonstration

■ Demonstration Source Code

- Project directory: Painter
- Bit stream used: Painter.sof
- Nios II Workspace: Painter \Software

■ Demonstration Batch File

Demo Batch File Folder: Painter \demo_batch

The demo batch file includes the following files:

- Batch File: test.bat, test_bashrc
- FPGA Configuration File: Painter.sof
- Nios II Program: Painter.elf

■ Demonstration Setup

1. Make sure Quartus II and Nios II are installed on your PC
2. Power on the DE2-115 board
3. Connect USB-Blaster to the DE2-115 board and install USB-Blaster driver if necessary
4. Execute the demo batch file "test.bat" under the batch file folder, Painter \demo_batch
5. After Nios II program is downloaded and executed successfully, you will see a painter GUI in the LCD. Figure 4-2 shows the GUI of the Painter Demo.

- The GUI is classified into three areas: Palette, Canvas, and Gesture. Users can select pen color from the color palette and start painting in the Canvas area. If gesture is detected, the associated gesture symbol is shown in the gesture area. To clear canvas content, press the

"Clear" button.

- Figure 4-3 shows the photo when users paint in the canvas area. Figure 4-4 shows the phone when counter-clockwise rotation gesture is detected. Figure 4-5 shows the photo when zoom-in gesture is detected.

