MQ-100/HD64464

HITACHI

LCD/CRT 2D Graphics
Subsystem

Data Book
Version 1.09

August 1998

P r e l i m i n a r y P r o d u c t I n f o r m a t i o n

Copyright Notice

Copyright 1998 MediaQ, Inc. ALL RIGHTS RESERVED.

This manual is copyrighted by MediaQ, Inc. You may not reproduce, transmit, transcribe, store in a retrieval system, or translate into any language or computer language, in any form or by any means - electronic, mechanical, magnetic, optical, chemical, manual, or otherwise - any part of this publication without the express written permission of MediaQ, Inc.

Restricted Rights Legend

Use, duplication, or disclosure by the Government is subject to restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at 252.277-7013.

Trademark Acknowledgment

MediaQ Logo is a registered trademark of MediaQ, Inc.

All other trademarks are the property of their respective holders.

Disclaimer

This document provides general information for the customer. MediaQ, Inc., reserves the right to modify the information contained herein as necessary and the customer should ensure that it has the most recent revision of the document. MediaQ makes no warranty for the use of its products and bears no responsibility for any errors which may appear in this document. The customer should be on notice that many different parties hold patents on products, components, and processes within the personal computer industry. Customers should ensure that their use of the products does not infringe upon any patents. MediaQ respects the patent rights of third parties and shall not participate in direct or indirect patent infringement.

Revision History

Revision	Date	By	Comment
1.07	7/20/98	SRV/bb	Frame Release.
1.08	7/29/98	MB	Revised all the Electricals.
1.09	8/27/98	SRV	Include HD64464

MQ-100/HD64464
LCD/CRT 2D Graphics Subsystem
Preliminary Product Information

The MQ-100/HD64464 is a single-chip LCD/CRT 2D graphics subsystem with 1MByte embedded DRAM. The MQ-100/HD64464 is designed to provide direct, high performance interface to the Hitachi SH-family of RISC Processors. The target market segments for the MQ-100/HD64464 are sub-notebooks based on the Windows® CE Operating System, Internet Appliances and Windows-based Terminals.

MQ-100/HD64464 Key Features:

- 64-Bit 2D Graphics Acceleration Engine
- Embedded 1-MByte SDRAM for Display Memory
- Bandwidth-Driven System Partitioning Reduces I/O Traffic and Saves Power
- Independent LCD and CRT Controllers
- Display Resolutions up to 1024x768; Pixel Depth up to 16-Bits per pixel
- Direct Interface to CRT monitors with On-Chip TrueColor DAC
- QView™ Display Functionality Enables Simultaneous and Dual Image Views on LCD / CRT
- Two 64x64 Hardware Cursors
- Programmable PWM Outputs control LCD Contrast and Brightness
- 3 On-Chip PLL Frequency Synthesizers
- Advanced Power Management Features with DynamiQ™ and Windows CE Power
- Management Support
- Fully Static Design Fabricated in CMOS Process

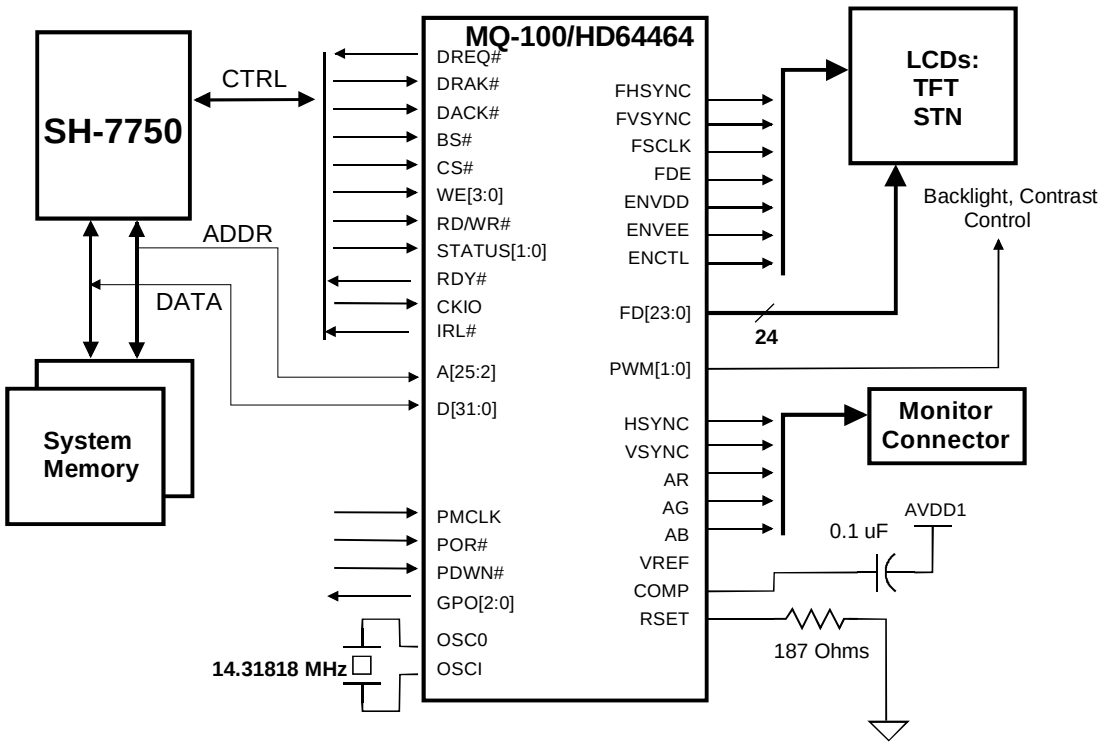


Figure 1: MQ-100/HD64464 System Implementation Diagram

1 MQ-100/HD64464 Overview

The MediaQ MQ-100/HD64464 is a high-performance single-chip accelerated 2D graphics display subsystem for Hitachi SH-7750 and SH-7709 based systems. An integrated 1-MByte frame buffer greatly reduces power consumption while dramatically improving graphics performance. The graphics engine and two graphics controllers share a high-bandwidth on-chip memory bus ensuring that graphics operations are not adversely affected by display refresh requirements. Placing the graphics frame buffer on-chip reduces power dissipation significantly while reducing chip count and improving the system form factor. Graphics controller 1 (GC1) produces control, timing and pixel data for a CRT display, and graphics controller 2 (GC2) produces control, timing and pixel data for a LCD display. The MQ-100/HD64464 provides a unique functionality called QView,^{TM} which lets the OEM configure the primary LCD display and the optional CRT display at different resolutions, refresh rates and pixel depths. For example, utilizing the QView^{TM} feature, a MQ-100/HD64464 based computing device can drive the LCD at a resolution of 640x480 8 bits per pixel (256 colors) at a refresh rate of 60 Hz while the CRT can be driven at a resolution of 800x600 8 bits per pixel (256 colors) at a refresh rate of 75 Hz. The QView^{TM} feature also allows the application software to show independent or separate images to be displayed on the primary and secondary display devices: LCD and CRT. The MQ-100/HD64464 is a very low-power design, employing asynchronous design techniques and dynamic gated clocking (DynamiQ^{TM}) to implement hardware dynamic power management, making the MQ-100/HD64464 ideal for portable and other low-power devices with demanding display requirements. Windows CE power management modes are directly supported, allowing the software to further reduce active system power consumption thus extending the battery life.

1.1 MQ-100/HD64464 Features

Superior Graphics Performance

- Integrated high-performance 64-bit 2D graphics engine
- 16-deep command FIFO provides efficient interface between CPU and graphics engine
- Independent 16-deep source image FIFO
- Hardware support for font color expansion
- High-speed interface to the internal frame buffer memory
- Graphics acceleration functions include:
    - Rectangular Source-copy block transfers (BitBLTs)
    - Transparent source-copy BitBLTs
    - Masked source-copy BitBLTs
    - Monochrome-to-color expansion on source and pattern data
    - Pattern and Rectangle Fills
    - Hardware Clipping
    - Panning and Scrolling
    - 256 Raster Ops (ROPs)
    - Horizontal Solid Line
    - Vertical Solid Line
    - Diagonal Solid Line (angle MUST be a multiple of 45 degrees)

Two Independent Graphics Controllers

- Full-featured CRT and LCD graphics controllers
- Support for 1, 2, 4, 8, 15, and 16-bits per pixel (bpp) modes
- Built-in triple 256x8 color lookup tables for 1-, 2-, 4- and 8-bpp modes
- Color lookup table can be used for gamma correction in 15 and 16-bpp modes
- Each graphics controller contains:
  - FIFO for data from display memory
  - Triple 256x8 color Look-Up Table (LUT)
  - Completely programmable display timing generator
  - 64x64 hardware cursor with 24-bit foreground/background colors, transparency or inverse
  - transparency

Dual-Display System Implementation with QView™ Technology

QView™ provides independent programmability of CRT and LCD parameters: resolution, color depth and refresh rate.

QView™ technology enhances the quality of the images in dual-display applications by providing separate refresh rates, resolutions and color depths for each display. Two graphics controllers are employed to provide optimal settings for each display device.

- The primary and secondary display devices can show the same image or separate images, constrained only by the amount of available video memory and the features supported by the O/S or device drivers
- Allows “panning” of an image on LCD and CRT display windows when same image is displayed; when different image is displayed, windows could be adjacent (left-right or top-bottom) or independent
- Automatic cursor switching between display windows provides smooth virtual desktop operation

Direct Interface to a Broad Range of Liquid Crystal Displays

- Supports XGA, SVGA and VGA passive matrix color dual-scan STN (D-STN) and single-scan STN (S-STN) Liquid Crystal Displays (LCD) with 8-bit, 16-bit and 24-bit physical interfaces
- Supports XGA, SVGA and VGA active matrix color TFT LCDs with 9-bit, 12-bit, 18-bit and 24-bit (1 pixel/clock) interfaces
- Built-in color-to-monochrome conversion to support monochrome TFT, S-STN, and D-STN panels
- Advanced dithering and Frame Modulation (FRC) algorithms to generate up to 256 gray levels on monochrome and color STN panels

Programmable Pulse Width Modulated (PWM) Outputs Control the Brightness and Contrast of the Liquid Crystal Display

- Wide range of PWM output clock frequencies: the source clock (which can be the oscillator clock, the bus clock or the power management clock) can be pre-divided by a factor ranging from 1 to 15
- Variable duty cycle: the output signal duty cycle is set by an 8-bit register, providing a range of duty cycles from 1/256 to 256/256

Direct Interface to CRT Monitor Displays

- Supports 640x480 (VGA) to 1024x768 (XGA) resolutions on CRT monitors with multiple refresh rates
- Built-in High Performance TrueColor DAC with monitor sense circuitry, blank pedestal and sync pedestal
- Cyclic Redundancy Check logic to insure datapath integrity for on-board diagnostics

Integrated Programmable Frequency Synthesizers Provide Independent Frequencies for Display and Memory Operation

- On-chip clocks are synthesized from a reference 12-25 MHz crystal Oscillator to provide three internal clocks.

High Performance System Solution

- Direct, “glueless” 32-bit interface to Hitachi SH-7750 and SH-7709 Microprocessors
- Burst-mode DMA is supported to optimize transfer of source image data to the Graphics Engine from main memory
- 64-bit Graphics Engine operates independently of the CPU, performing BitBLTs, rasterops (ROPs) and other graphics acceleration functions written to a command queue
- On-chip display memory greatly reduces memory contention and bottlenecks compared to systems that share system and display memory
- On-chip FIFOs are used extensively to ensure optimum bandwidth utilization of on-chip memory

Embedded 1-MByte SDRAM for Display Memory

- Lower power dissipation than conventional external frame buffers
- Low EMI emissions from reduced pin activity
- Single package display subsystem offers excellent system factor by eliminating external display memory

Advanced Power Management Features

- Compliant with the Microsoft OnNow Architecture Initiative: D0 (normal), D1, D2, and D3 (Sleep) power states are provided; transitions between states is accomplished via software
- D1 and D2 provide programmable “step down” states for selective functional shutdown
- Suspend mode: hardware suspend entered by asserting PDWN# input for maximum power savings
- Display memory contents can be preserved in all power saving modes provided the oscillator clock is maintained, supporting “instant on” capability
- Full-featured flat panel power sequencing with programmable intervals
- Flat panel sequencing compliant with Display Power Management Signaling (DPMS) Specification
- PWM activation can be tied to flat panel power sequencing to prevent out-of-sequence application of contrast or backlight control voltages
- Built-in DynamiQ™ power management: by dynamically gating clocks, inactive portions of the MQ-100/HD64464 are automatically shut down without any software intervention, optimizing power consumption in all operating modes

Built-in NAND Tree for Board-Level Test

Fully Static Design fabricated in CMOS Process

Packaged in 176 pin LQFP package with 0.5 mm pitch

2 Pin Description

Figure 2: MQ-100/HD64464 Pinout Diagram

The following are valid pin types:

Pin Type	Pin Type Description
I	Input pin.
O	Output pin.
I/O	Bidirectional input/output pin.
P	Power (VDD) pin.
G	Ground (GND) pin.
A	Analog input or output pin
AP	Analog power pin
AG	Analog ground pin

The following convention is also used in the signal names:

Note: A '#' indicates active low signal

2.1 Flat Panel Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
ENVDD	**O**	10	8	**Enable VDD** In panel power-up sequence, this signal is asserted first; while in panel power-down sequence, this signal is driven last (min. 32 ms after ENCTL). This signal is driven low during reset.
ENCTL (GPO0)	**O**	11	8	**Enable data and control signals** In panel power-up sequence, this signal is asserted minimum 32 ms after ENVDD and in panel power-down sequence, this signal is driven minimum 32 ms after ENVEE. If not used as ENCTL, this signal can be configured as a General Purpose Output Port Bit 0. This signal is driven low during reset.
ENVEE (GPO1)	**O**	9	8	**Enable VEE** In panel power-up sequence, this signal is asserted last (min. 32 ms after ENCTL). While in panel power-down sequence, this signal is driven first. If not used as Enable VEE, this signal can be configured as a General Purpose Output Port Bit 1. This signal is driven low during reset.
FD[23:0]	**O**	28-32, 34-38, 40- 43, 45-47, 49, 50, 52-55, 57	8 / 16	**Flat Panel Data (optional data inversion)** These signals provide data to the flat panel. They are active approximately one 32 KHz clock after ENCTL is asserted and they will be driven approximately one 32 KHz clock before ENCTL is driven. For STN panels, FD0 pin can be programmed to output shift clock, in which case, FSCLK pin will output pixel clock. These signals are driven low during reset, and when the flat panel is disabled, regardless of its polarity.
FDE **FMOD**	**O** **O**	24	8 / 16 8 / 16	**Flat Panel Data Enable (programmable polarity)** This signal indicates the valid data area. Timing of this signal is the same as flat panel data. This signal is normally used for active matrix (TFT) LCD. Modulation clock for STN LCD panels This signal is driven low during reset and when flat panel is disabled regardless of its polarity.
FDI	**O**	59	8 / 16	**Flat Panel Data Inversion** If flat panel data inversion is enabled, this signal will indicate whether the flat panel data is inverted or not. When FDI is high, the flat panel data is inverted and when FDI is low the flat panel data is not inverted. Timing of this signal is the same as flat panel data. This signal is driven low during reset.
FVSYNC **FLM**	**O** **O**	25	8 / 16 8/16	**Flat Panel Vertical Sync for TFT LCD panels (programmable polarity)** **FLM for STN LCD panels (programmable polarity)** This signal is driven low during reset and when flat panel is disabled regardless of its polarity.

FHSYNC **LP**	**O** **O**	27	8 / 16 8 / 16	**Flat Panel Horizontal Sync for TFT LCD panels (programmable polarity)** **LP for STN LCD panels (programmable polarity)** This signal is driven low during reset and when flat panel is disabled regardless of its polarity.
FSCLK	**O**	58	8 / 16	**Flat Panel Shift Clock** Rising edge of this clock is used to output flat panel data and control signals therefore falling edge of this clock is normally used to externally latch flat panel data and control signals.
PWM0 (GPO2)	**O**	2	4	**Pulse Width Modulation 0** This pulse width modulation clock is enabled just before flat panel control/data is enabled and deactivated just after flat panel control/data is enabled. This pulse can be used to control external contrast/brightness logic or backlight. If not used as PWM0, this signal can be used as General Purpose Output Port Bit 2.
PWM1 (GPO3)	**O**	1	4	**Pulse Width Modulation 1** This pulse width modulation clock is enabled just before flat panel control/data is enabled and deactivated just after flat panel control/data is enabled. This pulse can be used to control external contrast/brightness logic or backlight. If not used as PWM1, this signal can be used as General Purpose Output Port Bit 3.
GPIO [2:0]	**I/O**	8, 5, 3	4	**General Purpose I/Os.** These pins can be programmed to be either inputs or outputs. These pins are programmed as inputs during reset.
FVDD[3:0]	**P**	23, 33, 48, 56	-	**Flat Panel Interface Power Supply pins**
FGND[3:0]	**G**	26, 39, 51, 61	-	**Flat Panel Interface Ground Pins**

2.2 CRT Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
AR	**A**	80	16	**Analog Red** Output to the CRT monitor
AG	**A**	75	16	**Analog Green** Output to the CRT monitor
AB	**A**	87	16	**Analog Blue** Output to the CRT monitor
VSYNC	**O**	63	8	**Vertical Sync** Output to the CRT monitor. In NAND Chain test mode, this pin is the output of the NAND chain.

HSYNC	**O**	62	8	**Horizontal Sync** Output to the CRT monitor
VREFIO	**A**	85	-	**Voltage Reference** Internal voltage reference output, leave unconnected
FSADJ	**A**	73	-	**Full-Scale Adjust** This pin must be tied GND through a 186 ohm external resistor to set the full scale current value for the DAC outputs.
COMP	**A**	78	-	**Compensation** This pin is tied to one of the analog VDD pins through a 0.1 uF capacitor, such as AVDD1
AVDDC[4:1]	**AP**	86,79, 77,76		**Analog Power for CRT DAC**
AGNDC[4:1]	**AG**	88,83, 81,74		**Analog GND for CRT DAC**

2.3 CPU Interface

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
CKIO	**I**	93	-	**Bus Interface Clock** This signal is the bus clock generated by the CPU and is used as the reference for bus interface signals. The frequency range of this clock is 8 MHz to 66 MHz.
BS#	**I**	90	-	**Bus Cycle Start** CPU asserts this signal to request a bus cycle.
A[25:2]	**I**	141-148, 150-154, 156, 157, 159-164 166-169	-	**CPU Address Bus** This address bus is driven by the CPU for all read/write accesses.

D[31:0]	**I/O**	99, 101-104, 106-110, 112, 113, 115-120, 122-126, 128-136	4	**CPU Data Bus** This data bus is driven by the CPU during write accesses and driven by MQ-100/HD64464 during read accesses. These pins are tri-stated during reset and when the CPU is disabled (powered down).
CS#	**I**	89	-	**Chip Select** Chip Select signal from the CPU to the MQ-100/HD64464. MQ-100/HD64464 decodes the address presented on the CPU Address bus only when this signal is asserted.
WE[3:0]#	**I**	94, 96, 97, 98	-	**Write Enables[3:0]** These are the byte enables [3:0] used for writes from the CPU to the MQ-100/HD64464. Read operations are always 32-bit wide reads.
RD/WR#	**I**	91	-	**Read/Write Signal** This signal is valid through out the bus cycle. If it is at Logic High, a Read cycle takes place and if it is at Logic Low, a Write cycle takes place.
IRL#	**O**	137	4	**Interrupt Request Line** Under the assumption that the CPU Interrupt Lines IRL3-IRL0 are programmed to be in non-encoded mode (Setting IRL interrupt lines to be used as four independent interrupt sources), this signal is a request from MQ-100/HD64464. Active Low. Connected to one of the IRL lines on the CPU.
STATUS[1:0]	**I**	172, 170	-	**Status[1:0]** This is status of the processor.
WAIT# **RDY#**	**O**	173	4	**Wait (active low)** In SH-7750 systems, this signal is asserted by MQ-100/HD64464 to request additional hardware wait states, asking the CPU to wait until the MQ-100/HD64464 is ready to complete the requested bus cycle Ready (active low) In SH-7709 systems, this signal is deasserted by MQ-100/HD64464 to request additional hardware wait states when it is not ready to complete the requested bus cycle
DREQ#	**O**	138	4	**Data Transfer Request** Data Transfer Request Input from MQ-100/HD64464 to Channel 0 of the DMA Controller in the CPU.
DRAK#	**I**	139	-	**DMA Transfer Request Acknowledge** DMA Transfer Request Acknowledge from the Channel 0 of the DMA Controller in the CPU to the MQ-100/HD64464.
DACK#	**I**	140		**Strobe from Channel 0** Strobe output to MQ-100/HD64464 from Channel 0 of the DMA Controller in the CPU.

CVDD [5:1]	**P**	155, 165, 121, 111, 92		**CPU Interface Power pins**
CGND [5:1]	**G**	171, 158, 127, 114, 95		**CPU Interface Ground pins**

2.4 Power Management and Miscellaneous Pins

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
PDWN#	**I**	22		**Power Down** This is an active low input to the MQ-100/HD64464. This places the MQ-100/HD64464 in the lowest power state.
PMCLK	**I**	174		**Power Mode Clock** This is the 16.384KHz clock from the CPU, used to control the power state machine in the MQ-100/HD64464.
VDD	**P**	12, 100, 144		**Power Supply**
GND	**G**	17, 105, 149		**Ground**
VDDIO	**P**	4		**Additional Power input for pad I/O ring**
GNDIO	**G**	7		**Additional Ground input for pad I/O ring**
NC		13, 44, 60, 65, 76, 82, 84, 175, 176		**No Connect** These pins should not be connected in the system. They are reserved for future use. They must NOT be connected to VCC or GND or any other signal.

2.5 Clocks

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
OSCO	**A**	19		**Reference Clock Oscillator Output** This pin is connected to an external 12MHz to 25MHz crystal oscillator. The reference clock is used to generate a reference frequency for the three internal PLLs that generate Memory and Pixel clocks. The crystal value is dependent on the desired display resolution. The recommended value is 12.277MHz.
OSCI **L1CLK**	**A**	18		**Reference Clock Oscillator Input** This pin is connected to an external 12MHz to 25MHz crystal. The reference clock oscillator is used to generate reference frequency for the three internal PLLs that generate Memory and Pixel clocks. The crystal value is dependent on the desired display resolution. **PLL 1 Bypass Clock** When MQ-100/HD64464 is configured for test mode 10, PLL 1 (System Clock source) is bypassed and this input provides the clock that is normally provided by PLL 1. In normal mode, PLL 1 gets its input from the reference clock oscillator. When PMU control bit is set to a '1', L1CLK provides the input to PLL 1.
L2CLK	**I**	68		**PLL 2 Bypass Clock** When MQ-100/HD64464 is configured for test mode 10, PLL 2 (Primary Graphics Controller clock source) is bypassed and this input provides the clock that is normally provided by PLL 2. When not in test mode, PLL 2 gets its input from the OSCI pad. When PMU control bit is set to a '1', L2CLK provides the input to the PLL 2.
L3CLK	**I**	71		**PLL 3 Bypass Clock** When MQ-100/HD64464 is configured for test mode 10, PLL 3 is bypassed and this input provides the clock that is normally provided by PLL 3. When not in test mode, PLL 3 gets its input from the OSCI pad. When PMU control bit is set to a '1', L3CLK provides the input to the PLL 3.
AVDD[3:1]	**P**	72, 67, 66		**Analog Power Supply for the on-board PLLs.**
AGND[3:1]	**G**	70, 69, 64		**Analog GND for the on-board PLLs.**

2.6 Reset/Mode Control

Pin Name	Pin Type	Pin #	Drive (mA)	Pin Description
POR#	**I**	21		Power-On Reset This signal resets the whole system. It clears all the registers including RTC and its registers. It also puts the system in STANDBY mode.
EN7	**I**	20		Enable SH-7709 This signal should be tied to VDD to enable SH-7709 processor interface and disable SH-7750 processor interface. If it is tied to GND then SH-7750 processor interface is enabled and SH-7709 interface is disabled.
EDTMEN	**I**	6		Embedded DRAM Test Mode Enable Setting the EDTST pin to a '1' will enable DRAM test mode. This pin should be connected to ground when not under test.
TMEN	**I**	14		Test Mode Enable MQ-100/HD64464 has 5 different test modes. Setting the TESTMD pin to a '1', on reset, will put the chip in test mode. The appropriate test mode is selected based on the values on the TMD[1:0]pins
TMD[1:0]	**I**	15, 16		Test Mode Select These two pins select type of Test Mode for MQ-100/ HD64464. These two pins are used only when TESTMD pin is set to a '1' on reset. TMD[1:0] Description 00 PLL/OSC Testing Mode 01 DAC Test mode 10 NAND Tree Mode 11 Logic Function Test Mode

2.7 Pin Assignment Summary

2.7.1 Pin List Sorted By Pin Number

Table 2-1: Pin List Sorted by Pin Number

1	PWM1	**71**	L3CLK	**141**	A25
2	PWM0	**72**	AVDD3	**142**	A24
3	GPIO0	**73**	FSADJ	**143**	A23
4	VDDIO	**74**	AGNDC1	**144**	VDD
5	GPIO1	**75**	AG	**145**	A22
6	EDTMEN	**76**	AVDDC1	**146**	A21
7	GNDIO	**77**	AVDDC4	**147**	A20
8	GPIO2	**78**	COMP	**148**	A19
9	ENVEE	**79**	AVDDC2	**149**	GND
10	ENVDD	**80**	AR	**150**	A18
11	ENCTL	**81**	AGNDC2	**151**	A17
12	VDD	**82**	NC	**152**	A16
13	NC	**83**	AGNDC4	**153**	A16
14	TMEN	**84**	NC	**154**	A14
15	TMD1	**85**	VREFIO	**155**	CVDD5
16	TMD0	**86**	AVDDC3	**156**	A13
17	GND	**87**	AB	**157**	A12
18	OSCI	**88**	AGNDC3	**158**	CGND4
19	OSCO	**89**	CS#	**159**	A11
20	EN7	**90**	BS#	**160**	A10
21	POR#	**91**	RDWR#	**161**	A9
22	PDWN#	**92**	CVDD1	**162**	A8
23	FVDD3	**93**	CKIO	**163**	A7
24	FDE	**94**	WE3	**164**	A6
25	FVSYNC	**95**	CGND1	**165**	CVDD4
26	FGND3	**96**	WE2	**166**	A5
27	FHSYNC	**97**	WE1	**167**	A4
28	FD23	**98**	WE0	**168**	A3
29	FD22	**99**	D31	**169**	A2
30	FD21	**100**	VDD	**170**	STATUS1
31	FD20	**101**	D30	**171**	CGND5
32	FD19	**102**	D29	**172**	STATUSS0
33	FVDD2	**103**	D28	**173**	WAIT#/RDY#
34	FD18	**104**	D27	**174**	PMCLK
35	FD17	**105**	GND	**175**	NC
36	FD16	**106**	D26	**176**	NC
37	FD15	**107**	D25		
38	FD14	**108**	D24		
39	FGND2	**109**	D23		
40	FD13	**110**	D22		
41	FD12	**111**	CVDD2		
42	FD11	**112**	D21		
43	FD10	**113**	D20		
44	NC	**114**	CGND2		
45	FD9	**115**	D19		
46	FD8	**116**	D18		
47	FD7	**117**	D17		
48	FVDD1	**118**	D16		
49	FD6	**119**	D15		
50	FD5	**120**	D14		
51	FGND1	**121**	CVDD3		
52	FD4	**122**	D13		
53	FD3	**123**	D12		
54	FD2	**124**	D11		
55	FD1	**125**	D10		
56	FVDD0	**126**	D9		
57	FD0	**127**	CGND3		
58	FSCLK	**128**	D8		
59	FDI	**129**	D7		
60	NC	**130**	D6		
61	FGND0	**131**	D5		
62	HSYNC	**132**	D4		
63	VSYNC	**133**	D3		
64	AGND1	**134**	D2		
65	NC	**135**	D1		
66	AVDD1	**136**	D0		
67	AVDD2	**137**	IRL#		
68	L2CLK	**138**	DREQ#		
69	AGND2	**139**	DRAK#		
70	AGND3	**140**	DACK#		

2.7.2 Pin List Sorted By Signal Name

Table 2-2: Pin List Sorted by Signal Name

A10	**160**	D25	**107**	L3CLK	**71**
A11	**159**	D26	**106**	NC	**13**
A12	**157**	D27	**104**	NC	**44**
A13	**156**	D28	**103**	NC	**60**
A14	**154**	D29	**102**	NC	**65**
A15	**153**	D3	**133**	NC	**84**
A16	**152**	D30	**101**	NC	**175**
A17	**151**	D31	**99**	NC	**176**
A18	**150**	D4	**132**	OSCI	**18**
A19	**148**	D5	**131**	OSCO	**19**
A2	**169**	D6	**130**	PDWN#	**22**
A20	**147**	D7	**129**	PMCLK	**174**
A21	**146**	D8	**128**	POR#	**21**
A22	**145**	D9	**126**	PWM0	**2**
A23	**143**	DACK#	**140**	PWM1	**1**
A24	**142**	DRAK#	**139**	RDWR#	**91**
A25	**141**	DREQ#	**138**	STATUS0	**172**
A3	**168**	EDTMEN	**6**	STATUS1	**170**
A4	**167**	EN7	**20**	TMD0	**16**
A5	**166**	ENCTL	**11**	TMD1	**15**
A6	**164**	ENVDD	**10**	TMEN	**14**
A7	**163**	ENVEE	**9**	VDD	**12**
A8	**162**	FD0	**57**		
A9	**161**	FD1	**55**	VDD	**100**
AB	**87**	FD10	**43**	VDD	**144**
AG	**75**	FD11	**42**	VDDIO	**4**
AGND1	**64**	FD12	**41**	VREFIO	**85**
AGND2	**69**	FD13	**40**	VSYNC	**63**
AGND3	**70**	FD14	**38**	WAIT#/RDY#	**173**
AGNDC1	**74**	FD15	**37**	WE0	**98**
AGNDC2	**81**	FD16	**36**	WE1	**97**
AGNDC3	**83**	FD17	**35**	WE2	**96**
AGNDC4	**88**				
AR	**80**	FD18	**34**	WE3	**94**
AVDD1	**66**	FD19	**32**		
AVDD2	**67**	FD2	**54**		
AVDD3	**72**	FD20	**31**		
AVDDC1	**76**	FD21	**30**		
AVDDC2	**77**	FD22	**29**		
AVDDC3	**79**	FD23	**28**		
AVDDC4	**86**				
BS#	**90**	FD3	**53**		
CGND1	**95**	FD4	**52**		
CGND2	**114**	FD5	**50**		
CGND3	**127**	FD6	**49**		
CGND4	**158**	FD7	**47**		
CGND5	**171**	FD8	**46**		
CKIO	**93**	FD9	**45**		
COMP	**78**	FDE	**24**		
CS#	**89**	FDI	**59**		
CVDD1	**92**	FGND0	**61**		
CVDD2	**111**	FGND1	**51**		
CVDD3	**121**	FGND2	**39**		
CVDD4	**165**	FGND3	**26**		
CVDD5	**155**	FHSYNC	**27**		
D0	**136**	FSADJ	**73**		
D1	**135**	FSCLK	**58**		
D10	**125**	FVDD0	**56**		
D11	**124**	FVDD1	**48**		
D12	**123**	FVDD2	**33**		
D13	**122**	FVDD3	**23**		
D14	**120**	FVSYNC	**25**		
D15	**119**	GND	**17**		
D16	**118**				
D17	**117**	GND	**105**		
D18	**116**	GND	**149**		
D19	**115**	GNDIO	**7**		
D2	**134**	GPIO0	**3**		
D20	**113**	GPIO1	**5**		
D21	**112**	GPIO2	**8**		
D22	**110**	HSYNC	**62**		
D23	**109**	IRL#	**137**		
D24	**108**	L2CLK	**68**		

3 Functional Description

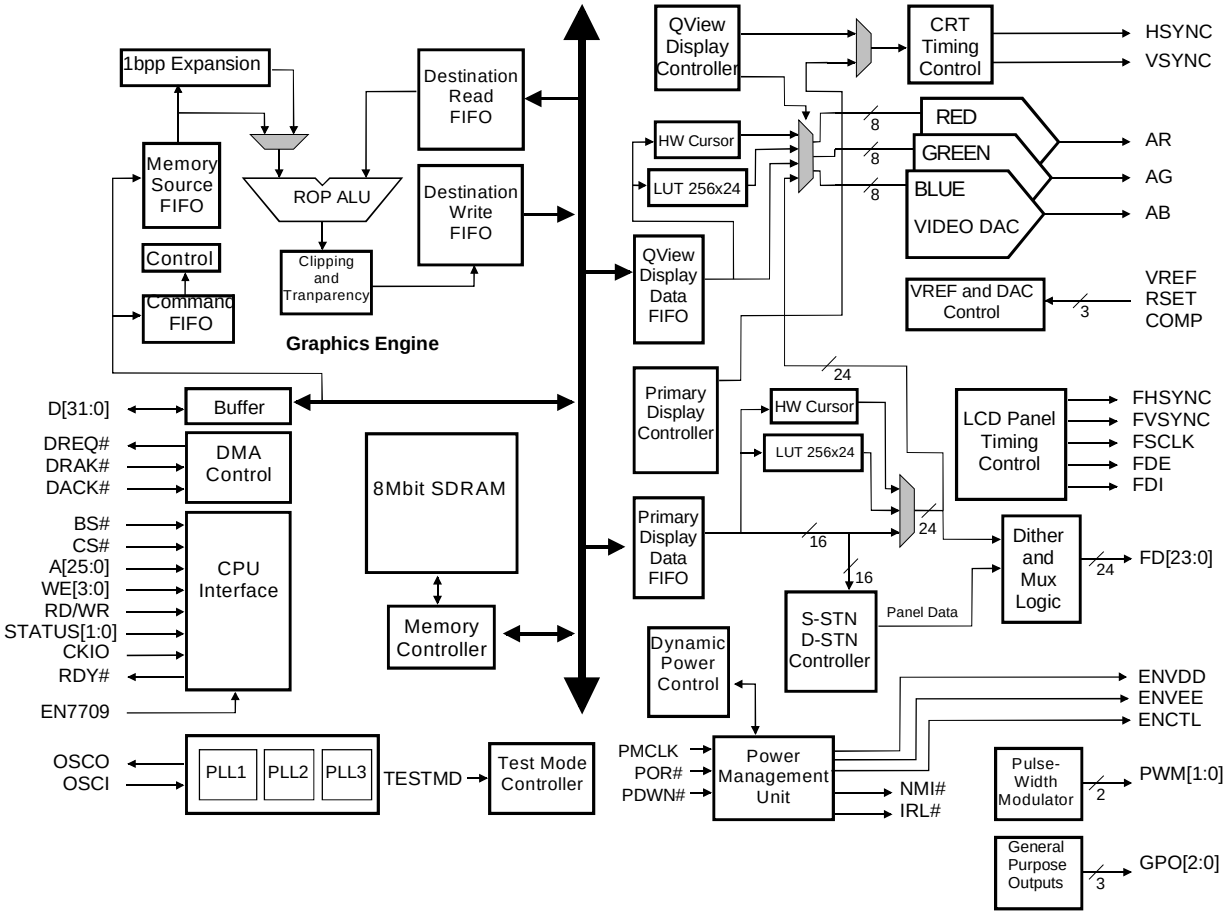


Figure 3A: MQ-100/HD64464 Block Diagram

The main functional components of the MQ-100/HD64464 are the two Graphics Controllers, the CRT interface, the Flat Panel interface, the Graphics Engine, the On-Chip Memory, the Memory Interface Unit, the CPU interface and the Power Management Unit. The diagram above shows how the different modules interact. The graphics controller gets image data from the on-chip memory and passes it along with timing information to either the CRT or Flat Panel interfaces (or both) for display. The images are placed in the memory by the Graphics Engine, either using image data from the external memory or by executing a series of graphics operations (such as area fill, block move, line draw, etc.). The on-chip memory is 128 bits wide, providing enough bandwidth to support two independent images displayed simultaneously on LCD and CRT. The CPU can read or write the on-chip memory for testing or diagnostics purposes but does not access the memory during normal operation.

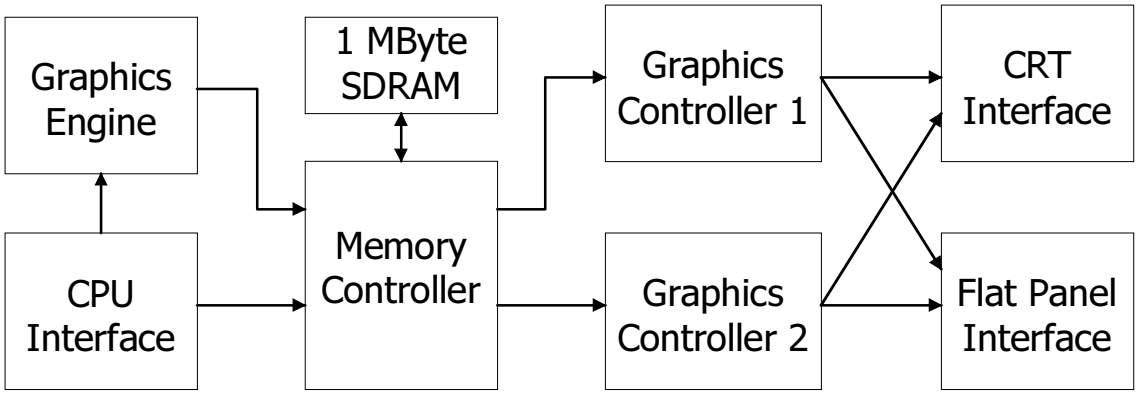


Figure 3B: MQ-100/HD64464 Functional Modules

The Power Management Unit (PMU) takes control of the MQ-100/HD64464 following reset and manages the power-up and power-down sequencing. It is must be active in order for the MQ-100/HD64464 to operate, and has it's own clock signal, allowing it to operate even when the rest of the MQ-100/HD64464 is shut down. The PMU provides the means for software to selectively shut down portions of the MQ-100/HD64464 to optimize power consumption.

