# 65550/554

Hardware Cursor and
Pop-up Window for the
HiQVideo™ Series

Application Note

Revision 1.0

March 1996

P R E L I M I N A R Y

# Revision History

| Revision | Date | By | Comment |
|---|---|---|---|
| 1.0 | 3/29/96 | BB/lc | Official Release |

This page intentionally left blank.

# Table of Contents

This page intentionally left blank.

# Hardware Cursor and Pop-up Window

## 1.0 Overview

Both HiQVideo™ Controllers (65550 and 65554) support a pair of hardware-accelerated cursors, called "cursor 1" and "cursor 2." Cursor 1 is normally used to provide the arrow pointer in most GUI applications and operating systems. Cursor 2 has no pre-assigned purpose, however it is assumed that it will be used to provide some form of pop-up window.

Off-screen memory in the frame buffer is used to provide the locations where the data for both cursor 1 and cursor 2 are kept.

Two sets of eight registers (XRA0-XRA7 for cursor 1, and XRA8-XRAF for cursor 2) provide the means to configure and position both cursors. In each set of eight registers, two are used to enable, disable, and configure each cursor. Another pair of registers from each set specifies the base address within the frame buffer memory where the cursor data is kept. Depending on the size of the cursor, up to 16 cursor patterns may be stored in the space allotted for cursor data, and within these same registers are four bits that are used to select which pattern is to be used. These four bits are double-buffered and synchronized to VSYNC to ensure that the transition from one pattern to another looks instantaneous on the display. The remaining four registers of each set are used to provide the X and Y coordinates to control the current location of each cursor relative to the upper left-hand corner of the display. These four registers are all double-buffered and synchronized to VSYNC. All four registers of the given cursor must be written, and in sequence (from XRA4 to XRA7 for cursor 1, and from XRAC to XRAF for cursor 2), before the position of the given cursor will be updated.

Two sets of four alternate color index positions added to the RAMDAC provide places in which the colors for each of the two cursors are specified (positions 0-3 for cursor 2 colors 0-3, and positions 4-7 for cursor 1 colors 0-3). These color index positions are accessed by the same sub-addressing scheme used to access the standard color index positions of the main RAMDAC palette, with the exception that a bit in the Pixel Pipeline Configuration Register 0 (XR80) must be set so that the alternate color index positions are accessible in place of the standard color index positions.

The first register in each set of eight (XRA0 for cursor 1, and XRA8 for cursor 2) is used to enable horizontal and/or vertical stretching, and blinking. The stretching features apply to flat panel displays only, and they function entirely independently of the horizontal and vertical stretching capabilities provided by the 65550 and 65554 for the main display image.

This same first register can also be used to disable the given cursor or to configure it for one of six possible modes. The main features which distinguish these modes from each other are the manner in which the cursor data is organized in memory and the meaning of the bits corresponding to each pixel position. The six possible modes are:

- 32 x 32 x 2bpp AND/XOR pixel plane mode
- 64 x 64 x 2bpp AND/XOR pixel plane mode
- 64 x 64 x 2bpp 4-color mode
- 64 x 64 x 2bpp 3-color and transparency mode
- 128 x 128 x 1bpp 2-color mode
- 128 x 128 x 1bpp 1-color and transparency mode

## 2.0 The 32 x 32 x 2bpp and 64 x 64 x 2bpp AND/XOR Pix el Plane Modes

These two modes are designed to follow the Microsoft Windows ™ cursor data plane structure, which provides two colors that may be used to draw the cursor, a third color for transparency (which allows the main display image behind the cursor to show through), and a fourth color for inverted transparency (which allows the main display image behind the cursor to show through, but with its color inverted). Each pixel position within the cursor is defined by the combination of two bits of data, each of which is stored in planes referred to as the "AND" plane and the "XOR" plane.