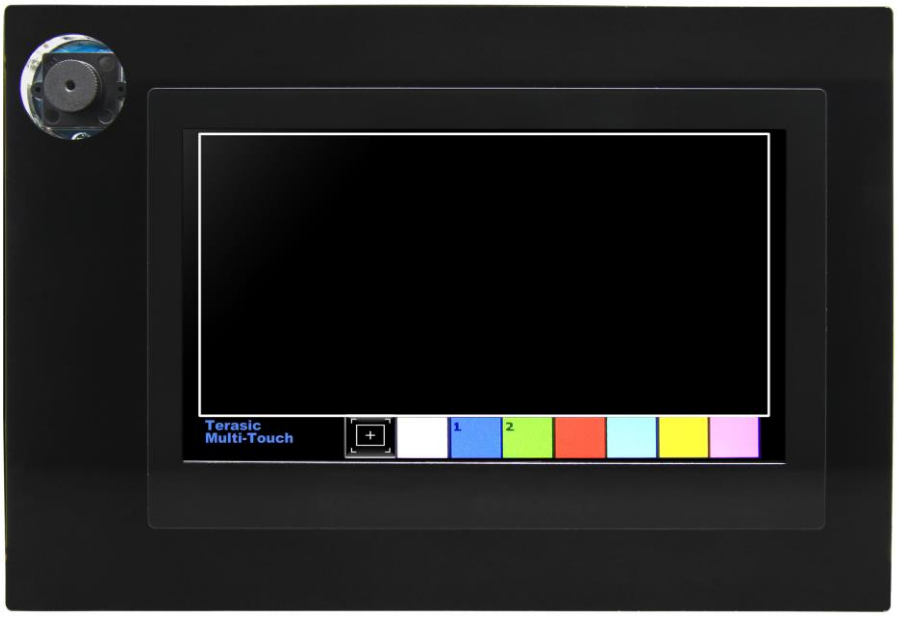


Figure 4-2 GUI of Painter Demo

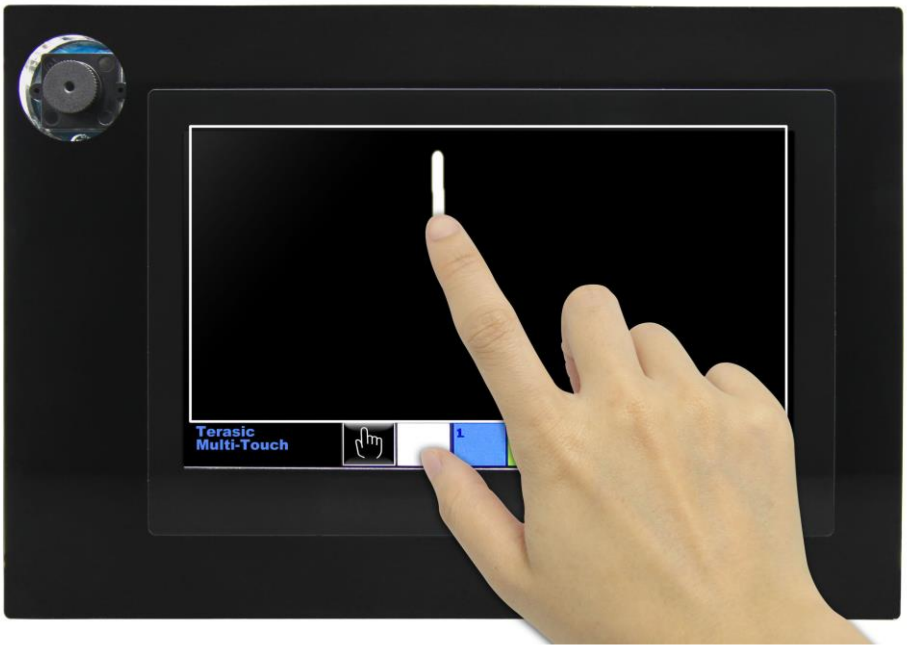


Figure 4-3 Single Touch Painting

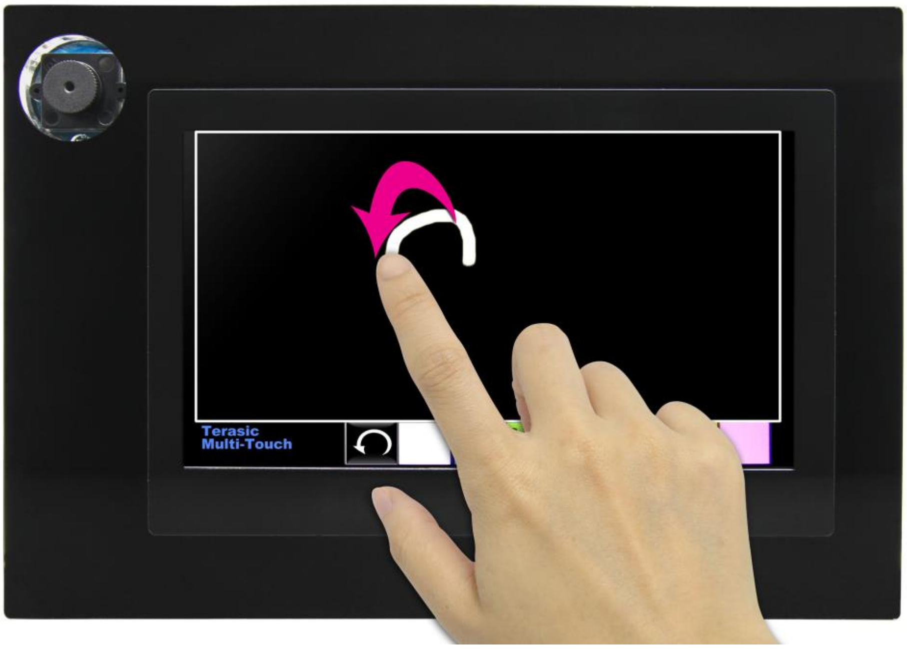


Figure 4-4 Counter-clockwise Rotation Gesture

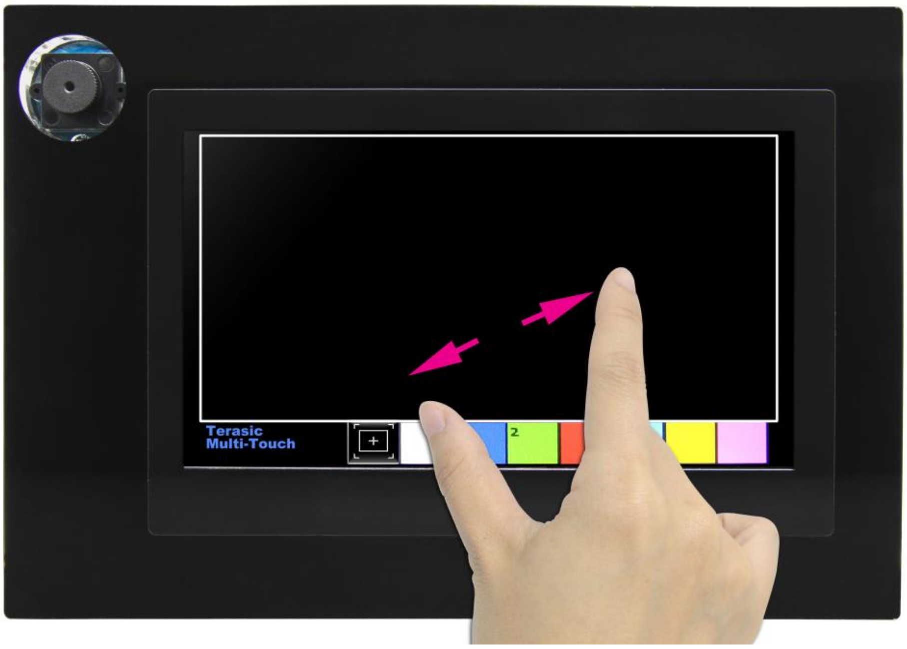


Figure 4-5 Zoom-in Gesture

Note: execute the test.bat under Picture_Viewer\demo_batch will automatically download the .sof and .elf file.

4.3 Picture Viewer

This demonstration shows a simple picture viewer implementation using Nios II-based SOPC system. It reads JPEG images stored on the SD card and displays them on the LCD. The Nios II CPU decodes the images and fills the raw result data into frame buffers in SDRAM. The VEEK-MT(DE2-115 + MTLC) will show the image the buffer being displayed points to. When users touch the LCD touch panel, it will proceed to display the next buffered image or last buffered image. Figure 4-6 shows the block diagram of this demonstration.

The Nios II CPU here takes a key role in the demonstration. It is responsible of decoding the JPEG images and coordinates the works of all the peripherals. The touch panel handling program uses the timer as a regular interrupter and periodically updates the sampled coordinates.

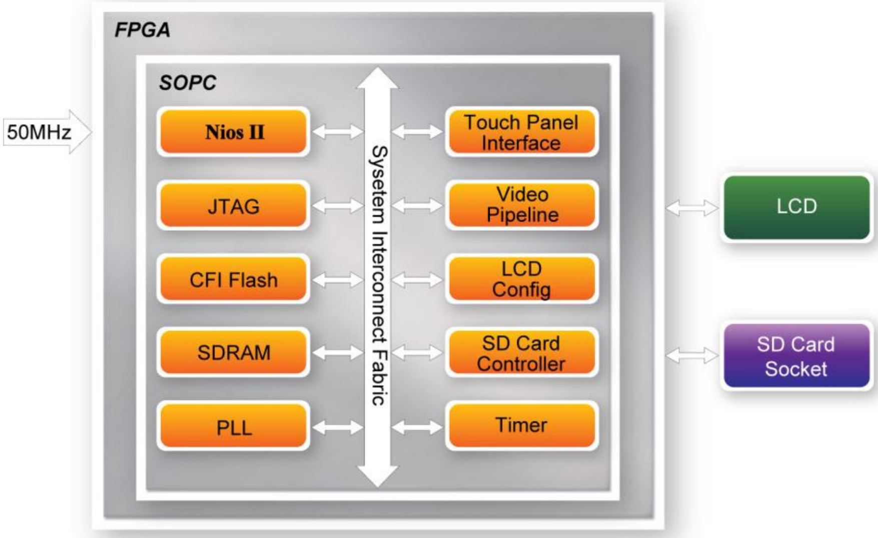


Figure 4-6 Block Diagram of the Picture Viewer Demonstration

■ Demonstration Source Code

- Project directory: Picture_Viewer
- Bit stream used: Picture_Viewer.sof
- Nios II Workspace: Picture_Viewer\Software