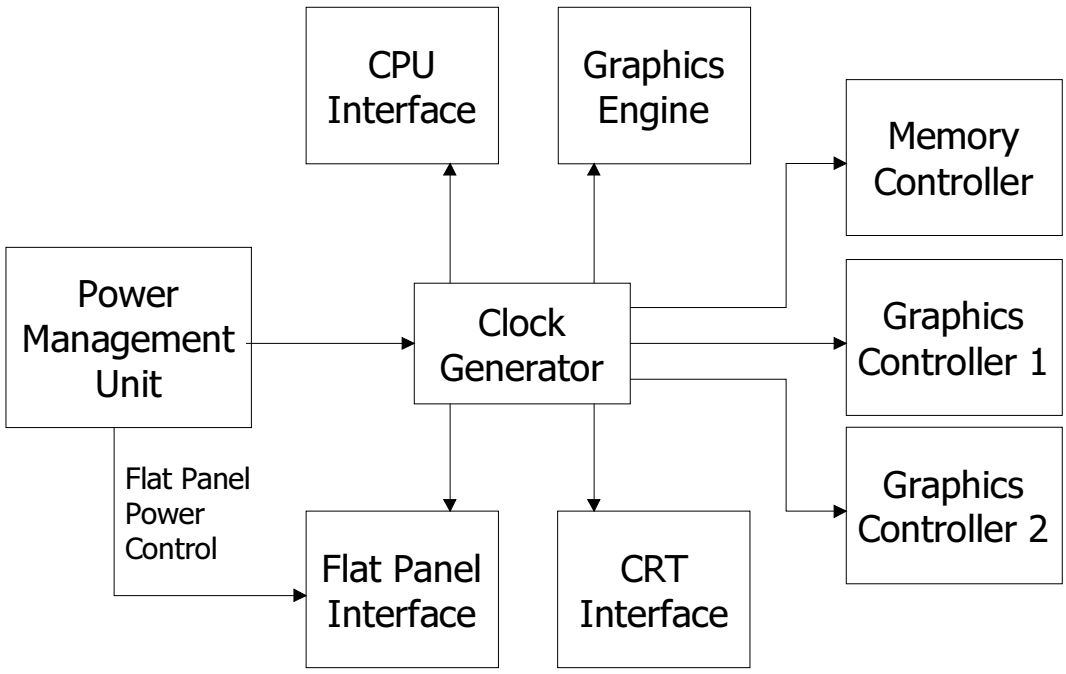


Figure 3C: MQ-100/HD64464 Clocking Scheme

3.1 CPU Interface

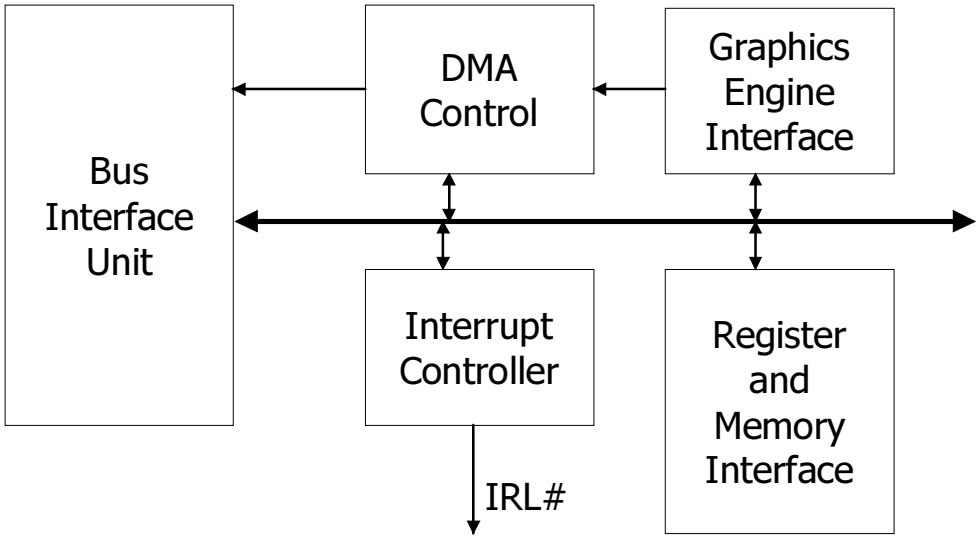


Figure 3D: MQ-100/HD64464 CPU Interface

3.1.1 Overview

The MQ-100/HD64464 connects directly to the Hitachi SH-7750 and SH-7709 processors. The bus interface is configured upon power-on reset (POR# is low) to operate with either SH-7750 or SH-7709 processors depending on the state of EN7 pin. If EN7 is low, the SH-7750 interface is enabled, otherwise the SH-7709 interface is enabled.

The CPU interface allows the SH-7709 or SH-7750 processor to access MQ-100/HD64464 registers and memory. It also performs DMA operations as needed to service the Graphics Engine. All MQ-100/HD64464 interrupt sources are gathered by the Interrupt Controller, which asserts IRL# pin whenever the General Interrupt Enable bit is enabled and one of the enabled interrupts is generated. Each interrupt source can be individually enabled or disabled (masked).

MQ-100/HD64464 register accesses require 2 wait states to complete. Memory access time will vary; the MQ-100/HD64464 uses the WAIT# (for SH-7709) and RDY# (for SH-7750) signals to indicate when the memory access operation is complete.

Figures 3E and 3F show the pin-to-pin connections for the MQ-100/HD64464 to the SH-7750 and to the SH-7709 configurations.

3.1.2 SH-7750 Interface

Interfacing the MQ-100/HD64464 to the SH-7750 CPU, the system designer must choose between the following options:

- No DMA, DMA channel 0 or DMA channel 1
- CS1# or CS4#
- MQ-100/HD64464 interrupt output (IRL#) can be connected directly to one of the four interrupt inputs (IRL[3:0]) of the SH-7750, or to a system interrupt priority encoder

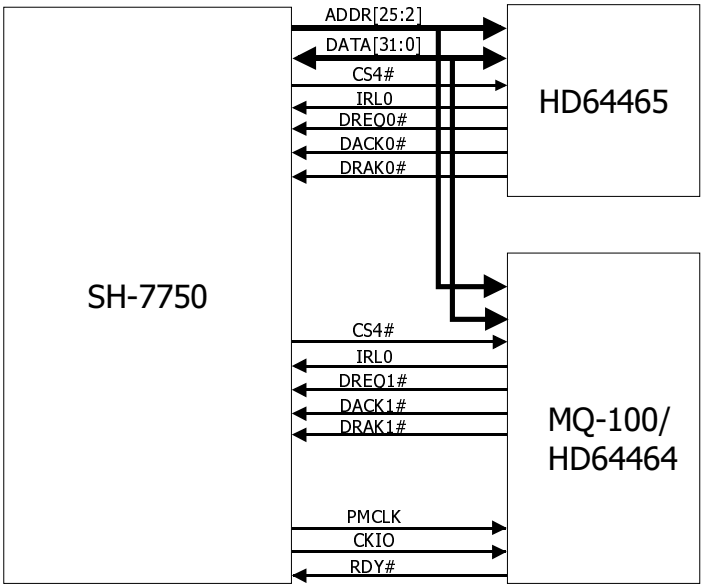


Figure 3E: SH-7750 to MQ-100/HD64464 Interface

3.1.3 SH-7709 Interface

Interfacing the MQ-100/HD64464 to the SH-7709, the system designer must choose between the following options:

- No DMA, DMA channel 0 or DMA channel 1
- The SH-7709 can accept external interrupts either directly via an I/O port bit or an IRQ# input, or indirectly as an interrupt request level on the IRL inputs from a priority encoder. The MQ-100/ HD64464 interrupt output (IRL#) can be connected directly to one of the PINT[15:0] or IRQ[5:0] bits, or to a system interrupt priority encoder which will supply the interrupt request level to IRL[3:0].

In SH-7709 systems, the MQ-100/HD64464 CS# input is always connected to SH-7709 CS4# output.

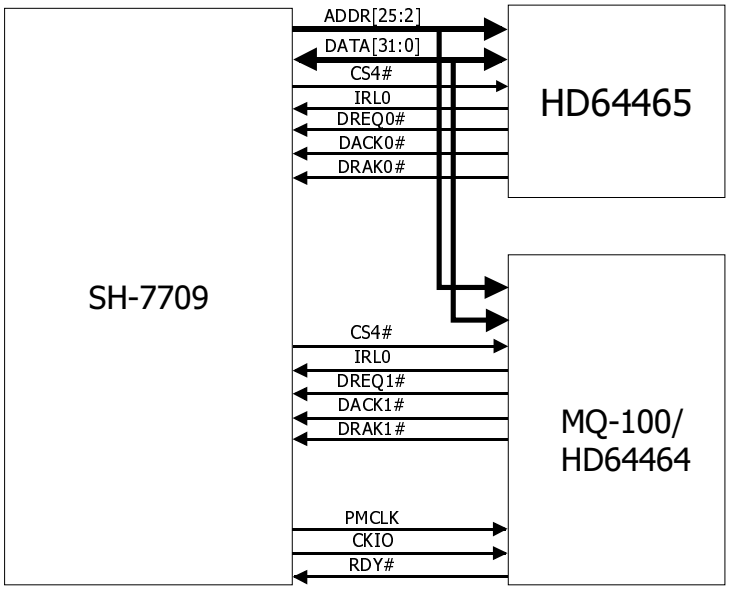


Figure 3F: SH-7709 to MQ-100/HD64464 Interface

3.1.4 Interrupt Controller

There are 16 possible interrupt sources in the MQ-100/HD64464:

Interrupts from Graphics Controller 1:
- Vertical Sync – Rising Edge
- Vertical Sync – Falling Edge
- Vertical Display Enable – Rising Edge
- Vertical Display Enable – Falling Edge

Interrupts from Graphics Controller 2:
- Vertical Sync – Rising Edge
- Vertical Sync – Falling Edge
- Vertical Display Enable – Rising Edge
- Vertical Display Enable – Falling Edge

Interrupts from Graphics Engine:
- Command FIFO Half Empty
- Command FIFO Empty
- Source FIFO Half Empty
- Source FIFO Empty
- Graphics Engine is IDLE

Interrupts from General Purpose I/O Port:
- Input from GPIO Pin 1. Low level or high level input can be programmed to generate interrupt
- Input from GPIO Pin 2. Low level or high level input can be programmed to generate interrupt
- Input from GPIO Pin 3. Low level or high level input can be programmed to generate interrupt

Each of these interrupt sources can be optionally masked (disabled). In addition, a global interrupt enable bit is provided.

When an enabled interrupt condition is detected, the corresponding bit in the interrupt status register is set. An interrupt is then signalled to the CPU by asserting the IRL# output pin provided the global interrupt enable bit is enabled. When the CPU responds to the interrupt, the interrupt handler reads the status register to determine the cause of the interrupt. The interrupt handler must then explicitly clear the interrupt by

writing a 1 to the corresponding bit in the status register. Status bits are “sticky”; they are set when the interrupt condition is detected and remain set until explicitly cleared. A status register reports which interrupts have been detected.

The active level of the IRL# output is programmable.

If the global enable bit is disabled, then the CPU checks the interrupt by polling the interrupt status register. A second status register containing the “raw” interrupt status is available. Instead of waiting for the IRL# signal, the driver software enters a loop during which it reads this “raw” interrupt status register, repeating the loop until the interrupt condition is seen. For this type of interrupt scheme, the sources to be polled to are masked off. Because masking an interrupt source prevents it’s status bit from being set, the “raw” or unmasked interrupt source register is required. This register will reflect the actual state of the interrupt source before the masking takes place.

Note that when using GPIO pins as interrupt sources, each pin must first be configured as an input by programming the GPIO control registers. The GPIO interrupt flags are level-triggered. If the GPIO pin is programmed as active high, the high level will set the interrupt; if it is programmed as active low, the low level will set the interrupt.

3.1.5 DMA Control Unit

Many graphics operations require data from the main system memory. The graphics device drivers may move this data directly into the graphics engine under software control using interrupts, or the MQ-100/HD64464 can be configured to use the SH-7750 internal DMA controller to move this data from system memory to the MQ-100/HD64464 Graphics Engine. The MQ-100/HD64464 DMA control unit monitors the status of the graphics engine FIFOs and generates DMA requests as needed.

3.2 Clocks and Oscillator

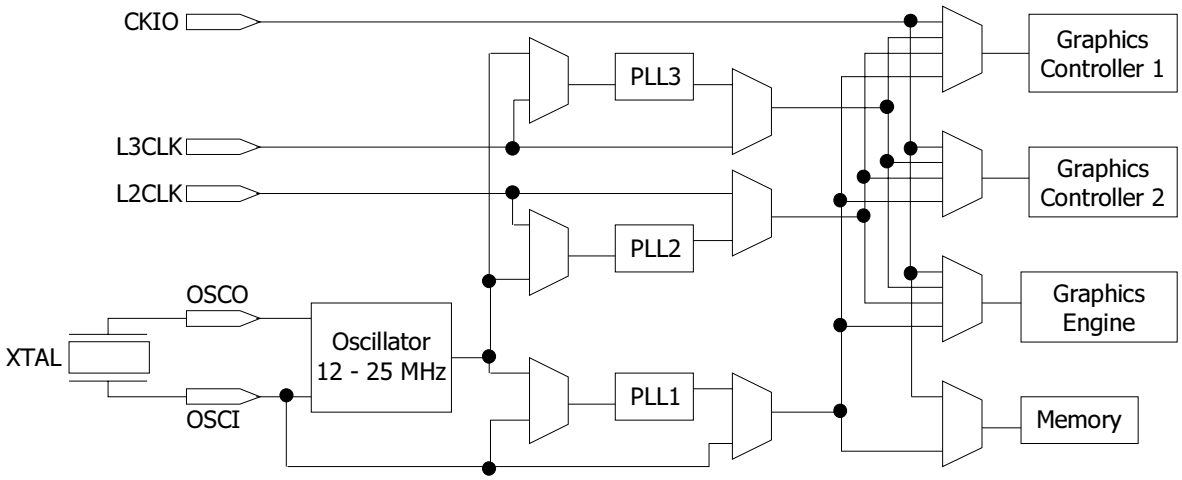


Figure 3G: Clocks and Oscillators

There are three on-chip PLLs that generate the display, memory and graphics engine clocks. The PLL reference clock normally comes from the on-chip clock oscillator, however, the oscillator circuit can be disabled and an external clock can be supplied to each of the PLLs (L1CLK, L2CLK and L3CLK). These clocks can also be used to bypass one or more PLLs if an external clock source is needed. Source selec-

tion is programmed in control registers PM05R, PM06R and PM07R.

Graphics Controller 1 and Graphics Controller 2 can get their clock from the CPU bus clock (CKIO), PLL1, PLL2, or PLL3. The CRT and Flat Panel Interface get their clocks from the graphics controller that is driving them.

The Graphics Engine can get its clock from the CPU bus clock (CKIO), PLL1, PLL2, or PLL3.

The on-chip memory system can use the output of PLL1 or the CPU bus clock (CKIO).

The CPU interface uses the CPU bus clock, CKIO.

The Power Management Unit's clock is PMCLK. The frequency of PMCLK is expected to be 16.384KHz (half the frequency of the RTC clock in SH-7709/SH-7750 systems)

3.3 Graphics Engine

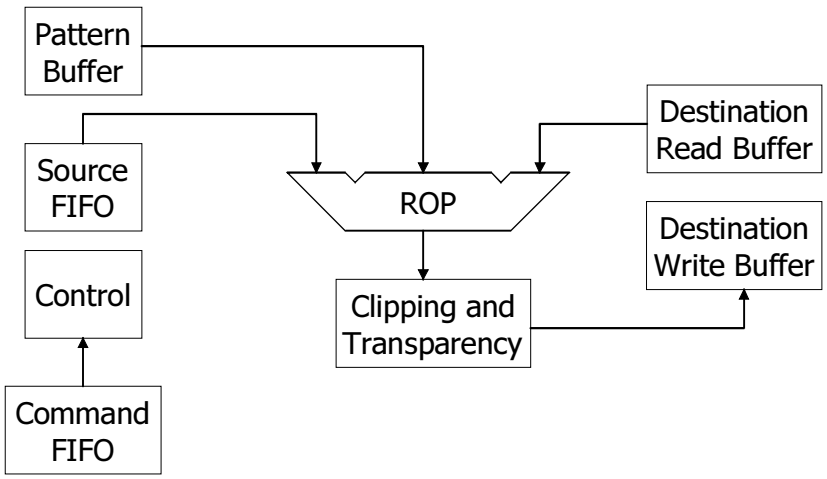


Figure 3H: Graphics Engine

3.3.1 Theory of Operation

The Graphics Engine is a specialized logic processor for 2D graphics operations such as BitBLTs and Raster-Ops (ROP), area fill and vertical and horizontal line drawing. It also provides hardware support for clipping, transparency, and font color expansion. MediaQ offers a Windows CE device driver package that uses the graphics engine to accelerate graphics and windowing performance in 8-bpp and 16-bpp graphics modes. The CPU is freed from most of the display rendering function with three main benefits:

- Accelerated graphics operations are not suspended while the CPU is busy, so that the end user does not see screen operations start and stop or slow down in most cases.
- Less power is consumed because the graphics engine is on the same chip as the display buffer. Graphics operations tend to involve the transfer of large amounts of data, and on-chip data transfers consume significantly less power than off chip transfers. So graphics engine operations use less power than the CPU would to do the same task.
- The CPU is free to perform time-critical or real-time operations, such as software modem while the Graphics Engine performs the graphics rendering.

3.3.1.1 Raster-Ops

A raster graphics system is one in which objects such as lines or characters are first “drawn” in a display memory organized as a Cartesian matrix of pixels, with a direct correspondence between a pixel’s position in the display memory and it’s position on the display. Each horizontal row of pixels is sometimes called a raster line. A raster operation (Raster-Op or ROP) is a graphics operation that is performed on a rectangular array of pixels. The graphics engine is a 3-operand (source, pattern, destination) raster engine which is capable of supporting all 256 raster operations as defined for Microsoft operating systems. Some basic ROPs are:

- Bit Block-Level Transfer (BitBLT): a rectangular area of pixels is moved from one location to another
- Area Fill: a rectangular area is filled with a background color
- Pattern Fill: an area is filled with a pattern stored in the 8x8 pixel Pattern Buffer
- Memory-to-Screen Transfer: a BitBLT from a source image to the display memory

Source data for the raster operation can come from the on-chip display memory or from the main system memory. Source data transfers from system memory are handled either by DMA or by a device driver running on the CPU. Source data can be either monochrome or color. Monochrome source data will be expanded to a foreground color or a background color.

Pattern data for the raster operation is an 8x8 pixel pattern programmed in the pattern buffer. The pattern can be either monochrome or color. Monochrome pattern pixels are expanded to a foreground color or a background color.

Destination data can also be used as one of the operands for raster operation which always comes from the display memory.

3.3.1.2 Clipping and Transparency

The process of “drawing” lines, windows, characters, and other objects, in the display memory is called rendering. A high-level description of the object (such as the endpoints of a line for example, or the height, width and origin of a window) is converted into pixels. Rendering is a complex task; it can sometimes be simplified if there is hardware support for clipping and transparency. The Graphics Engine provides the following support:

Clipping: a “bounding box” is defined, and only pixels inside this box will be affected by the graphics operation; any display modifications outside the bounding box are not written to the display memory; they are “clipped”.

Destination color transparency: a pixel value is designated as transparent; any pixel output from the Graphics Engine that has this value will not be written to the display memory.

Monochrome pattern transparency: either background color or foreground color in the mono pattern data can be defined as transparent. A destination pixel corresponding to the specified transparent pattern pixel will not be written to the display memory.

Both destination color transparency and monochrome pattern transparency can be simultaneously enabled in a single command. The two transparency controls will be ORed together.

3.3.1.3 Line Drawing

The Graphics Engine can draw vertical and horizontal lines. It can also draw diagonal lines that are at angles which are a multiple of 45 degrees (that is, with a slope of ± 1). Other types of lines must be rendered by the driver software. Note that the MQ-100/HD64464 device driver automatically applies acceleration when possible.

3.3.1.4 Monochrome to Color Expansion

The Graphics Engine supports pixel resolutions of 8 and 16-bits per pixel. If the source image or pattern image is monochrome (i.e., 1-bpp), it must be first expanded to match the required pixel resolution. The Graphics Engine will automatically expand monochrome source or pattern images according to the contents of the foreground and background color registers. When the monochrome source image is font data, this function is sometimes referred to as 'font expansion'. It is possible to have both source and pattern data be monochrome data but the same foreground and background colors will be used.

3.3.2 Graphics Engine Register Set

Please refer to Programming Information Section for a description of these registers.

3.4 Memory Sub-System

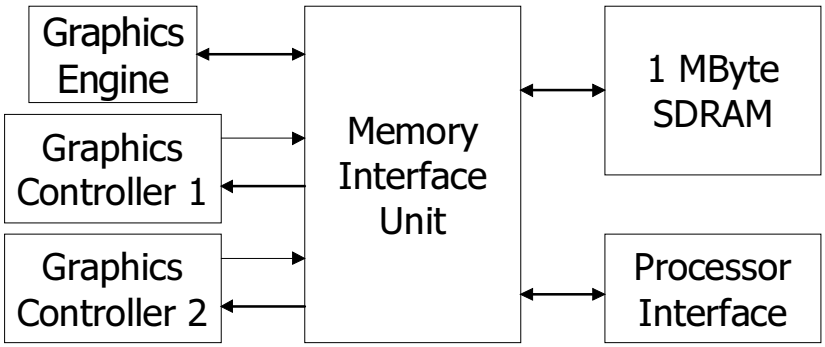


Figure 3I: Memory Sub-System

The Memory Interface Unit (MIU) arbitrates requests for memory access from CPU Interface, Graphics Engine, Graphics Controller 1 and 2, and Flat Panel Interface (for D-STN frame buffer read/write access). It also implements the memory sequencer to generate control signals for the internal frame buffer memory.

During power-down modes the memory contents will be lost unless memory refresh is enabled for the power-down mode by programming the control register in the Power Management Unit. To refresh the internal memory, reference oscillator clock is used and therefore if memory contents are to be maintained, the oscillator clock cannot be powered-down.

3.5 Display Section

The MQ-100/HD64464 modules related to displaying images are the two Graphics Controllers (GC1 and GC2), the LCD Flat Panel Interface (FPI) and the CRT interface (CRT). Figure 3J shows how these modules are connected. Each graphics controller contains a timing generator, a hardware cursor, a color lookup table and a display data FIFO. Either graphics controller can supply timing and pixel data to either the CRT or Flat Panel Interface (FPI). The choice of controller will be made by the device driver.

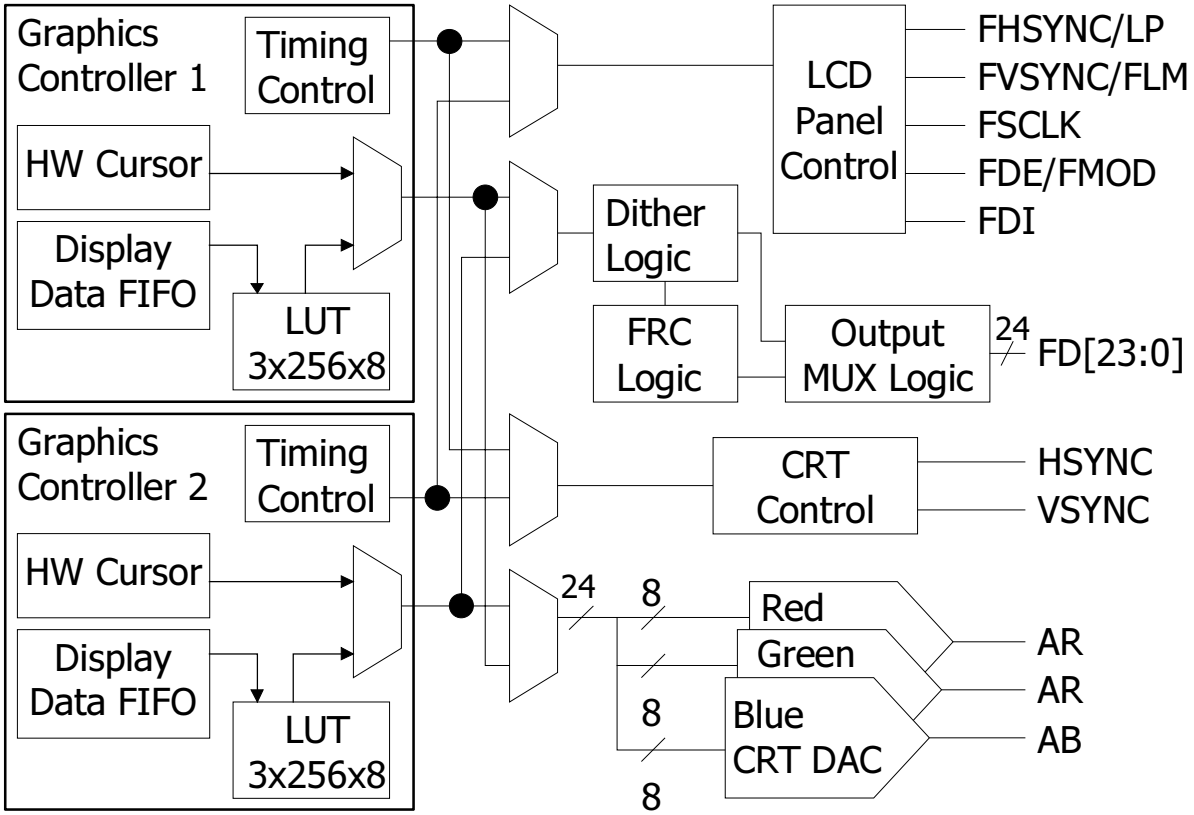


Figure 3J: Display Diagram

The MQ-100/HD64464 supports the following standard configurations:

- LCD Only: The display output is to LCD and the CRT Interface is turned off
- CRT Only: The display output is to a CRT and the Flat Panel Display Interface is turned off
- Simultaneous Display: A single graphics controller is displaying the same image at the same resolution, refresh rate and color depth on both CRT and LCD
- QView Dual Display: two graphics controllers are enabled at the same time to provide independent CRT and LCD outputs. Two QView options are available: Single-Image and Dual-Image

A single graphics controller can supply the same display resolution, color depth and refresh rate to both CRT and LCD simultaneously, so that the same image appears on both output devices. This is called Simultaneous Dual Display. If two independent images are to be displayed, or if the two displays require different screen resolution, color depth or refresh rates, one graphics controller drives the LCD and the other drives the CRT. This unique dual-display support is called QView Dual Display, and it allows OEMs to choose optimum screen resolution, refresh timing and color depths for each display device. Two QView display modes are provided: Single-Image QView and Dual-Image QView.

Table 3-1: Lists different configurations for LCD and CRT display utilizing QView :

Display Configuration	CRT Size	CRT Color Depth (bpp)	LCD Size	LCD Color Depth (bpp)
CRT and TFT/S-STN, Single Image	XGA	8	XGA	8
CRT and Color D-STN, Single Image	XGA	8	XGA	8
CRT and Color D-STN, LCD "window" on larger CRT display image	XGA	8	VGA	8
CRT and Mono D-STN, Single Image	XGA	8	XGA	8
CRT and TFT/S-STN, Single Image	SVGA	16	SVGA	16
CRT and Mono D-STN, Single Image	SVGA	16	SVGA	16
CRT and TFT/S-STN, Dual Images	SVGA	8	SVGA	8
CRT and TFT/S-STN, Single Image	VGA	16	VGA	16
CRT and Color D-STN, Single Image	VGA	16	VGA	16
CRT and TFT/S-STN, Dual Images	XGA	8	1/2VGA	8
CRT and Color D-STN, Dual Images	XGA	8	1/2VGA	8
CRT and Mono D-STN, Dual Images	SVGA	8	SVGA	8
CRT and TFT/S-STN, Dual Images and Color Depth	VGA	16	VGA	8
CRT and Color D-STN, Dual Images and Color Depths	VGA	16	VGA	8
CRT and Mono D-STN, Dual Images and Color Depth	VGA	16	VGA	8
CRT and TFT/S-STN, Dual Images	VGA	8	VGA	8
CRT and Color D-STN, Dual Images	VGA	8	VGA	8
CRT and Mono D-STN, Dual Images	VGA	8	VGA	8
CRT and ½ VGA S-STN, Dual Images	SVGA	16	½ VGA	4
CRT and VGA D-STN, Dual Images	SVGA	16	VGA	1

Each graphics controller generates the basic timing based on resolution and refresh rate that is required to produce a stable image on the display device. Within this active visible display area, the graphics controller can generate two display windows (only one can be displayed at any time) and a hardware cursor overlay. The size and position of the main and alternate windows are individually programmable, as is the color depth. The figure below illustrates an 800x600 configuration. The windows may be any size smaller than or equal to the size or the frame. This functionality can be used by system designers as a security feature or a screen saver or as a low power operating mode.

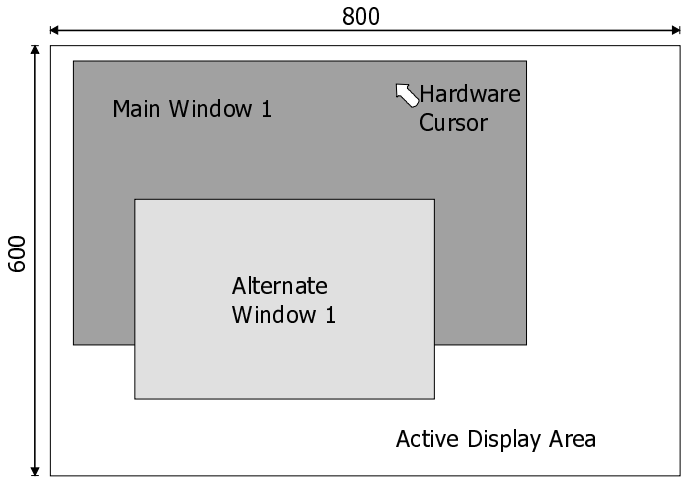


Figure 3K: Active Display Area

3.5.1 Theory of Operation

The two graphics controllers in the MQ-100/HD64464 (GC1 and GC2) generate all the pixel data and the basic timing signals based on resolution and refresh rate required to produce a stable image on the display devices. The timing signals provide the control signals used by the display device to line up the input pixel data with the proper XY position on the display. A set of registers in each graphics controller determines the total and active horizontal and vertical sizes of the display, and the start and end positions of the horizontal and vertical sync signals. These timing signals are used by the CRT and flat panel interfaces to generate the specific timing signals required for each type of display. Within the stable image frame set by these basic parameters, each graphics controller can be programmed to display a main display window or an alternate display window, and a hardware cursor. Figure 3K shows these windows for an 800 x 600 display driven by graphics controller 1 (note that normally the main window is full size). Examples demonstrating the programming of these registers may be found in the Programming Information section of this document.

Images are stored in the frame buffer as an array of pixels. Each pixel specifies the color for one point in the display. The MQ-100/HD64464 supports pixel depths of 1-, 2-, 4-, 8-, 15-, and 16-bits per pixel (bpp). These pixel depths correspond to 2, 4, 16, 256, 32,768 and 65,536 colors choices for each pixel, respectively. For 1-, 2-, 4- and 8-bpp modes, each pixel value corresponds to a 24-bit value stored in the color lookup table, with 8 bits each allocated to red, green and blue. As pixels are extracted from the display data FIFO; the pixel value is used as an index into the lookup table to select the 24-bit color value that will be displayed. For 15- and 16-bpp modes the color lookup table can be used as a gamma correction table, or it can be "bypassed" by programming a unity table.

Data is stored in the frame buffer in the following formats:

Bits	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1-bpp	P24	P25	P26	P27	P28	P29	P30	P31	P16	P17	P18	P19	P20	P21	P22	P23	P8	P9	P10	P11	P12	P13	P14	P15	P0	P1	P2	P3	P4	P5	P6	P7
2-bpp	P12	P13	P14	P15	P8	P9	P10	P11	P4	P5	P6	P7	P0	P1	P2	P3																
4-bpp	P6	P7	P4	P5	P2	P3	P0	P1																								
8-bpp	P3	P2	P1	P0																												
15-bpp (555)	R1	G1	B1	R0	G0	B0																										
16-bpp (565)	R1	G1	B1	R0	G0	B0																										

3.5.1.1 Display Clocks

The graphics controller clock sets the pixel output rate; one pixel is output for each clock during active display periods. All horizontal timing parameters are programmed in terms of display clock units. The clock for each graphics controller can be selected from the CPU bus clock or one of the three internal Phase-Locked Loop (PLL) circuits. The graphics controllers can be programmed with different clocks. Note that the graphics controllers can run asynchronously from each other and that both can be asynchronous to the rest of the MQ-100/HD64464 logic. FIFOs in each graphics controller decouple the display control unit from the internal display memory.

3.5.1.2 Pixel Generation

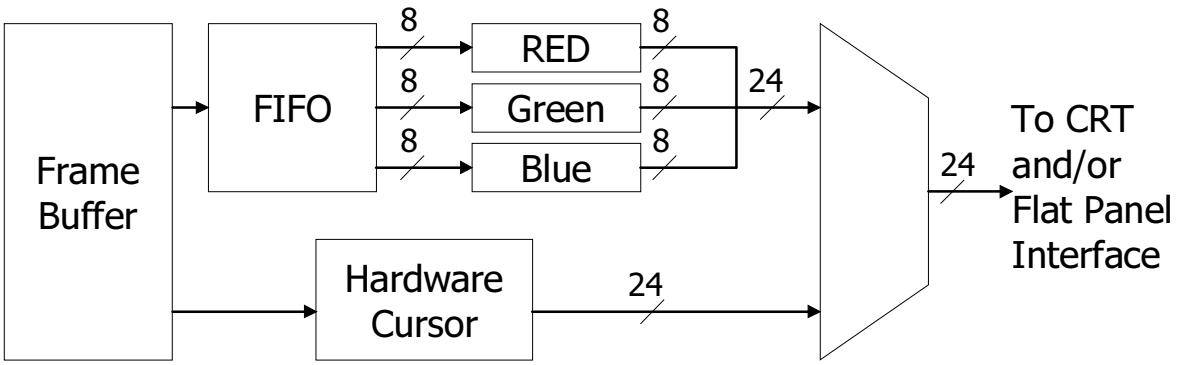


Figure 3L: Pixel Generation

Image data to be displayed is stored in the on-chip frame buffer. Each active graphics controller reads the image data into a FIFO before it is needed. Pixel data is taken from the FIFO one pixel at a time and input to the color lookup table for expansion into a 24-bit color. The 24-bit color value is then input to the selected display interfaces for output to the display device.

The hardware cursor may be overlayed on the pixel data stream, replacing the image data with the cursor image. The cursor image is also stored in the display memory. The cursor overlay is applied whenever the current pixel is inside the hardware cursor bounding box, as defined by the hardware cursor control registers.

3.5.1.3 Look-Up Table Expansion

The color lookup table (LUT) expands the pixel value to full 24-bit color, with 8 bits each for red, green and blue. The LUT is made up of three 256 x 8 tables, one each for red, green and blue. In 1-, 2-, 4-, and 8- bit per pixels, the pixel value is used as the table index to select three 8-bit color values. In 15- and 16-bit per pixel modes, the RGB color information is already contained in the pixel itself. The LUT can be used to enhance or adjust the initial color values, such as gamma color correction.

3.5.1.4 Hardware Cursor

The hardware cursor is dynamically overlayed on the pixel stream. The hardware cursor source is a 64 x 64 pixel image stored in the frame buffer. Multiple cursor images can be stored in the memory; a register sets the start address of the current cursor image. Each pixel in the hardware cursor image selects one of the following options for how that pixel is to be displayed:

- Background: use the cursor background color value
- Foreground: use the cursor foreground color value
- Transparent: the pixel from the underlying image will be used
- Inverse Transparency: the color corresponding to the underlying image pixel (after the lookup table) will be inverted and displayed

The horizontal and vertical position of the top left corner of the cursor is set by writing to a register. The cursor can be moved by simply writing a new value into this register. The new position will take effect in the next frame. If the hardware cursor is positioned such that it is partially or completely outside the active display area, hardware clipping is automatically done at the right and bottom edges. Clipping on the left and top edges is implemented in software by programming horizontal and vertical offsets. Cursor clipping is described in more detail in the Programming Information section.

3.5.1.4.1 Hardware Cursor Image Format

A 64x64 2-bpp hardware cursor image occupies exactly 1024 bytes of display memory, and each image must be aligned on a 1024-byte boundary. Each cursor pixel is described by two bits: an AND bit and an XOR bit. The two cursor bits are used as follows:

AND bit	XOR bit	Function
0	0	Cursor Background Color
0	1	Cursor Foreground Color
1	0	Transparent
1	1	Inverse Transparency

Cursor images are stored in memory in groups of 16 pixels. The two bits for a cursor pixel are stored as interleaved 16-bit words of XOR and AND information. The AND bits for the first 16 cursor pixels occupy the lower half of the first 32-bit word of the cursor image; the XOR bits for these pixels occupy the upper half of the 32-bit word:

Bits	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
First 32-bit	X15	X14	X13	X12	X11	X10	X9	X8	X7	X6	X5	X4	X3	X2	X1	X0	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

The next group of 16 cursor pixels is described by the next consecutive 32-word stored in the display memory:

First 32-bit	XOR word 0	AND word 0
Second 32-bit	XOR word 1	AND word 1
etc	:	:

Note: The X0 and A0 bits define the top left hardware cursor pixel, X1 and A1 define the next pixel in the top row, and so on.

3.5.2 Flat Panel Interface

The Flat Panel Interface (FPI) converts the output of the graphics controller into data that will be provided directly to the flat panel display. The FPI module supports direct interface to monochrome as well as color STN and TFT LCD panels.

MQ-100/HD64464 employs a proprietary frame rate control (FRC) algorithm that provides 16-levels of grayscaling on passive STN LCDs without dithering and up to 256 levels with dithering. The dithering algorithm (MediaQ® proprietary) is implemented to improve display quality on TFT panels and on STN panels. Sufficient programmability is built-in to allow fine-tuning of grayscale quality for STN panels. The MQ-100/HD64464 supports both single-scan (S-STN) and dual-scan (D-STN) passive LCD configurations. The following types of STN panels are supported:

- 4-bit, 8-bit, and 16-bit mono S-STN panel
- 4-bit, 8-bit, 12-bit, 16-bit and 24-bit color S-STN panel
- 8-bit and 16-bit mono D-STN panel
- 8-bit, 16-bit, and 24-bit color D-STN panel
- 4-bit, 6-bit, and 8-bit mono TFT panel
- 12-bit, 18-bit, and 24-bit color TFT panel

(The TFT interface supports panels with one pixel per shift clock only)

The maximum pixel clock frequency is 65 MHz for both TFT and STN panels.

Flat panel power sequencing is provided as part of the Power Management Unit (PMU) logic.

Two programmable pulse-width-modulation signals are provided on the PWM0 and PWM1 pins.

3.5.2.1 Flat Panel Interface External Signals

The MQ-100/HD64464 provides the control and data needed to drive a wide variety of flat panels. These outputs have a low drive (8 mA) and high drive (16 mA) capability and can drive most panels directly.

For TFT panels, the following signals are provided:

FVSYNC	**Vertical Sync signal**
FHSYNC	**Horizontal Sync signal**
FDE	**Data Enable:** indicates that the data being output is valid
FDI	**Data Inversion:** when high, indicates that the data being output is inverted
FSCLK	**Shift Clock:** data and timing signals are output by the MQ-100/HD64464 on the rising edge of FSCLK, display devices can use falling edge of FSCLK to latch these signals

For STN panels, the following signals are provided:

FLM	**First Line Mark**
LP	**Line Pulse**
FMOD	**Modulation Clock signal**
FSCLK	**Shift Clock:** data and timing signals are output by the MQ-100/HD64464 on the rising edge of FSCLK, display devices can use falling edge of FSCLK to latch these signals

The following signals are provided for all panel types:

FD[23:0]	**Data for the panel**
ENVDD	**Enable VDD.** During flat panel power-on sequencing this signal is the first one to be asserted and during power-off sequencing this signal is the last one to be deasserted.
ENCTL	**Enable data and control signals.** During flat panel power-on sequencing this signal is the second one to be asserted and during power-off sequencing this signal is the second one to be deasserted. If ENCTL is not required, this pin may be used as GPO0
ENVEE	**Enable VEE.** During flat panel power-on sequencing this signal is the last one to be asserted and during power-off sequencing this signal is the first one to be deasserted. If ENVEE is not required, this pin may be used as GPO1
PWM0	**Pulse width modulator output 0;** if PWM0 is not used, this pin may be used as GPO2
PWM1	**Pulse width modulator output 1;** if PWM0 is not used, this pin may be used as GPO3

3.5.2.2 Signal-To-Pin Mapping for Different Display Types

The pins used for the Flat Panel Interface serve different functions according to the panel type. The table below summarizes the pin uses for the supported panel types. The following legend explains the various table entries:

- Rx, Gx, Bx – Bit x of the red, green and blue color values, respectively for color TFT display
- URx, UGx, UBx – Bit x of the red, green and blue color value, respectively for color S-STN display or the upper panel of a color D-STN display
- LRx, LGx, LBx – Bit x of the red, green and blue color value, respectively for the lower panel of a color D-STN display
- Dx, Udx, LDx – Bit x of the gray value for mono TFT display, mono S-STN or the upper panel of a mono D-STN display, and the lower panel of a mono D-STN display, respectively
- FPCLK is the internal pixel clock generated by the graphics controller driving the flat panel interface
- FSCLK is the Shift Clock used to latch data by the panel.

3.5.2.2.1 Interface Signal Mapping for Color Panels

The following table shows the interface mapping for color TFT, D-STN and S-STN panels. Note that the interface for 12-bit/8-bit/4-bit color S-STN panels is the same as for 24-bit/16-bit/8-bit color D-STN panels without the lower panel data.