The tables below show how the cursor data is organized in memory for each of these two modes:

**Table 2-1: Memory Organization for the 32 x 32 x 2bpp AND/XOR Pixel Plane Mode**

| Offset | Plane | Pixels |
|---|---|---|
| 000h | AND | 31-0 on line 0 |
| 004h | AND | 31-0 on line 1 |
| 008h | XOR | 31-0 on line 0 |
| 00Ch | XOR | 31-0 on line 1 |
| 010h | AND | 31-0 on line 2 |
| 014h | AND | 31-0 on line 3 |
| ... | ... | ... |
| 0F0h | AND | 31-0 on line 30 |
| 0F4h | AND | 31-0 on line 31 |
| 0F8h | XOR | 31-0 on line 30 |
| 0FCh | XOR | 31-0 on line 31 |

**Table 2-2: Memory Organization for the 64 x 64 x 2bpp AND/XOR Pixel Plane Mode**

| Offset | Plane | Pixels |
|---|---|---|
| 000h | AND | 31-0 on line 0 |
| 004h | AND | 63-32 on line 0 |
| 008h | XOR | 31-0 on line 0 |
| 00Ch | XOR | 63-32 on line 0 |
| 010h | AND | 31-0 on line 1 |
| 014h | AND | 63-32 on line 1 |
| ... | ... | ... |
| 3F0h | AND | 31-0 on line 63 |
| 3F4h | AND | 63-32 on line 63 |
| 3F8h | XOR | 31-0 on line 63 |
| 3FCh | XOR | 63-32 on line 63 |

The meaning of each single bit in a given pixel position in the XOR plane changes depending on the bit in the corresponding position in the AND plane. If the value of the bit for a given pixel position in the AND plane is 0, then part of the cursor will be displayed at that pixel position, and the value of the corresponding bit in the XOR plane selects one of the two available cursor colors to be displayed there. Otherwise, if the value of the bit in the AND plane is 1, then that pixel position of the cursor will become transparent, allowing a pixel of the main display image behind the cursor to show through, and the value of the corresponding bit in the XOR plane chooses whether or not the color of the pixel of the main display image will be inverted. Table 2-3 summarizes this information.

**Table 2-3: Pixel Data for the 32 x 32 x 2bpp and 64 x 64 x 2bpp AND/XOR Pixel Plane Modes**

| AND Plane Pixel Data | XOR Plane Pixel Data | Color Displayed at the Corresponding Pixel Position |
|---|---|---|
| 0 | 0 | Cursor color 0 |
| 0 | 1 | Cursor color 1 |
| 1 | 0 | Transparent<br>Pixel of the main display image behind cursor shows through |
| 1 | 1 | Transparent, but inverted<br>Pixel of the main display image behind cursor shows through with inverted color |

## 3.0 The 64 x 64 x 2bpp 4-Color Mode

This mode provides four colors for use in drawing the cursor. There is no provision for transparency within the 64x64 pixel space occupied by the cursor. So, unless the image behind the cursor is the same color as one of the four colors used to draw the cursor, the cursor will be seen as having the shape of a square of 64x64 pixels. Each pixel position within the cursor is defined by the combination of two bits, each of which is stored in planes referred to as plane 0 and plane 1.

Table 3-1 shows how the cursor data is organized in memory and Table 3-2 shows the meaning of the two bits for each pixel position.

**Table 3-1: Memory Organization for the 64 x 64 x 2bpp 4-Color Mode**

| Offset | Plane | Pixels |
|---|---|---|
| 000h | 0 | 31-0 on line 0 |
| 004h | 0 | 63-32 on line 0 |
| 008h | 1 | 31-0 on line 0 |
| 00Ch | 1 | 63-32 on line 0 |
| 010h | 0 | 31-0 on line 1 |
| 014h | 0 | 63-32 on line 1 |
| ... | ... | ... |
| 3F0h | 0 | 31-0 on line 63 |
| 3F4h | 0 | 63-32 on line 63 |
| 3F8h | 1 | 31-0 on line 63 |
| 3FCh | 1 | 63-32 on line 63 |

**Table 3-2: Pixel Data for the 64 x 64 x 2bpp 4-Color Mode**

| Plane 0 Pixel Data | Plane 1 Pixel Data | Color Displayed at the Corresponding Pixel Position |
|---|---|---|
| 0 | 0 | Cursor color 0 |
| 0 | 1 | Cursor color 1 |
| 1 | 0 | Cursor color 2 |
| 1 | 1 | Cursor color 3 |

## 4.0 The 64 x 64 x 2bpp 3-Color and Transparency Mode

This mode provides three colors for use in drawing the cursor, and a fourth color for transparency (which allows the main display image behind the cursor to show through). Each pixel position within the cursor is defined by the combination of two bits, each of which is stored in planes referred to as plane 0 and plane 1.