■ Demonstration Batch File

Demo Batch File Folder: Picture_Viewer\demo_batch

The demo batch file includes the following files:

- Batch File: Picture_Viewer.bat, Picture_Viewer _bashrc
- FPGA Configuration File: Picture_Viewer.sof
- Nios II Program: Picture_Viewer.elf

■ Demonstration Setup

1. Format your SD card into FAT16 format
2. Place the jpg image files to the \jpg subdirectory of the SD card. For best display result, the image should have a resolution of 800x480 or the multiple of that
3. Insert the SD card to the SD card slot on the VEEK-MT
4. Load the bit stream into the FPGA on the VEEK-MT
5. Run the Nios II Software under the workspace Picture_Viewer\Software (Note*)
6. After loading the application you will see a slide show of pictures on the SD card.
7. The next image will be displayed after the delay period.
8. You can control the slide show as follows :

- Press Forward (⏩) to advance, Reverse (⏪) to go back to previous image, Play/Stop (⏯) to play the slide or stop it.
- On the top corner you will see the delay-period (seconds). You can increase or decrease the delay period by touching the + (➕) or – (➖) buttons.
- The max delay is 120 seconds, the min delay is 1 second, and the default delay is 10 seconds.
- You can hide the control buttons by clicking on the Hide button located at the top left corner of the touch screen. Touch anywhere on the screen to resume and to return to menu.

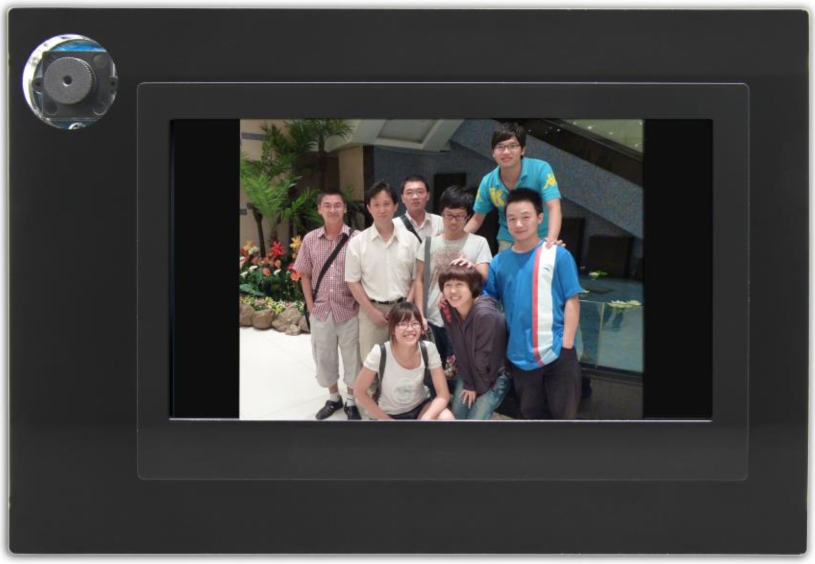


Figure 4-7 Picture Viewer Demonstration

Note: execute the Picture_Viewer.bat under Picture_Viewer\demo_batch will automatically download the .sof and .elf file.

4.4 Video and Image Processing

The Video and Image Processing (VIP) Example Design demonstrates dynamic scaling and clipping of a standard definition video stream in either National Television System Committee (NTSC) or Phase Alternation Line (PAL) format and picture-in-picture mixing with a background layer. The video stream is output in high resolution (800x480) LCD touch panel.

The example design demonstrates a framework for rapid development of video and image processing systems using the parameterizable MegaCore® functions that are available in the Video and Image Processing Suite. Available functions are listed in Table 4-1. This demonstration needs the Quartus II license file includes the VIP suite feature.

Table 4-1 VIP IP Cores Functions

IP MegaCore Function	*Description*
Frame Reader	Reads video from external memory and outputs it as a stream.
Control Synchronizer	Synchronizes the changes made to the video stream in real time between two functions.
Switch	Allows video streams to be switched in real time.
Color Space Converter	Converts image data between a variety of different color spaces such as RGB to YCrCb.
Chroma Resampler	Changes the sampling rate of the chroma data for image frames, for example from 4:2:2 to 4:4:4 or 4:2:2 to 4:2:0.
2D FIR Filter	Implements a 3 x 3, 5 x 5, or 7 x 7 finite impulse response (FIR) filter on an image data stream to smooth or sharpen images.
Alpha Blending Mixer	Mixes and blends multiple image streams—useful for implementing text overlay and picture-in-picture mixing.
Scaler	A sophisticated polyphase scaler that allows custom scaling and real-time updates of both the image sizes and the scaling coefficients.
Deinterlacer	Converts interlaced video formats to progressive video format using a motion adaptive deinterlacing algorithm. Also supports 'bob' and "weave" algorithms
Test Pattern Generator	Generates a video stream that contains still color bars for use as a test pattern.
Clipper	Provides a way to clip video streams and can be configured at compile time or at run time.
Color Plane Sequencer	Changes how color plane samples are transmitted across the Avalon-ST interface. This function can be used to split and join video streams, giving control over the routing of color plane samples.
Frame Buffer	Buffers video frames into external RAM. This core supports double or triple-buffering with a range of options for frame dropping and repeating.
2D Median Filter	Provides a way to apply 3 x 3, 5 x 5, or 7 x 7 pixel median filters to video images.
Gamma Corrector	Allows video streams to be corrected for the physical properties of display devices.
Clocked Video Input/Output	These two cores convert the industry-standard clocked video format (BT-656) to Avalon-ST video and vice versa.

These functions allow you to fully integrate common video functions with video interfaces, processors, and external memory controllers. The example design uses an Altera Cyclone® IV E EP4CE115F29 featured VEEK-MT (DE2-115 + MTLC).