Table 3-2. Interface Signal Mapping for Color Panels

Pin Name	Color TFT 24-bit	Color TFT 18-bit	Color TFT 12-bit	Color S-STN 16-bit	Color D-STN 24-bit	Color D-STN 16-bit	Color D-STN 8-bit
FD23	R7	R5	R3	UR0	UR0	UR0	UR0
FD22	R6	R4	R2	UR1	UR1	UR1	UR1
FD21	R5	R3	R1	UR2	UR2	UR2	
FD20	R4	R2	R0	UR3	UR3		
FD19	R3	R1		UR4	LR0	LR0	LR0
FD18	R2	R0		UR5	LR1	LR1	LR1
FD17	R1				LR2	LR2	
FD16	R0				LR3		
FD15	G7	G5	G3	UG0	UG0	UG0	UG0
FD14	G6	G4	G2	UG1	UG1	UG1	
FD13	G5	G3	G1	UG2	UG2	UG2	
FD12	G4	G2	G0	UG3	UG3		
FD11	G3	G1		UG4	LG0	LG0	LG0
FD10	G2	G0			LG1	LG1	
FD9	G1				LG2	LG2	
FD8	G0				LG3		
FD7	B7	B5	B3	UB0	UB0	UB0	UB0
FD6	B6	B4	B2	UB1	UB1	UB1	
FD5	B5	B3	B1	UB2	UB2		
FD4	B4	B2	B0	UB3	UB3		
FD3	B3	B1		UB4	LB0	LB0	LB0
FD2	B2	B0			LB1	LB1	
FD1	B1				LB2		

FD0	B0			FSCLK= FPCLK*3/16	LB3	FSCLK= FPCLK*3/8	FSCLK= FPCLK*3/4
FVSYNC	FVSYNC	FVSYNC	FVSYNC	FLM	FLM	FLM	FLM
FHSYNC	FHSYNC	FHSYNC	FHSYNC	LP	LP	LP	LP
FDE	FDE	FDE	FDE	FMOD	FMOD	FMOD	FMOD
FDI	FDI	FDI	-	-	-	-	-
FSCLK	FSCLK= FPCLK	FSCLK= FPCLK	FSCLK= FPCLK	FSCLK or FPCLK	FSCLK= FPCLK/4	FSCLK or FPCLK	FSCLK or FPCLK

3.5.2.2.2 Interface Signal Mapping for Mono Panels

The following table shows the interface mapping for 24-bit color S-STN and for mono panels. Note that the interface for 8-bit/4-bit mono S-STN panels is the same as for 16-bit/8-bit mono D-STN panels without the lower panel data.

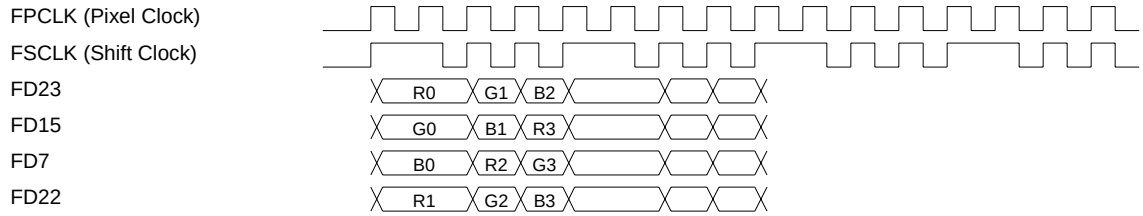
Table 3-3. Interface Signal Mapping for Mono Panels

Pin Name	Color S-STN 24-bit	Mono TFT 8-bit	Mono TFT 6-bit	Mono TFT 4-bit	Mono S-STN 16-bit	Mono D-STN 16-bit	Mono D-STN 8-bit
FD23	UR0						
FD22	UR1						
FD21	UR2						
FD20	UR3						
FD19	UR4						
FD18	UR5						
FD17	UR6						
FD16	UR7						
FD15	UG0				UD0	UD0	UD0
FD14	UG1				UD1	UD1	UD1
FD13	UG2				UD2	UD2	UD2
FD12	UG3				UD3	UD3	UD3
FD11	UG4				UD4	UD4	
FD10	UG5				UD5	UD5	
FD9	UG6				UD6	UD6	
FD8	UG7				UD7	UD7	
FD7	UB0	P7	P5	P3	UD8	LD0	LD0
FD6	UB1	P6	P4	P2	UD9	LD1	LD1
FD5	UB2	P5	P3	P1	UD10	LD2	LD2
FD4	UB3	P4	P2	P0	UD11	LD3	LD3
FD3	UB4	P3	P1		UD12	LD4	
FD2	UB5	P2	P0		UD13	LD5	
FD1	UB6	P1			UD14	LD6	
FD0	UB7	P0			UD15	LD7	
FVSYNC	FLM	FVSYNC	FVSYNC	FVSYNC	FLM	FLM	FLM
FHSYNC	LP	FHSYNC	FHSYNC	FHSYNC	LP	LP	LP
FDE	FMOD	FDE	FDE	FDE	FMOD	FMOD	FMOD
FDI	-	-	-	-	-	-	-
FSCLK	FSCLK= FPCLK	FSCLK = FPCLK	FSCLK = FPCLK	FSCLK = FPCLK	FSCLK= FPCLK/16	FSCLK = FPCLK/8	FSCLK = FPCLK/4

3.5.2.3 Timing Diagrams for S-STN Panels

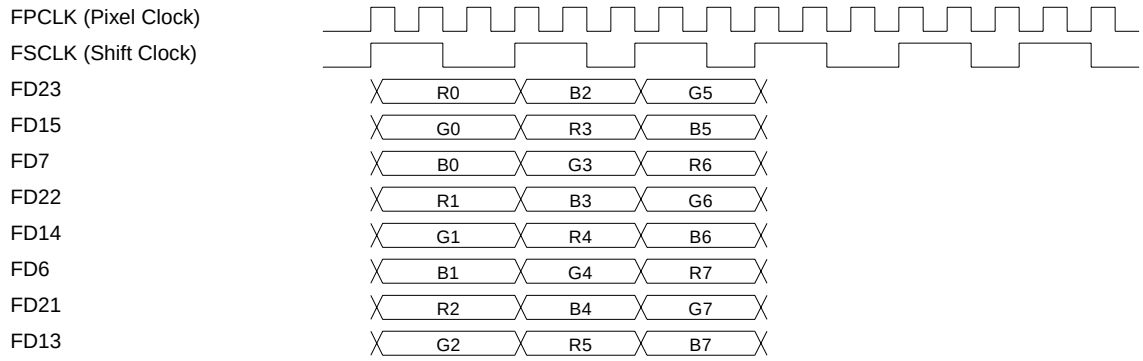
3.5.2.3.1 4-Bit Color S-STN

4-bit Color S-STN



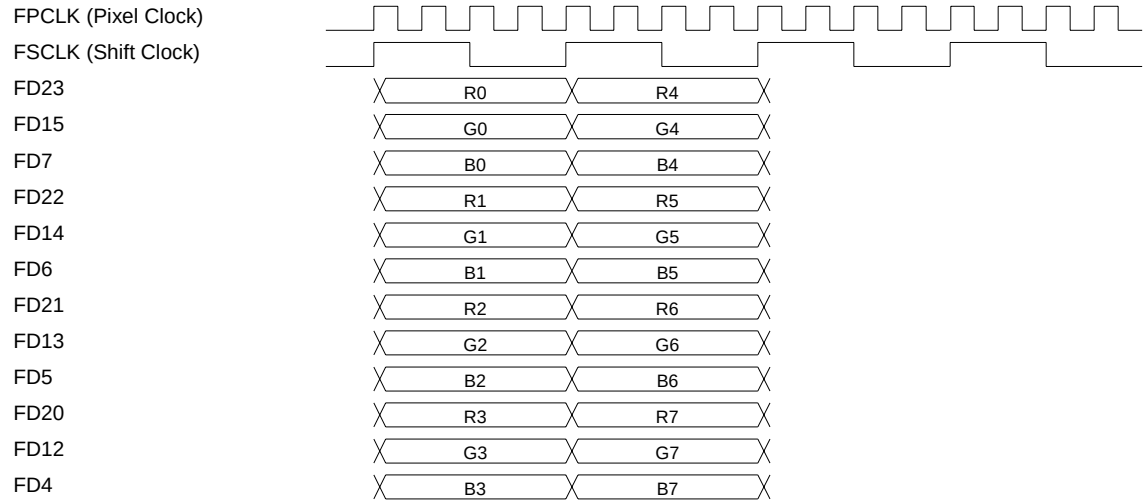
3.5.2.3.2 8-Bit Color S-STN

8-bit Color S-STN



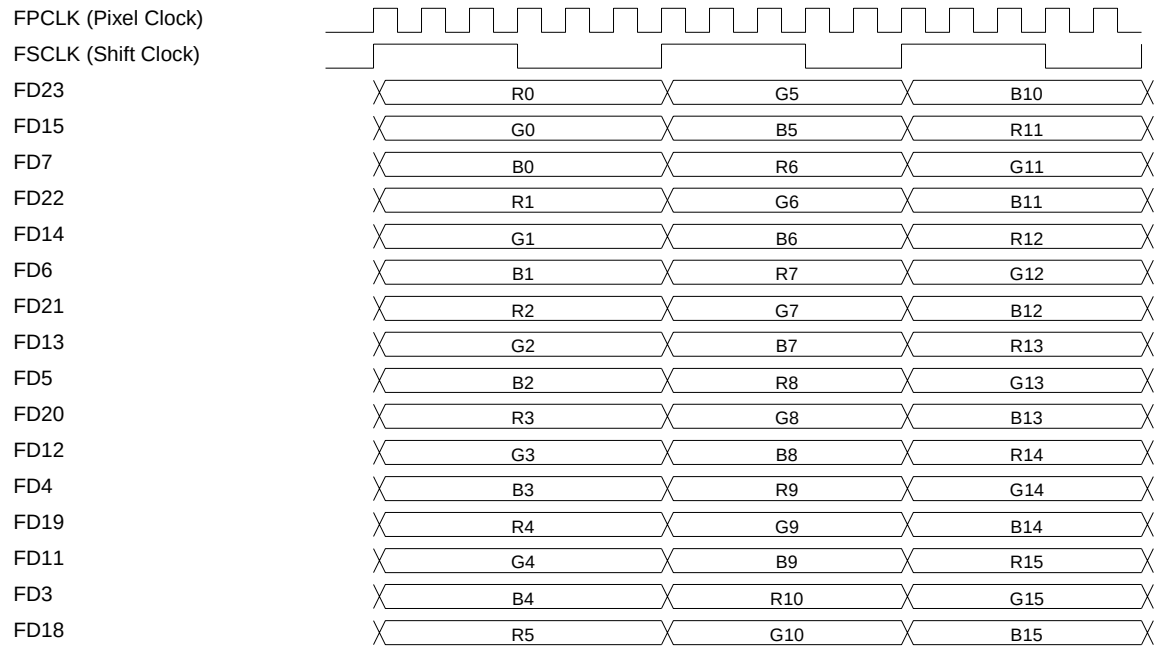
3.5.2.3.3 12-Bit Color S-STN

12-bit Color S-STN



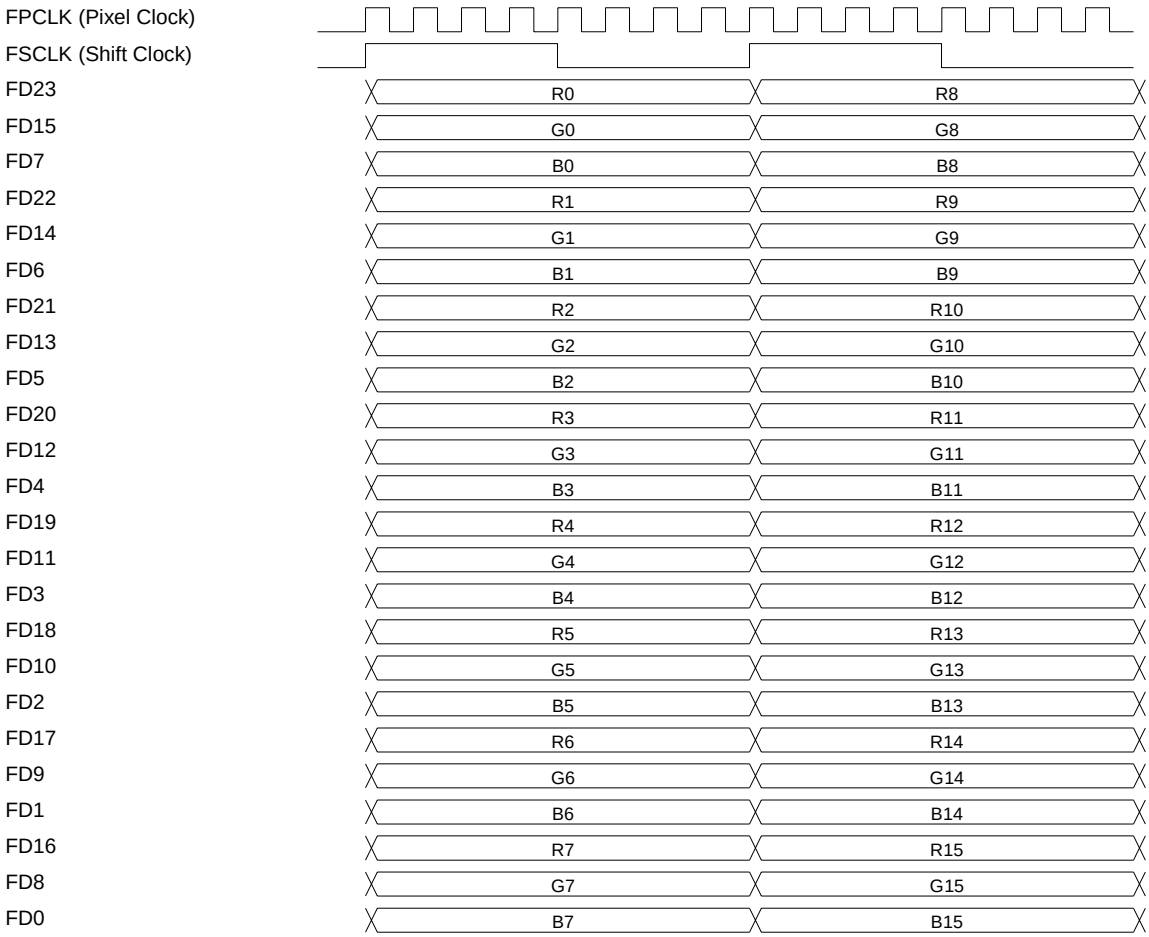
3.5.2.3.4 16-Bit Color S-STN

16-bit Color S-STN



3.5.2.3.5 24-Bit Color S-STN

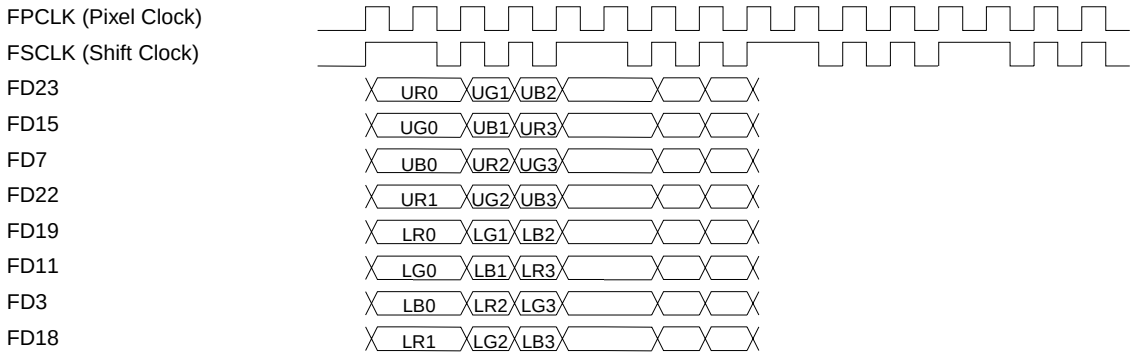
24-bit Color S-STN



3.5.2.4 Timing Diagrams for Color D-STN panels

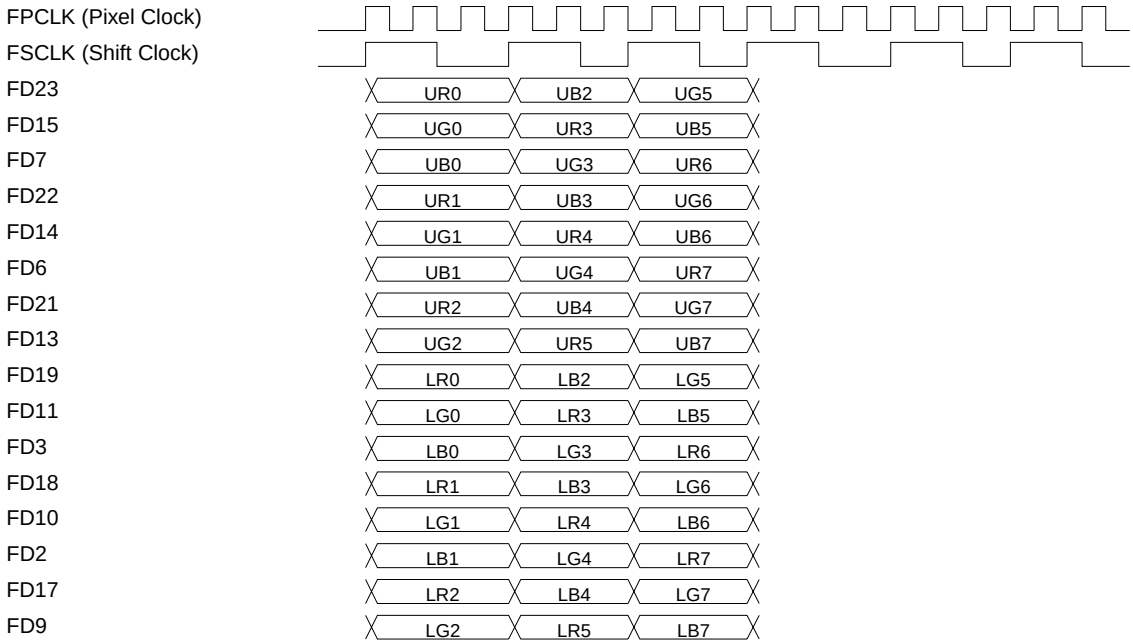
3.5.2.4.1 8-Bit Color D-STN

8-bit Color D-STN



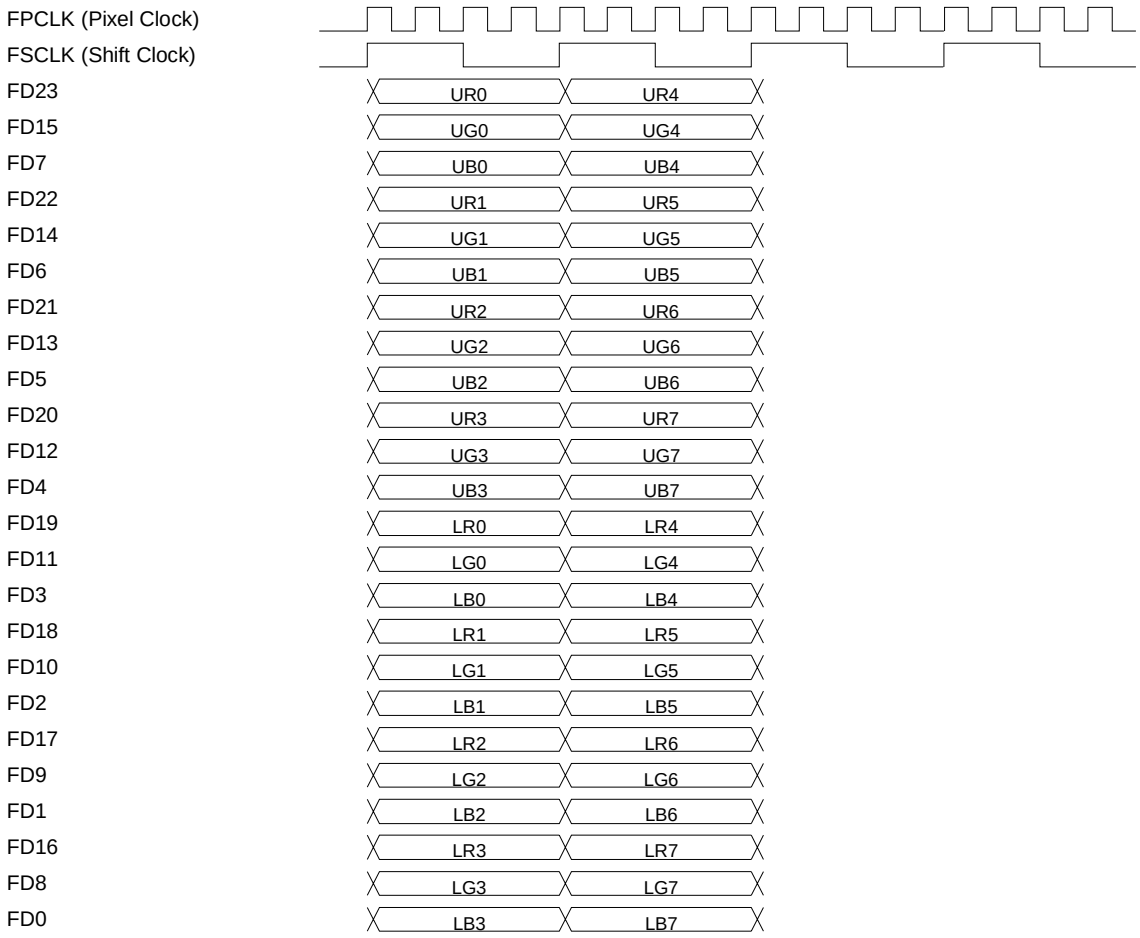
3.5.2.4.2 16-Bit Color D-STN

16-bit Color D-STN



3.5.2.4.3 24-Bit Color D-STN

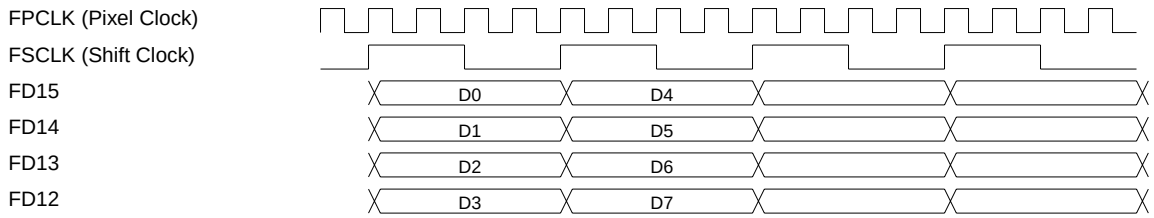
24-bit Color D-STN



3.5.2.5 Timing Diagrams for Mono S-STN Panels

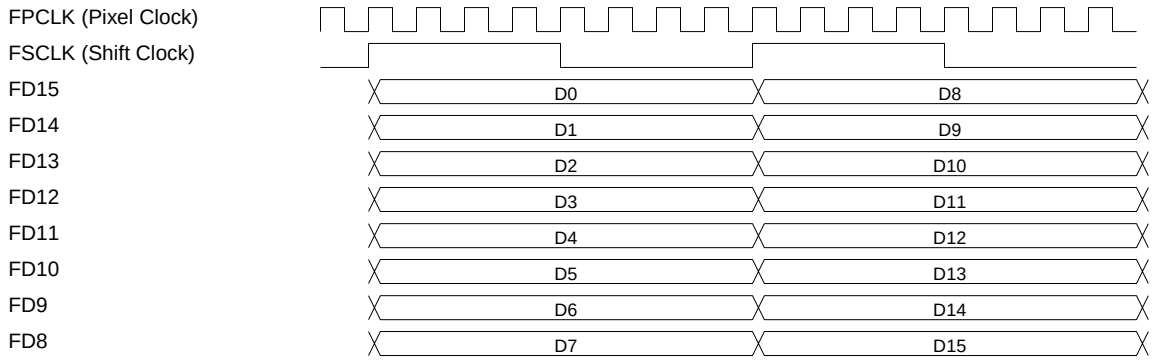
3.5.2.5.1 4-Bit Mono S-STN

4-bit Mono S-STN



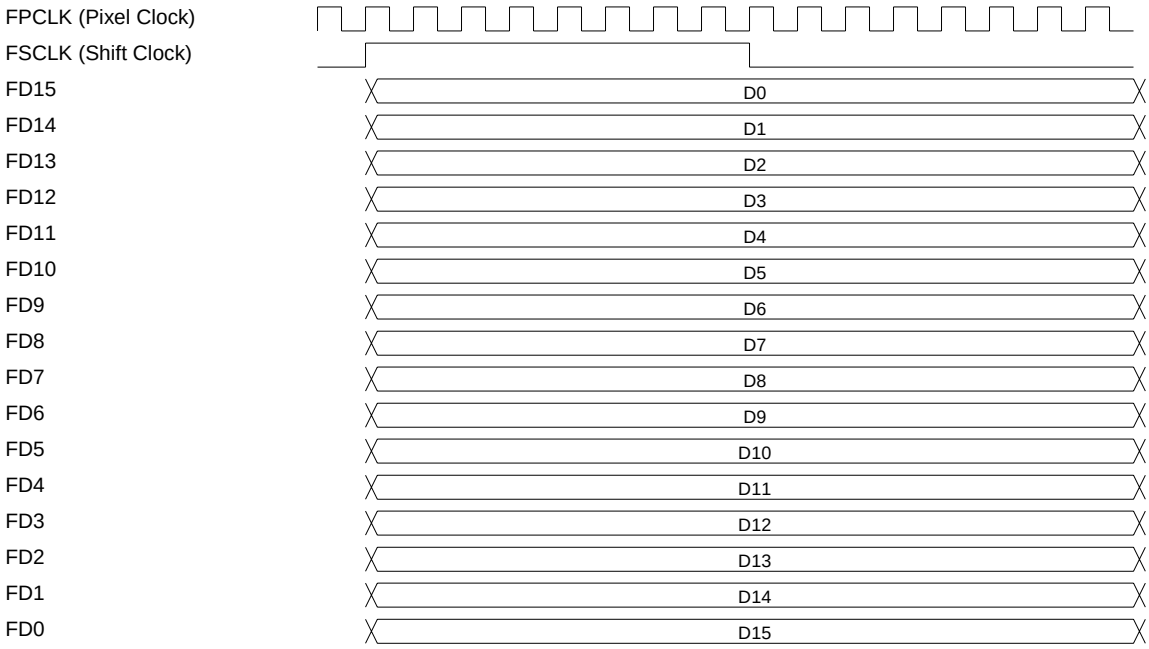
3.5.2.5.2 8-Bit Mono S-STN

8-bit Mono S-STN



3.5.2.5.3 16-Bit Mono S-STN

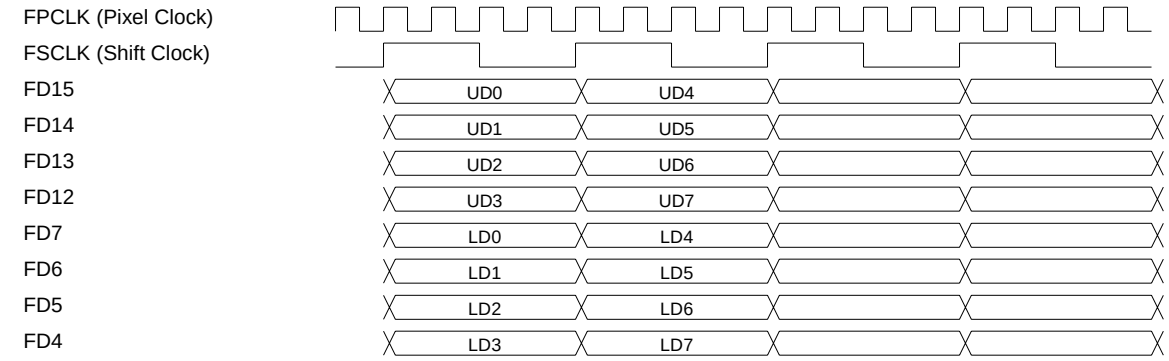
16-bit Mono S-STN



3.5.2.6 Timing Diagrams for Mono D-STN Panels

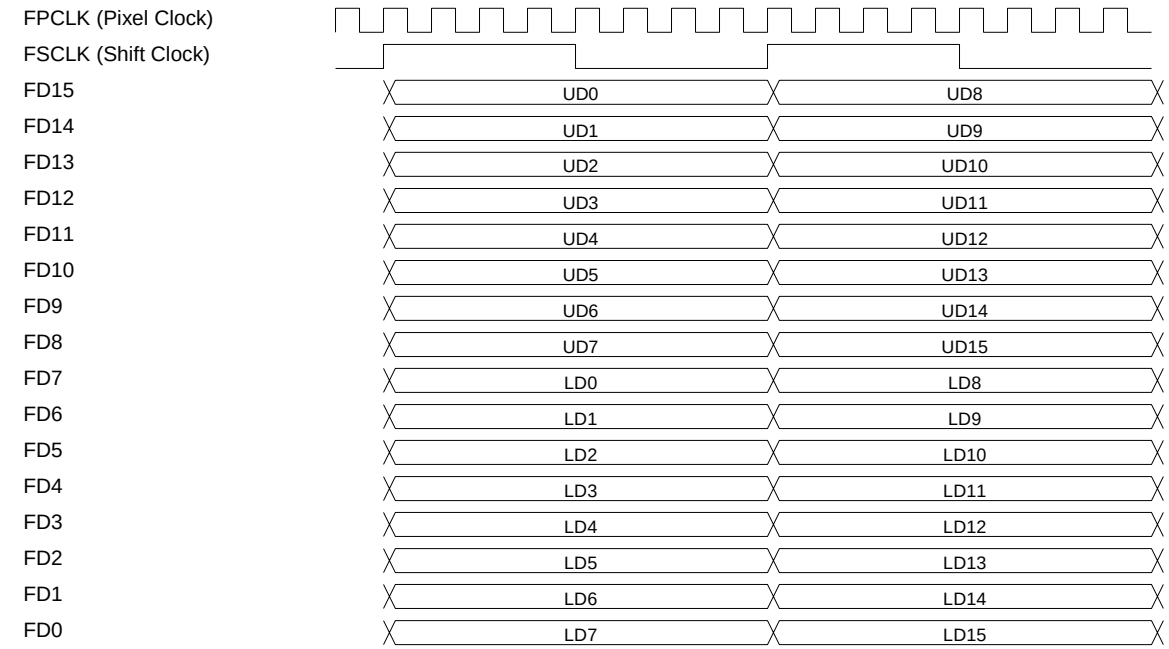
3.5.2.6.1 8-Bit Mono D-STN

8-bit Mono D-STN



3.5.2.6.2 16-Bit Mono D-STN

16-bit Mono D-STN



3.5.2.7 Pulse-Width Modulator

Two Pulse-Width Modulator (PWM) outputs are provided. Each output generates a square wave of programmable duty cycle and frequency. An external active or passive circuit can be used to generate a voltage output that is proportional to the duty cycle of the PWM output. This circuit allows software to control the contrast and backlight intensity of the LCD panel by varying the PWM output duty cycle.

The PWM control register contains separate clock and duty cycle fields for each PWM output, therefore allowing the two PWM outputs to operate independently. There are three possible sources for each of the PWM source clocks:

- CPU bus clock: From 25 MHz to 66 MHz depending on OEM design
- Oscillator clock: From 12 MHz to 25 MHz depending on OEM design
- Power management clock: 16.384 KHz

The source clock is further divided by a value ranging from 1 to 15 to produce the PWM input clock frequency. These options provide a wide range of PWM input clocks frequencies from 1092Hz to 66MHz. The PWM output frequency is always equal to the PWM input clock frequency divided by 256.

The duty cycle is set by an 8-bit field in the PWM control register, offering a range of duty cycles from 0.39% (=1/256) to 100% (256/256) with 0.39% increments. Programming this field with 127 will produce a 50% (128/256) duty cycle. Programming with 255 will produce a 100% (256/256) duty cycle which means that the output signal will be held at high level. Disabling the PWM logic will cause the PWM output to be driven low.

PWM0 or PWM1, or both, can be programmed to be part of the LCD panel power sequencing. When PWM outputs are included in the LCD power sequencing, they will be activated just before the ENCTL signal is activated, and will deactivated just after the ENCTL signal is deactivated. The PWM output will be driven low when it is deactivated.

3.5.2.7.1 Controlling Backlight and Contrast

Two very useful applications of the pulse-width modulator are software-controlled flat panel backlight and contrast. The user can provide input to the system using slider switches, arrow keys, touch pad movements, etc., to indicate the desired level of contrast or backlight. The system software can then program the PWM's pulse width and period to generate the duty cycle and frequency corresponding to the desired voltage. The use of PWM to control the contrast and backlight also allows system software to easily adjust contrast and backlight for power management.

3.5.2.8 General Purpose I/O Port

Three General Purpose I/O pins are provided on the MQ-100/HD64464. Each pin may be programmed as input or output.

When configured as inputs, input data from these pins is not synchronized to any MQ-100/HD64464 clocks, therefore the application software should read the register twice to make sure the signal was not changing state when it was read the first time.

3.5.2.9 General Purpose Output Port

Four bits of general-purpose output port are provided. These signals share pins with other signals as follows:

- GPO0 / ENCTL
- GPO1 / ENVEE

- GPO2 / PWM0
- GPO3 / PWM1

If the logic signal shared with an output port bit is not needed, the pin can be programmed as the output.

3.5.3 CRT Interface

The CRT interface consists of a triple 8-bit DAC that produces the red, green and blue analog outputs. These outputs are capable of directly driving a 75 ohm doubly-terminated load. Timing signals and pixel data from either graphics controller can be used to generate the HSYNC, VSYNC and monitor outputs required by the CRT. Monitor sense circuitry on each analog output allows software to determine whether there is a monitor attached, and whether it is monochrome or color.

3.5.3.1 CRT Control Signals

The CRT interface generates five signals:

- AR – Analog Red Output
- AG – Analog Green Output
- AB – Analog Blue Output
- HSYNC – Horizontal Sync Output
- VSYNC – Vertical Sync Output

The analog outputs can drive a 75 ohm doubly-terminated monitor directly. HSYNC and VSYNC are 3.3V outputs, therefore external level shift circuitry to 5V is required to prevent backpowering of the MQ-100/HD64464 by the CRT monitor.

3.5.3.2 Digital-to-Analog Conversion

The MQ-100/HD64464 contains three high-speed 8-bit DACs that convert the pixel color information into an analog RGB output. These analog components require some external bias components; refer to pinout section for more information.

3.5.3.3 Blanking and SYNC Pedestal

The MQ-100/HD64464 does not provide a BLANK output for the CRT. Blanking is accomplished using a blanking pedestal. When BLANK is provided separately, the output signal ranges from full black to full white. With a blanking pedestal, the voltage levels representing black to white are shifted up, and the new minimum level is used when blanking is needed. If the CRT does not have direct support for a blanking pedestal, the user can adjust the brightness until any visible scan lines are no longer visible.

A SYNC pedestal is also supported on the Green analog output.

3.5.3.4 Cyclic Redundancy Check (CRC)

A CRC function is provided for the data going to the CRT DACs which can be used for software diagnostics of the internal graphics controller datapath. The CRC value is computed based on the pixels for one display frame. Diagnostics software can read the CRC checksum value to verify integrity of the graphics controller datapath.

3.5.4 Graphics Controller Register Set

Please refer to the Programming Information Section for a description of these registers.

4 Power Management

4.1 Theory of Operation

The Power Management Unit (PMU) controls the power state transitions, internal power on/off sequencing including the LCD power sequencing. On power-up or reset, all the MQ-100/HD64464 logic modules, are in an inactive state. The first operation in the MQ-100/HD64464 initialization procedure must be to program the PMU to turn on the other functional modules. The MQ-100/HD64464 device drivers provide all requisite initialization.

Individual MQ-100/HD64464 logic modules employ DynamiQ^{TM} dynamic power management to minimize the MQ-100/HD64464 operating power. If a module is not in use, it may be turned off by setting the appropriate bit in the PMU control register.

The PMU supports five operational states. Three states (D0, D1, D2) are normal operating modes where some logic modules are powered down, the fourth and fifth states (D3 and D4) allow for powering down all of the MQ-100/HD64464 functional modules except for optional slow refresh of the internal memory and the power management state machine. Transitions between D0, D1, D2, and D3 are controlled by software; transitions in and out of D4 can only be controlled by the PDWN# pin. The five operating states are described below:

4.1.1 D0 State

Normal operating mode. In this state, all the functional modules on the MQ-100/HD64464 are fully operational. However, setting the appropriate control bits in the PMU control registers can enable or disable (power down) any of the functional modules.

4.1.2 D1 State

First level power saving mode. In this state, the CPU interface is active and therefore registers are accessible. PMU control register PM01R controls which functional modules are powered down in this mode.

4.1.3 D2 State

Second level power saving mode. In this state, the CPU interface is active and therefore registers are accessible. PMU control register PM02R controls which functional modules are powered down in this mode. **Programmer's Note:** PMU control register PM02R should be programmed such that the functional modules powered down in D2 state are a superset of those powered down in D1 state. This results in lower power dissipation in D2 state than in D1 state.

4.1.4 D3 State

This is the lowest power saving mode that can be entered or exited through software. In this state, most of the CPU interface is inactive. Only registers in the configuration space are accessible in this state. All other functional modules are forced into power down mode with possible exception of the internal memory refresh logic. When the Power-On Reset pin (POR#) is active (low), the chip will always enter this state. After POR# is deactivated (pulled high), the chip will remain in this state if Power Down (PDWN#) is inactive (high). In this state, system software can write to PMCSR register to revert to D0/D1/D2 state or the system can activate PDWN# pin to enter the D4 state. The first time the MQ-100/HD64464 enters the D3 state, the internal memory refresh is disabled. Subsequently, PM00R[16] bit can be set to enable internal memory refresh in D3 state.

4.1.5 D4 State

This is the lowest power mode on the MQ-100/HD64464 and can be activated only by forcing the PDWN# low. In this state, the CPU interface is completely disabled and is powered down along with all other functional modules with the exception of the internal memory refresh logic, which is

controlled by PM00R[17]. Immediately after the Power-On Reset pin (POR#) is forced high, PDWN# pin driven low (active), the MQ-100/HD64464 transitions from D3 state to D4 state and the internal memory refresh is disabled. This state can be exited by forcing the PDWN# pin high (inactive).

4.1.5.1 Software-Initiated Mode Transistions

Upon a Power-On Reset (POR# asserted and PDWN# is forced high), the MQ-100/HD64464 is in D3 state and may be transitioned to a different mode by writing to the PMCSR register. PMCSR register bits [1:0] control the power state (D0, D1, D2 or D3) in which the MQ-100/HD64464 is operating. Power State D4 can be entered only by forcing the PDWN# pin low (active).

4.1.5.2 Hardware Mode Transistions

D3 is the default state when POR# (Power-On Reset) is asserted.

The MQ-100/HD64464 enters the D4 state when PDWN# (Power Down Request) is asserted and POR# is not asserted, and will stay in D4 as long as PDWN# remains asserted. When PDWN# is deasserted, the PMU will return to the state designated in PMCSR[1:0].

If both POR# and PDWN# are asserted, the MQ-100/HD64464 will be reset and will enter D3. When POR# is deasserted, if PDWN# is still asserted the MQ-100/HD64464 will enter D4 state with memory refresh disabled.

4.1.6 Role of the Operating System in Managing Power

Although the current version of Windows CE does not explicitly support all of the MQ-100/HD64464 power management modes, the OEM can take advantage of the MQ-100/HD64464 power management through the system software drivers.

4.1.7 Role of the Device Driver in Managing Power

Each WindowsCE device driver implements the power management API. The device drivers for the MQ-100/HD64464 will take high-level power management commands from the OS and implement them by programming the device appropriately.

4.2 Power Management Unit

The Power Management Unit (PMU) is comprised of a state machine that is clocked by the Power Management Clock (PMCLK). Power to individual modules is turned on or off one at time. The LCD Power sequencing is also performed by the Power Management Unit.

A state machine clocked by a separate power clock (PMCLK) takes care of the power sequencing required when state transitions take place. The registers in the PMU module control most of the MQ-100/HD64464's clock control.

The PMU can enable or disable the following modules:

- Oscillator and PLLs
- Graphics Controller 1 (GC1)
- Graphics Controller 2 (GC2)
- Flat Panel Interface (FPI)
- CRT Interface (CRT)
- Memory Interface Unit (MIU)
- Graphics Engine (GE)
- CPU Interface (CPU)

The PMU also takes into account the dependency between the CRT interface or Flat Panel interface (FPI) and the graphics controller (which may be GC1 or GC2) that drives it. For example, if GC1 is used to drive the FPI, then the FPI will be powered down if GC1 is powered down. However, it is possible to power down the FPI without powering down the graphics controller.

4.2.1 Input Clocks

The power management state machine is clocked with the clock coming from the Power Management Clock pin (PMCLK). The frequency of the PMCLK is expected to be 16.384 KHz (½ the frequency of the RTC clock in SH-7709/SH-7750 systems).

