

1. Introduction

This document illustrates how to setup the painter demo on the DE10-Nano and the LT24 as shown in **Figure 1**. The demo is designed in Qsys and running by Nios II processor. Altera SPI IP in Qsys is used to retrieve the touch information from the touch screen. Terasic custom display component in Qsys is used to display image on the 2.4" LCD. This demo requires the following hardware.

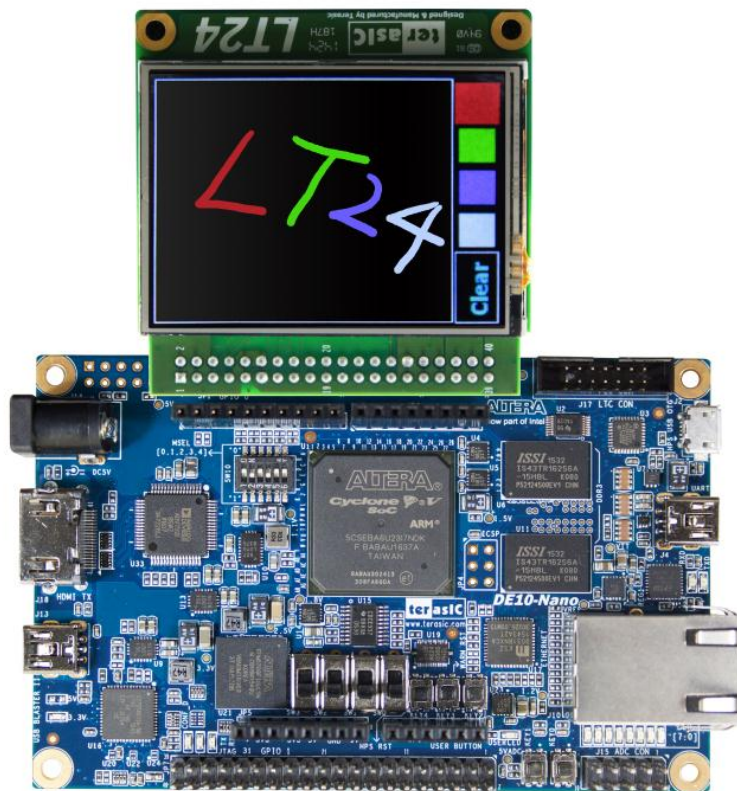


Figure 1 LT24 Painter Demo

2. System Requirements

The following items are required to perform this demonstration:

- DE10-Nano and power supply
- LT24 LCD touch module

3. Execute Demonstration

Please follow the procedures below to setup the demonstration:

1. Make sure both Quartus Prime and USB-Blaster II driver are installed on the host PC.
2. Power off the DE10-Nano board.
3. Connect a mini-USB cable to an UB2 port of the DE10-Nano and the host PC.
4. Mount the LT24 onto the 2x20 GPIO_0 expansion header of the DE10-Nano.
5. Power on the DE10-Nano Board.
6. Make sure Quartus Prime 16.0 or later is installed on your host PC.
7. Launch the “test.bat” from the folder demo_batch of the DE10_Nano_OCRAM_LT24_Painter Project.
8. Now, you should see the painter GUI on the LCD of LT24.

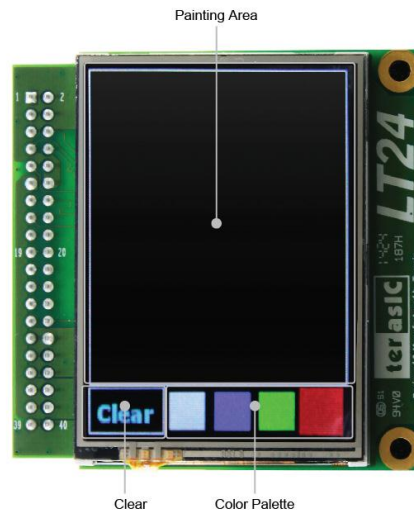


Figure 2 GUI of Painter demo

4. Project Description

Figure 3 shows the system block diagram of Painter demonstration. Terasic custom Qsys component - **LT24 LCD controller** is used to display 240(H) x 320(V) image. Its source code is located in the “/ip/LT24_Controller” folder of Painter demo project. Qsys built-in SPI controller is used to communicate with the AD7843 ADC via SPI interface to retrieve data from the touch screen. The Nios II program handles touch event and image display. It is stored in either on-chip memory or external memory. The LCD module should be initialized before sending image data to the LCD for image display.

The project is built by Quartus Prime 16.0.2 Standard Edition. If developers want to recompile the project, the same Quartus build is recommend for best compatibility.

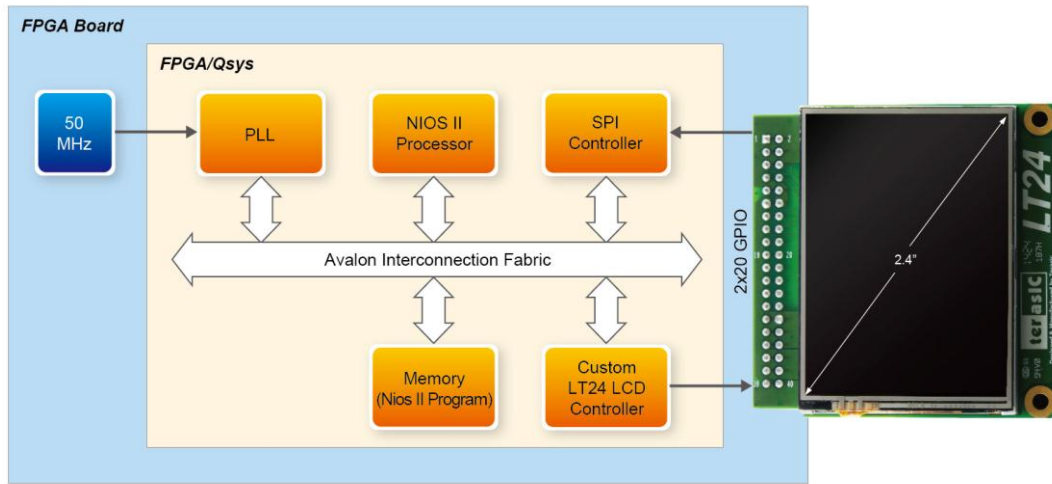


Figure 3 System Block Diagram