A video source is input through an analog composite port on VEEK-MT which generates a digital output in ITU BT656 format. A number of common video functions are performed on this input stream in the FPGA. These functions include clipping, chroma resampling, motion adaptive deinterlacing, color space conversion, picture-in-picture mixing, and polyphase scaling.

The input and output video interfaces on the VEEK-MT are configured and initialized by software running on a Nios® II processor. Nios II software demonstrates how to control the clocked video input, clocked video output, and mixer functions at run-time is also provided. The video system is implemented using the SOPC Builder system level design tool. This abstracted design tool provides an easy path to system integration of the video processing data path with a NTSC or PAL video input, VGA output, Nios II processor for configuration and control. The Video and Image Processing Suite MegaCore functions have common open Avalon-ST data interfaces and Avalon Memory-Mapped (Avalon-MM) control interfaces to facilitate connection of a chain of video functions and video system modeling. In addition, video data is transmitted between the Video and Image Processing Suite functions using the Avalon-ST Video protocol, which facilitates building run-time controllable systems and error recovery.

Figure 4-8 shows the Video and Image Processing block diagram.

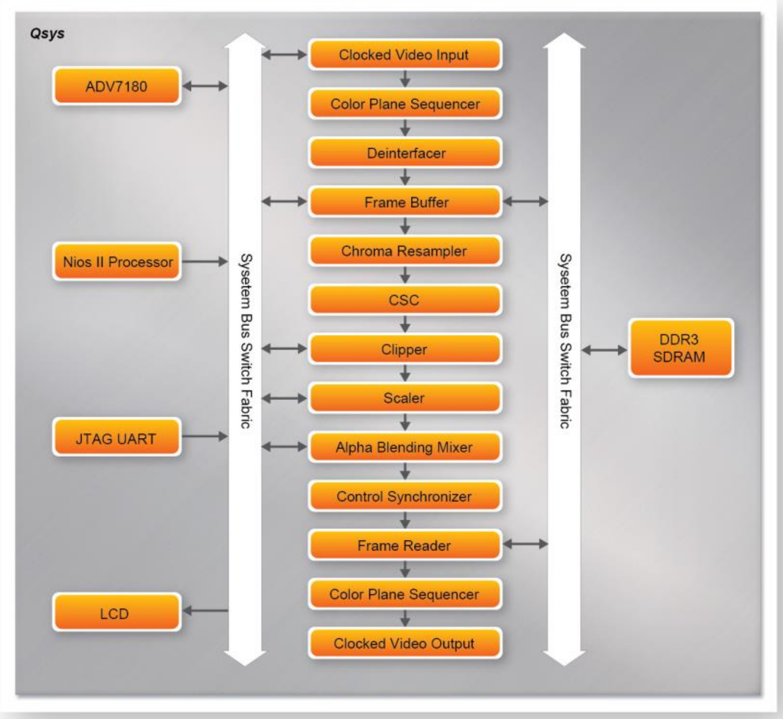


Figure 4-8 VIP Example SOPC Block Diagram (Key Components)

■ Demonstration Source Code

- Project directory: VIP
- Bit stream used: VIP.sof
- Nios II Workspace: VIP\Software

■ Demonstration Batch File

Demo Batch File Folder: VIP\demo_batch

The demo batch file includes the following files:

- Batch File: VIP.bat, VIP_bashrc
- FPGA Configuration File: VIP.sof
- Nios II Program: VIP.elf

■ Demonstration Setup

- Connect a DVD player's composite video output (yellow plug) to the Video-IN RCA jack (J12) of the VEEK-MT. The DVD player has to be configured to provide NTSC output or PAL output
- Connect the VGA output of the VEEK-MT to a VGA monitor (both LCD and CRT type of monitors should work)
- Load the bit stream into FPGA (note*)
- Run the Nios II IDE and choose VIP\Software as the workspace. Click on the Run button (note *)
- Press the screen of the VEEK-MT and drag the video frame box will result in scaling the playing window to any size, as shown in **Figure 4-9**

Note:

(1).Executing VIP\demo_batch\ VIP.bat will download .sof and .elf files.

(2).You may need additional Altera VIP suite Megacore license features to recompile the project.

Figure 4-10 illustrates the setup for this demonstration.