An oscillator circuit supplies the reference clock to the three on-chip PLLs. The reference clock for the second and third PLLs can also be supplied with external oscillator via the L2CLK and L3CLK input pins.

4.2.2 Power Sequencing

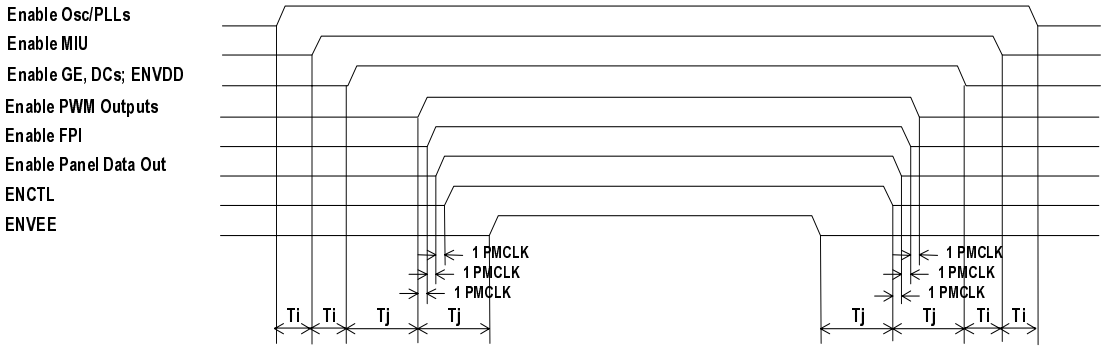


Figure 4A: Power Sequencing

The diagram above shows the order of power sequencing once the CPU interface is turned on, including the flat panel control. The delays between steps in the sequence (Ti and Tj in the above diagram) are programmable. The delay between steps in power change sequence is programmable in PM00R when PM00R[23] is set to 1. PM00R[23] is reset to 0 for test purposes on reset, and therefore software must initialize it to 1 for normal operation. Ti can be programmed to 16, 32, 64 or 128 PMCLK cycles and is controlled by PM00R[19:18]. Tj can be programmed to 512, 1024, 2048 or 4096 PMCLK cycles and is controlled by PM00R[21:20].

When transitioning from one device state to another the PMU will do the proper power-on or power-off sequencing when activating or deactivating a module. In general, the power-on sequence is:

1. CPU Interface
2. Clock oscillator and PLLs
3. Memory Interface Unit (MIU) for internal frame buffer
4. Graphics Controller 1, 2 and Graphics Engine
5. CRT DAC and Flat Panel Interface

Power-off sequencing is in the reverse order of the power-up sequencing.

When turning on or turning off the flat panel interface, proper flat panel power-on and power-off sequencing will also be performed via ENVDD, ENCTL, and ENVEE pins.

4.3 Power Management Register Set

Please refer to the Programming Information Section for details on the power management register set.

5 Programming Information

5.1 MQ-100/HD64464 Address Map

The MQ-100/HD64464 connects directly to Hitachi SH-7750 and SH-7709 microprocessors. These CPUs support a Logical Address Space of 4 GBytes and a Physical Address Space of 448 MBytes. The physical address space is decoded internally by the CPU, which generates 7 chip select signals and a 26-bit address bus. Each chip select region consists of 64 MBytes of address space.

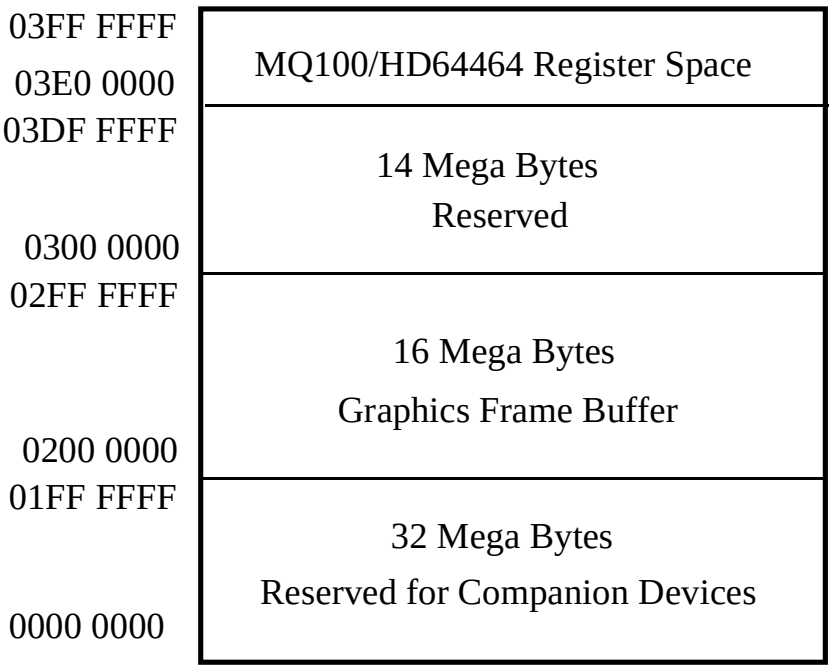


Figure 5A: Memory Map SH-7709 Interface

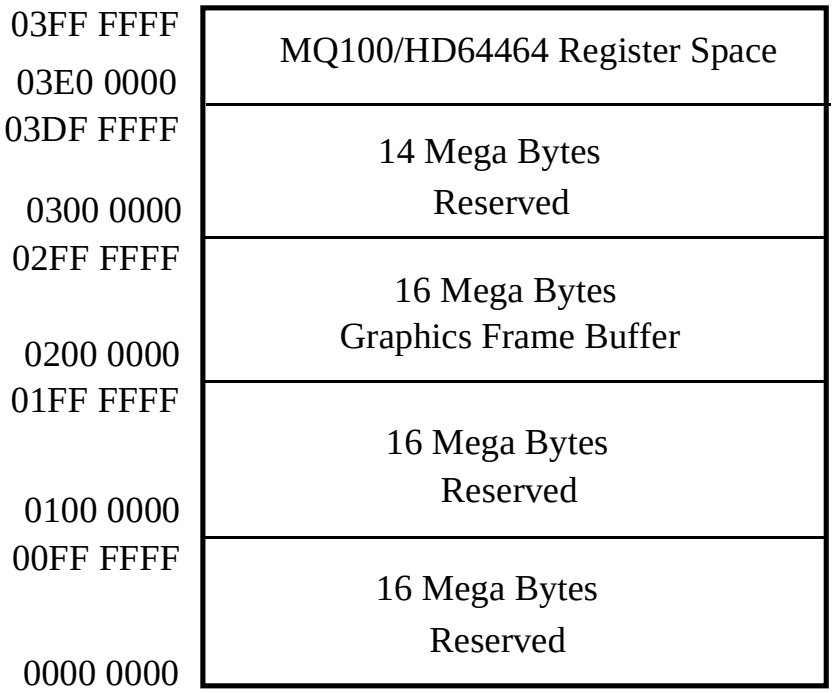


Figure 5B: Memory Map SH-7750 Interface

The MQ-100/HD64464 CS# input is connected to CS1# or CS4# output of the SH-7750 processor or to the CS4# output of the SH-7709 processor. The chip select region which is mapped to MQ-100/HD64464 must be programmed to be a 32-bit region in the processor. When the CPU selects the MQ-100/HD64464 by asserting MQ-100/HD64464 CS# pin, address bits A[25:2] are decoded to select a 32-bit MQ-100/HD64464 register or a 32-bit internal memory location. In the case of SH-7709, the MQ-100/HD64464 only uses the upper half of the 64-MByte space, leaving the lower 32 MBytes space usable for other devices, as shown below:

The register and memory addresses specified in this document are offsets within the 64-MByte region decoded by the MQ-100/HD64464; they are not the complete 32-bit addresses which are required by the application software. The upper address bits are determined by the chip select region to which the MQ-100/HD64464 is assigned.

In SH-7709 and SH-7750 systems, the upper half of the CS4# region is assigned to MQ-100/HD64464. Designers of SH-7750 systems can select either CS4# or CS1# to be connected to the CS# input of MQ-100/HD64464.

MQ-100/HD64464 registers are aligned on 32-bit boundaries. Registers are in the various MQ-100/HD64464 function modules. Each module is assigned a 8-KBytes address region within the upper 2 MBytes of internal address space. The address mapping of modules is shown in Table 5-1.

Table 5-1: Address Mapping of Modules

MQ-100/HD64464 Module	Address Range
Power Management + Clock Generation	x3E0 0000 to x3E0 1FFC
CPU Interface	x3E0 2000 to x3E0 3FFC
Memory Interface Unit	x3E0 4000 to x3E0 5FFC
Reserved	x3E0 6000 to x3E1 3FFC
Interrupt Controller	x3E1 4000 to x3E1 5FFC
Reserved	x3E1 6000 to x3E1 DFFC
Graphics Controller 1 and 2	x3E1 E000 to x3E1 FFFC
Graphics Engine	x3E2 0000 to x3E2 1FFC
Flat Panel Controller	x3E2 2000 to x3E2 3FFC
Color Palette 1	x3E2 4000 to x3E2 5FFC
Color Palette 2	x3E2 6000 to x3E2 7FFC
Reserved	x3E2 8000 to x3E2 9FFC
Configuration Registers	x3E2 A000 to x3E2 BFFC
Reserved	x3E2 C000 to x3FF FFFC

Register addresses in this document are specified as an index from the base address of the module in which they are to be found. The index is the offset from the base address of the module. For example, the Foreground Color Register has an index of 14 (hex) in the Graphics Engine module, which has a base address of x3E20000. This maps to an address of: x3E20014 for the Foreground Color Register.

5.2 REGISTER LOCATOR TABLE

Table 5-2: Register Locator Table

Table 5-2: Register Locator Table (Continued)

Section	Register	Register Description	Page #
5.5.41	GC2FR:	Window 2 Line Size Register	78
5.5.42	GC30R:	Hardware Cursor 2 Position	79
5.5.43	GC31R:	Hardware Cursor 2 Start Address & Offset Register	79
5.5.44	GC32R:	Hardware Cursor 2 Foreground Color Register	80
5.5.45	GC33R:	Hardware Cursor 2 Background Color Register	80
5.5.46	GC34R - GC3FR:	Reserved	80
5.5.47	C100R - C1FFR:	Color Palette 1	81
5.5.48	C200R - C2FFR:	Color Palette 2	82
5.6.1	FP00R:	Flat Panel Control Register	83
5.6.2	FP01R:	Flat Panel Output Pin Control Register	85
5.6.3	FP02R:	General Purpose Output Port Control Register	87
5.6.4	FP03R:	General Purpose I/O Port Control Register	88
5.6.5	FP04R:	STN Panel Control Register	89
5.6.6	FP05R:	Dual-STN Half-Frame Buffer Control Register	89
5.6.7	FP0FR:	Pulse Width Modulation Control Register	90
5.6.8	FP10R - FP2FR:	Frame-Rate Control Pattern Registers	91
5.6.9	FP30R - FP37R:	Frame-Rate Control Weight Registers	91
5.7.10	GE00R:	Primary Drawing Command Register	100
5.7.11	GE01R:	Primary Width and Height Register	102
5.7.12	GE02R:	Primary Base Address Register	102
5.7.13	GE03R:	Primary Source XY Register	103
5.7.14	GE04R:	Primary Destination XY Register	103
5.7.15	GE05R:	Primary Clip Left/Top Register	103
5.7.16	GE06R:	Primary Clip Right/Bottom Register	104
5.7.17	GE07R:	Primary Source and Pattern Offset Register	104
5.7.18	GE08:	Primary Foreground Color Register / Rectangle Fill Color	105
5.7.19	GE09R:	Primary Background Color Register	105
5.7.20	GE0AR:	Primary Mono Pattern Register 0	106
5.7.21	GE0BR:	Primary Mono Pattern Register 1	106
5.7.22	GE0CR:	Primary Source and Destination Stride Register	107
5.7.23	GE0DR:	Primary Base Address Register	108
5.7.24	GE1FR:	Test Mode Read Register	108
5.7.25	GE20R - GE3FR:	Primary Source Image Data Registers	109

Table 5-2: Register Locator Table (Continued)

Section	Register	Register Description	Page #
5.7.26	GE40 - GE5FR:	Primary Color Pattern Registers	110
5.7.27	GE80R:	Secondary Drawing Command Register	111
5.7.28	GE81R:	Secondary Width and Height Register	112
5.7.29	GE82R:	Secondary Base Address Register	112
5.7.30	GE83R:	Secondary Source XY Register	112
5.7.31	GE84R:	Secondary Destination XY Register	113
5.7.32	GE85R:	Secondary Clip Left/Top Register	113
5.7.33	GE86R:	Secondary Clip Right/Bottom Register	113
5.7.34	GE87R:	Secondary Source and Pattern Offset Register	114
5.7.35	GE88R:	Secondary Foreground Color Register / Rectangle Fill Color	115
5.7.36	GE89R:	Secondary Background Color Register	115
5.7.37	GE8AR:	Secondary Mono Pattern Register 0	115
5.7.38	GE8BR:	Secondary Mono Pattern Register 1	116
5.7.39	GE8CR:	Secondary Source and Destination Stride Register	116
5.7.40	GE8DR:	Secondary Base Address Register	117
5.7.41	GE20R-GE3FR	Source Image Data Registers	117
5.7.42	GEC0R - GEDFR:	Secondary Color Pattern Registers	117
5.8.2	CC00R:	CPU Control Register	118
5.8.3	CC01R:	GE and GE FIFO Status Register	119
5.8.4	IN00R:	Global Interrupt Control Register	120
5.8.5	IN01R:	Interrupt Mask Register	120
5.8.6	IN02R:	Interrupt Status Register	121
5.8.7	IN03R:	“Raw” Interrupt Status Register	122
5.9.1	MM00R:	Memory Interface Control Register	123
5.9.2	MM01R:	Memory Interface Control Register 2	124
5.9.3	MM02R:	Memory Interface Control 3	125
5.9.4	MM03R:	Memory Interface Control Register 4	126
5.10.1	PMR:	Power Management Unit Configuration Register	127
5.10.2	PMCSR:	Power Managent Control/Status Register	127
5.10.3	PM00R:	Power Management Miscellaneous Control Register	128
5.10.4	PM01R:	D1 State Control	131
5.10.5	PM02R:	D2 State Control	133
5.10.6	PM05R:	PLL1 Configuration Register	135
5.10.7	PM05R:	PLL2 Configuration Register	136
5.10.8	PM05R:	PLL3 Configuration Register	137

5.3 DEVICE INITIALIZATION

5.3.1 Recommended Initialization Procedure

Upon Power On Reset (with PDWN# forced high), the Power Management State is set to D3. The MQ-100/HD64464 will be disabled until the power-up sequence is completed. The MQ-100/HD64464 device driver then writes a value of 0 (for power management state D0) or 1 (for power management state D1) etc. The PMU takes some time to accomplish the transition of power management states, as power sequencing takes place internal to the MQ-100/HD64464 in the order dictated by the PMU. Initialization software must wait for the power-up sequencing to complete before further programming of the MQ-100/HD64464. Completion is indicated for example, when the PMCSR register reads 0, confirming that state D0 has been entered and the MQ-100/HD64464 is ready to be programmed.

The MQ-100/HD64464 functional modules each have configuration registers that must be set up before they can operate. These registers are described in detail in the individual programming information sections that follow.

5.4 Memory Configuration

No user configuration is required for the embedded memory in the MQ-100/HD64464. Memory Interface Unit control registers are defined in the Programming Information section for reference only. These registers should not be modified except by the MediaQ-supplied device drivers.

5.5 Graphics Controller Programming Information

Each MQ-100/HD64464 graphics controller contains a programmable timing generator for establishing a stable display area on the output device, and a programmable pixel serializer that fetches display data from the on-chip frame buffer, adds color information as needed and sends a stream of pixels to the display interface. The steps required to configure a graphics controller are outlined below.

The first step in programming the graphics controllers is to set the GC1 and GC2 configuration in registers GC00R and GC20R (please refer to the index for the page number of the detailed description of each register). This will determine the resolution, color depth, refresh rate and other parameters for each display output. To set a particular display mode, the first step is to select a clock and set the pixel clock frequency. Next, the graphics controller registers must be programmed with the horizontal and vertical size counts and the horizontal and vertical sync counts. Finally, the sizes of the main and alternate display windows must be programmed.

To set the clock frequency, the clock select and divisor fields in GC00R/GC20R must be set according to the following formula:

Pixel Clock = Source Clock ÷ FD_field ÷ SD_field

The source clock may be CKIO, PLL1, PLL2, or PLL3. The First Divisor (FD) divides the selected source clock by a factor from 1 to 6.5, and the Second Divisor (SD) further divides the clock by an 8-bit value. SD may range from 1 to 255; if it is programmed to 0, the pixel clock will be disabled and the clock generation logic will be shut down.

The horizontal and vertical parameters of the display output are set by programming fields in graphics controller registers, as shown in the following diagram for GC1:

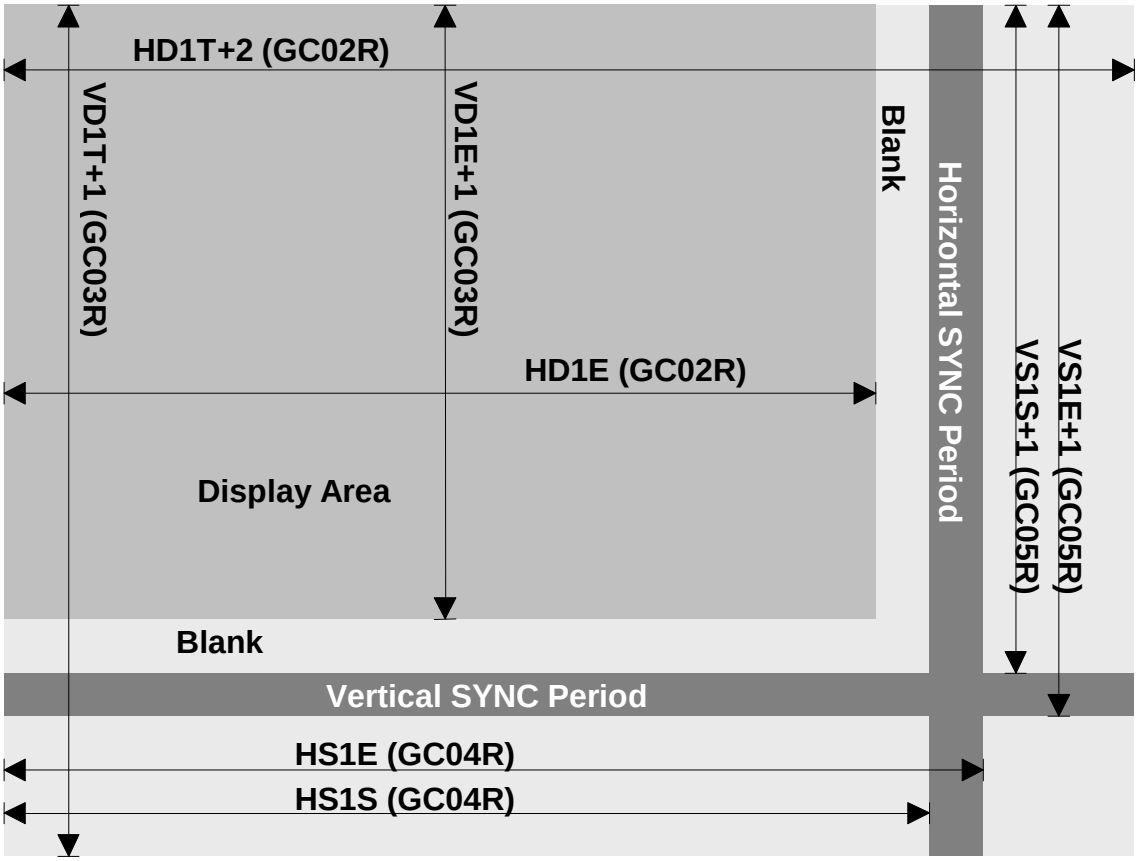


Figure 5C: Horizontal and Vertical Parameters

SYNC timings are programmed with respect to the display area. Horizontal SYNC timing is set in terms of a number of pixel clocks from the left side of the display area. Vertical SYNC timing is set in terms of a number of lines from the top of the display area. HSYNC and VSYNC parameters are diagrammed in the following two figures:

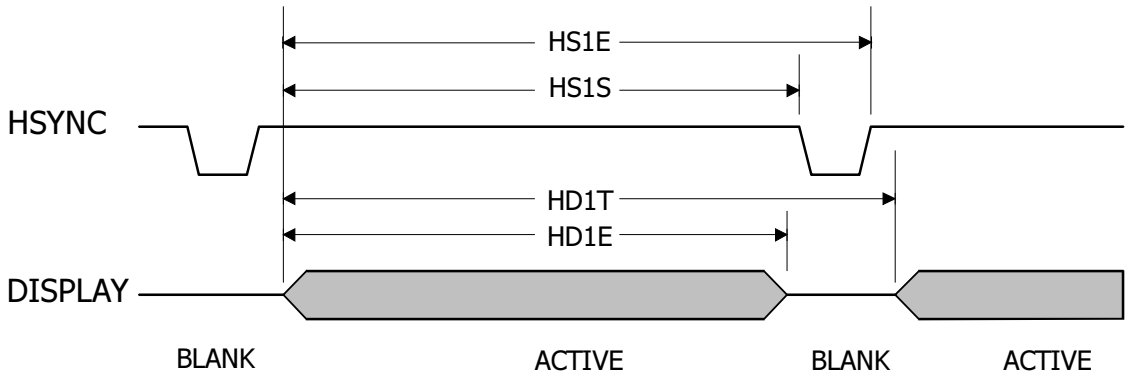


Figure 5D: HSYNC Parameters

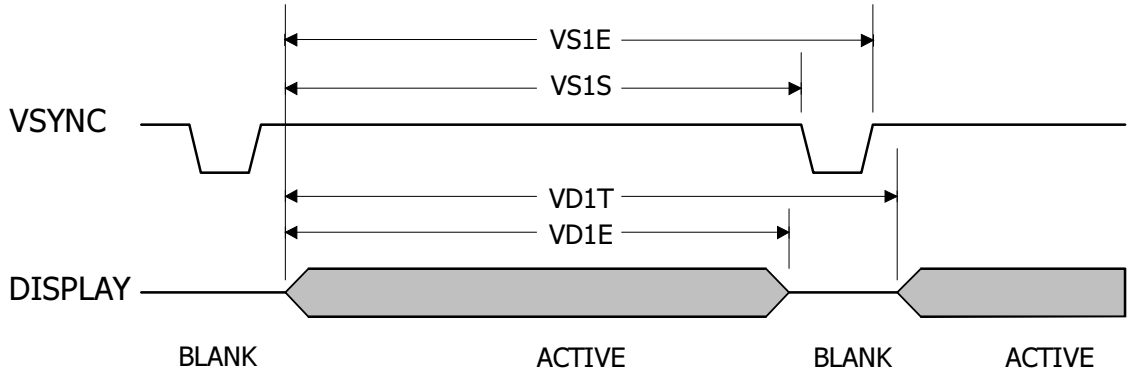


Figure 5E: VSYNC Parameters

Note that lines are themselves made up of a number of pixel clocks, the HD1E and HD1T fields in GC02R and GC22R set how many pixel clocks are in each line.

Once the display modes have been established, the output device(s) will show a stable display area. The graphics controllers must then be programmed to select and position the main and alternate windows. This is done by setting the source image address (GC0CR/GC2CR), line size (GC0FR/GC2FR), stride (GC0ER/GC2ER), and color depth (GC00R/GC20R). Note that either the main or alternate window may be active (not both at the same time), and that the active display window must lie within the fixed size of the display area. The display windows can be smaller than the fixed display size; in this case, the active display area outside the window will be blanked. Normally, the main window is programmed to the full size of the display, and the alternate window is used to display a smaller image.

Each pixel read from display memory passes through the Color Look-Up Table (LUT, also called Color Palette). The LUT must be programmed for all display modes. Each graphics controller has its own LUT, which is a 256 x 8 x 3 array. The pixel value is used to select an 8-bit red, 8-bit green and 8-bit blue color value that is passed to the CRT and/or Flat Panel Interfaces. The LUTs are programmed by writing to registers GC100R to GC1FFR for graphics controller 1 and GC200R to GC2FFR for graphics controller 2. For example, GC100R/200R should be written with the three 8-bit color values intended for pixel 0; GC101R/201R with the values for pixel 1, and so on to GC1FFR/2FFR for pixel 255.

The hardware cursor is controlled by GC10R/GC11R (cursor 1) and GC30R/GC31R. These registers set the respective cursor positions, the address of the cursor control images in off-screen display memory and the cursor clipping offsets. The horizontal and vertical clipping offsets specified in GC11R (for H/W cursor 1) and GC211R (for H/W cursor 2) allow the cursor to be scrolled smoothly off the left or top of the display. Scrolling off the right or bottom sides is accomplished by simply placing the cursor window so that it overhangs into the blanking region. Since this isn't possible on the left or top sides (the smallest horizontal/vertical position possible is 0/0), the two offsets are needed. A horizontal offset causes the cursor image to begin at that pixel instead of at pixel 0, in effect clipping all pixels to the left of the offset point. A vertical offset has a similar effect by clipping off any lines above the offset point.

5.5.1 CRT Interface

The CRT interface is configured by register GC01R. This register enables the CRT interface, sets the pin states in different operating modes, sets the configuration of the on-chip DACs, determines which graphics controller will supply data to the CRT interface, and provides the results of the monitor sense tests.

5.5.2 Flat Panel Interface

The Flat Panel Interface configuration is described in the Flat Panel Programming Information section.

5.5.3 Single-Display Configuration

The simplest type of application involves using one graphics controller to drive one type of output device.

5.5.4 Simultaneous Display Configuration

One graphics controller provides the pixel stream and display timing frame to both a CRT and a flat panel. The same data is displayed on both devices. Both display devices must be the same resolution and must operate at the same refresh rate.

5.5.5 QView™ Single Image

QView™ allows the CRT and flat panel to be run at different resolutions and refresh rates. This is an improvement over simultaneous display because it allows the CRT to operate at a higher refresh rate than the LCD panel, resulting in a more stable, sharper image.

5.5.6 QView™ Dual Image

This allows the end user to make the most effective use of the two displays. LCD and CRT both have an independent image, allowing different resolutions, color depths and refresh rates. Small LCDs do not limit the size of the CRT display or throttle the refresh rate of the CRT.

5.5.7 GC00R: Graphics Controller 1 Control Register

This register sets the following parameters for Graphics Controller 1:

- Controller Enable: turns on graphics controller when set; controller is powered down when cleared
- Vertical and Horizontal Reset: individual reset bits to initialize the display timing generator
- Primary Image Window Enable: frame buffer data is not displayed until this bit is set
- Alternate Image Window Enable: selects the alternate image window for display
- Color Depth: selects 1-, 2-, 4-, 8-, 15-, and 16-bits per pixel for the primary and alternate image windows
- Hardware Cursor Enable: turns the hardware cursor on/off
- Display Clock Configuration: several fields that together determine the pixel clock rate

GC1	GC00R - Graphics Controller 1 Control *Index: 00 Reset value: 0000-0000h*	
0	*Controller 1 Enable* Setting this bit will enable Graphics Controller 1. If this bit is reset, Controller 1 is powered down and reset.	
	0	Controller 1 is disabled.
	1	Controller 1 is enabled.
1	*Horizontal Counter 1 Reset* Setting this bit will perform software reset on Controller 1 horizontal counter and on the FIFO.	
	0	Controller 1 horizontal counter 1 is enabled.
	1	Controller 1 horizontal counter 1 is reset.
2	*Vertical Counter 1 Reset* Setting this bit will perform software reset on Controller 1 vertical counter.	
	0	Controller 1 vertical counter 1 is enabled.
	1	Controller 1 vertical counter 1 is reset.
3	*Image Window 1 Enable* Setting this bit will enable Window 1 (image window on Controller 1). If this bit is reset, Window 1 will not be displayed.	
	0	Window 1 is disabled.
	1	Window 1 is enabled.
6-4	*Graphics Color Depth (GCD)* This parameter specifies the number of bits/pixel for the graphics window for Controller 1 when alternate window is disabled (bit 11 is reset). Color palette is always used regardless of color depth.	
	000	1-bpp graphics.
	001	2-bpp graphics.
	010	4-bpp graphics.
	011	8-bpp graphics.
	100	15-bpp (RGB555) graphics.
	101	16-bpp (RGB565) graphics.
	others	Reserved.
7	*Reserved (R/W)* This bit must be programmed to 0.	
8	*Hardware Cursor 1 Enable* Setting this bit enables hardware cursor for Controller 1.	
	0	Hardware cursor for Controller 1 is disabled.
	1	Hardware cursor for Controller 1 is enabled.
10-9	*Reserved (R/W)* These bits must be programmed to 0.	
11	*Alternate Image Window 1 Enable* This bit is effective only when bit 3 is set. Setting this bit will enable alternate Window 1. Note that power management logic can also be programmed to enable alternate Window 1.	
	0	Alternate Window 1 is disabled. In this mode Graphics Color Depth, Horizontal Window 1 Start/End, Vertical Window 1 Start/End, Window 1 Start Address, Window 1 Stride, Window 1 Line Size parameters are used.
	1	Alternate Window 1 is enabled. In this mode, Alternate Graphics Color Depth, Alternate Horizontal Window 1 Start/End, Alternate Vertical Window 1 Start/End, Alternate Window 1 Start Address, Alternate Window 1 Stride, Alternate Window 1 Line Size, and Alternate Window 1 Palette Index parameters are used. Note that color palette programming may have to be adjusted if the alternate window is running at different color depth.
14-12	*Alternate Graphics Color Depth (AGCD)* This parameter specifies the number of bits/pixel for the graphics window for Controller 1 when alternate window is enabled (bit 11 is set). Color palette is always used regardless of color depth.	

	000	1-bpp graphics.
	001	2-bpp graphics.
	010	4-bpp graphics.
	011	8-bpp graphics.
	100	15-bpp (RGB555) graphics.
	101	16-bpp (RGB565) graphics.
	others	Reserved.
15	*Reserved (R/W)* This bit must be programmed to 0.	
17-16	*G1RCLK Source* These bits select the clock source for the controller 1 root clock (G1RCLK).	
	00	G1RCLK source is bus clock.
	01	G1RCLK source is PLL1.
	10	G1RCLK source is PLL2.
	11	G1RCLK source is PLL3.
18	*Test Mode 0* Setting this bit enables factory test mode 0 for Controller 1. This bit must be programmed to 0 in normal operation.	
	0	Test mode 0 for Controller 1 is disabled.
	1	Test mode 0 for Controller 1 is enabled.
19	*Test Mode 1* Setting this bit enables factory test mode for Controller 1. This bit must be programmed to 0 in normal operation.	
	0	Test mode 1 for Controller 1 is disabled.
	1	Test mode 1 for Controller 1 is enabled.
22-20	*G1MCLK First Clock Divisor (FD1)* This parameter and bits 31-24 are used to generate controller 1 master clock (G1MCLK) from the root clock. This parameter specifies the divisor value for the first stage clock divider to generate G1MCLK. Note that the duty cycle for the output clock is not balanced except for the case where FD1 is 1.	
	000	FD1 = 1.
	001	FD1 = 1.5.
	010	FD1 = 2.5.
	011	FD1 = 3.5.
	100	FD1 = 4.5.
	101	FD1 = 5.5.
	110	FD1 = 6.5.
	111	Reserved.
23	*Reserved (R/W)* This bit must be programmed to 0.	
31-24	*G1MCLK Second Clock Divisor (SD1)* This parameter specifies the divisor value for the second stage clock divider which is used to divide generate G1MCLK. The divisor values ranges from 1 to 255. If this parameter is set to 0, G1MCLK is disabled and G1MCLK clock generation logic is powered down. G1MCLK = G1RCLK / FD1 / SD1.	

5.5.8 GC01R: CRT Interface Control Register

This register contains the control fields for configuring the CRT Interface.

GC1	GC01R - Graphics Controller 1 CRT Control *Index: 04 Reset value: xxxx-0000h*	
1-0	*CRT DAC Enable* These bits control CRT DAC output.	
	X0	CRT DAC is disabled. CRT HSYNC and CRT VSYNC will be powered down.
	01	CRT DAC is enabled and driven by Controller 1. Controller 1 must also be enabled.
	11	CRT DAC is enabled and driven by Controller 2. Controller 2 must also be enabled.
2	*CRT HSYNC Output during Power Down mode* This bit controls output on CRT HSYNC pin when Graphics Controller is powered down when bits 5-4 are set to 00.	
	0	CRT HSYNC output is forced low during power down mode and bits 5-4 are set to 00.
	1	CRT HSYNC output is Power Management clock (PMCLK) during power down mode and bits 5-4 are set to 00.
3	*CRT VSYNC Output during Power Down mode* This bit controls output on CRT VSYNC pin when Graphics Controller is powered down and bits 7-6 are set to 00.	
	0	CRT VSYNC output is forced low during power down mode and bits 7-6 are set to 00.
	1	CRT VSYNC output is Power Management clock (PMCLK) clock during power down mode and bits 7-6 are set to 00.
5-4	*CRT HSYNC Control* These bits control output on CRT HSYNC pin. Note that an external level shifter is needed on CRT HSYNC to convert from 3.3V to 5V.	
	00	Normal operation. If CRT DAC is powered down then bit 2 controls output on this pin.
	01	CRT HSYNC output pin is forced low.
	10	CRT HSYNC output pin is forced high.
	11	Reserved.
7-6	*CRT VSYNC Control* These bits control output on CRT VSYNC pin. Note that an external level shifter is needed on CRT VSYNC to convert from 3.3V to 5V.	
	00	Normal operation. If CRT DAC is powered down, then bit 3 controls output on this pin.
	01	CRT VSYNC output pin is forced low.
	10	CRT VSYNC output pin is forced high.
	11	Reserved.
8	*CRT HSYNC Polarity* This bit controls polarity of CRT HSYNC pin. This bit is effective only when CRT DAC is enabled (not in power down mode) and bits 5-4 are 00.	
	0	CRT HSYNC output is active high.
	1	CRT HSYNC output is active low.
9	*CRT VSYNC Polarity* This bit controls polarity of CRT VSYNC pin. This bit is effective only when CRT DAC is enabled (not in power down mode) and bits 7-6 are 00.	
	0	CRT VSYNC output is active high.
	1	CRT VSYNC output is active low.
10	*Sync Pedestal Enable* This bit enables Sync Pedestal on CRT DAC (green only). When Sync pedestal is enabled, the DAC output current increases by approximately 7.62 mA in non-Sync area. Sync pedestal is applied only to Green DAC only.	
	0	Sync pedestal is disabled.
	1	Sync pedestal is enabled.
11	*Blank Pedestal Enable* This bit enables Blank Pedestal on CRT DAC. When Blank pedestal is enabled, the DAC output current increases by approximately 1.45 mA in non-blank area.	

	0	Blank pedestal is disabled.
	1	Blank pedestal is enabled.
12	*Reserved (R/W)* This bit must be programmed to 0.	
13	*VREF Select* This bit controls voltage reference for the DAC. For driving CRT, the required voltage reference is 1.235V. With Sync and Blank pedestals disabled, the peak DAC current is 17.62 mA. With both Sync and Blank pedestals enabled, the peak DAC current is 26.67 mA	
	0	Internal VREF.
	1	External VREF.
14	*Monitor Sense Enable* This bit controls monitor sense circuit on the CRT DAC. Each of the Red, Green and Blue DACs has its own monitor sense circuit.	
	0	CRT DAC monitor sense circuit is disabled.
	1	CRT DAC monitor sense circuit is enabled.
15	*Constant Output Enable* This bit forces bits 23-16 to be output on Red, Green, and Blue DAC. This can be used when monitor sense is enabled.	
	0	Normal output on Red, Green, and Blue DAC.
	1	Bits 23-16 is output on Red, Green, and Blue DAC.
23-16	*Monitor Sense DAC Output Level* This parameter specifies the output level value on Red, Green and Blue DACs when monitor sensing is enabled (bit 4 = 1).	
24	*Blue DAC Sense Result (Read Only)* This bit returns the result of monitor sense circuit for the Blue CRT DAC.	
	0	Blue DAC is loaded.
	1	Blue DAC is not loaded.
25	*Green DAC Sense Result (Read Only)* This bit returns the result of monitor sense circuit for the Green CRT DAC.	
	0	Green DAC is loaded.
	1	Green DAC is not loaded.
26	*Red DAC Sense Result (Read Only)* This bit returns the result of monitor sense circuit for the Red CRT DAC.	
	0	Red DAC is loaded.
	1	Red DAC is not loaded.
31-27	*Reserved (0)*	

5.5.9 GC02R: Horizontal Display Timing Control Register 1

This register set the horizontal dimensions of the total size, the active area and the blanking area for the output of graphics controller 1.

GC1	**GC02R - Horizontal Display 1 Control** *Index: 08 Reset value: XXXX-XXXXH*
11-0	*Horizontal Display 1 Total (HD1T)* This parameter specifies the total length of horizontal display and blank area in terms of number of pixels. This also marks the start of the next display line with respect to the start of the current line. Constraint: HD1T ≥ HS1E + 4 where HS1E is Horizontal Sync 1 End (see GC04R). Programmed value = actual value – 2.
15-12	*Reserved (0)*
27-16	*Horizontal Display 1 End (HD1E)* This parameter specifies the horizontal size of the display area in terms of number of pixels. For flat panel, this parameter must be programmed to the horizontal panel size. Programmed value = actual value.
31-28	*Reserved (0)*

5.5.10 GC03R: Vertical Display Timing Control Register 1

This register set the vertical dimensions of the total size, the active area and the blanking area for the output of graphics controller 1.

GC1	**GC03R - Vertical Display 1 Control** *Index: 0C Reset value: XXXX-XXXXH*
11-0	*Vertical Display 1 Total (VD1T)* This parameter specifies the total height of Vertical display and blank area in terms of number of lines. This also marks the start of the next display frame with respect to the start of the current frame. Constraint: VD1T ≥ VS1E + 1 where VS1E is Vertical Sync 1 End (see GC05R). Programmed value = actual value – 1
15-12	*Reserved (0)*
27-16	*Vertical Display 1 End (VD1E)* This parameter specifies the Vertical size of the display area in terms of number of lines. For flat panel, this parameter must be programmed to the Vertical panel size. Programmed value = actual value – 1.
31-28	*Reserved (0)*

5.5.11 GC04R: Horizontal SYNC Control Register 1

This register defines the width and starting position for the Horizontal SYNC output of Graphics Controller 1.

GC1	**GC04R - Horizontal Sync 1 Control** *Index: 10 Reset value: XXXX-XXXXH*
11-0	*Horizontal Sync 1 Start (HS1S)* This parameter specifies the start position of Horizontal Sync pulse in terms of number of pixels with respect to the left edge of the display area. Constraint: HS1S ≥ HD1E + 4. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Horizontal Sync 1 End (HS1E)* This parameter specifies the end position of Horizontal Sync pulse in terms of number of pixels with respect to the left edge of the display area. Constraint: HS1E ≥ HS1S + 4. Programmed value = Horizontal Sync 1 Start + Horizontal Sync 1 Width.
31-28	*Reserved (0)*

5.5.12 GC05R: Vertical SYNC Control Register 1

This register defines the width and start position of the Vertical SYNC output of Graphics Controller 1.

GC1	**GC05R - Vertical Sync 1 Control** *Index: 14 Reset value: XXXX-XXXXH*
11-0	*Vertical Sync 1 Start (VS1S)* This parameter specifies the start position of Vertical Sync pulse in terms of number of lines with respect to the top edge of the display area. Constraint: VS1S ≥ VD1E + 1. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Sync 1 End (VS1E)* This parameter specifies the end position of Vertical Sync pulse in terms of number of lines with respect to the top edge of the display area. Constraint: VS1E ≥ VD1S + 1. Programmed value = Vertical Sync 1 Start + Vertical Sync 1 Height.
31-28	*Reserved (0)*

5.5.13 GC06R and GC07R: Reserved

These registers are reserved for future use.

5.5.14 GC08R: Graphics Controller 1 Main Window Horizontal Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the horizontal start position and width of the main window within the display frame output from Graphics Controller 1.

