



# Integrating Custom Displays with LogicLoader

## Application Note 298

Mike Aanenson

Logic Product Development

Published: July 2005

### Abstract

This application note describes how to set up the LCD controller's registers and how to create a custom video mode in LogicLoader.

This file contains source code, ideas, techniques, and information (the Information) which are Proprietary and Confidential Information of Logic Product Development, Inc. This information may not be used by or disclosed to any third party except under written license, and shall be subject to the limitations prescribed under license.

No warranties of any nature are extended by this document. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed license or agreement to purchase or lease equipments. The only warranties made by Logic Product Development, if any, with respect to the products described in this document are set forth in such license or agreement. Logic Product Development cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

Logic Product Development may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering the subject matter in this document. Except as expressly provided in any written agreement from Logic Product Development, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

© Copyright 2005, Logic Product Development, Inc. All Rights Reserved.

### REVISION HISTORY

REV	EDITOR	DESCRIPTION	LOGICLOADER VER.	APPROVAL	DATE
A	Mike Aanenson	Release	2.0	MAA	6/29/05
B	James Wicks	Doc Edit, Section 2.1	2.0	JAW	7/22/05

# 1 Introduction

Logic Product Development's LogicLoader is capable of integrating a custom display with the specific LCD controller used on a product. The custom video mode(s) feature is useful in a number of scenarios. For example:

By using LogicLoader, an engineer could demonstrate the capabilities of three different displays to his manager. Since LogicLoader can save a mode for each display on the configuration block, each display could play the same bitmap off of a CompactFlash card.

An engineer could display a splash-screen with her company's logo -- while the OS is booting -- by determining the proper configuration of the LCD controller and saving the mode to LogicLoader's configuration block.

## 1.1 LCD Controller Registers

In order to integrate a custom display, one must determine the proper configuration of the LCD controller so that the timing signals meet display specification requirements.

Typically, the LCD controller has several registers that need to be set to generate the proper timing signals, including: DCLK, HSYNC, VSYNC, data signals, and, depending on the display type, several other signals. LogicLoader not only allows one to write to these registers, but also to save a custom video mode that can be used after power has been cycled.

## 1.2 Requirements

The steps outlined in this document will work as expected if the following assumptions are correct:

- LogicLoader version 2.0.1 or higher is running
- The display is properly connected to the Logic Product Development product
- All cables are properly connected
- The display is properly powered
- The correct LCD controller's guide document is available (note: this may be the on-board processor user guide)
- The user knows how to write to a register inside an IC
- The user understands LCD controllers and LCD timing signals (this document is not a guide to LCD controllers or LCD timing signals)

## 2 Integrating Custom Displays with LogicLoader

1. Create the configuration space in LogicLoader if this has not been done before. See example below.

The configuration space is a block of Flash immediately following LogicLoader that is set aside for saving information such as scripts and display modes. For more information, see the *LogicLoader User's Manual*.

```
losh> config CREATE
creating new config: offset: 0x0 pos: 0
total size: 0x40000, address 0x40000
copying block 262144 bytes from 00040000 to addr: c0056bd4
copying config 18300 bytes from c0052434 to addr: c0056bd4
dev_config burn good (262144 bytes)
```

2. Initialize one of the current video modes in LogicLoader.

A list of supported modes can be found by using this command: `losh> help video-init`. Note: selecting a mode with similar width, height, and timing characteristics may allow for fewer LCD Controller register changes.

```
losh> video-init 8 16
video-init display: width: 240 height: 320 bpp: 16 disp: 8
```

3. Create a new mode for the custom display.

The name should be kept short as there is a limit of characters. In this example, 'test' is the name used for the new mode. Also, the width (640) and height (480) are defined in this example.

```
losh> config V test 640 480
located free video index 0.
total size: 0x40000, address 0x40000
copying block 262144 bytes from 00040000 to addr: c0052588
copying config 100 bytes from c0052500 to addr: c00569e0
dev_config burn good (262144 bytes)
```

4. Close the currently open video mode and initialize the new display mode.

The 'video-init' command will set up all of the LCD controller registers but does not enable the display.

```
losh> video-close
losh> video-init test 16
video-init display: width: 640 height: 480 bpp: 16 disp: test
```

5. View the current values that are set in the LCD controller registers by using the 'config v test' command.

The 'config v test' command prints the current state of the configurable LCD controller registers and the saved values for the video mode. Each register can be found in the User Guide for the specific LCD controller in use. *Note:* the LCD controller guide -- *located on the processor manufacturer's website* -- is typically in the on-board processor User Guide.

This example shows the Sharp LH7A404 processor's internal LCD controller on Logic's LH7A404-11 card engine. The values shown are specific to the Logic product being used.

```
losh> config v test
ali: 80001000 clcdc: 80003000
ali->setup : 0x00002ef0
ali->control : 0x00000000
ali->timing1 : 0x00000000
ali->timing2 : 0x00000000
clcdc->lcdtiming0 : 0x0e143c38
clcdc->lcdtiming1 : 0x075f013f
clcdc->lcdtiming2 : 0x00ef300e
clcdc->lcdtiming3 : 0x00000000
clcdc->lcdupbase : 0xc0200000
clcdc->lcdlpbase : 0x00000000
clcdc->lcdctrl : 0x00010028
clcdc->lcdoverflow : 0x00000000
dev_config->screen[0]: test x: 640 y: 480
stored video regs: 00044458
ali->setup : 0x00002ef0
ali->control : 0x00000000
ali->timing1 : 0x00000000
ali->timing2 : 0x00000000
clcdc->lcdtiming0 : 0x0e143c38
clcdc->lcdtiming1 : 0x075f013f
clcdc->lcdtiming2 : 0x00ef300e
clcdc->lcdtiming3 : 0x00000000
clcdc->lcdupbase : 0xc0200000
clcdc->lcdlpbase : 0x00000000
clcdc->lcdctrl : 0x00010028
clcdc->lcdoverflow : 0x00000000
```

6. Determine values for the registers, by using the User Guide for the processor's internal LCD controller, and write the new addresses to the registers.