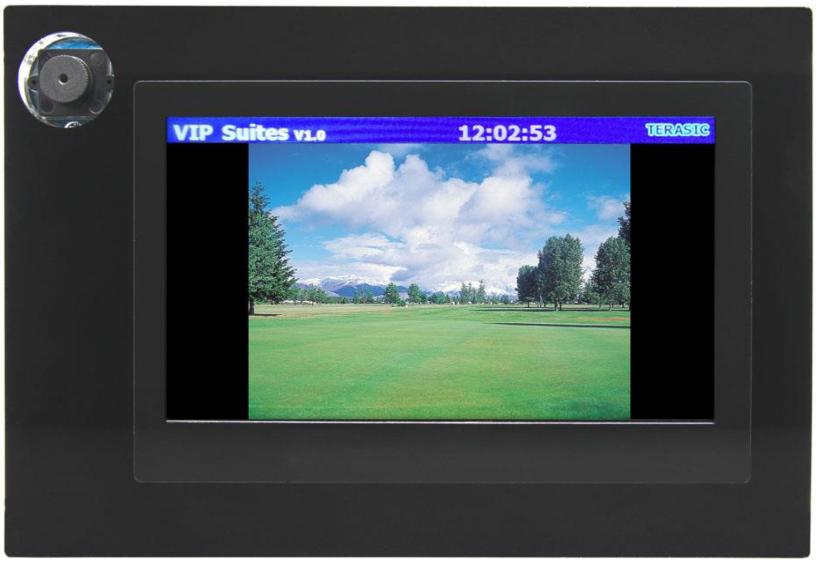


Figure 4-9 VIP Demonstration

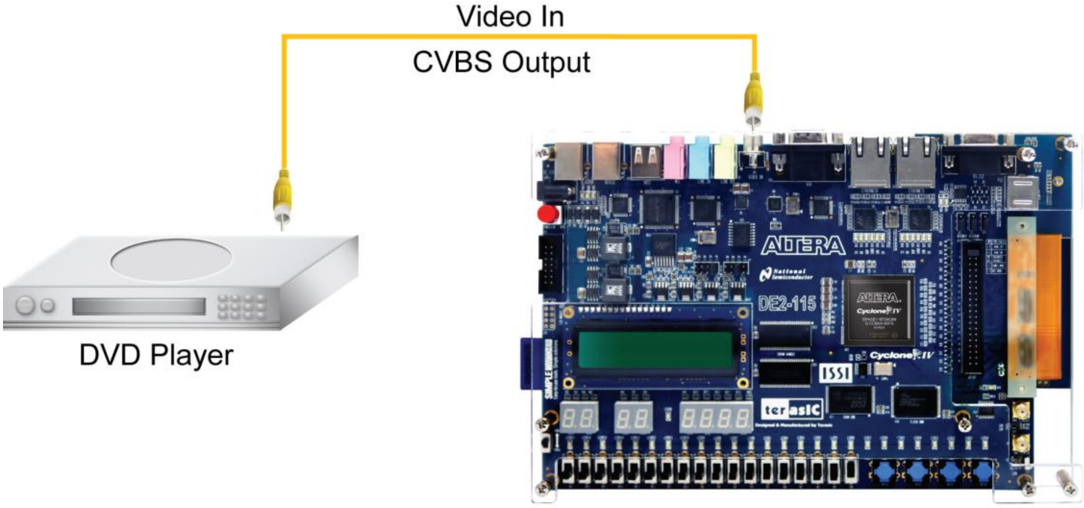


Figure 4-10 Setup for the VIP Demonstration

4.5 Camera Application

This demonstration shows a digital camera reference design using the 5 megapixel CMOS sensor and 7-inch LCD modules on the VEEK-MT (MTLC+DE2-115). The CMOS sensor module sends the raw image data to FPGA on the DE2-115 board, the FPGA on the board handles image processing part and converts the data to RGB format to display on the LCD module. The I2C Sensor Configuration module is used to configure the CMOS sensor module. Figure 4-11 shows the block diagram of the demonstration.

As soon as the configuration code is downloaded into the FPGA, the I2C Sensor Configuration block will initial the CMOS sensor via I2C interface. The CMOS sensor is configured as follow:

- Row and Column Size: 800 * 480
- Exposure time: Adjustable
- Pix clock: MCLK*2 = 25*2 = 50MHz
- Readout modes: Binning
- Mirror mode: Line mirrored

According to the settings, we can calculate the CMOS sensor output frame rate is about 44.4 **fps**.

After the configuration, The CMOS sensor starts to capture and output image data streams, the CMOS sensor Capture block extracts the valid pix data streams based on the synchronous signals from the CMOS sensor. The data streams are generated in Bayer Color Pattern format. So it's then converted to RGB data streams by the RAW2RGB block.

After that, the Multi-Port SDRAM Controller acquires and writes the RGB data streams to the SDRAM which performs as a frame buffer. The Multi-Port SDRAM Controller has two write ports and read ports also with 16-bit data width each. The writing clock is the same as CMOS sensor pix clock, and the reading clock is provided by the LCD Controller, which is 33MHz.

Finally, the LCD controller fetches the RGB data from the buffer and displays it on the LCD panel continuously. Because the resolution and timing of the LCD is compatible with WVGA@800*480, the LCD controller generates the same timing and the frame rate can achieve about 25 **fps**.

For the objective of a better visual effect, the CMOS sensor is configured to enable the left right mirror mode. User could disable this functionality by modifying the related register value being written to CMOS controller chip.

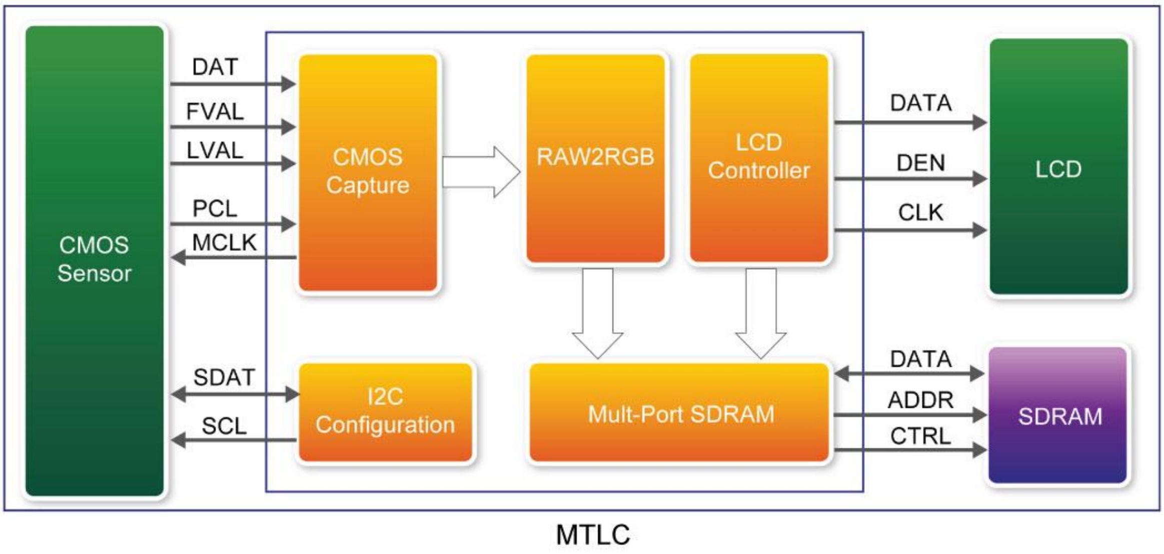


Figure 4-11 Block Diagram of the Digital Camera Design

■ Demonstration Source Code

- Project directory: Camera
- Bit stream used: Camera.sof

■ Demonstration Batch File

Demo Batch File Folder: Camera\demo_batch

The demo batch file includes the following files:

- Batch File: test.bat
- FPGA Configuration File: Camera.sof

■ Demonstration Setup

- Load the bit stream into FPGA by executing the batch file ‘test.bat’ under Camera\demo_batch\ folder
- The system enters the FREE RUN mode automatically. Press KEY[0] on the DE2-115 board to reset the circuit
- Press KEY[2] to take a shot of the photo; you can press KEY[3] again to switch back to FREE RUN mode and you should be able to see whatever the camera captures on the LCD display
- User can use the SW[0] and KEY[1] to set the exposure time for brightness adjustment of the image captured. When SW[0] is set to Off, the brightness of image will be increased as KEY[1] is pressed longer. If SW[0] is set to On, the brightness of image will be decreased as KEY[1] is pressed shorter
- User can use SW[17] to mirror image of the line. However, remember to press KEY[0] after toggle SW[17]

Note: execute the test.bat under Camera\demo_batch will automatically download the .sof file.

Table 4-2 summarizes the functional keys of the digital camera. **Figure 4-12** gives a run-time photograph of the demonstration.

Table 4-2 The functional keys of the digital camera demonstration

Component	***Function Description***
KEY[0]	**Reset circuit**
KEY[1]	**Set the new exposure time (use with SW[0])**
KEY[2]	**Trigger the Image Capture (take a shot)**
KEY[3]	**Switch to Free Run mode**
SW[0]	**Off: Extend the exposure time**
	On: Shorten the exposure time
SW[17]	**Mirror mode**
HEX[7:0]	**Frame counter (Display ONLY)**

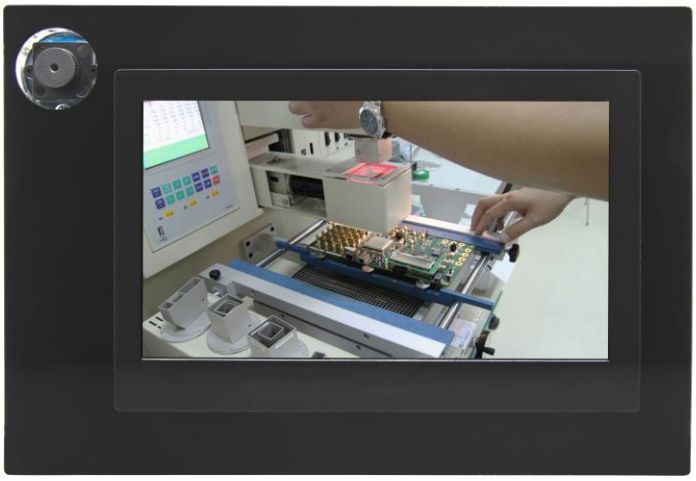


Figure 4-12 Screen Shot of the Camera Demonstration

4.6 Video and Image Processing for Camera

The Video and Image Processing (VIP) for Camera Example Design demonstrates dynamic scaling and clipping of a standard definition video stream in RGB format and picture-in-picture mixing with a background layer. The video stream is output in high resolution (800×480) on LCD touch panel.

The example design demonstrates a framework for rapid development of video and image processing systems using the parameterizable MegaCore® functions that are available in the Video and Image Processing Suite. Available functions are listed in Table 4-2. This demonstration needs the Quartus II license file includes the VIP suite feature.

These functions allow you to fully integrate common video functions with video interfaces, processors, and external memory controllers. The example design uses an Altera Cyclone® IV E EP4CE115F29 featured on the VEEK-MT (MTLC+DE2-115).

A video source is input through the CMOS sensor on VEEK-MT which generates a digital output in RGB format. A number of common video functions are performed on this input stream in the FPGA. These functions include clipping, chroma resampling, motion adaptive deinterlacing, color space conversion, picture-in-picture mixing, and polyphase scaling.

The input and output video interfaces on the VEEK-MT are configured and initialized by software running on a Nios® II processor. Nios II software demonstrates how to control the clocked video input, clocked video output, and mixer functions at run-time is also provided. The video system is implemented using the SOPC Builder system level design tool. This abstracted design tool provides an easy path to system integration of the video processing data path with a NTSC or PAL video input, VGA output, Nios II processor for configuration and control. The Video and Image Processing Suite MegaCore functions have common open Avalon-ST data interfaces and Avalon Memory-Mapped (Avalon-MM) control interfaces to facilitate connection of a chain of video functions and video system modeling. In addition, video data is transmitted between the Video and Image Processing Suite functions using the Avalon-ST Video protocol, which facilitates building run-time controllable systems and error recovery.

For the objective of a better visual effect, the CMOS sensor is configured to enable the left right mirror mode. User could disable this functionality by modifying the related register value being written to CMOS controller chip.

Figure 4-13 shows the Video and Image Processing block diagram.