Table 4-1 shows how the cursor data is organized in memory and Table 4-2 shows the meaning of the two bits for each pixel position.

**Table 4-1: Memory Organization for the 64 x 64 x 2bpp 3-Color and Transparency Mode**

| Offset | Plane | Pixels |
|---|---|---|
| 000h | 0 | 31-0 on line 0 |
| 004h | 0 | 63-32 on line 0 |
| 008h | 1 | 31-0 on line 0 |
| 00Ch | 1 | 63-32 on line 0 |
| 010h | 0 | 31-0 on line 1 |
| 014h | 0 | 63-32 on line 1 |
| ... | ... | ... |
| 3F0h | 0 | 31-0 on line 63 |
| 3F4h | 0 | 63-32 on line 63 |
| 3F8h | 1 | 31-0 on line 63 |
| 3FCh | 1 | 63-32 on line 63 |

**Table 4-2: Pixel Data for the 64 x 64 x 2bpp 3-Color and Transparency Mode**

| Plane 0 Pixel Data | Plane 1 Pixel Data | Color Displayed at the Corresponding Pixel Position |
|---|---|---|
| 0 | 0 | Cursor color 0 |
| 0 | 1 | Cursor color 1 |
| 1 | 0 | Transparent<br>Pixel of the image behind cursor shows through |
| 1 | 1 | Cursor color 3 |

## 5.0 The 128 x 128 x 1bpp 2-Color Mode

This mode provides two colors for use in drawing the cursor. There is no provision for transparency within the 128x128 pixel space occupied by the cursor, so unless the image behind the cursor is the same color as one of the two colors used to draw the cursor, the cursor will be seen as having the shape of a square of 128x128 pixels.

Table 5-1 shows how the cursor data is organized in memory and Table 5-2 shows the meaning of the bit for each position.

**Table 5-1: Memory Organization for the 128 x 128 x 1bpp 2-Color Mode**

| Offset | Pixels |
|---|---|
| 000h | 31-0 on line 0 |
| 004h | 63-32 on line 0 |
| 008h | 95-64 on line 0 |
| 00Ch | 127-96 on line 0 |
| 010h | 31-0 on line 1 |
| 014h | 63-32 on line 1 |
| ... | ... |
| 7F0h | 31-0 on line 127 |
| 7F4h | 63-32 on line 127 |
| 7F8h | 95-64 on line 127 |
| 7FCh | 127-96 on line 127 |

**Table 5-2: Pixel Data for the 128 x 128 x 1bpp 2-Color Mode**

| Pixel Data Bit | Color Displayed at the Corresponding Pixel Position |
|---|---|
| 0 | Cursor color 2 |
| 1 | Cursor color 3 |

## 6.0 The 128 x 128 x 1bpp 1-Color and Transparency Mode

This mode provides one color for use in drawing the cursor, and a second color for transparency (which allows the image behind the cursor to show through).

Table 6-1 shows how the cursor data is organized in memory and Table 6-2 shows the meaning of the bit for each position.

**Table 6-1: Memory Organization for the 128 x 128 x 1bpp 1-Color and Transparency Mode**

| Offset | Pixels |
|---|---|
| 000h | 31-0 on line 0 |
| 004h | 63-32 on line 0 |
| 008h | 95-64 on line 0 |
| 00Ch | 127-96 on line 0 |
| 010h | 31-0 on line 1 |
| 014h | 63-32 on line 1 |
| ... | ... |
| 7F0h | 31-0 on line 127 |
| 7F4h | 63-32 on line 127 |
| 7F8h | 95-64 on line 127 |
| 7FCh | 127-96 on line 127 |

**Table 6-2: Pixel Bit Definitions for the 128 x 128 x 1bpp 1-Color and Transparency Mode**

| Pixel Data Bit | Color Displayed at the Corresponding Pixel Position |
|---|---|
| 0 | Transparent<br>Pixel of the image behind cursor shows through |
| 1 | Cursor color 2 |

Chips and Technologies, Inc.
2950 Zanker Road
San Jose, California 95134
Phone: 408-434-0600
FAX: 408-894-2080

Title: Hardware Cursor and Pop Up Window for the HiQVideo™ Series.
Publication No.: AN98
Stock No.: 020098-000
Revision No.: 1.0
Date: 3/29/96