GC1	**GC08R - Horizontal Window 1 Control** *Index: 20 Reset value: XXXX-XXXXH*
11-0	*Horizontal Window 1 Start (HW1S)* This parameter specifies the horizontal start position of the graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is disabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Horizontal Window 1 End (HW1E)* This parameter specifies the horizontal end position of graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is disabled. Constraint: HW1E ≥ HW1S + 4. Programmed value = Horizontal Window 1 Start + Horizontal Window 1 Width.
31-28	*Reserved (0)*

5.5.15 GC09R: Graphics Controller 1 Main Window Vertical Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the vertical start position and height of the main window within the display frame output from Graphics Controller 1.

GC1	**GC09R - Vertical Window 1 Control** *Index: 24 Reset value: XXXX-XXXXH*
11-0	*Vertical Window 1 Start (VW1S)* This parameter specifies the vertical start position of the graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is disabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Window 1 End (VW1E)* This parameter specifies the vertical end position of graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is disabled. Constraint: VW1E ≥ VW1S + 1. Programmed value = Vertical Window 1 Start + Vertical Window 1 Height.
31-28	*Reserved (0)*

5.5.16 GC0AR: Graphics Controller 1 Alternate Window Horizontal Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the horizontal start position and width of the alternate window within the display frame output from Graphics Controller 1.

GC1	**GC0AR - Alternate Horizontal Window 1 Control** *Index: 28 Reset value: XXXX-XXXXH*
11-0	*Alternate Horizontal Window 1 Start (AHW1S)* This parameter specifies the horizontal start position of the graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is enabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Alternate Horizontal Window 1 End (AHW1E)* This parameter specifies the horizontal end position of graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is enabled. Constraint: AHW1E ≥ AHW1S + 4. Programmed value = Alternate Horizontal Window 1 Start + Alternate Horizontal Window 1 Width.
31-28	*Reserved (0)*

5.5.17 GC0BR: Graphics Controller 1 Alternate Window Vertical Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the vertical start position and height of the alternate window within the display frame output from Graphics Controller 1.

GC1	**GC0BR - Alternate Vertical Window 1 Control** *Index: 2C Reset value: XXXX-XXXXH*
11-0	*Alternate Vertical Window 1 Start (AVW1S)* This parameter specifies the vertical start position of the graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is enabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Alternate Vertical Window 1 End (AVW1E)* This parameter specifies the vertical end position of graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is enabled. Constraint: AVW1E ≥ AVW1S + 1. Programmed value = Alternate Vertical Window 1 Start + Alternate Vertical Window 1 Height.
31-28	*Reserved (0)*

5.5.18 GC0CR: Main Window 1 Start Address

This register is programmed with the on-chip display memory byte address of the first pixel in the image to be displayed within the main window of graphics controller 1. Note that since this value is not in terms of pixels, the first pixel of the image should always be byte-aligned.

GC1	**GC0CR - Window 1 Start Address** *Index: 30 Reset value: XXXX-XXXXH*
19-0	*Window 1 Start Address (W1SA)* This parameter specifies the start address of the graphics image window in the display frame buffer when alternate window is disabled. This is a byte address. Note that if the mode is less than 8 bits/pixel then the start address is restricted to pixels on byte boundary.
31-20	*Reserved (0)*

5.5.19 GC0DR: Alternate Window 1 Start Address

This register is programmed with the on-chip display memory byte address of the first pixel in the image to be displayed within the alternate window of graphics controller 1. Note that since this value is not in terms of pixels, the first pixel of the image should always be byte-aligned.

This register also specifies the Palette Index for the alternate window. Since the Color Look-Up Table (Palette) contains 256 entries for each color, pixel sizes less than 8 bits/pixel do not fully specify a lookup table entry. The Palette Index field is used as the upper bits and the pixel value as the lower bits to form a full 8-bit lookup table address. The Palette Index for the Main window is always 0; providing a programmable Palette Index for the alternate window allows different color maps to be used for the main and alternate windows.

GC1	**GC0DR - Alternate Window 1 Start Address** *Index: 34 Reset value: XXXX-XXXXH*
19-0	*Alternate Window 1 Start Address (AW1SA)* This parameter specifies the start address of the graphics image window in the display frame buffer when alternate window is enabled. This is a byte address. Note that if the mode is less than 8 bits/pixel then the start address is restricted to pixels on byte boundary.
24-20	*Reserved (0)*
31-25	*Alternate Window 1 Palette Index (AW1PI)* This parameter specifies the upper bits of palette index when alternate window is enabled in 1-bpp, 2-bpp, 4-bpp alternate graphics color depth. If alternate graphics color depth is 1-bpp then bits 31-25 are used for palette index bits 7-1. If alternate graphics color depth is 2-bpp then bits 31-26 are used for palette index bits 7-2. If alternate graphics color depth is 4-bpp then bits 31-28 are used for palette index bits 7-4.

5.5.20 GC0ER: Window 1 Stride Register

GC0ER sets the stride for the main and alternate windows of graphics controller 1. The “stride” is the distance in bytes between consecutive lines of an image in the display memory. Lines may be contiguous (i.e., the first pixel of the next line immediately follows the last pixel of the current line) but may also not be contiguous (as when a smaller window of a larger image is being displayed). In either case, the stride should be a multiple of 16 bytes.

GC1	**GC0ER - Window 1 Stride** Index: 38 Reset value: XXXX-XXXXH
15-0	*Window 1 Stride (AW1ST)* This parameter specifies the distance from the start of current line to the start of next line when alternate window is disabled. This is specified in terms of number of bytes. It is recommended that this parameter be specified in terms of frame buffer bus width (128 bits).
31-16	*Alternate Window 1 Stride (AW1ST)* This parameter specifies the distance from the start of current line to the start of next line when alternate window is enabled. This is specified in terms of number of bytes. It is recommended that this parameter be specified in terms of frame buffer bus width (128 bits).

5.5.21 GC0FR: Window 1 Line Size Register

GC0FR sets the line size in bytes for the main and alternate windows of graphics controller. This is the number of bytes that will be read from the display memory before starting to fetch the next line.

GC1	**GC0FR - Window 1 Line Size** *Index: 3C Reset value: XXXX-XXXXH*
13-0	*Window 1 Line Size (W1LS)* This parameter specifies the amount of data to be fetched from display frame buffer for each graphics image line when alternate window is disabled. This is specified in terms of number of bytes and subtracted by 1. Programmed value = (number of pixels per line * bits per pixel / 8 - 1) rounded up to nearest integer.
15-14	*Reserved (0)*
29-16	*Alternate Window 1 Line Size (AW1LS)* This parameter specifies the amount of data to be fetched from display frame buffer for each graphics image line when alternate window is enabled. This is specified in terms of number of bytes and subtracted by 1. Programmed value = (number of pixels per line * bits per pixel / 8 - 1) rounded up to nearest integer.
31-30	*Reserved (0)*

5.5.22 GC10R: Hardware Cursor 1 Position

GC10R sets the horizontal and vertical positions of the hardware cursor in graphics controller 1. To avoid flicker when a new position is written, the cursor will not be moved during the active display time (the update will take effect at the start of the next Vertical SYNC).

GC1	**GC10R - Hardware Cursor 1 Position** *Index: 40 Reset value: XXXX-XXXXH* Both Horizontal and Vertical Cursor 1 Start must be changed at the same time. This register takes effect on the next falling edge of Controller 1 Vertical Sync.
11-0	*Horizontal Cursor 1 Start (HC1S)* This parameter specifies the horizontal start position of the hardware cursor, in terms of number of pixels with respect to the left edge of the display area. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Cursor 1 Start (VC1S)* This parameter specifies the vertical start position of the hardware cursor, in terms of number of lines with respect to the top edge of the display area. Programmed value = actual value.
31-28	*Reserved (0)*

5.5.23 GC11R: Hardware Cursor 1 Start Address and Offset Register

GC11R sets the display memory byte address for the 1024-byte cursor image. It also sets the horizontal and vertical pixel offsets used for clipping the cursor on the left and top edges of the display window.

GC1	**GC11R - Hardware Cursor 1 Start Address and Offset** *Index: 44 Reset value: XXXX-XXXXH* Both Horizontal and Vertical Cursor 1 Offset must be changed at the same time and they take effect on the next falling edge of Controller 1 Vertical Sync.
9-0	*Hardware Cursor 1 Start Address (HC1SA)* This parameter specifies bits 19-10 of the start address of the hardware cursor in the display frame buffer. Note that the start address is restricted to 1K byte boundary, therefore the lower 10 bits of this parameter is forced to 0.
15-10	*Reserved (0)*
21-16	*Horizontal Cursor 1 Offset (HC1O)* This parameter specifies the horizontal offset of the hardware cursor, in terms of number of pixels with respect to the left edge of the cursor area. Hardware cursor horizontal offset, specifies the first horizontal cursor pixel to be displayed. This parameter is used for left-edge clipping of the hardware cursor.
23-22	*Reserved (0)*
29-24	*Vertical Cursor 1 Offset (VC1O)* This parameter specifies the vertical offset of the hardware cursor, in terms of number of lines with respect to the top edge of the cursor area. Hardware cursor vertical offset specifies the first vertical cursor line to be displayed. This parameter is used for top-edge clipping of the hardware cursor.
31-30	*Reserved (0)*

5.5.24 GC12R: Hardware Cursor 1 Foreground Color Register

GC12R sets the foreground color that will be used when the hardware cursor pixel is displayed on top of the background image. The color value is specified in 24-bit RGB format.

GC1	**GC12R - Hardware Cursor 1 Foreground Color** *Index: 48 Reset value: XXXX-XXXXH*
23-0	*Hardware Cursor 1 Foreground Color (HC1FC)* This parameter specifies the hardware cursor foreground color in the RGB format. Bits 23-16 store the red data, bits 15-8 stores the green data, and bits 7-0 stores the blue data. In case of monochrome mode, only the green data is used.
31-24	*Reserved (0)*

5.5.25 GC13R: Hardware Cursor 1 Background Color Register

GC13R sets the background color that will be used when the hardware cursor pixel is displayed on top of the background image. The color value is specified in 24-bit RGB format.

GC1	**GC13R - Hardware Cursor 1 Background Color** *Index: 4C Reset value: XXXX-XXXXH*
23-0	*Hardware Cursor 1 Background Color (HC1BC)* This parameter specifies the hardware cursor foreground color in the RGB format. Bits 23-16 store the red data, bits 15-8 stores the green data, and bits 7-0 stores the blue data. In case of monochrome mode, only the green data is used.
31-24	*Reserved (0)*

5.5.26 GC14R Through GC1FR: Reserved

These registers are reserved for future use.

5.5.27 GC20R: Graphics Controller 2 Control Register

This register sets the following parameters for Graphics Controller 2:

- Controller Enable: turns on graphics controller when set; controller is powered down when cleared
- Vertical and Horizontal Reset: individual reset bits to initialize the display timing generator
- Primary Image Window Enable: frame buffer data is not displayed until this bit is set
- Alternate Image Window Enable: selects the alternate image window for display
- Color Depth: selects 1-, 2-, 4-, 8-, 15-, and 16-bits per pixel for the primary and alternate image windows
- Hardware Cursor Enable: turns the hardware cursor on/off
- Display Clock Configuration: several fields that together determine the pixel clock rate

GC2	**GC20R - Graphics Controller 2 Control** *Index: 80 Reset value: 0000-0000h*	
0	*Controller 2 Enable* Setting this bit will enable Graphics Controller 2. If this bit is reset, Controller 2 is powered down and reset.	
	0	Controller 2 is disabled.
	1	Controller 2 is enabled.
1	*Horizontal Counter 2 Reset* Setting this bit will perform software reset on Controller 2 horizontal counter and on the FIFO.	
	0	Controller 2 horizontal counter 2 is enabled.
	1	Controller 2 horizontal counter 2 is reset.
2	*Vertical Counter 2 Reset* Setting this bit will perform software reset on Controller 2 vertical counter.	
	0	Controller 2 vertical counter 2 is enabled.
	1	Controller 2 vertical counter 2 is reset.
3	*Image Window 2 Enable* Setting this bit will enable Window 2 (image window on Controller 2). If this bit is reset, Window 2 will not be displayed.	
	0	Window 2 is disabled.
	1	Window 2 is enabled.
6-4	*Graphics Color Depth (GCD)* This parameter specifies the number of bits/pixel for the graphics window for Controller 2 when alternate window is disabled (bit 11 is reset). Color palette is always used regardless of color depth.	
	000	1-bpp graphics.
	001	2-bpp graphics.
	010	4-bpp graphics.
	011	8-bpp graphics.
	100	15-bpp (RGB555) graphics.
	101	16-bpp (RGB565) graphics.
	others	Reserved.
7	*Reserved (R/W)* This bit must be programmed to 0.	
8	*Hardware Cursor 2 Enable* Setting this bit enables hardware cursor for Controller 2.	
	0	Hardware cursor for Controller 2 is disabled.
	1	Hardware cursor for Controller 2 is enabled.
10-9	*Reserved (R/W)* These bits must be programmed to 0.	

11	*Alternate Image Window 2 Enable* This bit is effective only when bit 3 is set. Setting this bit will enable alternate Window 2. Note that power management logic can also be programmed to enable alternate Window 2.	
	0	Alternate Window 2 is disabled. In this mode Graphics Color Depth, Horizontal Window 2 Start/End, Vertical Window 2 Start/End, Window 2 Start Address, Window 2 Stride, Window 2 Line Size parameters are used.
	1	Alternate Window 2 is enabled. In this mode, Alternate Graphics Color Depth, Alternate Horizontal Window 2 Start/End, Alternate Vertical Window 2 Start/End, Alternate Window 2 Start Address, Alternate Window 2 Stride, Alternate Window 2 Line Size, and Alternate Window 2 Palette Index parameters are used. Note that color palette programming may have to be adjusted if the alternate window is running at different color depth.
14-12	*Alternate Graphics Color Depth (AGCD)* This parameter specifies the number of bits/pixel for the graphics window for Controller 2 when alternate window is enabled (bit 11 is set). Color palette is always used regardless of color depth.	
	000	1-bpp graphics.
	001	2-bpp graphics.
	010	4-bpp graphics.
	011	8-bpp graphics.
	100	15-bpp (RGB555) graphics.
	101	16-bpp (RGB565) graphics.
	others	Reserved.
15	*Reserved (R/W)* This bit must be programmed to 0.	
17-16	*G2RCLK Source* These bits select the clock source for the controller 2 root clock (G2RCLK).	
	00	G2RCLK source is bus clock.
	01	G2RCLK source is PLL1.
	10	G2RCLK source is PLL2.
	11	G2RCLK source is PLL3.
18	*Test Mode 0* Setting this bit enables factory test mode 0 for Controller 2. This bit must be programmed to 0 in normal operation.	
	0	Test mode 0 for Controller 2 is disabled.
	1	Test mode 0 for Controller 2 is enabled.
19	*Test Mode 1* Setting this bit enables factory test mode for Controller 2. This bit must be programmed to 0 in normal operation.	
	0	Test mode 1 for Controller 2 is disabled.
	1	Test mode 1 for Controller 2 is enabled.
22-20	*G2MCLK First Clock Divisor (FD1)* This parameter and bits 31-24 are used to generate controller 2 master clock (G2MCLK) from the root clock. This parameter specifies the divisor value for the first stage clock divider to generate G2MCLK. Note that the duty cycle for the output clock is not balanced except for the case where FD1 is 1.	
	000	FD1 = 1.
	001	FD1 = 1.5.
	010	FD1 = 2.5.
	011	FD1 = 3.5.
	100	FD1 = 4.5.
	101	FD1 = 5.5.
	110	FD1 = 6.5.
	111	Reserved.
23	*Reserved (R/W)* This bit must be programmed to 0.	

31-24	*G2MCLK Second Clock Divisor (SD1)* This parameter specifies the divisor value for the second stage clock divider which is used to divide generate G2MCLK. The divisor values ranges from 1 to 255. If this parameter is set to 0, G2MCLK is disabled and G2MCLK clock generation logic is powered down. G2MCLK = G2RCLK / FD1 / SD1.

5.5.28 GC21R: Graphics Controller CRC Result Register

This register contains the result of the CRC calculation on the display data output.

GC2	**GC21R - Graphics Controller CRC Control** *Index: 80 Reset value: xxxx-xx00h*	
0	*Controller CRC Enable* Setting this bit enables CRC logic. GC01R[11] must be set to 1 when CRC is enabled to get the correct CRC result.	
	0	CRC disabled.
	1	CRC enabled for graphics controller data going to the DAC.
1	*CRC Input Data Control* Setting this bit selects input data for CRC logic.	
	0	Wait for 1 VSYNC before capturing data for CRC logic.
	1	Wait for 2 VSYNC before capturing data for CRC logic.
3-2	*CRC Output Select* These bits select CRC output data to be read in bits 19-8. Bit 0 must still be set when reading the CRC result. The CRC result will be set to 0 when bit 0 is reset.	
	00	
	01	
	10	
	11	
7-4	*Reserved (R/W)* This bit must be programmed to 0.	
29-8	*CRC Result (Read Only)* When this register is read, it returns the result of the CRC logic from the datapath selected by bits 3-2.	
31-30	*Reserved (0)* These bits must be programmed to 0.	

5.5.29 GC22R: Horizontal Display Timing Control Register 2

This register set the horizontal dimensions of the total size, the active area and the blanking area for the output of graphics controller 2.

GC2	**GC22R - Horizontal Display 2 Control** *Index:88 Reset value: XXXX-XXXXH*
11-0	*Horizontal Display 2 Total (HD2T)* This parameter specifies the total length of horizontal display and blank area in terms of number of pixels. This also marks the start of the next display line with respect to the start of the current line. Constraint: HD2T ≥ HS2E + 4 where HS2E is Horizontal Sync 2 End (see GC24R). Programmed value = actual value – 2.
15-12	*Reserved (0)*
27-16	*Horizontal Display 2 End (HD2E)* This parameter specifies the horizontal size of the display area in terms of number of pixels. For flat panel, this parameter must be programmed to the horizontal panel size. Programmed value = actual value.
31-28	*Reserved (0)*

5.5.30 GC23R: Vertical Display Timing Control Register 2

This register sets the vertical dimensions of the total size, the active area and the blanking area for the output of graphics controller 2.

GC2	**GC23R - Vertical Display 2 Control** *Index: 8C Reset value: XXXX-XXXXH*
11-0	*Vertical Display 2 Total (VD2T)* This parameter specifies the total height of Vertical display and blank area in terms of number of lines. This also marks the start of the next display frame with respect to the start of the current frame. Constraint: VD2T ≥ VS2E + 1 where VS2E is Vertical Sync 2 End (see GC25R). Programmed value = actual value – 1
15-12	*Reserved (0)*
27-16	*Vertical Display 2 End (VD2E)* This parameter specifies the Vertical size of the display area in terms of number of lines. For flat panel, this parameter must be programmed to the Vertical panel size. Programmed value = actual value – 1.
31-28	*Reserved (0)*

5.5.31 GC24R: Horizontal SYNC Control Register 2

This register defines the width and starting position for the Horizontal SYNC output of Graphics Controller 2.

GC2	**GC24R - Horizontal Sync 2 Control** *Index: 90 Reset value: XXXX-XXXXH*
11-0	*Horizontal Sync 2 Start (HS2S)* This parameter specifies the start position of Horizontal Sync pulse in terms of number of pixels with respect to the left edge of the display area. Constraint: HS2S ≥ HD2E + 4. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Horizontal Sync 2 End (HS2E)* This parameter specifies the end position of Horizontal Sync pulse in terms of number of pixels with respect to the left edge of the display area. Constraint: HS2E ≥ HS2S + 4. Programmed value = Horizontal Sync 2 Start + Horizontal Sync 2 Width.
31-28	*Reserved (0)*

5.5.32 GC25R: Vertical SYNC Control Register 2

This register defines the width and start position of the Vertical SYNC output of Graphics Controller 2.

GC2	**GC25R - Vertical Sync 2 Control** *Index: 94 Reset value: XXXX-XXXXH*
11-0	*Vertical Sync 2 Start (VS2S)* This parameter specifies the start position of Vertical Sync pulse in terms of number of lines with respect to the top edge of the display area. Constraint: VS2S ≥ VD2E + 1. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Sync 2 End (VS2E)* This parameter specifies the end position of Vertical Sync pulse in terms of number of lines with respect to the top edge of the display area. Constraint: VS2E ≥ VD2S + 1. Programmed value = Vertical Sync 2 Start + Vertical Sync 2 Height.
31-28	*Reserved (0)*

5.5.33 GC26R and GC27R: Reserved

These registers are reserved for future use.

5.5.34 GC28R: Graphics Controller 2 Main Window Horizontal Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the horizontal start position and width of the main window within the display frame output from Graphics Controller 2.

GC2	**GC28R - Horizontal Window 2 Control** *Index: A0 Reset value: XXXX-XXXXH*
11-0	*Horizontal Window 2 Start (HW2S)* This parameter specifies the horizontal start position of the graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is disabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Horizontal Window 2 End (HW2E)* This parameter specifies the horizontal end position of graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is disabled. Constraint: HW2E ≥ HW2S + 4. Programmed value = Horizontal Window 2 Start + Horizontal Window 2 Width.
31-28	*Reserved (0)*

5.5.35 GC29R: Graphics Controller 2 Main Window Vertical Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the vertical start position and height of the main window within the display frame output from Graphics Controller 2.

GC2	**GC29R - Vertical Window 2 Control** *Index: A4 Reset value: XXXX-XXXXH*
11-0	*Vertical Window 2 Start (VW2S)* This parameter specifies the vertical start position of the graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is disabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Window 2 End (VW2E)* This parameter specifies the vertical end position of graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is disabled. Constraint: VW2E ≥ VW2S + 1. Programmed value = Vertical Window 2 Start + Vertical Window 2 Height.
31-28	*Reserved (0)*

5.5.36 GC2AR: Graphics Controller 2 Alternate Window Horizontal Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the horizontal start position and width of the alternate window within the display frame output from Graphics Controller 2.

GC2	**GC2AR - Alternate Horizontal Window 2 Control** *Index: A8 Reset value: XXXX-XXXXH*
11-0	*Alternate Horizontal Window 2 Start (AHW2S)* This parameter specifies the horizontal start position of the graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is enabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Alternate Horizontal Window 2 End (AHW2E)* This parameter specifies the horizontal end position of graphics image window in terms of number of pixels with respect to the left edge of the display area when alternate window is enabled. Constraint: AHW2E ≥ AHW2S + 4. Programmed value = Alternate Horizontal Window 2 Start + Alternate Horizontal Window 2 Width.
31-28	*Reserved (0)*

5.5.37 GC2BR: Graphics Controller 2 Alternate Window Vertical Control Register

Each graphics controller supports a main display window and an alternate display window. This register defines the vertical start position and height of the alternate window within the display frame output from Graphics Controller 2.

GC2	**GC2BR - Alternate Vertical Window 2 Control** *Index: AC Reset value: XXXX-XXXXH*
11-0	*Alternate Vertical Window 2 Start (AVW2S)* This parameter specifies the vertical start position of the graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is enabled. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Alternate Vertical Window 2 End (AVW2E)* This parameter specifies the vertical end position of graphics image window in terms of number of lines with respect to the top edge of the display area when alternate window is enabled. Constraint: AVW2E ≥ AVW2S + 1. Programmed value = Alternate Vertical Window 2 Start + Alternate Vertical Window 2 Height.
31-28	*Reserved (0)*

5.5.38 GC2CR: Main Window 2 Start Address

This register is programmed with the on-chip display memory byte address of the first pixel in the image to be displayed within the main window of graphics controller 2. Note that since this value is not in terms of pixels, the first pixel of the image should always be byte-aligned.

GC2	**GC2CR - Window 2 Start Address** *Index: B0 Reset value: XXXX-XXXXH*
19-0	*Window 2 Start Address (W2SA)* This parameter specifies the start address of the graphics image window in the display frame buffer when alternate window is disabled. This is a byte address. Note that if the mode is less than 8 bits/pixel then the start address is restricted to pixels on byte boundary.
31-20	*Reserved (0)*

5.5.39 GC2DR: Alternate Window 2 Start Address

This register is programmed with the on-chip display memory byte address of the first pixel in the image to be displayed within the alternate window of graphics controller 2. Note that since this value is not in terms of pixels, the first pixel of the image should always be byte-aligned.

This register also specifies the Palette Index for the alternate window. Since the Color Look-Up Table (Palette) contains 256 entries for each color, pixel sizes less than 8 bits/pixel do not fully specify a lookup table entry. The Palette Index field is used as the upper bits and the pixel value as the lower bits to form a full 8-bit lookup table address. The Palette Index for the Main window is always 0; providing a programmable Palette Index for the alternate window allows different color maps to be used for the main and alternate windows.

GC2	**GC2DR - Alternate Window 2 Start Address** *Index: B4 Reset value: XXXX-XXXXH*
19-0	*Alternate Window 2 Start Address (AW2SA)* This parameter specifies the start address of the graphics image window in the display frame buffer when alternate window is enabled. This is a byte address. Note that if the mode is less than 8 bits/pixel then the start address is restricted to pixels on byte boundary.
24-20	*Reserved (0)*
31-25	*Alternate Window 2 Palette Index (AW2PI)* This parameter specifies the upper bits of palette index when alternate window is enabled in 1-bpp, 2-bpp, 4-bpp alternate graphics color depth. If alternate graphics color depth is 1-bpp then bits 31-25 are used for palette index bits 7-1. If alternate graphics color depth is 2-bpp then bits 31-26 are used for palette index bits 7-2. If alternate graphics color depth is 4-bpp then bits 31-28 are used for palette index bits 7-4.

5.5.40 GC2ER: Window 2 Stride Register

GC2ER sets the stride for the main and alternate windows of graphics controller 2. The “stride” is the distance in bytes between consecutive lines of an image in the display memory. Lines may be contiguous (i.e., the first pixel of the next line immediately follows the last pixel of the current line) but may also not be contiguous (as when a smaller window of a larger image is being displayed). In either case, the stride should be a multiple of 16 bytes.

GC2	**GC2ER - Window 2 Stride** *Index: B8 Reset value: XXXX-XXXXH*
15-0	*Window 2 Stride (AW2ST)* This parameter specifies the distance from the start of current line to the start of next line when alternate window is disabled. This is specified in terms of number of bytes. It is recommended that this parameter be specified in terms of frame buffer bus width (128 bits).
31-16	*Alternate Window 2 Stride (AW2ST)* This parameter specifies the distance from the start of current line to the start of next line when alternate window is enabled. This is specified in terms of number of bytes. It is recommended that this parameter be specified in terms of frame buffer bus width (128 bits).

5.5.41 GC2FR: Window 2 Line Size Register

GC2FR sets the line size in bytes for the main and alternate windows of graphics controller. This is the number of bytes that will be read from the display memory before starting to fetch the next line.

GC2	**GC2FR - Window 2 Line Size** *Index: BC Reset value: XXXX-XXXXH*
13-0	*Window 2 Line Size (W2LS)* This parameter specifies the amount of data to be fetched from display frame buffer for each graphics image line when alternate window is disabled. This is specified in terms of number of bytes and subtracted by 1. Programmed value =(number of pixels per line * bits per pixel / 8-1) rounded up to nearest integer.
15-14	*Reserved (0)*
29-16	*Alternate Window 2 Line Size (AW2LS)* This parameter specifies the amount of data to be fetched from display frame buffer for each graphics image line when alternate window is enabled. This is specified in terms of number of bytes and subtracted by 1. Programmed value=(number of pixels per line * bits per pixel / 8-1) rounded up to nearest integer.
31-30	*Reserved (0)*

5.5.42 GC30R: Hardware Cursor 2 Position

GC30R sets the horizontal and vertical positions of the hardware cursor in graphics controller 2. To avoid flicker when a new position is written, the cursor will not be moved during the active display time (the update will take effect at the start of the next Vertical SYNC).

GC2	**GC30R - Hardware Cursor 2 Position** *Index: C0 Reset value: XXXX-XXXXH* Both Horizontal and Vertical Cursor 2 Start must be changed at the same time. This register takes effect on the next falling edge of Controller 2 Vertical Sync.
11-0	*Horizontal Cursor 2 Start (HC2S)* This parameter specifies the horizontal start position of the hardware cursor, in terms of number of pixels with respect to the left edge of the display area. Programmed value = actual value.
15-12	*Reserved (0)*
27-16	*Vertical Cursor 2 Start (VC2S)* This parameter specifies the vertical start position of the hardware cursor, in terms of number of lines with respect to the top edge of the display area. Programmed value = actual value.
31-28	*Reserved (0)*

5.5.43 GC31R: Hardware Cursor 2 Start Address and Offset Register

GC31R sets the display memory byte address for the 1024-byte cursor image. It also sets the horizontal and vertical pixel offsets used for clipping the cursor on the left and top edges of the display window.

GC2	**GC31R - Hardware Cursor 2 Start Address and Offset** *Index: C4 Reset value: XXXX-XXXXH* Both Horizontal and Vertical Cursor 2 Offset must be changed at the same time and they take effect on the next falling edge of Controller 2 Vertical Sync.
9-0	*Hardware Cursor 2 Start Address (HC2SA)* This parameter specifies bits 19-10 of the start address of the hardware cursor in the display frame buffer. Note that the start address is restricted to 1K byte boundary, therefore the lower 10 bits of this parameter is forced to 0.
15-10	*Reserved (0)*
21-16	*Horizontal Cursor 2 Offset (HC2O)* This parameter specifies the horizontal offset of the hardware cursor, in terms of number of pixels with respect to the left edge of the cursor area. Hardware cursor horizontal offset, specifies the first horizontal cursor pixel to be displayed. This parameter is used for left-edge clipping of the hardware cursor.
23-22	*Reserved (0)*
29-24	*Vertical Cursor 2 Offset (VC2O)* This parameter specifies the vertical offset of the hardware cursor, in terms of number of lines with respect to the top edge of the cursor area. Hardware cursor vertical offset specifies the first vertical cursor line to be displayed. This parameter is used for top-edge clipping of the hardware cursor.
31-30	*Reserved (0)*

5.5.44 GC32R: Hardware Cursor 2 Foreground Color Register

GC32R sets the foreground color that will be used when the hardware cursor pixel is displayed on top of the background image. The color value is specified in 24-bit RGB format.

GC2	**GC32R - Hardware Cursor 2 Foreground Color** *Index: C8 Reset value: XXXX-XXXXH*
23-0	*Hardware Cursor 2 Foreground Color (HC2FC)* This parameter specifies the hardware cursor foreground color in the RGB format. Bits 23-16 store the red data, bits 15-8 stores the green data, and bits 7-0 stores the blue data. In case of monochrome mode, only the green data is used.
31-24	*Reserved (0)*

5.5.45 GC33R: Hardware Cursor 2 Background Color Register

GC33R sets the background color that will be used when the hardware cursor pixel is displayed on top of the background image. The color value is specified in 24-bit RGB format.

GC2	**GC33R - Hardware Cursor 2 Background Color** *Index: CC Reset value: XXXX-XXXXH*
23-0	*Hardware Cursor 2 Background Color (HC2BC)* This parameter specifies the hardware cursor foreground color in the RGB format. Bits 23-16 store the red data, bits 15-8 stores the green data, and bits 7-0 stores the blue data. In case of monochrome mode, only the green data is used.
31-24	*Reserved (0)*

5.5.46 GC34R Through GC3FR: Reserved

These registers are reserved for future use.

5.5.47 C100R Through C1FFR: Color Palette 1

These 256 registers specify the color lookup table for graphics controller 1. In the Main Window 1, the index to the color look-up table is generated as follows:

8-bpp mode: All 8 bits are used for indexing to look up the color value.

4-bpp mode: 4 Most significant bits are 0, and the least significant bits have the 4-bit pixel value

2-bpp mode: 6 Most significant bits are 0, and the least significant bits have the 2-bit pixel value

1-bpp mode: 7 Most significant bits are 0, and the least significant bits have the 1-bit pixel value

When displayed in Alternate Window 1, the Alternate Palette Index is used in addition to the pixel value. Depending on the color depth of the video mode, the new palette index is is calculated by combining the most significant 7 bits of GC0DR and the pixel values are as below:

8-bpp mode: All 8 bits are used for indexing to look up the color value

4-bpp mode: 4 Most significant bits are GC0DR[31:28], and the least significant bits have the 4-bit pixel value

2-bpp mode: 6 Most significant bits are GC0DR[31:27], and the least significant bits have the 4-bit pixel value

1-bpp mode: 7 Most significant bits are GC0DR[31:26], and the least significant bits have the 4-bit pixel value

Color Palette entries are in 24-bit RGB format

C1	**C100R to C1FFR - Graphics Controller 1 Color Palette** *Index: 400-7FC Reset value: XXXX-XXXXH* **This register is**
7-0	*Blue Color Palette* This parameter specifies the blue color values.
15-8	*Green / Gray Color Palette* This parameter specifies the green color values in color modes or the gray level values in mono-chrome modes.
23-16	*Red Color Palette* This parameter specifies the red color values.
31-24	*Reserved (0)*

5.5.48 C200R Through C2FFR: Color Palette 2

These 256 registers specify the color lookup table for graphics controller 1. In the Main Window 1, the index to the color look-up table is generated as follows:

8-bpp mode: All 8 bits are used for indexing to look up the color value.

4-bpp mode: 4 Most significant bits are 0, and the least significant bits have the 4-bit pixel value

2-bpp mode: 6 Most significant bits are 0, and the least significant bits have the 2-bit pixel value

1-bpp mode: 7 Most significant bits are 0, and the least significant bits have the 1-bit pixel value

When displayed in Alternate Window 1, the Alternate Palette Index is used in addition to the pixel value. Depending on the color depth of the video mode, the new palette index is is calculated by combining the most significant 7 bits of GC2DR and the pixel values are as below:

8-bpp mode: All 8 bits are used for indexing to look up the color value

4-bpp mode: 4 Most significant bits are GC2DR[31:28], and the least significant bits have the 4-bit pixel value

2-bpp mode: 6 Most significant bits are GC2DR[31:27], and the least significant bits have the 4-bit pixel value

1-bpp mode: 7 Most significant bits are GC2DR[31:26], and the least significant bits have the 4-bit pixel value

C2	**C200R to C2FFR - Graphics Controller 2 Color Palette** **Index: 800-BFC Reset value: XXXX-XXXXH**
7-0	*Blue Color Palette* This parameter specifies the blue color values.
15-8	*Green / Gray Color Palette* This parameter specifies the green color values in color modes or the gray level values in mono-chrome modes.
23-16	*Red Color Palette* This parameter specifies the red color values.
31-24	*Reserved (0)*

5.6 Flat Panel Interface Programming Information

The Flat Panel Interface requires very little configuration. Once the panel characteristics (such as type and data word width) are set, the interface operates without further programming. The dithering and frame-rate control logic is programmed by the MediaQ-supplied device drivers using proprietary algorithms.

5.6.1 FP00R: Flat Panel Control Register

FP	FP00R – Flat Panel Control *Index: 0 Reset value: 0000-0000h*	
1-0	*Flat Panel Interface Enable* Setting this bit will enable flat panel interface logic. If this bit is reset, flat panel interface logic is powered down and reset.	
	X0	Flat panel interface is disabled. This will trigger flat panel power down sequencing.
	01	Flat panel interface is enabled and driven by Controller 1. Controller 1 must also be enabled. This will trigger flat panel power up sequencing.
	11	Flat panel interface is enabled and driven by Controller 2. Controller 2 must also be enabled. This will trigger flat panel power up sequencing.
3-2	*Flat Panel Type* These bits indicate type of flat panel.	
	00	TFT panel.
	01	S-STN panel (single-panel single-drive).
	1x	D-STN panel (dual-panel dual-drive).
4	*Mono/Color Panel Select* This bit selects mono/color panel.	
	0	Color panel.
	1	Mono panel. NTSC color conversion is applied to convert RGB color data to mono-chrome data.
7-5	*Flat Panel Interface* This parameter specifies the number of bits/pixel for the graphics window for Controller 1. Color palette is always used regardless of color depth.	
	000	TFT: 4-bit mono or 12-bit color interface. S-STN: 4-bit mono or 4-bit color interface. D-STN: 8-bit mono or 8-bit color interface.
	001	TFT: 6-bit mono or 18-bit color interface. S-STN: 8-bit mono or 8-bit color interface. D-STN: 16-bit mono or 16-bit color interface.
	010	TFT: 8-bit mono or 24-bit color. S-STN: 12-bit color interface. D-STN: 24-bit color interface.
	011	S-STN: 16-bit mono or 16-bit color interface.
	100	S-STN 24-bit color interface
	Others	Reserved
9-8	*Dither Pattern* These bits select dither pattern.	
	00	Dither is disabled.
	01	Dither pattern 1.
	10	Dither pattern 2.
	11	Dither pattern 3.
11-10	*Dither Pattern Adjust 1* These bits may modify the selected dither pattern. The modification depends on the selected dither pattern. Some combination may not have any impact. The default value of '00' is expected to produce the best result.	
14-12	*Dither Base Color* These bits select the number of bits to be dithered.	
	000	8 bits (dithering is disabled).

	001	Reserved.
	010	2 bits.
	011	3 bits. This setting should be used with 8-level FRC for STN panels.
	100	4 bits. This setting should be used with 16-level FRC for STN panels.
	101	Reserved.
	110	6 bits.
	111	Reserved.
15	*Alternate Window Control* Setting this bit controls dithering and FRC in Alternate Window mode.	
	0	Dither and FRC can be enabled when Alternate Window is enabled.
	1	Disable dither and FRC when Alternate Window is enabled.
17-16	*FRC Control* These bits control grayscaling for STN panels. These bits have no effect on TFT panels.	
	00	2-level FRC. This setting essentially disables the FRC. The FRC logic will take the most significant bits of input color data as FRC output data.
	01	4-level FRC.
	10	8-level FRC.
	11	16-level FRC.
19-18	*Reserved (R/W)* This bit must be programmed to 0.	
22-20	*Dither Pattern Adjust 2* These bits may modify the selected dither pattern. The modification depends on the selected dither pattern. Some combination may not have any impact. For TFT panels, the default value of '000' is expected to produce the best result. For STN panels, programming this parameter to the value of FP00R[17:16] + 1 is expected to produce the best result.	
27-23	*Reserved (R/W)* These bits must be programmed to 0.	
28	*Test Mode 0* Setting this bit enables factory test mode for FPI module. This bit must be programmed to 0 in normal operation.	
	0	Test mode for FPI is disabled.
	1	Test mode for FPI is enabled.
29	*Test Mode 1* Setting this bit enables factory test mode for FPI module. This bit must be programmed to 0 in normal operation.	
	0	Test mode for FPI is disabled.
	1	Test mode for FPI is enabled.
30	*Test Mode 2* Setting this bit enables factory test mode for FPI module. This bit must be programmed to 0 in normal operation.	
	0	Test mode for FPI is disabled.
	1	Test mode for FPI is enabled.
31	*Test Mode 3* Setting this bit enables factory test mode for FPI module. This bit must be programmed to 0 in normal operation.	
	0	Test mode for FPI is disabled.
	1	Test mode for FPI is enabled.