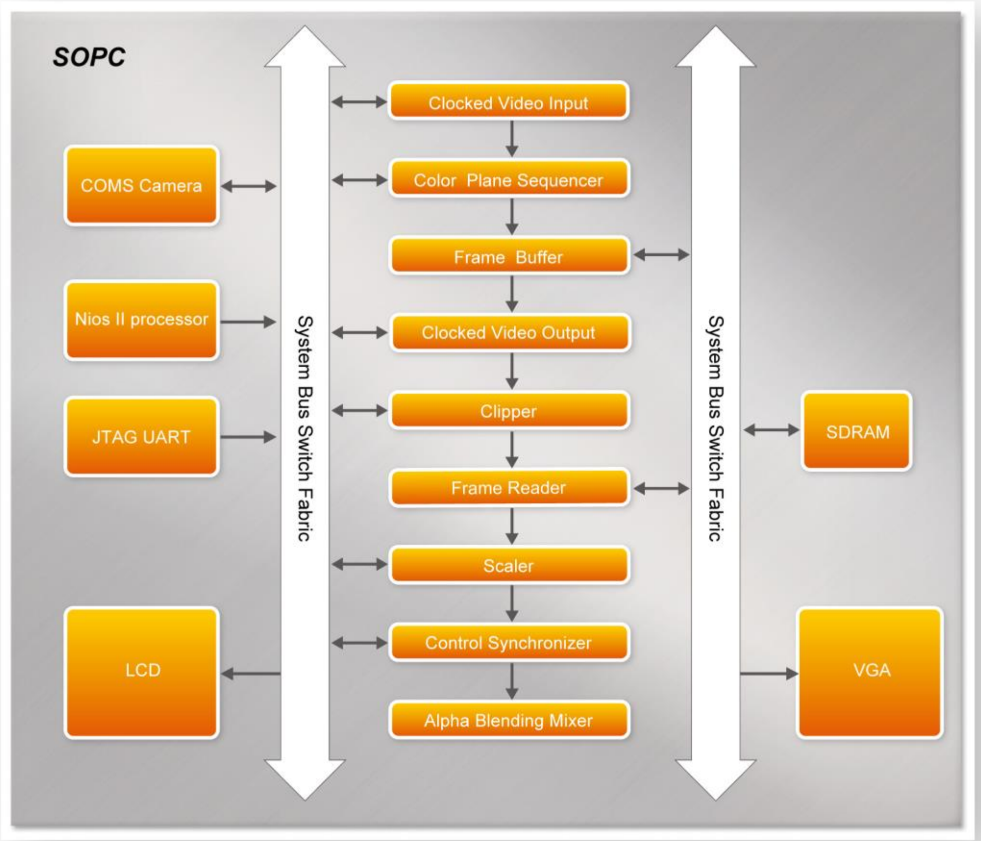


Figure 4-13 VIP Camera Example SOPC Block Diagram (Key Components)

■ Demonstration Source Code

- Project directory: VIP_Camera
- Bit stream used: VIP_Camera.sof
- Nios II Workspace: VIP_Camera \Software

■ Demonstration Batch File

Demo Batch File Folder: VIP_Camera\demo_batch

The demo batch file includes the following files:

- Batch File: VIP_Camera.bat, VIP_Camera _bashrc
- FPGA Configuration File: VIP_Camera.sof
- Nios II Program: VIP_Camera.elf

■ Demonstration Setup

- Connect the VGA output of the VEEK-MT to a VGA monitor (both LCD and CRT type of monitors should work)
- Load the bit stream into FPGA (note*)
- Run the Nios II and choose VIP_Camera\Software as the workspace. Click on the Run button (note *)
- The system enters the FREE RUN mode automatically. Press KEY[0] on the DE2-115 board to reset the circuit
- Press KEY[2] to stop run; you can press KEY[3] again to switch back to FREE RUN mode and you should be able to see whatever the camera captures on the VGA display
- User can use SW[17] to mirror image of the line. However, remember to press KEY[0] after toggle SW[17]
- Press and drag the video frame box will result in scaling the playing window to any size, as shown in Figure 4-14

Note:

(1).Execute VIP_Camera\demo_batch\VIP_CameraA.bat will download .sof and .elf files.

(2).You may need additional Altera VIP suite Megacore license features to recompile the project.

Figure 4-14 illustrates the setup for this demonstration.

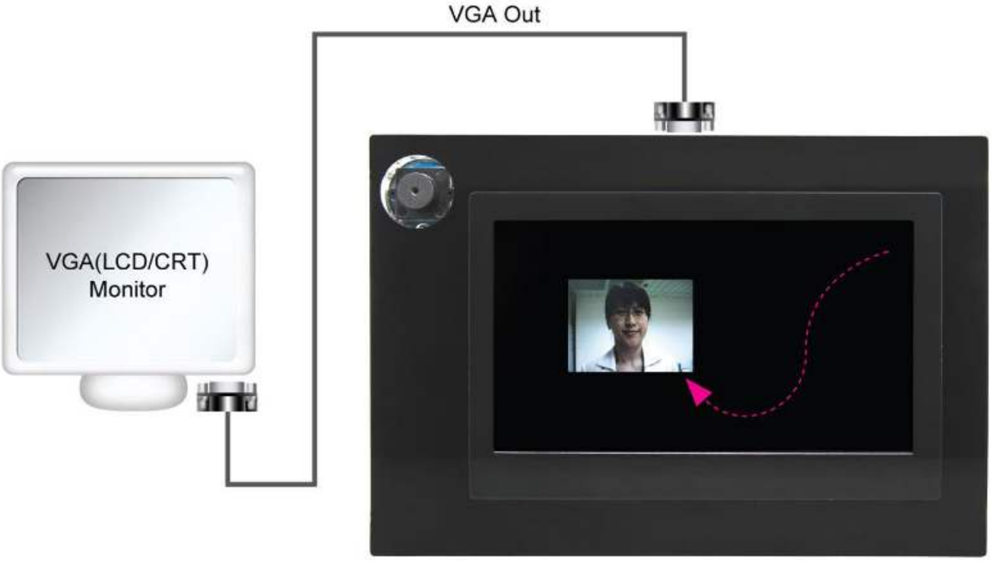


Figure 4-14 Setup for the VIP_Camera demonstration

4.7 Digital Accelerometer Demonstration

This demonstration shows a bubble level implementation based on a digital accelerometer. We use I^{2}C protocol to control the ADXL345 digital accelerometer, and the APDS-9300 Miniature Ambient Light Photo Sensor. The LCD displays the interface. When tilting the VEEK-MT (MTLC+DE2-115), the ADXL345 measures the static acceleration of gravity. In the Nios II software, the change of angle in the x-axis and y-axis is computed, and shown as angle data in the LCD display. The value of light sensor will change as the brightness changes around the light-sensor.

Figure 4-15 shows the hardware system block diagram of this demonstration. The system is clocked by an external 50MHz oscillator. Through the internal PLL module, the generated 150MHz clock is used for Nios II processor and other components, and there is also 10MHz for low-speed peripherals.