For general purposes, only the registers containing values for Horizontal Sync, Vertical Sync, Dot Clock, Display Type and signal polarization need to be modified. This step is the most time-consuming because each LCD controller must be studied to learn what each register controls.

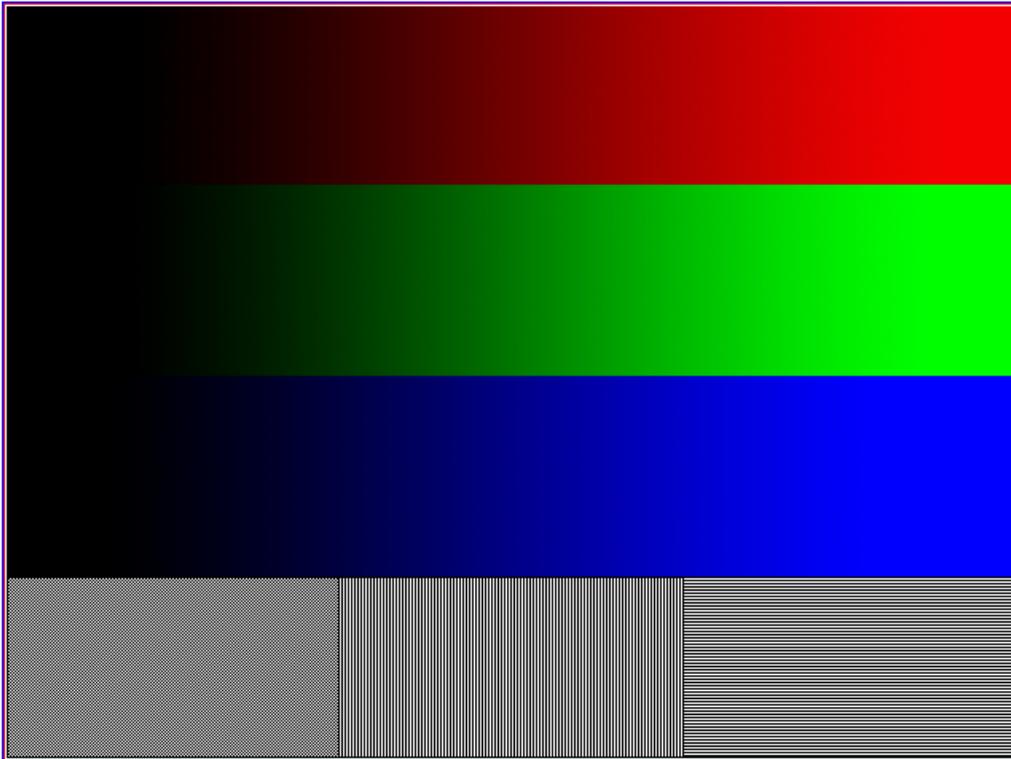
This example shows the Sharp LH7A404 processor's internal LCD controller, which has 32 bit registers.

```
losh> w /w 0x80003000 0x59320f9c
losh> w /w 0x80003004 0x0a5e1ddf
losh> w /w 0x80003008 0x027f1802
```

7. Turn on the display and put a test image up on the screen using the following commands:

```
losh> video-on  
losh> draw-test
```

The 'draw-test' command displays an image that fits the width, height, and bpp of the open video mode. The image, presented in the figure below, divides the display into 4 horizontal rows. The top three rows contain red, green, and blue bars that show the stepped gradients of each color from dark to light. The bottom row contains three boxes with black and white line patterns.



**Figure 2.1: The 'draw-test' command**

8. Locate additional changes that need to be made to the LCD controller register values.

In this example, one more change is made to a timing register. In actual testing, it may take several hours to set up the registers for a custom display. A good knowledge of the LCD controller, and how the registers affect the timing parameters to the display, is very important.

```
losh> w 0x80003004 0x1a5e1ddf
```

9. Save the register changes.

```
losh> config V test 640 480
replacing screen: testlocated free video index 0.
total size: 0x40000, address 0x40000
copying block 262144 bytes from 00040000 to addr: c0052588
copying config 100 bytes from c0052500 to addr: c00569e0
dev_config burn good (262144 bytes)
```

10. Ensure that the display comes up with the saved data when the system is powered off and on by using the 'video-open test 16' or the 'video-init test 16; video-on;' command.

```
losh> video-open test 16
video-open display: width: 640 height: 480 bpp: 16 disp: test
```

11. A mode for the custom display has been successfully added to LogicLoader.

To test the new video mode further, use the 'bitmap' command to load a .bmp file from a CF card or from the YAFFS file system.

```
losh> video-open test 16
video-open display: width: 640 height: 480 bpp: 16 disp: test
losh> bitmap TEST.BMP
```

### 3 Further Reading and Information

- LCD Controller User Guide
- Logic Product Development "Application Note 161 "Interfacing LCD's to Logic's SDK Board"
- Logic Product Development's *LogicLoader User's Manual*
- Logic Product Development's *LogicLoader Command Description Manual*
- Logic Product Development's *Hardware Specification* for the specific product in use
- Sharp Microelectronics of the America's LCD site: <http://www.sharpsma.com>. This site contains documents with helpful LCD information.