5.6.2 FP01R: Flat Panel Output Pin Control Register

FP	FP01R – Flat Panel Pin Control *Index: 4 Reset value: 0000-0000h*	
0	*Disable Flat Panel pins* Setting this bit disables all flat panel data and control (FD[23:0], FDE, FDI, FHSYNC, FVSYNC, and FSCLK) pins.	
	0	Flat panel data and control signals pins are enabled.
	1	Flat panel data and control signals pins are driven low.
1	*Flat Panel Data Inversion Enable (TFT panel only)* This bit enables FDI pin. This bit should be set only for 18-bit or 24-bit TFT panels. This bit should be set for color TFT panel only and should be reset for mono TFT panel. Setting this bit will ensure that not more than half of the output data lines will toggle at any time during active data output. The number of lines that can toggle depends on the smaller of TFT interface width (FP00R[7:5]) and dither base color bits (FP00R[14:12]).	
	0	Flat panel data inversion is disabled and FDI pin is driven low.
	1	Flat panel data inversion is enabled and output on FDI pin.
2	*Reserved (R/W)* This bit must be programmed to 0.	
3	*Flat panel AC Modulation Enable* This bit enables FMOD signal to be output on FDE pin. FMOD signal is normally used only on some STN panels but this bit is effective regardless of panel type (TFT/STN).	
	0	FMOD is disabled and FDE pin outputs flat panel display enable signal.
	1	FMOD is enabled and output on FDE pin.
4	*Output Shift Clock on FD0 pin (STN panel only)* Setting this bit will output shift clock on FD0 pin and pixel clock on FSCLK pin. Setting this bit is effective only for STN panels.	
	0	FD0 pin will be used to output data and FSCLK will output shift clock.
	1	FD0 pin will be used to output shift clock and FSCLK will output pixel clock.
5	*FSCLK Output Enable.* This bit enables FSCLK pin for most panels. For some STN panels that use FD0 for shift clock, FSCLK pin may be disabled to save power by resetting this bit.	
	0	FSCLK pin is forced low regardless of its polarity setting.
	1	FSCLK pin outputs either shift clock or pixel clock depending on panel type.
6	*TFT Shift Clock Select* This bit is effective for TFT panels only. Setting this bit will essentially divide shift clock by 2 for panels that require data on both falling and rising edges.	
	0	Output data can be latched externally on falling edge of shift clock (FSCLK).
	1	Output data can be latched externally on both rising and falling edges of shift clock (FSCLK).
7	*Shift Clock Mask* This bit controls the shift clock output (FSCLK) when valid display data is not being output (such as during horizontal and vertical blanking).	
	0	Allow shift clock to run during non-display area.
	1	Force shift clock low during non-display area.
8	*STN LP Control* This bit is effective for STN panel only. This bit controls STN LP (FHSYNC pin) and FSCLK output during non-display area.	
	0	STN LP (FHSYNC pin) is enabled during vertical blank. Shift clock (on FSCLK or FD0 pin) will also be enabled during vertical blank time. Bit 9 is only effective for this setting.
	1	STN LP (FHSYNC pin) is disabled during vertical blank time. Shift clock (on FSCLK or FD0 pin) will also be disabled during vertical blank time.

9	*STN Shift Clock Control (STN panel only)* This bit is effective for STN panel only when bit 8 is reset to 0. This bit controls shift clock output during first line of non-display area. Note that shift clock can be output on either FSCLK pin or FD0 pin.	
	0	Shift clock is enabled during the first line of vertical blank.
	1	Shift clock is disabled during the first line of vertical blank.
10	*STN Extra LP Enable (STN Panel Only)* This bit is effective for STN panels and only when bit 8 is set to 1. When this bit is enabled, extra LP is generated at the end of the first line of vertical blanking time.	
11	*Reserved (R/W)* These bits must be programmed to 0.	
12	*Flat Panel Shift Clock Drive Strength* This bit controls drive strength of FSCLK pin.	
	0	8 mA.
	1	16 mA.
13	*FD0 Drive Strength* This bit controls drive strength of FD0 pin. This is useful when FD0 pin is used to output shift clock.	
	0	8 mA.
	1	16 mA.
14	*Flat Panel Data Drive Strength* This bit controls drive strength of FD[23:1], FDE, and FDI pins.	
	0	8 mA.
	1	16 mA.
15	*Reserved (R/W)* This bit must be programmed to 0.	
16	*Flat Panel Data bit 0 Polarity* This bit controls polarity of flat panel data bit 0 (FD0) pin. This is effective regardless of whether FD0 is used to output data bit 0 or shift clock.	
	0	FD0 output is active high. If FD0 is used to output shift clock then flat panel data output is on rising edge of FD0 and should be latched externally with falling edge of FD0.
	1	FD0 output is active low. If FD0 is used to output shift clock then flat panel data output is on falling edge of FD0 and should be latched externally with rising edge of FD0.
17	*Flat Panel Data Inversion* This bit controls polarity of most flat panel data pins (FD[23:1]). This bit is effective only when the flat panel is enabled. FD0 polarity is controlled by bit 16.	
	0	Flat Panel Data output is active high.
	1	Flat Panel Data output is active low.
18	*Flat Panel Data Enable (FDE) Polarity* This bit controls polarity of FDE pin. This bit is effective only when the flat panel is enabled.	
	0	FDE output is active high.
	1	FDE output is active low.
19	*Flat Panel Horizontal Sync (FHSYNC) Polarity* This bit controls polarity of FHSYNC pin. This bit is effective only when the flat panel is enabled.	
	0	FHSYNC output is active high.
	1	FHSYNC output is active low.
20	*Flat Panel Vertical Sync (FVSYNC) Polarity* This bit controls polarity of FVSYNC pin. This bit is effective only when the flat panel is enabled.	
	0	FVSYNC output is active high.
	1	FVSYNC output is active low.
21	*Flat Panel Shift Clock (FSCLK) Polarity* This bit controls the polarity of FSCLK output pin; it is effective only when the flat panel is enabled. This bit is effective regardless of whether FSCLK pin is used to output shift clock or pixel clock. This bit has no effect on shift clock output on FD0 pin.	

	0	FSCLK output is active high. Data is output on rising edge of FSCLK and should be latched externally with falling edge of FSCLK.
	1	FSCLK output is active low. Data is output on falling edge of FSCLK and should be latched externally with rising edge of FSCLK.
23-22	*Reserved (R/W)* These bits must be programmed to 0.	
26-24	*Flat Panel Shift Clock (FSCLK) Delay* This parameter provides programmable delay on FSCLK output pin ranging from 0 to 7 ns, typical. This bit has no effect on shift clock output on FD0 pin.	
31-27	*Reserved (0)* These bits do not exist (not implemented).	

5.6.3 FP02R: General Purpose Output Port Control Register

FP	**FP02R – Flat Panel General Purpose Output Control** *Index:8 Reset value: xx00-0000h*	
1-0	*GPO0 Enable* This bit controls ENCTL pin.	
	00	ENCTL pin is used for Enable CTL output.
	01	ENCTL pin is used as general-purpose output 0 (GPO0).
	1x	Reserved.
3-2	*GPO1 Enable* This bit controls ENVEE pin.	
	00	ENVEE pin is used for Enable VEE output.
	01	ENVEE pin is used as general-purpose output 1 (GPO1).
	1x	Reserved.
5-4	*GPO2 Enable* This bit controls PWM0 pin.	
	00	PWM0 pin is used for PWM0 output.
	01	PWM0 pin is used as general-purpose output 2 (GPO2).
	1x	Reserved.
7-6	*GPO3 Enable* This bit controls PWM1 pin.	
	00	PWM1 pin is used for PWM1 output.
	01	PWM1 pin is used as general-purpose output 3 (GPO3).
	1x	Reserved.
15-8	*Reserved (R/W)* These bits must be programmed to 0.	
16	*GPO0 Data* This bit is effective when GPO0 Enable bit is set.	
	0	ENCTL pin is driven low when GPO0 Enable bit is set.
	1	ENCTL pin is driven high when GPO0 Enable bit is set.
17	*GPO1 Data* This bit is effective when GPO1 Enable bit is set.	
	0	ENVEE pin is driven low when GPO1 Enable bit is set.
	1	ENVEE pin is driven high when GPO1 Enable bit is set.
18	*GPO2 Data* This bit is effective when GPO2 Enable bit is set.	
	0	PWM0 pin is driven low when GPO2 Enable bit is set.
	1	PWM0 pin is driven high when GPO2 Enable bit is set.
19	*GPO3 Data* This bit is effective when GPO3 Enable bit is set.	

	0	PWM1 pin is driven low when GPO3 Enable bit is set.
	1	PWM1 pin is driven high when GPO3 Enable bit is set.
23-20	*Reserved (R/W)* These bits must be programmed to 0.	
31-24	*Reserved (0)* These bits do not exist (not implemented).	

5.6.4 FP03R: General Purpose I/O Port Control Register

FP	**FP03R – Flat Panel General Purpose Input Output Control** *Index:C Reset value: xx00-0000h*	
1-0	*GPIO0 Enable* This bit controls GPIO0 pin.	
	00	GPIO0 pin is used as general-purpose input.
	01	GPIO0 pin is used as general-purpose output.
	10	GPIO0 pin is used to output PLL 1 clock.
	11	Reserved.
3-2	*GPIO1 Enable* This bit controls GPIO1 pin.	
	00	GPIO1 pin is used as general-purpose input.
	01	GPIO1 pin is used as general-purpose output.
	10	GPIO1 pin is used to output PLL 2 clock.
	11	Reserved.
5-4	*GPIO2 Enable* This bit controls GPIO2 pin.	
	00	GPIO2 pin is used as general-purpose input.
	01	GPIO2 pin is used as general-purpose output.
	10	GPIO2 pin is used to output PLL 3 clock.
	11	Reserved.
15-6	*Reserved (R/W)* These bits must be programmed to 0.	
16	*GPIO0 Output Data* This bit is effective when GPIO0 is programmed as general-purpose output.	
	0	GPIO0 pin is driven low.
	1	GPIO0 pin is driven high.
17	*GPIO1 Output Data* This bit is effective when GPIO1 is programmed as general-purpose output.	
	0	GPIO1 pin is driven low.
	1	GPIO1 pin is driven high.
18	*GPIO2 Output Data* This bit is effective when GPIO2 is programmed as general-purpose output.	
	0	GPIO2 pin is driven low.
	1	GPIO2 pin is driven high.
23-19	*Reserved (R/W)* These bits must be programmed to 0.	
24	*GPIO0 Input Data* This bit is effective when GPIO0 is programmed as general-purpose input.	
	0	GPIO0 pin is driven low.
	1	GPIO0 pin is driven high.
25	*GPIO1 Input Data* This bit is effective when GPIO1 is programmed as general-purpose input.	
	0	GPIO1 pin is driven low.

	1	GPIO1 pin is driven high.
26	*GPIO2 Input Data* This bit is effective when GPIO2 is programmed as general-purpose input.	
	0	GPIO2 pin is driven low.
	1	GPIO2 pin is driven high.
31-27	*Reserved (0)* These bits do not exist (not implemented).	

5.6.5 FP04R: STN Panel Control Register

FP	FP04R - STN Panel Control *Index: 10 Reset value: 0000-0000h*
7-0	*FRC Tuning Control bits 7-0* This parameter specifies tuning value for FRC algorithm.
15-8	*FRC Tuning Control bits 15-8* This parameter specifies tuning value for FRC algorithm.
23-16	*Reserved (R/W)* This parameter specifies tuning value for FRC algorithm.
30-24	*Flat Panel Modulation Period* This parameter specifies the period of the FMOD signal for STN panels. FMOD signal is generated either using line clock or frame clock depending on setting of bit 31. Note: Programmed value = actual value – 1.
31	*Flat Panel Modulation Clock Select* This bit controls the clock used to generate FMOD signal.
0	FMOD is generated using frame clock.
1	FMOD is generated using line clock.

5.6.6 FP05R: Dual-STN Half-Frame Buffer Control Register

FP	FP05R - D-STN Frame Buffer Control *Index: 14 Reset value: XXXX-XXXXH*
12-0	*D-STN Frame Buffer Start Address bits 19-7* This parameter specifies start address for D-STN frame buffer. This address will be incremented for subsequent writes to the D-STN frame buffer. The address is a byte address but the least significant 7 bits of this parameter (address bits 6-0) are forced to 0 so that the address is aligned on 128-byte boundary.
15-13	*Reserved (R/W)* These bits must be programmed to 0.
31-16	*D-STN Frame Buffer Size bits 19-4* This parameter specifies end address for D-STN frame buffer that is calculated by adding the D-STN frame buffer size to the start address and then subtracting by 1. The frame buffer size depends on panel size (display size) at 3-bits/pixel for color D-STN and 1 bit/pixel for mono D-STN panel. This is byte address but the least significant 4 bits of this parameter (address bits 3-0) are forced to 0 so that this parameter is practically in terms of 128-bit unit. Programmed value = D-STN Frame Buffer Start Address + D-STN Frame Buffer Size – 1. D-STN Frame Buffer size is calculated as: (vertical display size * line size * 8). Where: line size = (horizontal display size * bits-per-pixel / 128) rounded up.

5.6.7 FP0FR: Pulse Width Modulation Control Register

Register FP0FR configures the individual source clock and pre-divide values for each PWM output, and whether the output is part of the general flat panel power sequencing cycle. Once a PWM output has been configured, the duty cycle will be varied by changing the 8-bit duty cycle field in this register.

FP	**FP0FR - Pulse Width Modulation (PWM) Control** *Index: 3C Reset value: 0000-0000h*	
1-0	*PWM 0 Source Clock* These bits control the source of PWM 0 clock.	
	00	PWM 0 signal is generated using oscillator clock.
	01	PWM 0 signal is generated using bus clock.
	10	PWM 0 signal is generated using power management clock.
	11	Reserved.
2	*PWM 0 Sequencing* This bit controls PWM 0 sequencing when PWM 0 is enabled.	
	0	PWM 0 sequencing is tied to flat panel power sequencing. PWM 0 signal will be generated one PMCLK cycle before flat panel data/control signals are enabled, and it will be deactivated one PMCLK cycle after flat panel data/control signals are disabled.
	1	PWM 0 generation is *not* tied to flat panel power sequencing. PWM 0 signal will always be generated when it is enabled.
3	*Reserved (R/W)* This bit must be programmed to 0.	
7-4	*PWM 0 Clock Pre-Divider* This parameter specifies the divisor value for the PWM 0 clock. The divisor value ranges from 1 to 15. If this parameter is set to 0, PWM 0 source clock is disabled and therefore PWM 0 generation logic is powered down.	
15-8	*PWM 0 Duty Cycle* This parameter specifies the number of clocks high time for PWM 0 pulse. Note that the period of PWM 0 signal is always 256 clocks. If this parameter is programmed to 0, then the PWM 0 will be static low signal and if this parameter is programmed to FFh then the PWM 0 will be static high signal. If enabled, the PWM 0 signal starts one PDCLK cycle before the ENCTL pin is activated (high) and ends one PDCLK cycle after the ENCTL pin is deactivated (low). During period where PWM 0 is inactive, the PWM 0 signal will also be driven low.	
17-16	*PWM 1 Source Clock* These bits control the source of PWM 1 clock.	
	00	PWM 1 signal is generated using oscillator clock.
	01	PWM 1 signal is generated using bus clock.
	10	PWM 1 signal is generated using power management clock.
	11	Reserved.
18	*PWM 1 Sequencing* This bit controls PWM 1 sequencing when PWM 1 is enabled.	
	0	PWM 1 sequencing is tied to flat panel power sequencing. PWM 1 signal will be generated one PMCLK cycle before flat panel data/control signals are enabled, and it will be deactivated one PMCLK cycle after flat panel data/control signals are disabled.
	1	PWM 1 generation is *not* tied to flat panel power sequencing. PWM 1 signal will always be generated when it is enabled.
19	*Reserved (R/W)* This bit must be programmed to 0.	
23-20	*PWM 1 Clock Pre-Divider* This parameter specifies the divisor value for the PWM 1 clock. The divisor value ranges from 1 to 15. If this parameter is set to 0, PWM 1 source clock is disabled and therefore PWM 1 generation logic is powered down.	

31-24	*PWM 1 Duty Cycle* This parameter specifies the number of clock high time for PWM 1 pulse. Note that the period of PWM 1 signal is always 256 clocks. If this parameter is programmed to 0 then the PWM 1 will be static low signal and if this parameter is programmed to FFh then the PWM 1 will be static high signal. If enabled, the PWM 1 signal starts one PGCLK cycle before the ENCTL pin is activated (high) and ends one PGCLK cycle after the ENCTL pin is deactivated (low). During period where PWM 1 is inactive, the PWM 1 signal will also be driven low.

5.6.8 FP10R to FP2FR: Frame-Rate Control Pattern Registers

FP	**FP10R to FP2FR - FRC Pattern** *Index: 40-BC Reset value: XXXX-XXXXH*
31-0	*FRC Pattern* These registers specify FRC patterns. FRC utility will be provided to change this pattern.

5.6.9 FP30R to FP37R: Frame-Rate Control Weight Registers

These registers should not be written to. They are used by the MQ-100/HD64464 device drivers.

FP	**FP30R to FP37R - FRC Weight** *Index: C0-DC Reset value: XXXX-XXXXH*
31-0	*FRC Pattern* These registers specify FRC weights. FRC utility will be provided to change this data.

5.7 Graphics Engine Programming Information

The MQ-100/HD64464 2D Graphics Engine supports various bit-block-level transfer (BitBLT) operations, with a choice of 256 ROPs. Engine operation can be triggered by writing either the Drawing Command Register or the Destination X/Y Register. The hardware-assisted clipping rectangle can be used for global clipping of BitBLT, line and text output operation. Arbitrary line drawing is not currently supported in 2D engine, but it does support horizontal, vertical and 45-degree angle line drawing. Note that the DrvTextOut entry point to draw text output is not supported in Window CE 2.0 display driver architecture. Instead, BitBLTs are used for general text output.

Following are some example GE programs. It is not expected that users of MQ-100/HD64464 will program the GE directly. These examples are provided for reference only. The GE register set is described in detail in the next section.

5.7.1 Rectangle Fill (Solid Fill)

This program segment draws a rectangle filled with a solid color.

```
Destination coordinate = DestX, DestY
Width W and height H
SolidColor

WaitCmdFIFO(4);               // Need 4 command FIFO entries
GE01[11-0] = W;
GE01[27-16] = H;              // BLT width and height
GE02[11-0] = DestX;
GE02[27-16] = DestY;          // Destination X and Y
GE08[15-0]=SolidColor;        // Solid color in Foreground Color Register
GE00[7-0]=rop;        // Set up rop code
GE00[10-8]=Rectangle Fill;    // This should trigger engine operation
```

5.7.2 Regular BitBLT – Screen to Screen

This BitBLT operation is used to move a rectangle block of pixels from one area on the screen to another location. When the source and destination overlay, special handling is required. The program segments below show how each case is handled.

5.7.2.1Overlapping Rectangles

There are four overlapping cases to consider while programming the X and Y direction of a pixel transfer,

Case 1 :

SrcX >= DestX and SrcY >= DestY
Pixels are moved from left to right and top to bottom
X DIR=0 and Y DIR=0
Suitable for non-overlapping BLT case

Case 2 :

SrcX < DestX and SrcY < DestY
Pixels are moved from right to left and bottom to top
X DIR=1 and Y DIR=1

Case 3 :

SrcX >= DestX and SrcY < DestY
Pixels are moved from left to right and bottom to top
X DIR=0 and Y DIR=1

Case 4 :

SrcX < DestX and SrcY >= DestY
Pixels are moved from right to left and top to bottom
X DIR=1 and Y DIR=0

Example:

```
Source coordinate = SrcX, SrcY
Destination coordinate = DestX, DestY
Width W and height H

WaitCmdFIFO(4);               // Wait for 4 command FIFO entries
GE03[11- 0] = SrcX;
GE03[27-16] = SrcY;           // Source X and Y
GE01[11- 0] = W;
GE01[27-16] = H;              // Width and height for the BLT
GE02[11- 0] = DestX;
GE02[27-16] = DestY;          // Destination X and Y
GE00[ 7- 0] = rop;            // Set up rop code
GE00[   11] = XDIR;
GE00[   12] = YDIR;
GE00[   13] = 0;              // Source is in screen
GE00[10- 8] = BitBLT;         // This should trigger engine operation
```

5.7.3 Regular BitBLT – Memory to Screen

This BitBLT operation is used to transfer a bitmap image from system memory to the screen. The source image can be a color or monochrome bitmap. (See Monochrome-to-Color Expansion BitBLT for details on the monochrome source bitmap case).

Example: Color source bitmap

```
Source data pointer = pScrData
Destination coordinate = DestX, DestY
Width W and height H

WaitCmdFIFO(4);                     // Wait for enough entries
GE01[11- 0]    = W;
GE01[27-16]    = H;                 // Width and height for the BLT
GE02[11- 0]    = DestX;
GE02[27-16]    = DestY;             // Destination X and Y
GE07[ 2- 0]    = PhaseofSource      // Byte Offset. See below
GE00[ 7- 0]    = rop;               // Set up rop code
GE00[   13]    = 1;                 // Source is in system memory
GE00[   14]    = 0;                 // Color source
GE00[10- 8]    = BitBLT;            // This should trigger engine operation

// Start pumping enough DWORD count of data to Source Image Data Register, GE20
To calculate the correct number of DWORDs for the Source Image Data Register
```

per scanline:

```
# of DWORD per line = (Width * BytesPerPixel + 3 + PhaseOfSource) mod 4

Where
Width          = width in pixels of each scanline
BytesPerPixel  = ½ (4bpp), 1 (8bpp) and 2 (16bpp)
PhaseOfSource  = 0, 1, 2 or 3 (Byte Offset within a DWORD)
```

If the number of DWORDs per scanline is an odd number, the driver has to pump one more DWORD so that the total data block is 64-bit aligned. (For monochrome source data, a different calculation is used. See Monochrome-to-Color Expansion BitBLT for this case.)

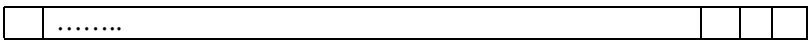
The driver must to ensure that enough Source Image Data FIFO entries are free before writing the pixel data by polling the Source/Command FIFO Status Register to determine number of available FIFO entries. The size of the Source Image Data FIFO is 16x64, i.e. 16 deep by 64 bits. The following table summarizes the FIFO entry and pixel data relationship,

Color Depth	Empty FIFO (16x64)	Pixels Count
8BPP	128 bytes	128 pixels
16BPP	128 bytes	64 pixels

The driver writes first non double-word aligned data to Source Image Data FIFO if the source bitmap in the system memory is not dword-aligned, followed by integral number of dwords, and then the last dword that might not contain full 4-byte data. GE07[2-0] is set up to indicate leading PhaseOfSource (or Byte Offset). For 32-bit system memory interface, only bits 1-0 are used. For 64-bit system memory interface bits 2-0 are used.

The following is an example to write a scanline of 8BPP source bitmap data,

3 4 5 …….. 47 48 49 50



To send bitmap pixel data to Source Image Data Register,

1. send first DWORD (with GE07[1-0] set to 3, 11 in binary), then
2. send 11 complete DWORDs, and finally,
3. send last DWORD (2D engine knows how many bytes are valid)

5.7.4 Monochrome-to-Color Expansion BitBLT

This function can be used to perform a text-expansion BitBLT. In Windows CE applications, this operation is the next most useful form of hardware acceleration after Rectangle Fill and Source Copy.

One variation of this type of BitBLT is the Transparent Monochrome-to-Color Expansion BitBLT or Masked Solid-Fill BitBLT. See the section on “Transparent BitBLT” for more detail.

```
Destination coordinate = DestX, DestY
Width W and height H
Foreground color = SolidColor1
Background color = SolidColor2

WaitCmdFIFO(6);                    // Need 6 command FIFO entries
```

```
GE01[11- 0] = W;
GE01[27-16] = H;                    // Width and height for the BLT
GE02[11- 0] = DestX;
GE02[27-16] = DestY;                // Destination X and Y
GE07[ 2- 0] = BitPhaseOfSource      // Bit offset within a byte.
GE07[ 5- 3] = BytePhaseOfSource     // Byte offset within a DWORD
GE08[15- 0] = SolidColor1;          // Foreground Color Register
GE09[15- 0] = SolidColor2;          // Background Color Register

// Set up Drawing Command Register
GE00[ 7- 0] = rop;                  // Set up rop code
GE00[   13] = 1;                    // Source is in system memory
GE00[   14] = 1;                    // Mono Source
GE00[   16] = 0;                    // Transparency NOT enable
GE00[10- 8] = bit BLT;              // This should trigger engine operation

// Start pumping data to Source Image Data Register. (see below for details)
```

To calculate the correct number of DWORDs for Source Image Data Register per scanline:

```
# of DWORD per line = (Width + (BytePhaseOfSource * 8) + BitPhaseOfSource + 31) mod 32

Where
    Width             = width in pixels of each scanline
    BitPhaseOfSource  = 0,1,2,3,4,5,6 or 7 (Bit Offset within a Byte)
    BytePhaseOfSource = 0, 1, 2 or 3 (Byte Offset within a DWORD)
```

If the number of DWORDs per scanline is an odd number, the driver has to pump one more DWORD so that the total data block is 64-bit aligned. An empty Source Image Data FIFO allows 128 bytes of data to be pumped through which is equivalent to a maximum of 1024 pixels in this monochrome source bitmap case.

5.7.5 Transparent BLT

Two types of transparency operations can be performed; one using a monochrome bitmap (Monochrome Transparency) and the other using a color compare register (Color Transparency).

Color Transparency

The source can come from either screen or system memory in this case. The Color Compare Register is used to determine which destination pixel will be overwritten.

Example A : Color transparency from screen

```
Source coordinate = SrcX, SrcY
Destination coordinate = DestX, DestY
Width W and height H
Color compare = SolidColor

WaitCmdFIFO(5);               // Need 5 command FIFO entries
GE03[11-0] = SrcX;
GE03[27-16] = SrcY;           // Source X and Y – programmed first
GE01[11-0] = W;
GE01[27-16] = H;              // Width and height for the BLT
GE02[11-0] = DestX;
GE02[27-16] = DestY;          // Destination X and Y
GE04[15-0] = SolidColor;      // Color Compare Register
```

```
// Set up Drawing Command Register
GE00[7-0]=rop;        // Set up rop code
GE00[11] = X DIR;
GE00[12] = Y DIR;             // See screen to screen BLT for detail
GE00[13]=0;                   // Source is in screen
GE00[14] = 0;                 // Color Source
GE00[16] = 1;         // Transparency Enable
GE00[17] = 0 (equal) or 1 (not equal)
GE00[10-8]=BitBLT;            // This should trigger engine operation
```

Example B : Color transparency from bitmap in system memory

```
Destination coordinate = DestX, DestY
Width W and height H
Color compare = SolidColor

WaitCmdFIFO(5);               // Need 5 command FIFO entries
GE01[11-0] = W;
GE01[27-16] = H;              // Width and height for the BLT
GE02[11-0] = DestX;
GE02[27-16] = DestY;          // Destination X and Y
GE04[15-0] = SolidColor;      // Color Compare Register
GE07[2-0] = PhaseOfSource;    // Byte Offset

// Set up Drawing Command Register
GE00[7-0] = rop;              // Set up rop code
GE00[13] = 1;                 // Source is in system memory
GE00[14] = 0;                 // Color Source
GE00[16] = 1;         // Transparency Enable
GE00[17] = 0 (equal) or 1 (not equal)
GE00[10-8]=bit BLT;           // This should trigger engine operation

// Start pumping data to Source Image Data Register (see Memory to Screen
// BLT above for details).
```

Monochrome Transparency

This is the Transparent Monochrome-to-Color Expansion BitBLT (also called Masked Solid Fill BitBLT.) Monochrome transparency causes each pixel to be written or blocked depending on the input data bit being one or zero. A color register (foreground color or background color) will contain a solid color to be written into the frame buffer.

If the Transparency Polarity bit (GE00[17]) is 0, a ONE in the input data bits will write the post-ROP result of foreground color to destination, and ZERO makes the destination pixel retain its prior value. One common usage for this type of BitBLT is to put foreground solid colored texts over an arbitrary existing background. This is also called ‘transparent text’ capability. The source data is always from system memory in this case.

Example

```
Destination coordinate = DestX, DestY
Width W and height H
Foreground color = SolidColor (or Background color = SolidColor if GE00[17] is to be 1)

WaitCmdFIFO(5);                       // Need 5 command FIFO entries
GE01[11- 0] = W;
GE01[27-16] = H;                      // Width and height for the BLT
GE02[11- 0] = DestX;
GE02[27-16] = DestY;                  // Destination X and Y
```

```
GE07[ 2- 0] = BitPhaseOfSource;       // Bit offset within a byte
GE07[ 5- 3] = BytePhaseOfSource;      // Byte offset
GE08[15- 0] = SolidColor;             // Foreground Color Register
```

5.7.6 Pattern BitBLT

The Pattern BLT is a common BitBLT operation in Windows CE applications.

Example A: Color Pattern BitBLT

```
8BPP Screen mode
Destination coordinate = DestX, DestY
Width W and height H
Pattern data : ColorPattern[64] of 8 x 8 8BPP color patterns


WaitCmdFIFO(16);              // Empty command FIFO entries
GE40 = ColorPattern[3-0];
GE41 = ColorPattern[7-4];
.
.
GE4f = ColorPattern[63-60];

WaitCmdFIFO(4);               // Need 4 more command FIFO entries
GE01[11- 0]    = W;
GE01[27-16]    = H;
GE02[11- 0]    = DestX;
GE02[27-16]    = DestY;
GE07[ 8- 6]    = Pattern BitBLT order at x-direction;
GE07[11- 9]    = Pattern BitBLT order at y-direction;
GE00[   15]    = 0;           // color pattern
GE00[ 7- 0]    = rop;         // Set up rop code
GE00[10- 8]    = Pattern BLT; // This should trigger engine operation
```

Example B: Mono Pattern BitBLT

```
Destination coordinate = DestX, DestY
Width W and height H
Pattern data : MonoPattern[8] of 8 x 8 1BPP monochrome patterns


WaitCmdFIFO(6);               // Need 6 command FIFO entries
GE0a = MonoPattern[3:0];
GE0b = MonoPattern[7:4];

GE01[11-0] = W;
GE01[27-16] = H;
GE02[11-0] = DestX;
GE02[27-16] = DestY;
GE07[8-6] = Pattern BitBLT order at x-direction;
GE07[11-9] = Pattern BitBLT order at y-direction;
GE00[15] = 1;          // mono pattern
GE00[7-0]=rop;         // Set up rop code
GE00[10-8]=Pattern BLT;       // This should trigger engine operation
```

5.7.7 Regular BitBLT - Screen to Memory

This operation is currently not supported by the MQ-100/HD64464 Graphics Engine. Direct frame buffer access is provided, which is an acceptable alternative to accomplish screen bitmap transfer to memory operation. Typical usage of screen to memory BitBLT is often seen as a user browses through window pull-down menus.

5.7.8 Line Drawing

The MQ100 does not implement a full line-drawing engine. However, the display driver can accelerate horizontal and vertical solids lines with by using Rectangle Fill with a width of 1.

45- and 135-degree diagonal solid lines can be accelerated using a similar technique

5.7.9 Graphics Engine Register Definition

The graphics engine commands rely on parameters that are set in GE registers. To allow the device drivers to set up the next operation while the current one is in progress, two sets of parameter registers are provided. While one set is in used, the other can be set up for the next operation. When the next operation starts and that set of registers is in use, the first set can be set up for the next operation.

The *primary* register set is in double-word address range xx00h – xx7Fh, and the *secondary* register set is in xx80h – xxFFh. All the registers are programmed through the Command FIFO except the CPU Source FIFO write registers, which are written directly. The CPU Source FIFO write registers are in a 32 double-word address range. The primary Source FIFO registers are in the double-word address range xx20h – xx3Fh, and the secondary source FIFO registers are in the range xxA0h – xxCFh. The color pattern registers are also written separately. There are two 8x8x16-bpp color pattern registers in the register address range xx40h – xx5Fh (primary) and xxC0h – xxDFh (secondary).

The following shows the double-word address space for Graphics Engine registers:

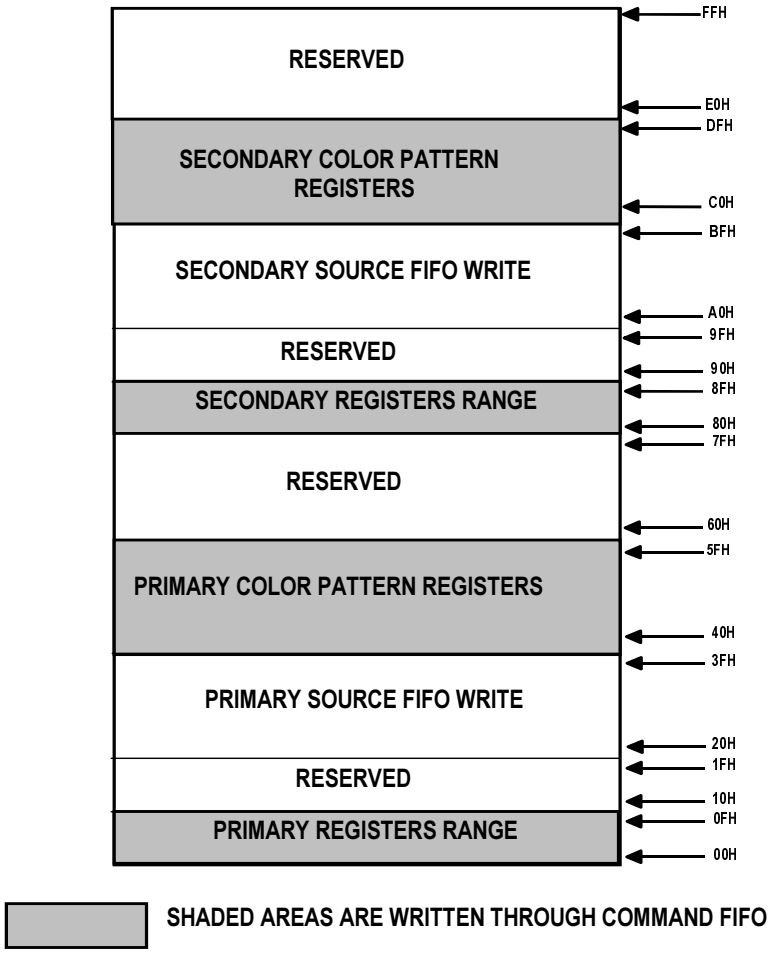


Figure 5F: Double-Word Address Space for Graphics Engine Registers

5.7.10 GE00R: Primary Drawing Command Register

GE00R defines the drawing command to be performed by the Graphics Engine.

GE	GE00R - Primary Drawing Command Register Index: Power on Reset value: 0000-0000h	
7-0	*Raster Operation (ROP)* This defines the 8-bit raster operation (ROP). All 256 possible ROPs are supported.	
10-8	*Command Type* These bits define the supported drawing commands.	
	000	NOP No Operation.
	001	Rectangle Fill The position, height and width of the rectangle are defined. This is essentially a BitBLT operation with all source data forced to foreground color. Raster engine is enabled and mono/color pattern can also be included.
	010	BitBLT A rectangle of defined location, height and width is transferred to another location in memory. Source memory is defined by bit 13 of this register. Screen-to-screen and Memory-to-screen BLT are supported. The logic supports 3-operand (pattern, source, destination) raster engine so that all 256 ROPs specified in bits 7-0 are supported. Mono source and/or mono pattern can also be supported.
	Others	Reserved
11	*X Direction* This bit defines the direction of transfer for the X coordinate. This bit must be set to 0 for mono source.	
	0	Positive X direction (left-to-right drawing direction). X coordinates for source and destination width counters and the address registers within a line, get incremented after transfer of each pixel.
	1	Negative X direction (right-to-left drawing direction). X coordinates for source and destination width counters and the address registers within a line, get de-incremented after transfer of each pixel.
12	*Y Direction* This bit defines the direction of transfer for the Y coordinate. This bit must be set to 0 for mono source.	
	0	Positive Y direction (top-to-bottom drawing direction). Height counters and address registers, for source and destination get incremented after transfer of each line.
	1	Negative Y direction (bottom-to-top drawing direction). Height counters and address registers, for source and destination get decremented after transfer of each line.
13	*Source Memory* This bit defines the source memory for BitBLT operation. Source data can come from either screen (display memory) or from memory (system memory).	
	0	Source window is in screen.
	1	Source window is in memory. CPU must write source data to the Source FIFO.
14	*Mono source* This bit specifies whether source data is mono (1-bpp) or color.	
	0	Color source. Source data has the same color depth as destination data.
	1	Mono source. Source data is 1-bpp and will be expanded to the value stored in either the foreground or background color registers.
15	*Mono Pattern* This bit specifies whether pattern data is mono (1-bpp) or color.	
	0	Color pattern. Pattern data has the same color depth as destination data. 8x8 color pattern is programmed in the Color Pattern registers.

	1	Mono pattern. Pattern data is 1-bpp and will be expanded to the value stored in the Foreground or Background Color registers. 8x8 mono pattern is programmed in the Mono Pattern registers 0 and 1.
16	*Destination Transparency Enable* Setting this bit enables transparency depending on destination (output) color data.	
	0	Destination transparency is disabled.
	1	Destination transparency is enabled. The output results of the raster operation is compared with color compare registers and the transparency depends on bit 17.
17	*Destination Transparency Polarity* This bit defines polarity for destination transparency.	
	0	Destination data is not updated (transparent) if this data is equal to Color Compare register value.
	1	Destination data is not updated (transparent) if this data is *not* equal to Color Compare register value.
18	*Mono Pattern Transparency Enable* Setting this bit enables transparency depending on mono pattern data (if bit 15 of this register is set) or mono source data (if bit 14 of this register is set). If both these bits are set, this bit will enable mono source transparency.	
	0	Mono pattern transparency is disabled.
	1	Mono pattern transparency is enabled. Either foreground color or background color is defined as transparent depending on bit 19.
19	*Mono Pattern Transparency Polarity* This bit defines polarity for mono pattern transparency.	
	0	Background color for mono pattern is transparent.
	1	Foreground color for mono pattern is transparent.
22-20	*Reserved (R/W)* These bits must be programmed to 0.	
23	*Solid Mono Pattern* Setting this bit forces all pattern data to foreground color. This bit is effective only if mono pattern is programmed (bit 15 is set to 1).	
	0	Mono pattern data is specified by contents of Mono Pattern registers.
	1	Solid mono pattern. All pattern data is forced to the value of Foreground Color register.
24	*Diagonal Line Draw* Setting this bit will automatically adjust Destination Line Stride by 1 or -1 (depending on drawing direction) to draw diagonal line.	
	0	Diagonal Line Draw disabled.
	1	Diagonal Line Draw enabled.
24	*Reserved (R/W)* This bit must be programmed to 0.	
25	*Source Byte Offset Control* For mono/color source data from system memory, this bit is used to indicate whether the most significant bit of source byte offset is forced to 0 or not.	
	0	Most significant bit of source byte offset is forced to 0. The first pixel of each line must start in the lower 4 bytes of CPU Source FIFO data.
	1	Most significant bit of source byte offset is not forced to 0. The first pixel of each line can start in any byte of CPU Source FIFO data.
26	*Enable Clipping* Setting this bit enables clipping. Clipping area is defined as a rectangle. Destination (output) pixels outside the clipping rectangle will not be written to the destination.	
	0	Clipping is disabled.
	1	Clipping is enabled.
27	*Auto Execute* Setting this bit enables auto-execution of 2D command after Destination XY register is written.	
	0	The Graphics Engine command execution starts after this register is written.
	1	The Graphics Engine command execution starts after Destination XY register is written.

28	*Wait for Vertical Blank of Graphics Controller 1* Setting this bit will delay command execution until the falling edge of Vertical Display Enable of Graphics Controller 1.	
	0	Execute command immediately if bit 29 is not set.
	1	Command is executed after falling edge of Vertical Display Enable of Graphics Controller 1 is detected.
29	*Wait for Vertical Blank of Graphics Controller 2* Setting this bit will delay command execution until the falling edge of Vertical Display Enable of Graphics Controller 2.	
	0	Execute command immediately if bit 28 is not set.
	1	Command is executed after falling edge of Vertical Display Enable of Graphics Controller 2 is detected.
31-30	*Reserved (R/W)* These bits must be programmed to 0.	

5.7.11 GE01R: Primary Width and Height Register

GE01R is the window Width and Height register and defines the width and the height of the source and destination windows.

GE	**GE01R - Width and Height Register** Index: Power on Reset value: XXXX-XXXXH This register specifies the width and height of source and destination windows.
11-0	*Source/Destination Window Width* This parameter defines the number of pixels in each source/destination line.
15-12	*Reserved* These bits are not implemented.
27-16	*Source/Destination Window Height* This parameter defines the number of lines in the source/destination window.
31-28	*Reserved* These bits are not implemented.

5.7.12 GE02R: Primary Destination Address Register

GE02R sets the X and Y coordinates of the destination window.

GE	**GE02R - Destination XY Register** Index: Power on Reset value: xxxx-xxxxxh This register defines the X and Y starting position of the destination window.
11-0	*Destination X Position* This parameter defines the X start position of destination window.
15-12	*Reserved* These bits are not implemented.
27-16	*Destination Y Position* This parameter defines the Y start position of destination window.
31-28	*Reserved* These bits are not implemented.

5.7.13 GE03R: Primary Source XY Register

GE03R sets the X and Y coordinates of the source window**.**

GE	**GE03R - Source XY Register** Index: Power on Reset value: XXXX-XXXXH This register defines the X and Y starting position of the source window.
11-0	*Source X Position* This parameter defines the X start position of source window.
15-12	*Reserved* These bits are not implemented.
27-16	*Source Y Position* This parameter defines the Y start position of source window.
31-28	*Reserved* These bits are not implemented.