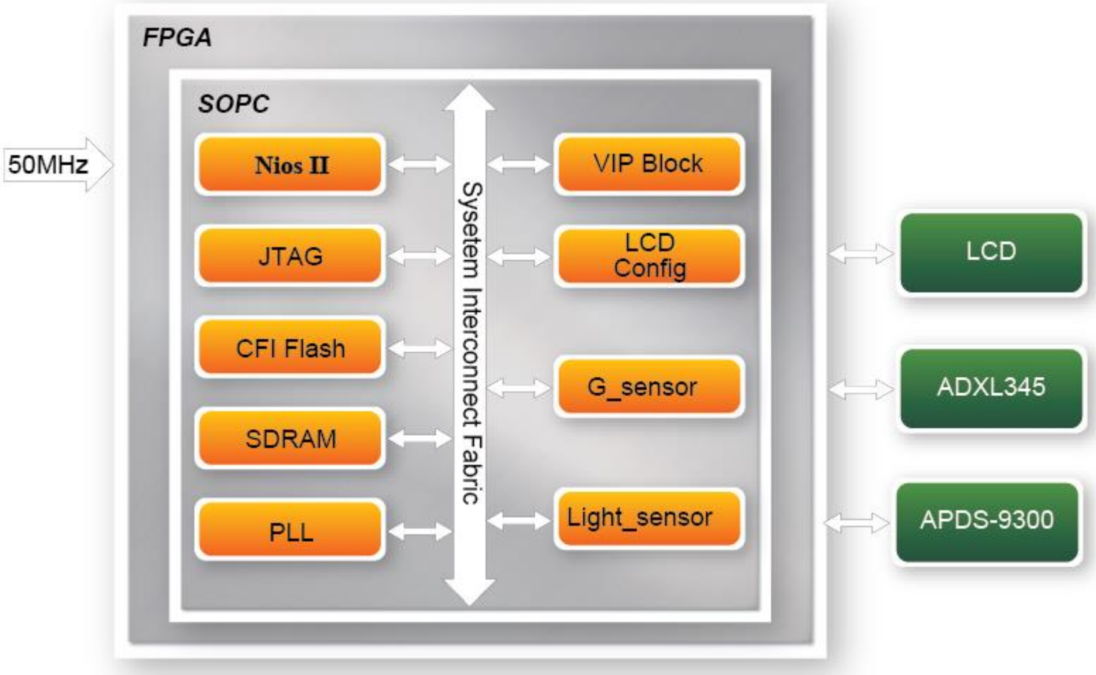


Figure 4-15 Block Diagram of the Digital Accelerometer Demonstration

■ Demonstration Source Code

- Project directory: G_sensor
- Bit stream used: G_sensor.sof
- Nios II Workspace: G_sensor\Software

■ Demonstration Batch File

Demo Batch File Folder: G_sensor\demo_batch

The demo batch file includes the following files:

- Batch File: G_sensor.bat, test_bashrc
- FPGA Configuration File: G_sensor.sof
- Nios II Program: G_sensor.elf

■ Demonstration Setup

- Load the bit stream into the FPGA on the VEEK-MT.
- Run the Nios II software under the workspace *G_sensor\Software* (Note*).
- After the Nios II program is downloaded and executed successfully, a prompt message will be displayed in nios2-terminal: “its ADXL345’s ID = e5”.
- Tilt the VEEK-MT to all directions, and you will find that the angle of the accelerometer and value of light sensor will change. When turning the board from -80^o to -10^o and from 10^o to 80^o in Y-axis, or from 10^o to 80^o and from -80^o to -10^o in Y-axis, the image will invert **Figure 4-16** shows the demonstration in action.

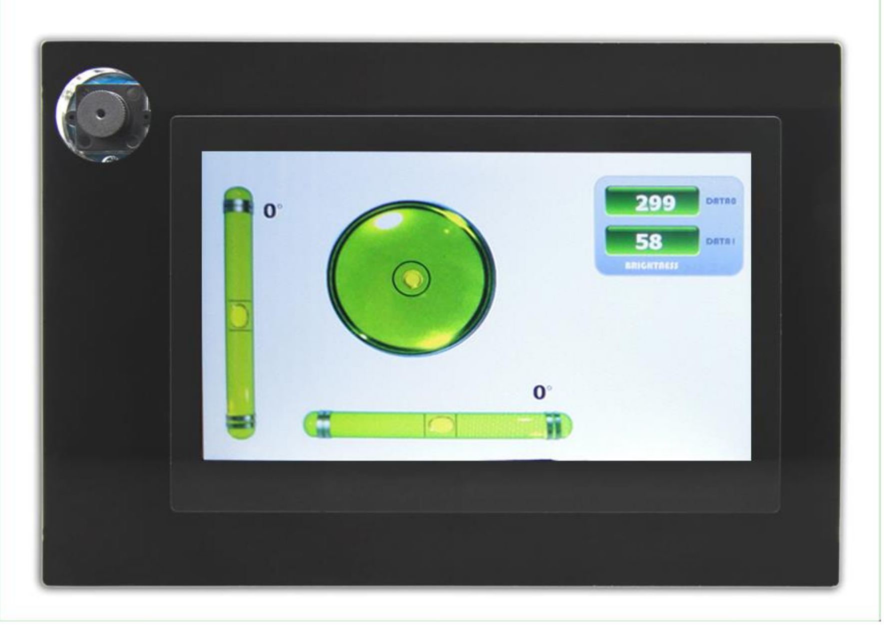


Figure 4-16 Digital Accelerometer Demonstration

Note: Execute G_sensor \demo_batch\test.bat to download .sof and .elf files.

Chapter 5

Appendix

5.1 Revision History

Version	*Change Log*
V1.0	Initial Version (Preliminary)

5.2 Copyright Statement

We will continue to provide interesting examples and labs on our MTLC webpage. Please visit mtlc.terasic.com for more information.