5.7.14 GE04R: Primary Color Compare Register

GE04R sets pixel value for transparency. Whenever a pixel matches this register, the pixel will be treated as transparent.

GE	**GE04R - Color Compare Register** Index: Power on Reset value: XXXX-XXXXH This register is used to determine the destination (output) color for destination color transparency.
15-0	*Destination Transparent Color* This is either 8-bpp or 16-bpp color for destination color transparency.
31-16	*Reserved* These bits are not implemented.

5.7.15 GE05R: Primary Clip Left/Top Register

GE05R specifies the Left Edge and the Top Edge of the clipping window. All destination (output) pixels outside the clippling rectangle will be clipped.

GE	**GE05R - Clip Left/Top Register** Index: Power on Reset value: XXXX-XXXXH
11-0	*Left Edge of Clipping Rectangle* This parameter specifies the left edge of the clipping rectangle. All destination (output) pixels with a X coordinate less than this value will not be written.
15-12	*Reserved* These bits are not implemented.
27-16	*Top Edge of Clipping Rectangle* This parameter specifies the top edge of the clipping rectangle. All destination (output) pixels with a Y coordinate less than this value will not be written.
31-28	*Reserved* These bits are not implemented.

5.7.16 GE06R: Primary Clip Right/Bottom Register

GE06R specifies the Right Edge and the Bottom Edge of the clipping window. All destination (output) pixels outside the clippling rectangle will be clipped.

GE	GE06R - Clip Right/Bottom Register Index: Power on Reset value: XXXX-XXXXH
11-0	*Right Edge of Clipping Rectangle* This parameter specifies the right edge of the clipping rectangle. All destination (output) pixels with a X coordinate larger than this value will not be written.
15-12	*Reserved* These bits are not implemented.
27-16	*Bottom Edge of Clipping Rectangle* This parameter specifies the bottom edge of the clipping rectangle. All destination (output) pixels with a Y coordinate less than this value will not be written.
31-28	*Reserved* These bits are not implemented.

5.7.17 GE07R: Primary Source and Pattern Offset Register

GE07R defines the offset values for CPU source data and for pattern data. Offsets are used when the source image or pattern is not aligned. The offsets indicate where the first pixel can be found. Alignment sizes vary with pixel mode.

GE	GE07 - Source and Pattern Offset Register Index: Power on Reset value: XXXX-XXXXH	
2-0	*Initial Color Source Byte Offset or Mono Source Bit Offset* This parameter is used to indicate the starting byte for the first line of color source data from system memory or it indicates the starting bit (in the starting byte) for mono source data for each line of source data coming from either memory or screen. If used as color source byte offset, this value will be adjusted with the source byte offset adjustment value (bits 14-12) for subsequent lines. For mono source, this parameter is used to specify bit offset for mono source data coming from both memory and screen. Drawing Command Register bit 25 can be used to force the most significant bit of Source Byte Offset to 0.	
	000	Color source first line starts at byte 0 or mono source line starts at bit 0.
	001	Color source first line starts at byte 1 or mono source line starts at bit 1.
	010	Color source first line starts at byte 2 or mono source line starts at bit 2.
	011	Color source first line starts at byte 3 or mono source line starts at bit 3.
	100	Color source first line starts at byte 4 or mono source line starts at bit 4.
	101	Color source first line starts at byte 5 or mono source line starts at bit 5.
	110	Color source first line starts at byte 6 or mono source line starts at bit 6.
	111	Color source first line starts at byte 7 or mono source line starts at bit 7.
5-3	*Initial Mono Source Byte Offset* This parameter is used to indicate starting byte for first line of mono source data from system memory. This parameter is not used if source data is from screen. This value will be adjusted with the source byte offset adjustment value (bits 14-12) for subsequent lines. Drawing Command Register bit 25 can be used to force the most significant bit of Source Byte Offset to 0.	
	000	Mono source line starts at byte 0.
	001	Mono source line starts at byte 1.
	010	Mono source line starts at byte 2.
	011	Mono source line starts at byte 3.
	100	Mono source line starts at byte 4.
	101	Mono source line starts at byte 5.

	110	Mono source line starts at byte 6.
	111	Mono source line starts at byte 7.
8-6	*Mono/Color Pattern Horizontal Offset* This parameter is used to indicate starting pixel for mono or color pattern data for each line of pattern data.	
	000	Pattern data line starts at pixel 0.
	001	Pattern data line starts at pixel 1.
	010	Pattern data line starts at pixel 2.
	011	Pattern data line starts at pixel 3.
	100	Pattern data line starts at pixel 4.
	101	Pattern data line starts at pixel 5.
	110	Pattern data line starts at pixel 6.
	111	Pattern data line starts at pixel 7.
11-9	*Mono/Color Pattern Vertical Offset* This parameter is used to indicate starting line for mono or color pattern data.	
	000	Pattern data starts from line 0.
	001	Pattern data starts from line 1.
	010	Pattern data starts from line 2.
	011	Pattern data starts from line 3.
	100	Pattern data starts from line 4.
	101	Pattern data starts from line 5.
	110	Pattern data starts from line 6.
	111	Pattern data starts from line 7.
14-12	*Source Byte Offset Adjustment* This parameter is used to adjust mono/color source byte offset at the beginning of every source line from system memory. This value will be added to the initial mono/color source byte offset value at the beginning of every line after the first line.	
31-15	*Reserved* These bits are not implemented.	

5.7.18 GE08: Primary Foreground Color Register / Rectangle Fill Color

GE08R defines the non-transparent foreground color when the mono source/pattern is '1'. This color value is also used as source data for the Rectangle Fill command.

GE	**GE08R - Foreground Color Register / Rectangle Fill Color** Index: Power on Reset value: XXXX-XXXXH
15-0	*Foreground Color* This is 8-bpp or 16-bpp foreground color value. Bits 7-0 are used for 8-bpp mode. Foreground color for mono pattern can be defined as transparent, in which case this value is ignored.
31-16	*Reserved* These bits are not implemented.

5.7.19 GE09R: Primary Background Color Register

GE09R defines the non-transparent background color when the mono source/pattern bit is '0'.

GE	**GE09R - Background Color Register** Index: Power on Reset value: XXXX-XXXXH
15-0	*Background Color* This is 8-bpp or 16-bpp background color value. Bits 7-0 are used for 8-bpp mode. Background color for mono pattern can be defined as transparent, in which case this value is ignored.
31-16	*Reserved* These bits are not implemented.

5.7.20 GE0AR: Primary Mono Pattern Register 0

GE0AR contains the first four lines of the mono pattern data.

GE	**GE0AR - Mono Pattern Register 0** Index: Power on Reset value: XXXX-XXXXH
7-0	*Line 0 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
15-8	*Line 1 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
23-16	*Line 2 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
31-24	*Line 3 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.

5.7.21 GE0BR: Primary Mono Pattern Register 1

GE0BR contains the second four lines of mono pattern data.

GE	**GE0BR - Mono Pattern Register 1** Index: Power on Reset value: XXXX-XXXXH
7-0	*Line 4 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
15-8	*Line 5 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
23-16	*Line 6 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
31-24	*Line 7 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.

5.7.22 GE0CR: Primary Source and Destination Stride Register

GE0CR defines the stride for source lines when source data comes from display memory, and for destination lines and color depth.

GE	**GE0CR - Source and Destination Stride Register.** Index: Power on Reset value: XXXX-XXXXH	
11-0	*Source Line Stride* This parameter specifies the number of bytes from the first pixel of a source line to the first pixel of the next source line when mono or color source data comes from screen (display memory). If source data comes from CPU or in a Rectangle Fill command, this parameter is not used. This parameter is *not* used to calculate the address of the first pixel of mono/color source window when the source comes from screen; destination line stride is used instead.	
15-12	*Reserved* These bits are not implemented.	
27-16	*Destination Line Stride* This parameter specifies the number of bytes from the first pixel of a line to the first pixel of the next line for destination data. This parameter is also used to calculate the address of the first pixel of the destination window and the first pixel of mono/color source window if source comes from screen (display memory).	
28	*Reserved* This bit is not implemented.	
30-29	*Color Depth* This parameter defines the number of bits per pixel (bpp) for destination data and for color source/ pattern. This parameter is also used to calculate the address of first pixel of mono source window from screen.	
	00	8 bpp.
	01	16 bpp.
	1x	Reserved.
31	*Reserved* This bit is not implemented.	

5.7.23 GE0DR: Primary Base Address Register

GE0DR is the base (start) address of the image in the display memory.

GE	GE0DR - Base Address Register Index: Power on Reset value: XXXX-XXXXH	
19-0	*Base Address* This parameter defines the start address of the image in the display memory. This is a byte address but the lower 3 bits are internally forced to zeros, therefore this parameter is aligned to quad-word boundary.	
28-20	*Reserved* These bits are not implemented.	
29	*Test Mode Enable.* Setting this bit will enable test mode which allows the internal ALU to be read through register GE1FR. This bit is used for factory testing only.	
	0	Test mode disabled.
	1	Test mode enabled.
31-30	*Test Mode Control* These bits are used when test mode is enabled.	
	00	The address calculation results appear in GE1FR register.
	01	The clipping left and right results appear in GE1FR register.
	10	The clipping top and bottom results appear in GE1FR register.
	11	Reserved.

5.7.24 GE1FR: Test Mode Read Register

GE1FR is a read-only register which can be used to read internal status in test mode. If test mode is disabled, this register will return an unpredictable value.

GE	GE1FR - Test Mode Read Register (Read Only) Index: Power on Reset value: XXXX-XXXXH	
	For address calculation results: (GE0DR[31:30] are set to 00)	
	19-0	Contains the source address results from the ALU.
	31-20	Don't care, set to 0.
	For clipping left and right: (GE0DR[31:30] are set to 01)	
	8-0	Contains the left difference between shcleft and shdstx.
	15-9	Don't care, set to 0.
	24-16	Contains the right difference between shcright and shdstx.
	31-25	Don't care, set to 0.
	For clipping top and bottom results: (GE0DR[31:30] are set to 10)	
	11-0	Contains the top difference between shctop and shdsty.
	15-12	Don't care, set to 0.
	27-16	Contains the bottom difference between shcbot and shdsty.
	31-28	Don't care, set to 0.

5.7.25 GE20R- GE3FR: Primary Source Image Data Registers

GE20R-GE3FR are the Image data Registers. CPU write to this address range will write the data to the CPU Source FIFO for memory-to-screen BLT. The Source FIFO can also be accessed through GEA0R to GEBFR. ***The software driver must make sure that the end of each source data line falls on a quad-word boundary; otherwise, one additional dummy write must be written.***

GE	**GE20R to GE3FR - Source Image Data Register (Write Only)** Index: Power on Reset value: XXXX-XXXXH
31-0	*CPU Source Image Data*

5.7.26 GE40R-GE5FR: Primary Color Pattern Registers

GE40R-G5FR are the registers that define the Primary Color Pattern. They store the 8x8 color pattern data in packed mode for the primary command pipeline.

In 8-bpp mode, each register stores 4 pixels of pattern data. The first line of pattern data is stored in GE40R (pixel 0 to 3) and GE41R (pixel 4 to 7) and the last line of pattern data is stored in GE4ER and GE4FR. Pixel 0 and pixel 4 are stored in least significant byte and pixel 3 and pixel 7 are stored in most significant byte.

In 16-bpp mode, each register stores 2 pixels of pattern data. The first line of pattern data is stored in GE40R (pixel 0 and 1), GE41R (pixel 2 and 3), GE42R (pixel 4 and 5), and GE43R (pixel 6 and 7). The last line of pattern data is stored in GE5CR to GE5FR. Pixels 0 or 2 or 4 or 6 are stored in bits 15-0 and pixels 1 or 3 or 5 or 7 are stored in bits 31-16

GE	**GE40R to GE5FR - Primary Color Pattern Register** Index: Power on Reset value: XXXX-XXXXH
31-0	*Primary Color Pattern Data*

5.7.27 GE80R: Secondary Drawing Command Register

GE80R defines the drawing command to be performed by the Graphics Engine.

GE	**GE80R - Secondary Drawing Command Register** Index: Power on Reset value: 0000-0000h	
7-0	*Raster Operation (ROP)* This defines the 8-bit raster operation (ROP). All 256 possible ROPs are supported.	
10-8	*Command Type* These bits define the supported drawing commands.	
	000	NOP No Operation.
	001	Rectangle Fill The position, height and width of the rectangle are defined. This is essentially a BitBLT operation with all source data forced to foreground color. Raster engine is enabled and mono/color pattern can also be included.
	010	BitBLT A rectangle of defined location, height and width is transferred to another location in memory. Source memory is defined by bit 13 of this register. Screen-to-screen and Memory-to-screen BLT are supported. The logic supports 3-operand (pattern, source, destination) raster engine so that all 256 ROPs specified in bits 7-0 are supported. Mono source and/or mono pattern can also be supported.
	Others	Reserved

11	*X Direction* This bit defines the direction of transfer for the X coordinate. This bit must be set to 0 for mono source.	
	0	Positive X direction (left-to-right drawing direction). X coordinates for source and destination width counters and the address registers within a line, get incremented after transfer of each pixel.
	1	Negative X direction (right-to-left drawing direction). X coordinates for source and destination width counters and the address registers within a line, get decremented after transfer of each pixel.
12	*Y Direction* This bit defines the direction of transfer for the Y coordinate. This bit must be set to 0 for mono source.	
	0	Positive Y direction (top-to-bottom drawing direction). Height counters and address registers, for source and destination get incremented after transfer of each line.
	1	Negative Y direction (bottom-to-top drawing direction). Height counters and address registers, for source and destination get decremented after transfer of each line.
13	*Source Memory* This bit defines the source memory for BitBLT operation. Source data can come from either screen (display memory) or from memory (system memory).	
	0	Source window is in screen.
	1	Source window is in memory. CPU must write source data to the Source FIFO.
14	*Mono source* This bit specifies whether source data is mono (1-bpp) or color.	
	0	Color source. Source data has the same color depth as destination data.
	1	Mono source. Source data is 1-bpp and will be expanded to the value stored in either the foreground or background color registers.
15	*Mono Pattern* This bit specifies whether pattern data is mono (1-bpp) or color.	
	0	Color pattern. Pattern data has the same color depth as destination data. 8x8 color pattern is programmed in the Color Pattern registers.
	1	Mono pattern. Pattern data is 1-bpp and will be expanded to the value stored in the Foreground or Background Color registers. 8x8 mono pattern is programmed in the Mono Pattern registers 0 and 1.
16	*Destination Transparency Enable* Setting this bit enables transparency depending on destination (output) color data.	
	0	Destination transparency is disabled.
	1	Destination transparency is enabled. The output results of the raster operation is compared with color compare registers and the transparency depends on bit 17.
17	*Destination Transparency Polarity* This bit defines polarity for destination transparency.	
	0	Destination data is not updated (transparent) if this data is equal to Color Compare register value.
	1	Destination data is not updated (transparent) if this data is *not* equal to Color Compare register value.
18	*Mono Pattern Transparency Enable* Setting this bit enables transparency depending on mono pattern data (if bit 15 of this register is set) or mono source data (if bit 14 of this register is set). If both these bits are set, this bit will enable mono source transparency.	
	0	Mono pattern transparency is disabled.
	1	Mono pattern transparency is enabled. Either foreground color or background color is defined as transparent depending on bit 19.
19	*Mono Pattern Transparency Polarity* This bit defines polarity for mono pattern transparency.	
	0	Background color for mono pattern is transparent.
	1	Foreground color for mono pattern is transparent.

22-20	*Reserved (R/W)* These bits must be programmed to 0.	
23	*Solid Mono Pattern* Setting this bit forces all pattern data to foreground color. This bit is effective only if mono pattern is programmed (bit 15 is set to 1).	
	0	Mono pattern data is specified by contents of Mono Pattern registers.
	1	Solid mono pattern. All pattern data is forced to the value of Foreground Color register.
24	*Diagonal Line Draw* Setting this bit will automatically adjust Destination Line Stride by 1 or -1 (depending on drawing direction) to draw diagonal line.	
	0	Diagonal Line Draw disabled.
	1	Diagonal Line Draw enabled.
24	*Reserved (R/W)* This bit must be programmed to 0.	
25	*Source Byte Offset Control* For mono/color source data from system memory, this bit is used to indicate whether the most significant bit of source byte offset is forced to 0 or not.	
	0	Most significant bit of source byte offset is forced to 0. The first pixel of each line must start in the lower 4 bytes of CPU Source FIFO data.
	1	Most significant bit of source byte offset is not forced to 0. The first pixel of each line can start in any bytes of CPU Source FIFO data.
26	*Enable Clipping* Setting this bit enables clipping. Clipping area is defined as a rectangle. Destination (output) pixels outside the clipping rectangle will not be written to the destination.	
	0	Clipping is disabled.
	1	Clipping is enabled.
27	*Auto Execute* Setting this bit enables auto-execution of 2D command after Destination XY register is written.	
	0	The Graphics Engine command execution starts after this register is written.
	1	The Graphics Engine command execution starts after Destination XY register is written.
28	*Wait for Vertical Blank of Graphics Controller 1* Setting this bit will delay command execution until the falling edge of Vertical Display Enable of Graphics Controller 1.	
	0	Execute command immediately if bit 29 is not set.
	1	Command is executed after falling edge of Vertical Display Enable of Graphics Controller 1 is detected.
29	*Wait for Vertical Blank of Graphics Controller 2* Setting this bit will delay command execution until the falling edge of Vertical Display Enable of Graphics Controller 2.	
	0	Execute command immediately if bit 28 is not set.
	1	Command is executed after falling edge of Vertical Display Enable of Graphics Controller 2 is detected.
31-30	*Reserved (R/W)* These bits must be programmed to 0.	

5.7.28 GE81R: Secondary Width and Height Register

GE81R is the window width and Height register and defines the width and the height of the source and destination windows.

GE	**GE81R - Width and Height Register** Index: Power on Reset value: XXXX-XXXXH This register specifies the width and height of source and destination windows.
11-0	*Source/Destination Window Width* This parameter defines the number of pixels in each source/destination line.
15-12	*Reserved* These bits are not implemented.
27-16	*Source/Destination Window Height* This parameter defines the number of lines in the source/destination window.
31-28	*Reserved* These bits are not implemented.

5.7.29 GE82R: Destination XY Register

GE82R sets the X and Y coordinates of the destination window.

GE	**GE82R - Destination XY Register** Index: Power on Reset value: xxxx-xxxxxh This register defines the X and Y starting position of the destination window.
11-0	*Destination X Position* This parameter defines the X start position of destination window.
15-12	*Reserved* These bits are not implemented.
27-16	*Destination Y Position* This parameter defines the Y start position of destination window.
31-28	*Reserved* These bits are not implemented.

5.7.30 GE83R: Secondary Source XY Register

GE83R sets the X and Y coordinates of the source window**.**

GE	**GE83R - Source XY Register** Index: Power on Reset value: XXXX-XXXXH This register defines the X and Y starting position of the source window.
11-0	*Source X Position* This parameter defines the X start position of source window.
15-12	*Reserved* These bits are not implemented.
27-16	*Source Y Position* This parameter defines the Y start position of source window.
31-28	*Reserved* These bits are not implemented.

5.7.31 GE84R: Secondary Color Compare Register

GE84R sets pixel value for transparency. Whenever a pixel matches this register, the pixel will be treated as transparent.

GE	**GE84R - Color Compare Register** Index: Power on Reset value: XXXX-XXXXH This register is used to determine the destination (output) color for destination color transparency.
15-0	*Destination Transparent Color* This is either 8-bpp or 16-bpp color for destination color transparency.
31-16	*Reserved* These bits are not implemented.

5.7.32 GE85R: Secondary Clip Left/Top Register

GE85R specifies the left edge and the top edge of the clipping window. All destination (output) pixels outside the clippling rectangle will be clipped.

GE	**GE85R - Clip Left/Top Register** Index: Power on Reset value: XXXX-XXXXH
11-0	*Left Edge of Clipping Rectangle* This parameter specifies the left edge of the clipping rectangle. All destination (output) pixels with a X coordinate less than this value will not be written.
15-12	*Reserved* These bits are not implemented.
27-16	*Top Edge of Clipping Rectangle* This parameter specifies the top edge of the clipping rectangle. All destination (output) pixels with a Y coordinate less than this value will not be written.
31-28	*Reserved* These bits are not implemented.

5.7.33 GE86R: Secondary Clip Right/Bottom Register

GE86R specifies the right edge and the bottom edge of the clipping window. All destination (output) pixels outside the clippling rectangle will be clipped.

GE	**GE86R - Clip Right/Bottom Register** Index: Power on Reset value: XXXX-XXXXH
11-0	*Right Edge of Clipping Rectangle* This parameter specifies the right edge of the clipping rectangle. All destination (output) pixels with a X coordinate larger than this value will not be written.
15-12	*Reserved* These bits are not implemented.
27-16	*Bottom Edge of Clipping Rectangle* This parameter specifies the bottom edge of the clipping rectangle. All destination (output) pixels with a Y coordinate less than this value will not be written.
31-28	*Reserved* These bits are not implemented.

5.7.34 GE87R: Secondary Source and Pattern Offset Register

GE87R defines the offset values for CPU source data and for pattern data. Offsets are used when the source image or pattern is not aligned. The offsets indicates where the first pixel can be found. Alignment ***sizes vary with pixel modes.***

GE	GE87 - Source and Pattern Offset Register Index: Power on Reset value: XXXX-XXXXH	
2-0	*Initial Color Source Byte Offset or Mono Source Bit Offset* This parameter is used to indicate the starting byte for the first line of color source data from system memory or indicates the starting bit (in the starting byte) for mono source data for each line of source data coming from either memory or screen. If used as color source byte offset, this value will be adjusted with the source byte offset adjustment value (bits 14-12) for subsequent lines. For mono source, this parameter is used to specify bit offset for mono source data coming from both memory and screen. Drawing Command Register bit 25 can be used to force the most significant bit of Source Byte Offset to 0.	
	000	Color source first line starts at byte 0 or mono source line starts at bit 0.
	001	Color source first line starts at byte 1 or mono source line starts at bit 1.
	010	Color source first line starts at byte 2 or mono source line starts at bit 2.
	011	Color source first line starts at byte 3 or mono source line starts at bit 3.
	100	Color source first line starts at byte 4 or mono source line starts at bit 4.
	101	Color source first line starts at byte 5 or mono source line starts at bit 5.
	110	Color source first line starts at byte 6 or mono source line starts at bit 6.
	111	Color source first line starts at byte 7 or mono source line starts at bit 7.
5-3	*Initial Mono Source Byte Offset* This parameter is used to indicate starting byte for first line of mono source data from system memory. This parameter is not used if source data is from screen. This value will be adjusted with the source byte offset adjustment value (bits 14-12) for subsequent lines. Drawing Command Register bit 25 can be used to force the most significant bit of Source Byte Offset to 0.	
	000	Mono source line starts at byte 0.
	001	Mono source line starts at byte 1.
	010	Mono source line starts at byte 2.
	011	Mono source line starts at byte 3.
	100	Mono source line starts at byte 4.
	101	Mono source line starts at byte 5.
	110	Mono source line starts at byte 6.
	111	Mono source line starts at byte 7.
8-6	*Mono/Color Pattern Horizontal Offset* This parameter is used to indicate starting pixel for mono or color pattern data for each line of pattern data.	
	000	Pattern data line starts at pixel 0.
	001	Pattern data line starts at pixel 1.
	010	Pattern data line starts at pixel 2.
	011	Pattern data line starts at pixel 3.
	100	Pattern data line starts at pixel 4.
	101	Pattern data line starts at pixel 5.
	110	Pattern data line starts at pixel 6.
	111	Pattern data line starts at pixel 7.
11-9	*Mono/Color Pattern Vertical Offset* This parameter is used to indicate starting line for mono or color pattern data.	
	000	Pattern data starts from line 0.
	001	Pattern data starts from line 1.
	010	Pattern data starts from line 2.
	011	Pattern data starts from line 3.

	100	Pattern data starts from line 4.
	101	Pattern data starts from line 5.
	110	Pattern data starts from line 6.
	111	Pattern data starts from line 7.
14-12	*Source Byte Offset Adjustment* This parameter is used to adjust mono/color source byte offset at the beginning of every source line from system memory. This value will be added to the initial mono/color source byte offset value at the beginning of every line after the first line.	
31-15	*Reserved* These bits are not implemented.	

5.7.35 GE88R: Secondary Foreground Color Register / Rectangle Fill Color

GE88R defines the non-transparent foreground color when the mono source/pattern is '1'. This color value is also used as source data for the Rectangle Fill command.

GE	**GE88R - Foreground Color Register / Rectangle Fill Color** Index: Power on Reset value: XXXX-XXXXH
15-0	*Foreground Color* This is 8-bpp or 16-bpp foreground color value. Bits 7-0 are used for 8-bpp mode. Foreground color for mono pattern can be defined as transparent, in which case this value is ignored.
31-16	*Reserved* These bits are not implemented.

5.7.36 GE89R: Secondary Background Color Register

GE89R defines the non-transparent background color when the mono source/pattern bit is '0'.

GE	**GE89R - Background Color Register** Index: Power on Reset value: XXXX-XXXXH
15-0	*Background Color* This is 8-bpp or 16-bpp background color value. Bits 7-0 are used for 8-bpp mode. Background color for mono pattern can be defined as transparent, in which case this value is ignored.
31-16	*Reserved* These bits are not implemented.

5.7.37 GE8AR: Secondary Mono Pattern Register 0

GE8AR contains the first four lines of the mono pattern data.

GE	**GE8AR - Mono Pattern Register 0** Index: Power on Reset value: XXXX-XXXXH
7-0	*Line 0 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
15-8	*Line 1 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
23-16	*Line 2 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
31-24	*Line 3 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.

5.7.38 GE8BR: Secondary Mono Pattern Register 1

GE8BR contains the second four lines of mono pattern data.

GE	**GE8BR - Mono Pattern Register 1** Index: Power on Reset value: XXXX-XXXXH
7-0	*Line 4 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
15-8	*Line 5 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
23-16	*Line 6 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.
31-24	*Line 7 of mono pattern.* Pixel 0 is the most significant bit and pixel 7 is the least significant bit.

5.7.39 GE8CR: Secondary Source and Destination Stride Register

GE8CR defines the stride for source lines when source data comes from display memory, and for destination lines and color depth.

GE	**GE8CR - Source and Destination Stride Register.** Index: Power on Reset value: XXXX-XXXXH	
11-0	*Source Line Stride* This parameter specifies the number of bytes from the first pixel of a source line to the first pixel of the next source line when mono or color source data comes from screen (display memory). If source data comes from CPU or in a Rectangle Fill command, this parameter is not used. This parameter is *not* used to calculate the address of the first pixel of mono/color source window when the source comes from screen; destination line stride is used instead.	
15-12	*Reserved* These bits are not implemented.	
27-16	*Destination Line Stride* This parameter specifies the number of bytes from the first pixel of a line to the first pixel of the next line for destination data. This parameter is also used to calculate the address of the first pixel of the destination window and the first pixel of mono/color source window if source comes from screen (display memory).	
28	*Reserved* This bit is not implemented.	
30-29	*Color Depth* This parameter defines the number of bits per pixel (bpp) for destination data and for color source/ pattern. This parameter is also used to calculate the address of first pixel of mono source window from screen.	
	00	8 bpp.
	01	16 bpp.
	1x	Reserved.
31	*Reserved* This bit is not implemented.	

5.7.40 GE8DR: Secondary Base Address Register

GE8DR is the base (start) address of the image in the display memory.

GE	GE8DR - Base Address Register Index: Power on Reset value: XXXX-XXXXH	
19-0	*Base Address* This parameter defines the start address of the image in the display memory. This is a byte address but the lower 3 bits are internally forced to zeros, therefore this parameter is aligned to quad-word boundary.	
28-20	*Reserved* These bits are not implemented.	
29	*Test Mode Enable.* Setting this bit will enable test mode which allows the internal ALU to be read through register GE1FR. This bit is used for factory testing only.	
	0	Test mode disabled.
	1	Test mode enabled.
31-30	*Test Mode Control* These bits are used when test mode is enabled.	
	00	The address calculation results appear in GE1FR register.
	01	The clipping left and right results appear in GE1FR register.
	10	The clipping top and bottom results appear in GE1FR register.
	11	Reserved.

5.7.41 GE20R-GE3FR: Source Image Data Registers

GEA0R-GEBFR are the Image data Registers. CPU writes to this address range will write the data to the CPU Source FIFO for memory-to-screen BLT. The Source FIFO can also be accessed through GE20R to GE3FR. ***The software driver must make sure that the end of each source data line falls on a quad-word boundary; otherwise, one additional dummy write must be written.***

GE	GE20R to GE3FR - Source Image Data Register (Write Only) Index: Power on Reset value: XXXX-XXXXH
31-0	*CPU Source Image Data*

5.7.42 GEC0R-GEDFR: Secondary Color Pattern Registers

GEC0R-GEDFR are the registers that define the Secondary Color Pattern. These registers store the 8x8 color pattern data in packed mode for secondary command pipeline.

In 8-bpp mode, each register stores 4 pixels of pattern data. The first line of pattern data is stored in GEC0R (pixel 0 to 3) and GEC1R (pixel 4 to 7) and the last line of pattern data is stored in GECER and GECFR. Pixel 0 and pixel 4 are stored in least significant byte and pixel 3 and pixel 7 are stored in most significant byte.

In 16-bpp mode, each register stores 2 pixels of pattern data. The first line of pattern data is stored in GEC0R (pixel 0 and 1), GEC1R (pixel 2 and 3), GEC2R (pixel 4 and 5), and GEC3R (pixel 6 and 7). The last line of pattern data is stored in GEDCR to GEDFR. Pixels 0 or 2 or 4 or 6 are stored in bits 15-0 and pixels 1 or 3 or 5 or 7 are stored in bits 31-16.

GE	GEC0R to GEDFR - Secondary Color Pattern Register Index: Power on Reset value: XXXX-XXXXH
31-0	*Secondary Color Pattern Data*

5.8 CPU Interface Programming Information

5.8.1 CPU Register Definition

The CPU Interface address space includes GE and GE FIFO Status and Interrupt Controller registers, as well as the CPU interface registers.

5.8.2 CC00R: CPU Control Register

CC00R	**CPUCTRL Register** Index: Reset value: 0000-0000h	
0	*Color Palette Write Control*	
	0	Software has to write to Color Palette 1 and Color Palette 2 separately.
	1	All writes to Color Palette 1 also are applicable to Color Palette 2.
1	*Software Reset*	
	0	Reset to all the modules is deasserted.
	1	Reset to all the modules except the CIF is asserted.
2	*MIU Read Register Generator*	
	0	Read request is generated by the C1F to the MIU only when there is a cache miss. C1F keeps track of the CPU writes to the Frame Buffer to validate the cached data. C1F guarantees cache coherency only if the Frame Buffer is accessed by the CPU. In case GE modifies the data in the Frame Buffer, then C1F does not guarantee the correct data being read in case the previously cached data is from the same address (which was modified by the GE) and the CPU tries to read the address
	1	Setting this bit forces a read request by the C1F to the M1U, regardless of the validity bits. This bit gets automatically cleared once the read request is made. Software writes a 1 to this bit whenever it reads the data from the Frame Buffer modified by the GE. It writes a 1 only for the first read. For subsequent reads, after one GE command is executed, software need not set this bit.
31-3	*Reserved*	

5.8.3 CC01R: GE and GE FIFO Status Register

CC01R	Source FIFO/Command FIFO/GE Status Register Index: Reset value: 0000-0000h	
4-0	*Command FIFO Status Register (CFS). Command FIFO is 16 deep.*	
	00000	Command FIFO is EMPTY
	00001	Command FIFO has 1 location free
	00010	Command FIFO has 2 locations free
	00011	Command FIFO has 3 locations free
	00100	Command FIFO has 4 locations free
	00101	Command FIFO has 5 locations free
	00110	Command FIFO has 6 locations free
	00111	Command FIFO has 7 locations free
	01000	Command FIFO has 8 locations free
	01001	Command FIFO has 9 locations free
	01010	Command FIFO has 10 locations free
	01011	Command FIFO has 11 locations free
	01100	Command FIFO has 12 locations free
	01101	Command FIFO has 13 locations free
	01110	Command FIFO has 14 locations free
	01111	Command FIFO has 15 locations free
	10000	Command FIFO is FULL
7-5	*Reserved*	
12-8	*Source FIFO Status Register (SFS). Source FIFO is 16 - deep.*	
	00000	Source FIFO is EMPTY
	00001	Source FIFO has 1 location free
	00010	Source FIFO has 2 locations free
	00011	Source FIFO has 3 locations free
	00100	Source FIFO has 4 locations free
	00101	Source FIFO has 5 locations free
	00110	Source FIFO has 6 locations free
	00111	Source FIFO has 7 locations free
	01000	Source FIFO has 8 locations free
	01001	Source FIFO has 9 locations free
	01010	Source FIFO has 10 locations free
	01011	Source FIFO has 11 locations free
	01100	Source FIFO has 12 locations free
	01101	Source FIFO has 13 locations free
	01110	Source FIFO has 14 locations free
	01111	Source FIFO has 15 locations free
	10000	Source FIFO is FULL
15-13	Reserved	
16	*Busy*	
	0	Graphics Engine is IDLE.
	1	Graphics Engine is busy. This is set as long as the Command FIFO associated with the Graphics engine has any commands left in it. This signal is provided by the Graphics Engine.
23-17	Reserved	
31-24	*Reserved*	

5.8.4 IN00R: Global Interrupt Control Register

This register configures the active level of the IRL# output and the GPIO inputs (when configured as interrupts). It also provides the the global interrupt enable.

IN00R	**Global Interrupt Control Register** Index: Reset value: 0000-0000h	
0	*Interrupt Enable Bit*	
	0	Interrupt to the CPU from MQ100 is Disabled.
	1	Interrupt to the CPU from MQ100 is Enabled.
1	*Polarity of the Interrupt Pin*	
	0	Interrupt is active Low.
	1	Interrupt is active High.
2	*GPIO Interrupt Polarity – GPIO 1 – Level Triggered Interrupt*	
	0	Interrupt happens when GPIO Pin 1 is at level 0.
	1	Interrupt happens when GPIO Pin 1 is at level 1.
3	*GPIO Interrupt Polarity – GPIO 2 – Level Triggered Interrupt*	
	0	Interrupt happens when GPIO Pin 2 is at level 0.
	1	Interrupt happens when GPIO Pin 2 is at level 1.
4	*GPIO Interrupt Polarity – GPIO 3 – Level Triggered Interrupt*	
	0	Interrupt happens when GPIO Pin 3 is at level 0.
	1	Interrupt happens when GPIO Pin 3 is at level 1.

5.8.5 IN01R: Interrupt Mask Register

This register contains the individual enable bits for all interrupt sources.

IN01R	**Interrupt Mask Register** Index: Reset value: 0000-0000h	
0	*Graphics Controller 1 – Vertical Sync Enable – Rising Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
1	*Graphics Controller 1 – Vertical Sync Enable – Falling Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
2	*Graphics Controller 1 – Vertical Display Enable – Rising Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
3	*Graphics Controller 1 – Vertical Display Enable – Falling Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
4	*Graphics Controller 2 – Vertical Sync Enable – Rising Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
5	*Graphics Controller 2 – Vertical Sync Enable – Falling Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
6	*Graphics Controller 2 – Vertical Display Enable – Rising Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
7	*Graphics Controller 2 – Vertical Display Enable – Falling Edge*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
8	*Command FIFO Half Empty*	
	0	Interrupt is Masked

	1	Interrupt is Not Masked
9	*Command FIFO Empty*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
10	*Source FIFO half Empty*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
11	*Source FIFO Empty*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
12	*Graphics Engine is IDLE*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
13	*GPIO Pin 1 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
14	*GPIO Pin 2 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked
15	*GPIO Pin 3 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt is Masked
	1	Interrupt is Not Masked

5.8.6 IN02R: Interrupt Status Register

This register shows which interrupts are currently active. Only unmasked interrupts are reported. Once an unmasked interrupt event has been accepted and posted, it is the responsibility of the software to clear it by writing a 1 to the Interrupt Status Register.

IN02R	**Interrupt Status Register** Index: Reset value: 0000-0000h	
0	*Graphics Controller 1 – Vertical Sync Enable – Rising Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
1	*Graphics Controller 1 – Vertical Sync Enable – Falling Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
2	*Graphics Controller 1 – Vertical Display Enable – Rising Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
3	*Graphics Controller 1 – Vertical Display Enable – Falling Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
4	*Graphics Controller 2 – Vertical Sync Enable – Rising Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
5	*Graphics Controller 2 – Vertical Sync Enable – Falling Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
6	*Graphics Controller 2 – Vertical Display Enable – Rising Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
7	*Graphics Controller 2 – Vertical Display Enable – Falling Edge*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
8	*Command FIFO Half Empty*	

	0	Interrupt Event is False.
	1	Interrupt Event is True.
9	*Command FIFO Empty*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
10	*Source FIFO half Empty*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
11	*Source FIFO Empty*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
12	*Graphics Engine is IDLE*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
13	*GPIO Pin 1 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
14	*GPIO Pin 2 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.
15	*GPIO Pin 3 – (Multiplexed with Flat panel Signals)*	
	0	Interrupt Event is False.
	1	Interrupt Event is True.

5.8.7 IN03R: “Raw” Interrupt Status Register

This register reports the status of all interrupt sources prior to masking. The interrupt flag bits for the GPIO are not latched or synchronized; if the register is read when the interrupt source is changing state, the value reported may be incorrect. Software can correct for this possible problem by reading this register two or three times until a stable value is read.

IN03R	**Interrupt Pin Raw Status Register** Index: Reset value: 0000-0000h
0	*Graphics Controller 1 – Vertical Sync Enable*
1	*Reserved – Returns a value of 0 when read*
2	*Graphics Controller 1 – Vertical Display Enable*
3	*Reserved – Returns a value of 0 when read*
4	*Graphics Controller 2 – Vertical Sync Enable*
5	*Reserved – Returns a value of 0 when read*
6	*Graphics Controller 2 – Vertical Display Enable*
7	*Reserved – Returns a value of 0 when read*
8	*Graphics Engine Busy Signal*
9	*Source FIFO Empty*
10	*Source FIFO Half Empty*
11	*Command FIFO Empty*
12	*Command FIFO Half Empty*
13	*GPIO Pin 1*
14	*GPIO Pin 2*
15	*GPIO Pin 3*

5.9 Memory Interface Unit Programming Information

This information is provided for reference only. These registers should not be modified except by the MediaQ-supplied device drivers.

5.9.1 MM00R: Memory Interface Control Register

	MM00R - Memory Interface Control 1 Index: Reset value: XXXX-XXX0h	
0	*MIU enable bit* Setting this bit enables MIU. If this bit is reset, MIU is powered down.	
	0	MIU is powered down (reset value).
	1	MIU is enabled.
1	*MIU reset disable bit* This register bit is used to reset MIU asynchronously. Before changing any MIU register bit (except MM00R-0 and MM01R-0), this bit should be programmed as '0'.	
	0	MIU reset is enabled (reset value)
	1	MIU reset is disabled.
2	*Embedded DRAM mode/state register reset disable bit* This register bit is used to drive the embedded DRAM hardware reset (asynchronous) input in order to reset the internal DRAM's control logic. This bit should be kept at low value at least 1us. MIU reset disable bit should be programmed as '1' at least 200us after embedded DRAM mode/ state register reset disable bit is programmed as '1'.	
	0	Internal DRAM hardware reset is enabled (reset value)
	1	Internal DRAM hardware reset is disabled.
3	*Memory clock source* This bit specifies the clock source for the memory clock.	
	1	Bus clock is used as memory clock source (reset value).
	0	PLL 1 output is used as memory clock source.
4	*Memory slow refresh disable bit* This bit controls the refreshing during MIU power down mode for D0 mode only.	
	0	Memory refreshing is enabled during MIU power down mode.
	1	Memory refreshing is disabled during MIU power down mode.
5	*Memory burst refresh enable* This bit controls the burst refreshing before entering and after existing slow refresh mode. For normal operation, this bit must be set to '0'	
	0	Burst refreshing is disabled.
	1	Burst refreshing is enabled.
7-6	*Reserved (R/W)*	
18-8	*Normal memory refresh request time interval in terms of PLL reference clock.* This parameter specifies the time interval between the consecutive memory refresh requests in terms of PLL reference clock; 16 ms normal refresh. The formula for this parameter is as following: ((16 / (Refclk period in ns * (Refresh end address – refresh start address)) * (10 ** 6)) - 5	
23-19	*Reserved (R/W)*	
31-24	*Slow refresh request time interval in terms of PLL reference clock.* This parameter specifies the time interval between the consecutive memory refresh requests in terms of PLL reference clock; 64 ms slow refresh. The formula for this parameter is as following: ((64 / (Refclk period in ns * (Refresh end address – refresh start address)) * (10 ** 6)) - 5	

5.9.2 MM01R: Memory Interface Control Register 2

	MM01R - Memory Interface Control 2 Index: Reset value: XXXXXXXXH	
3-0	*Internal Memory clock delay control* This parameter specifies the delay amount from the clock going into embedded DRAM to the internal memory clock.	
	0000	0 ns delay.
	0001	.5 ns delay.
	0010	1.0 ns delay.
	0011	1.5 ns delay.
	0100	2.0 ns delay
	0101	2.5 ns delay.
	0110	3.0 ns delay.
	0111	7.0 ns delay.
	1000	3.5 ns delay
	1001	4.0 ns delay.
	1010	4.5 ns delay.
	1011	5.0 ns delay.
	1100	5.5 ns delay
	1101	6.0 ns delay.
	1110	6.5 ns delay.
	1111	7.0 ns delay.
16-4	*Start memory refresh row address for embedded DRAM refreshing. MIU generates memory refresh request by starting the row address specified with these bits.* This value corresponds to CPU address [19:7].	
29-17	*End memory refresh row address for embedded DRAM refreshing. MIU generates memory refresh request by starting from the row address specified with the above bits. And it continues the refresh until it refreshes the row address specified with these bits. After the end row is refreshed, MIU restarts memory refreshing from the start refresh row address.* This value corresponds to CPU address [19:7].	
31-30	*Delay control bits from read latch clock to embedded DRAM clock.*	
	00	0 ns delay.
	01	.5 ns delay.
	10	1.0 ns delay.
	11	1.5 ns delay.

5.9.3 MM02R: Memory Interface Control 3

	MM02R-Memory Interface Control 3 Index: Reset value: XXXXXXXXH	
1-0	*Burst count for display refresh memory cycles* These bits control the size of the burst for the display refresh memory cycles in a memory page; meaning that if MIU starts display refresh data fetching, it will fetch until it gets (burst count * 128) bits of data, or until it hits the page break, or it reaches the line end which ever becomes true first.	
	00	Burst size is two.
	01	Burst size is four.
	10	Burst size is six.
	11	Burst size is eight.
3-2	*Burst count for STN read memory cycles* These bits control the size of the burst for the STN read memory cycles in a memory page.	
	00	Burst size is two.
	01	Burst size is four.
	10	Burst size is six.
	11	Burst size is eight.
5-4	*Burst count for STN write memory cycles* These bits control the size of the burst for the STN write memory cycles in a memory page.	
	00	Burst size is two.
	01	Burst size is four.
	10	Burst size is six.
	11	Burst size is eight.
7-6	*Burst count for Graphics engine read/write memory cycles* These bits control the size of the burst for the graphics engine read/write memory cycles in a memory page.	
	00	Burst size is two.
	01	Burst size is four.
	10	Burst size is six.
	11	Burst size is eight.
9-8	*Burst count for CPU write memory cycles* These bits control the size of the burst for the CPU write memory cycles in a memory page.	
	00	Burst size is two.
	01	Burst size is four.
	10	Burst size is six.
	11	Burst size is eight.
13-10	*GC1 display refresh FIFO threshold* Whenever the number of filled locations in GC1 display refresh FIFOs is less than this threshold, the memory request for GC1 is generated in order to fetch the display refresh data for GC1. Legal values are from 0001 to 1011.	
17-14	*GC2 display refresh FIFO threshold* Whenever the number of filled locations in GC2 display refresh FIFO is less than this threshold, the memory request for GC2 is generated in order to fetch the display refresh data for GC2. Legal values are from 0001 to 1011.	
21-18	*STN read FIFO threshold* Whenever the number of filled locations in STN read FIFO is less than this threshold, the memory request for STN read is generated in order to fetch the DSTN frame buffer data. Legal values are from 0001 to 1001.	
25-22	*STN write FIFO threshold* Whenever the number of filled locations in STN write FIFO is equal to or more than this threshold, the memory request for STN write is generated in order to write the DSTN frame buffer data. Legal values are from 0001 to 1001.	

28-26	*GE Source read FIFO threshold* Whenever the number of filled locations in GE source read FIFO is less than this threshold, the memory request for GE source read is generated in order to fetch the source data for GE. Legal values are from 001 to 111.
31-29	*GE Destination read FIFO threshold* Whenever the number of filled locations in GE destination read FIFO is less than this threshold, the memory request for GE source read is generated in order to fetch the source data for GE. Legal values are from 001 to 111.

5.9.4 MM03R: Memory Interface Control Register 4

	MM03R -Memory Interface Control 4 Index: Reset value: XXXXXXXXH	
2-0	*Bank activate (without precharge) command to bank close command timing interval control* These bits control the timing interval in memory clocks from the bank activate (without precharge) command to bank close (precharge) time interval. This parameter should be >= 60 ns.	
	000	Reserved.
	001	Reserved.
	010	3 memory clocks.
	011	4 memory clocks.
	100	5 memory clocks.
	101	6 memory clocks.
	110	7 memory clocks.
4-3	*Bank activate (without precharge) command to read/write command timing interval control* These bits control the timing interval in memory clocks from the bank activate command to read/ write command when the previous state of the bank has been inactive (bank close mode). This parameter should be >= 30 ns.	
	00	Reserved.
	01	2 memory clocks.
	10	3 memory clocks.
	11	4 memory clocks.
6-5	*Bank close command to bank activate command timing interval control* These bits control the timing interval in memory clocks from the bank close command to bank activate command for the same bank. This parameter should be >= 30 ns.	
	00	Reserved.
	01	2 memory clocks.
	10	3 memory clocks.
	11	4 memory clocks.
31-7	*Reserved (R/W)*	

5.10 Power Management Unit Programming Information

5.10.1 PMR: Power Management Unit Configuration Register

PC40R	PMR – Power Management Register Index: 40h in Configuration space Reset value: 0621-0001h This register is part of the configuration space and always accessible when chip power down (PDWN#) pin is high (inactive). This is the first register to control power states. The second register is Power Management Control/Status Register (PMCSR).
7-0	*Capability ID (Read Only)* These bits are set to 01h.
15-8	*Next Item Pointer (Read Only)* These bits are set to 00h.
31-16	*Power Management Capabilities (Read Only)* These bits are set to 0621h. D2 and D1 functions are supported and device specific initialization is required following transition to D0 state.

5.10.2 PMCSR: Power Managent Control/Status Register

On Power On Reset, the Power Management State is set to D3. The MQ-100/HD64464 will be disabled until the power-up sequence is completed. Software must start the power-up sequence by writing a 0 to this register; this will request the Power Management Unit (PMU) to bring the MQ-100/HD64464 up to state D0. The PMU takes some time to accomplish this, as each portion of the device and the LCD panel interface must be powered-up sequentially. Initialization software should wait for the power-up process to complete before programming the MQ-100/HD64464. Completion is indicated when the PMCSR register reads 0, confirming that state D0 has been entered and the MQ-100/HD64464 is ready to be programmed.

PC44R	PMCSR – Power Management Control/Status Register Index: 44h in Configuration space Reset value: 0000-0003h This register is part of configuration space and always accessible when chip power down (PDWN#) pin is high (inactive). In D3 state, all register write to the chip will be directed to this register regardless of the address and byte enables. This register is reset by chip power-on reset (POR# low).	
1-0	*Power State (Read/Write)* These bits control device power states when power is applied.	
	00	D0 State. This is normal operation mode. CPU interface is fully active and all blocks can be enabled using their corresponding enable bits.
	01	D1 State. In this mode, part of CPU interface is enabled and other modules can be optionally enabled/disabled as controlled by PM01R register.
	10	D2 State. In this mode, part of CPU interface is enabled and other modules can be optionally enabled/disabled as controlled by PM02R register.

	11	D3 State. This is the deepest power down mode, which is controllable by software. This state is entered when power-on reset is applied (POR# is low). PDWN# pin must be pulled high before POR# is deactivated for the chip to remain in this state. If PDWN# is still low after POR# is deactivated, then the chip will transition to D4 State. In this state, all modules in the chip are powered down except for a small portion of the CPU interface, which is needed to monitor register writes to the chip. In this state, only configuration registers are accessible if the bus clock is active. The power management clock is expected to be on. The following pins are monitored: CKIO, FRAME#, CS#. When FRAME# and CS# are active in D3 state, then DATA pins [1:0] are enabled and latched into this register bit.
31-2	*Reserved (Read Only)* These bits are set to 0's.	

5.10.3 PM00R: Power Management Miscellaneous Control Register

PM	PM00R – Power Management Miscellaneous Control *Index: 00 Reset value: xxxx-xx00h*	
0	*Oscillator Enable* This bit controls clock oscillator (OSCCLK). Note that OSCCLK must be enabled when any of the PLLs are enabled.	
	0	Clock oscillator is powered down.
	1	Clock oscillator can be enabled. This setting can be overwritten in D1 and D2 states. In D3 and D4 states, the clock oscillator will be disabled if memory refresh is not required and it will be enabled if memory refresh is required.
1	*PLL 1 Enable* This bit controls PLL 1. This PLL can be used to supply clock for Memory Interface Unit (MIU) and other blocks.	
	0	PLL 1 is powered down.
	1	PLL 1 can be enabled. This setting can be overwritten in D1 and D2 states. This PLL is powered down in D3 and D4 states.
2	*PLL 2 Enable* This bit controls PLL 2. This PLL can be used to supply clock for Controller 1 or Controller 2.	
	0	PLL 2 is powered down.
	1	PLL 2 can be enabled. This setting can be overwritten in D1 and D2 states. This PLL is powered down in D3 and D4 states.
3	*PLL 3 Enable* This bit controls PLL 3. This PLL can be used to supply clock for Controller 1 or Controller 2.	
	0	PLL 3 is powered down.
	1	PLL 3 can be enabled. This setting can be overwritten in D1 and D2 states. This PLL is powered down in D3 and D4 states.
4	*Reserved (R/W)* This bit must be programmed to 0.	
5	*Power State Status Control* This bit controls the read value fo the Power State bits (PMCSR[1:0]).	
	0	PMCSR[1-0] reflect the power state after power sequencing is done. Software can therefore read PMCSR[1-0] to verify that power sequencing is already completed.
	1	PMCSR[1-0] reflect the power state immediately after it is programmed (before power sequencing is completed).
7-6	*Memory Refresh Clock Control* This bit controls generation of internal memory refresh clock (REFCLK) from oscillator clock (OSCCLK).	
	00	Internal memory refresh is clocked by oscillator clock.

	01	Internal memory refresh is clocked by oscillator clock divided by 2.
	10	Internal memory refresh is clocked by oscillator clock divided by 4
	11	Internal memory refresh is clocked by oscillator clock divided by 8.
8	*Graphics Engine Enable* This bit controls Graphics Engine.	
	0	Graphics Engine is powered down. All GE related logic (state machines, Command FIFO, Source FIFOs, etc) are reset.
	1	Graphics Engine can be enabled. This setting can be overwritten in D1 and D2 states. In D3 and D4 states, Graphics Engine will be disabled. This bit can normally be set all the time because the Graphics Engine can be automatically powered down when there is no activity (see bit 9).
9	*Graphics Engine Force Busy (Global)* This bit is effective when bit 8 is set. Setting this bit will disable the global dynamic power management logic for the Graphics Engine such that the engine will not be shut off automatically whenever it is idle.	
	0	Graphics Engine clock is powered down when there is no activity.
	1	Graphics Engine clock is always running.
10	*Graphics Engine Force Busy (Local)* This bit is effective when bit 8 is set. Setting this bit will disable the local dynamic power management for the Graphics Engine pipelines such that the individual pipeline will not be shut off automatically whenever it is idle.	
	0	Graphics Engine pipeline is powered down when there is no activity.
	1	Graphics Engine pipeline is always running.
12-11	*Graphics Engine Clock Select* These bits select Graphics Engine (GE) clock.	
	00	GE is driven by bus interface clock.
	01	GE is driven by PLL 1 clock
	10	GE is driven by PLL 2 clock
	11	GE is driven by PLL 3 clock.
13	*Graphics Engine Command FIFO Reset* This bit is effective independent of bit 8.	
	0	Graphics Engine Command FIFO Enabled.
	1	Graphics Engine Command FIFO Enabled.
14	*Graphics Engine CPU Source FIFO Reset* This bit is effective independent of bit 8.	
	0	Graphics Engine CPU Source FIFO Enabled.
	1	Graphics Engine CPU Source FIFO is Reset.
15	*Reserved (R/W)* This bit must be programmed to 0.	
16	*D3 Memory Refresh* This bit controls the refresh of internal frame buffer in D3 State. Note that in D3 State, the internal frame buffer memory is generally powered down, however refresh to the internal frame buffer memory can still be enabled by setting this register bit.	
	0	Frame buffer memory is not refreshed in D3 State.
	1	Frame buffer memory is refreshed in D3 State. Clock to frame buffer memory will be forced on in D3 State. If frame buffer memory clock is supplied by internal PLL1, then this PLL must be enabled and a rather significant amount of power may still be consumed.
17	*D4 Memory Refresh* This bit controls the refresh of internal frame buffer in D4 State. Note that in D4 State, the internal frame buffer memory is generally powered down, however, refresh to the internal frame buffer memory can still be enabled by setting this register bit.	
	0	Frame buffer memory is not refreshed in D4 state.

	1	Frame buffer memory is refreshed in D4 state. Clock to frame buffer memory will be forced on in D4 state. If frame buffer memory clock is supplied by internal PLL 1, then this PLL must be enabled, and a rather significant amount of power may still be consumed.
19-18	*General power sequencing interval* These bits control the time allocated to power-up the oscillator/PLL before MIU is enabled and the time allocated to power-up MIU before other blocks are enabled. This parameter is approximately applicable for power down sequencing also except for the fact that the order of the sequence is reversed. This parameter is specified in terms of Power Management Clock (PMCLK) cycles.	
	00	16 PMCLK cycles
	01	32 PMCLK cycles
	10	64 PMCLK cycles
	11	128 PMCLK cycles
21-20	*Flat Panel power sequencing interval* These bits control the time interval used for FP power-up or power-down sequencing. This is basically half of the time between ENVDD rising edge and ENVEE rising edge during flat panel power-up, or half the time between ENVEE falling edge and ENVDD falling edge during flat panel power-down. This parameter is specified in terms of Power Management Clock (PMCLK) cycles.	
	00	512 PMCLK cycles.
	01	1024 PMCLK cycles.
	10	2048 PMCLK cycles.
	11	4096 PMCLK cycles.
22	*Reserved (R/W)* This bit must be programmed to 0.	
23	*Fast Power Sequencing* Setting this bit will shorten power-sequencing intervals to one PMCLK cycle for boot-up or for testing purpose. This will override bits 21-18.	
	0	Fast power sequencing enabled.
	1	Fast power sequencing disabled.
25-24	*Power State (Read Only)* These bits return the same value as PMCSR[1:0] when read. This status is updated at the end of power sequencing.	
	00	D0 State.
	01	D1 State.
	10	D2 State.
	11	D3 State.
26	*Power Sequencing Active Status (Read Only)* This bit indicates status of power sequencing state machine. When an event that triggers power sequencing (write to PMCSR[1:0], enabling/disabling MIU, enabling/disabling flat panel) occurs, this bit will be set high within 4 PMCLK cycles. The delay in setting this bit is caused by internal synchronization of the trigger event to PMCLK. Because of this delay, it is recommended that the software checks that this bit is high after the trigger event, and then search this bit going from high to low.	
	0	Power sequencing is not in progress.
	1	Power sequencing is in progress.
31-27	*Reserved (0)* These bits do not exist (not implemented).	

5.10.4 PM01R: D1 State Control

PM	PM01R – D1 State Control *Index: 01 Reset value: 0000-0000h*	
0	*D1 Oscillator Enable* This bit controls clock oscillator (OSCCLK) in D1 State. Note that the clock oscillator must be enabled when any of the PLLs are enabled or when slow refresh for the internal memory is enabled.	
	0	Clock oscillator is powered down in D1 State.
	1	Clock oscillator can be enabled in D1 State.
1	*D1 PLL 1 Enable* This bit controls PLL 1 in D1 State.	
	0	PLL 1 is powered down in D1 State.
	1	PLL 1 can be enabled in D1 State.
2	*D1 PLL 2 Enable* This bit controls PLL 2 in D1 State.	
	0	PLL 2 is powered down in D1 State.
	1	PLL 2 can be enabled in D1 State.
3	*D1 PLL 3 Enable* This bit controls PLL 3 in D1 State.	
	0	PLL 3 is powered down in D1 State.
	1	PLL 3 can be enabled in D1 State.
4	*D1 Memory Interface Unit (MIU) Enable* This bit controls Memory Interface Unit (MIU) in D1 State.	
	0	MIU is powered-down in D1 State. Graphics Engine and both Controller 1 and Controller 2 will also be powered down. When MIU is powered down, slave read/write access to display frame buffer and to GE Command and Source FIFO cannot be executed. Similarly bus master (DMA) read/write accesses cannot be executed.
	1	MIU can be enabled in D1 State.
5	*D1 Memory Refresh Enable* This bit controls internal memory refresh when MIU is powered-down in D1 State.	
	0	Internal memory is not refreshed in D1 State when MIU is powered-down. **Note:** oscillator clock must be supplied in order to refresh the internal memory.
	1	Internal memory is refreshed in D1 State when MIU is powered-down.
6	*D1 Graphics Engine (GE) Enable* This bit controls Graphics Engine (GE) in D1 State.	
	0	GE is powered down in D1 State. Power management software must make sure that Graphics Engine is idle prior to going to power down state. If the engine is not idle prior to entering power down state, then frame buffer may be corrupted and may have to be redrawn at a later point.
	1	GE can be enabled in D1 State.
7	*Reserved (R/W)* This bit must be programmed to 0.	
8	*D1 CRT Enable.* This bit controls CRT display in D1 State. Note that CRT display can also be powered down by powering down the Graphics Controller that drives the CRT. This bit may be used only if the same controller is used to drive both CRT and flat panel and only the CRT needs to be powered down in this state.	
	0	CRT display is powered down in D1 State. This does not power down the Graphics Controller that drives the CRT.
	1	CRT display can be enabled in D1 State.

9	*D1 Flat Panel Enable.* This bit controls Flat Panel display in D1 State. Note that flat panel display can also be powered down by powering down the Graphics Controller that drives the flat panel. This bit may be used only if the same controller is used to drive both CRT and flat panel and only the flat panel needs to be powered down in this state.	
	0	Flat panel is powered down in D1 State. This does not power down the Graphics Controller that drives the flat panel.
	1	Flat panel can be enabled in D1 State.
10	*Reserved (R/W) for D1 TV Enable* This bit must be programmed to 0.	
15-11	*Reserved (R/W)* These bits must be programmed to 0.	
16	*D1 Controller 1 Enable* This bit controls Controller 1 in D1 State.	
	0	Controller 1 is powered down in D1 State. This will also power down Window 1, Cursor 1, and Overlay 1. If Controller 1 is driving CRT or Flat panel, then the CRT or Flat Panel will also be powered down.
	1	Controller 1 can be enabled in D1 State.
17	*D1 Window 1 Enable* This bit controls Window 1 in D1 State.	
	0	Window 1 is powered down in D1 State. This will power down both Window 1 and Alternate Window 1.
	1	Window 1 can be enabled in D1 State if bit 16 is 1.
18	*D1 Alternate Window 1 Enable* This bit controls Alternate Window 1 in D1 State.	
	0	Alternate Window 1 is not enabled in D1 State if bit 16 is 1 and bit 17 is 1.
	1	Alternate Window 1 is enabled in D1 State if bit 16 is 1 and bit 17 is 1.
19	*D1 Cursor 1 Enable* This bit controls Cursor 1 in D1 State.	
	0	Cursor 1 is powered down in D1 State.
	1	Cursor 1 can be enabled in D1 State if bit 16 is 1.
20	*Reserved (R/W) for D1 Overlay 1 Enable* This bit controls Overlay 1 in D1 State.	
	0	Overlay 1 is powered down in D1 State.
	1	Overlay 1 can be enabled in D1 State if bit 16 is 1.
23-21	*Reserved (R/W)* These bits must be programmed to 0.	
24	*D1 Controller 2 Enable* This bit controls Controller 2 in D1 State.	
	0	Controller 2 is powered down in D1 State. This will also power down Window 2, Cursor 2, and Overlay 2. If Controller 2 is driving CRT or Flat panel, then the CRT or Flat Panel will also be powered down.
	1	Controller 2 can be enabled in D1 State.
25	*D1 Window 2 Enable.* This bit controls Window 2 in D1 State.	
	0	Window 2 is powered down in D1 State. This will power down both Window 2 and Alternate Window 2.
	1	Window 2 can be enabled in D1 State if bit 24 is 1.
26	*D1 Alternate Window 2 Enable.* This bit controls Alternate Window 2 in D1 State.	
	0	Alternate Window 2 is not enabled in D1 State if bit 24 is 1 and bit 25 is 1.
	1	Alternate Window 2 is enabled in D1 State if bit 24 is 1 and bit 25 is 1.
27	*D1 Cursor 2 Enable.* This bit controls Cursor 2 in D1 State.	

	0	Cursor 2 is powered down in D1 State.
	1	Cursor 2 can be enabled in D1 State if bit 24 is 1.
31-28	*Reserved (R/W)* These bits must be programmed to 0.	

5.10.5 PM02R: D2 State Control

PM	PM02R – D2 State Control *Index: 02h Reset value: 0000-0000h*	
0	*D2 Oscillator Enable* This bit controls clock oscillator (OSCCLK) in D2 State. Note that the oscillator must be enabled when any of the PLLs are enabled or when slow refresh for the internal memory is enabled.	
	0	Clock oscillator is powered down in D2 State.
	1	Clock oscillator can be enabled in D2 State.
1	*D2 PLL 1 Enable* This bit controls PLL 1 in D2 State.	
	0	PLL 1 is powered down in D2 State.
	1	PLL 1 can be enabled in D2 State.
2	*D2 PLL 2 Enable* This bit controls PLL 2 in D2 State.	
	0	PLL 2 is powered down in D2 State.
	1	PLL 2 can be enabled in D2 State.
3	*D2 PLL 3 Enable* This bit controls PLL 3 in D2 State.	
	0	PLL 3 is powered down in D2 State.
	1	PLL 3 can be enabled in D2 State.
4	*D2 Memory Interface Unit (MIU) Enable* This bit controls Memory Interface Unit (MIU) in D2 State.	
	0	MIU is powered-down in D2 State. Graphics Engine and both Controller 1 and Controller 2 will also be powered down. When MIU is powered down, slave read/write access to display frame buffer and to GE Command and Source FIFO cannot be executed. Similarly bus master (DMA) read/write accesses cannot be executed.
	1	MIU can be enabled in D2 State.
5	*D2 Memory Refresh Enable* This bit controls internal memory refresh when MIU is powered-down in D2 State.	
	0	Internal memory is not refreshed in D2 State when MIU is powered-down. Note oscillator clock must be supplied in order to refresh the internal memory.
	1	Internal memory is not refreshed in D2 State when MIU is powered-down.
6	*D2 Graphics Engine (GE) Enable* This bit controls Graphics Engine (GE) in D2 State.	
	0	GE is powered down in D2 State. Power management software must make sure that Graphics Engine is idle prior to going to power down state. If the engine is not idle prior to entering power down state, then frame buffer may be corrupted and may have to be redrawn at a later point.
	1	GE can be enabled in D2 State.
7	*Reserved (R/W)* This bit must be programmed to 0.	
8	*D2 CRT Enable.* This bit controls CRT display in D2 State. Note that CRT display can also be powered down by powering down the Graphics Controller that drives the CRT. This bit may be used only if the same controller is used to drive both CRT and flat panel and only the CRT needs to be powered down in this state.	

	0	CRT display is powered down in D2 State. This does not power down the Graphics Controller that drives the CRT.
	1	CRT display can be enabled in D2 State.
9	*D2 Flat Panel Enable.* This bit controls Flat Panel display in D2 State. Note that flat panel display can also be powered down by powering down the Graphics Controller that drives the flat panel. This bit may be used only if the same controller is used to drive both CRT and flat panel and only the flat panel needs to be powered down in this state.	
	0	Flat panel is powered down in D2 State. This does not power down the Graphics Controller that drives the flat panel.
	1	Flat panel can be enabled in D2 State.
10	*Reserved (R/W) for D2 TV Enable* This bit must be programmed to 0.	
15-11	*Reserved (R/W)* These bits must be programmed to 0.	
16	*D2 Controller 1 Enable* This bit controls Controller 1 in D2 State.	
	0	Controller 1 is powered down in D2 State. This will also power down Window 1, Cursor 1, and Overlay 1. If Controller 1 is driving CRT or Flat panel then the CRT or Flat Panel will also be powered down.
	1	Controller 1 can be enabled in D2 State.
17	*D2 Window 1 Enable* This bit controls Window 1 in D2 State.	
	0	Window 1 is powered down in D2 State. This will power down both Window 1 and Alternate Window 1.
	1	Window 1 can be enabled in D2 State if bit 16 is 1.
18	*D2 Alternate Window 1 Enable* This bit controls Alternate Window 1 in D2 State.	
	0	Alternate Window 1 is not enabled in D2 State if bit 16 is 1 and bit 17 is 1.
	1	Alternate Window 1 is enabled in D2 State if bit 16 is 1 and bit 17 is 1.
19	*D2 Cursor 1 Enable* This bit controls Cursor 1 in D2 State.	
	0	Cursor 1 is powered down in D2 State.
	1	Cursor 1 can be enabled in D2 State if bit 16 is 1.
20	*Reserved (R/W) for D2 Overlay 1 Enable* This bit controls Overlay 1 in D2 State.	
	0	Overlay 1 is powered down in D2 State.
	1	Overlay 1 can be enabled in D2 State if bit 16 is 1.
23-21	*Reserved (R/W)* These bits must be programmed to 0.	
24	*D2 Controller 2 Enable* This bit controls Controller 2 in D2 State.	
	0	Controller 2 is powered down in D2 State. This will also power down Window 2, Cursor 2, and Overlay 2. If Controller 2 is driving CRT or Flat panel then the CRT or Flat Panel will also be powered down.
	1	Controller 2 can be enabled in D2 State.
25	*D2 Window 2 Enable.* This bit controls Window 2 in D2 State.	
	0	Window 2 is powered down in D2 State. This will power down both Window 2 and Alternate Window 2.
	1	Window 2 can be enabled in D2 State if bit 24 is 1.
26	*D2 Alternate Window 2 Enable.* This bit controls Alternate Window 2 in D2 State.	
	0	Alternate Window 2 is not enable in D2 State if bit 24 is 1 and bit 25 is 1.

	1	Alternate Window 2 is enable in D2 State if bit 24 is 1 and bit 25 is 1.
27	*D2 Cursor 2 Enable.* This bit controls Cursor 2 in D2 State.	
	0	Cursor 2 is powered down in D2 State.
	1	Cursor 2 can be enabled in D2 State if bit 24 is 1.
31-28	*Reserved (R/W)* These bits must be programmed to 0.	

5.10.6 PM05R: PLL1 Configuration Register

PM	**PM05R – PLL 1 Programming** *Index: 05 Reset value: xxxx-0000h*	
0	*Reserved (R/W)* This bit must be programmed to 0.	
1	*PLL 1 Bypass* This bit optionally bypasses the PLL 1.	
	0	PLL 1 output is not bypassed.
	1	PLL 1 is bypassed. Oscillator clock output is used instead. In this case, PLL 1 should be disabled to save power.
3-2	*Reserved (R/W)* These bits must be programmed to 0.	
5-4	*PLL 1 Current Gain* These bits control PLL1 current gain.	
	00	This setting is used if total multiplier ratio is between 1 to 4.
	01	This setting is used if total multiplier ratio is between 4 to 8.
	10	This setting is used if total multiplier ratio is between 8 to 16.
	11	This setting is used if total multiplier ratio is between 16 to 32.
7-6	*PLL 1 Voltage Gain* These bits control PLL1 voltage gain.	
	00	This setting is used if VCO frequency is 35 to 100 MHz.
	01	This setting is used if VCO frequency is 100 to 150 MHz.
	10	This setting is used if VCO frequency is 150 to 200 MHz.
	11	This setting is used if VCO frequency is 200 to 250 MHz.
10-8	*PLL 1 Multiplier Factor* This parameter specifies multiplication factor for PLL 1. Programmed value = actual value – 1.	
15-11	*Reserved (R/W)* These bits must be programmed to 0.	
31-16	*Reserved (0)* These bits do not exist (not implemented).	

5.10.7 PM06R: PLL2 Configuration Register

PM	PM06R – PLL 2 Programming *Index: 6 Reset value: xxxx-0000h*	
0	*PLL 2 Reference Clock Source* This bit selects reference clock for PLL 2.	
	0	PLL 2 reference clock is the oscillator clock (OSCCLK).
	1	PLL 2 reference clock is P2CLK input.
1	*PLL 2 Bypass* This bit optionally bypasses the PLL 2.	
	0	PLL 2 output is not bypassed.
	1	PLL 2 is bypassed. P2CLK input is used instead. In this case, bit 0 has no effect and PLL 2 should be disabled to save power.
3-2	*Reserved (R/W)* These bits must be programmed to 0.	
5-4	*PLL 2 Current Gain* These bits control PLL2 current gain.	
	00	This setting is used if total multiplier ratio is between 1 to 4.
	01	This setting is used if total multiplier ratio is between 4 to 8.
	10	This setting is used if total multiplier ratio is between 8 to 16.
	11	This setting is used if total multiplier ratio is between 16 to 32.
7-6	*PLL 2 Voltage Gain* These bits control PLL2 voltage gain.	
	00	This setting is used if VCO frequency is 35 to 100 MHz.
	01	This setting is used if VCO frequency is 100 to 150 MHz.
	10	This setting is used if VCO frequency is 150 to 200 MHz.
	11	This setting is used if VCO frequency is 200 to 250 MHz.
10-8	*PLL 2 Multiplier Factor* This parameter specifies multiplication factor for PLL 2. Programmed value = actual value – 1.	
15-11	*Reserved (R/W)* These bits must be programmed to 0.	
31-16	*Reserved (0)* These bits do not exist (not implemented).	

5.10.8 PM05R: PLL3 Configuration Register

PM	PM07R – PLL 3 Programming *Index: 07 Reset value: xxxx-0000h*	
0	*PLL 3 Reference Clock Source* This bit selects reference clock for PLL 3.	
	0	PLL 2 reference clock is the oscillator clock (OSCCLK).
	1	PLL 2 reference clock is P3CLK input.
1	*PLL 3 Bypass* This bit optionally bypasses the PLL 3.	
	0	PLL 3 output is not bypassed.
	1	PLL 3 is bypassed. P3CLK input is used instead. In this case bit 0 has no effect and PLL 3 should be disabled to save power.
3-2	*Reserved (R/W)* These bits must be programmed to 0.	
5-4	*PLL 3 Current Gain* These bits control PLL3 current gain.	
	00	This setting is used if total divider ratio is between 1 to 4.
	01	This setting is used if total divider ratio is between 4 to 8.
	10	This setting is used if total divider ratio is between 8 to 16.
	11	This setting is used if total divider ratio is between 16 to 32.
7-6	*PLL 3 Voltage Gain* These bits control PLL3 voltage gain.	
	00	This setting is used if VCO frequency is 35 to 100 MHz.
	01	This setting is used if VCO frequency is 100 to 150 MHz.
	10	This setting is used if VCO frequency is 150 to 200 MHz.
	11	This setting is used if VCO frequency is 200 to 250 MHz.
10-8	*PLL 3 Multiplier Factor* This parameter specifies multiplication factor for PLL 3. Programmed value = actual value – 1.	
15-11	*Reserved (R/W)* These bits must be programmed to 0.	
31-16	*Reserved (0)* These bits do not exist (not implemented).	

6 Electrical Specifications

Table 6-1: Absolute Maximum Conditions

Symbol	Parameter	Min	Max	Units
V_{CC}	Supply Voltage	-0.5	3.3	V
V_I	Input Voltage	-0.5	3.6	V
T_{STG}	Storage Temperature	-40	125	°C

Note: Permanent device damage may occur if Absolute Maximum Rating are exceeded. Operation must be restricted to the conditions under Normal Operating Conditions.

Table 6-2: Normal Operating Conditions

Symbol	Parameter	Min	Typical	Max	Units
V_{CC}	Supply Voltage	3.0	3.3	3.6	V
T_A	Ambient Temperature	0	—	70	°C

Table 6-3: DAC Characteristics

Symbol	Parameter	Notes	Min	Typical	Max	Units
I_O	Full Scale Output Current	R_{SET}=187Ω and 37.5Ω Load	–	17.2	–	mA
	Full Scale Error		–	–	± 5	%
	DAC to DAC Correlation		–	1.27	–	%
	DAC Linearity		±2	–	–	LSB

Note: These values apply under normal operating conditions unless otherwise noted.

Table 6-4: DC Characteristics

Symbol	Parameter	Notes	Min	Max	Units
P_D	Power Dissipation	All VCCs at 3.3V MCLK=66MHz, DCLK=94.5MHz	–	1.3	W
I_{IL}	Input Leakage Current		–100	+100	µA
I_{OZ}	Output Leakage Current	High Impedance	–100	+100	µA
V_{IL}	Input Low Voltage	All input pins	–0.5	0.8	V
V_{IH}	Input High Voltage	All input pins	0.6xVcc	5.5	V
V_{OL}	Output Low Voltage	Under max load per table 16-5 (3.3V)	–	0.5	V
V_{OH}	Output High Voltage	Under max load per table 16-5 (3.3V)	0.7xVcc	–	V

Note: These values apply under normal operating conditions unless otherwise noted.

Table 6-5: DC Drive Characteristics

Symbol	Parameter	Output Pins	DC Test Conditions	Min	Units
I_{OL}	Output Low Current	D[31:0], IRL#, WAIT#, DREQ#, GPIO[2:0], PWM[1:0]	$V_{OUT} \leq V_{OL}$ and V_{CC}=3.3V	4	mA
		ENVDD, ENCTL, ENVEE, FD[23:0], FDE, FDI, FLM, LP, FSCLK, VSYNC, HSYNC		8	mA
		AR, AG, AB		12	mA
I_{OH}	Output High Current	D[31:0], IRL#, WAIT#, DREQ#, GPIO[2:0], PWM[1:0]	$V_{OUT} \geq V_{OL}$ and V_{CC}=3.3V	4	mA
		ENVDD, ENCTL, ENVEE, FD[23:0], FDE, FDI, FLM, LP, FSCLK, VSYNC, HSYNC		8	mA
		AR, AG, AB		12	mA

Note: These values apply under normal operating conditions unless otherwise noted.

Table 6-6: AC Test Conditions

Symbol	Parameter	Minimum	Maximum	Units
V_{CC}	Supply Voltage	3.15	3.6	V
V_{TEST}	All AC parameters	1.5	1.5	V
V_{IL}	Input low voltage (10% of V_{CC})	0.1 V_{CC}	0.1 V_{CC}	V
V_{IH}	Input high voltage (90% of V_{CC})	V_{CC} - 0.1	V_{CC} - 0.1	V
T_R	Maximum input rise time	3	3	ns
T_F	Maximum input fall time	2	2	ns

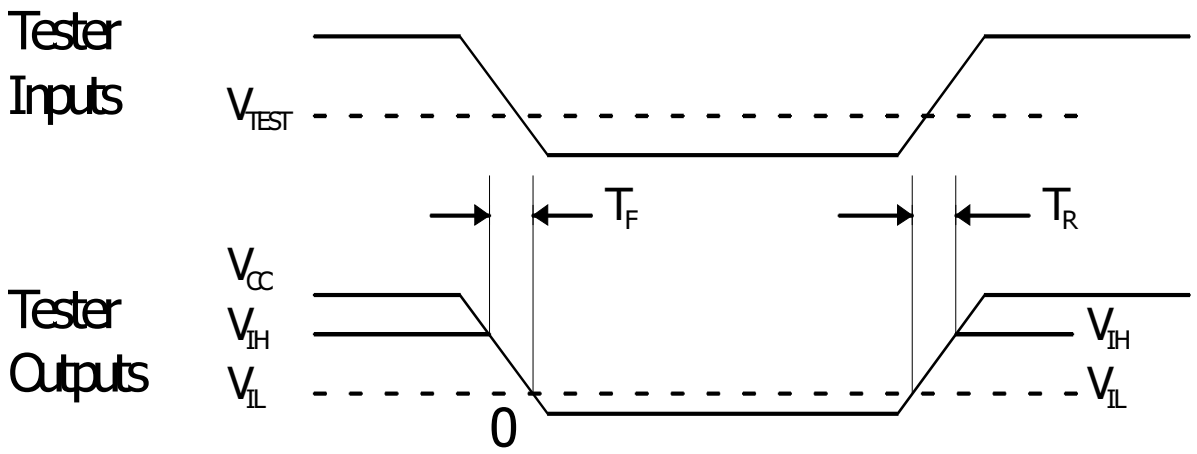


Figure 6A: AC Test Timing

Table 6-7: AC Timing Characteristics - Reference Clock

Symbol	Parameter	Notes	Min	Typical	Max	Units
F_{REF}	Reference Frequency		10	12.288	20	MHz
T_{HI} /T_{REF}	Reference Clock Duty Cycle		40	–	60	%

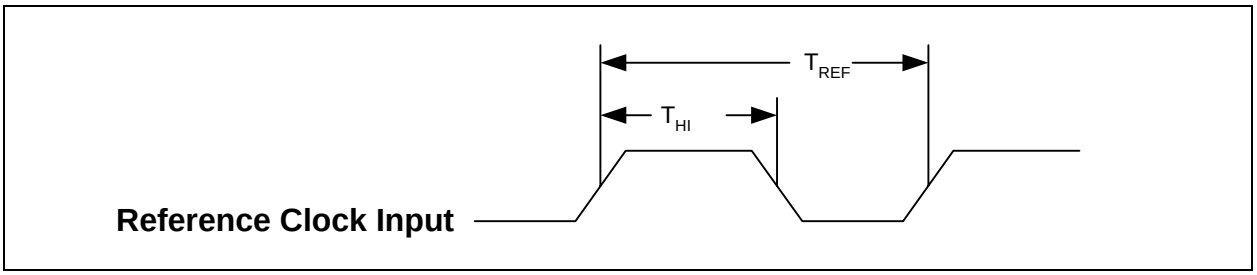


Figure 6B: Reference Clock Timing

Table 6-8: AC Timing Characteristics - Clock Generator

Symbol	Parameter	Notes	Min	Typical	Max	Units
F_{DCLK}	DCLK Frequency		–	–	94.5	MHz
F_{MCLK}	MCLK Frequency		–	–	66	MHz

Table 6-9: AC Timing Characteristics - Reset

Symbol	Parameter	Notes	Min	Max	Units
T_{IPR}	Reset Inactive from Power Stable	See **Note 1**	1	–	ms
T_{ORS}	Reset Inactive from Ext. Osc. Stable		0	–	ms
T_{RES}	Minimum Reset Pulse Width	See **Note 2**	1	–	ms
T_{RSR}	Reset Rise Time	measured 0.1Vcc to 0.9Vcc	–	20	ns
T_{RSO}	Reset Active to Output Float Delay		–	40	ns
T_{CSU}	Configuration Setup Time	See **Note 3**	20	–	ns
T_{CHD}	Configuration Hold Time		5	–	ns

Note 1: This parameter includes time for internal voltage stabilization of all sections of the chip, start-up and stabilization of the internal clock synthesizer, and setting of all internal logic to a known state.

Note 2: This parameter includes time for the internal clock synthesizer to reset to its default frequency and time to set all internal logic to a known state. It assumes power is stable and the internal clock synthesizer is already operating at some stable frequency.

Note 3: This parameter specifies the setup time to latch reliably the state of the configuration bits. Changes in some configuration bits may take longer to stabilize inside the chip (such as internal clock synthesizer-related bits 4 and 5). The recommended configuration bit setup time is T_{RES} to insure that the chip is in a completely stable state when Reset goes inactive.

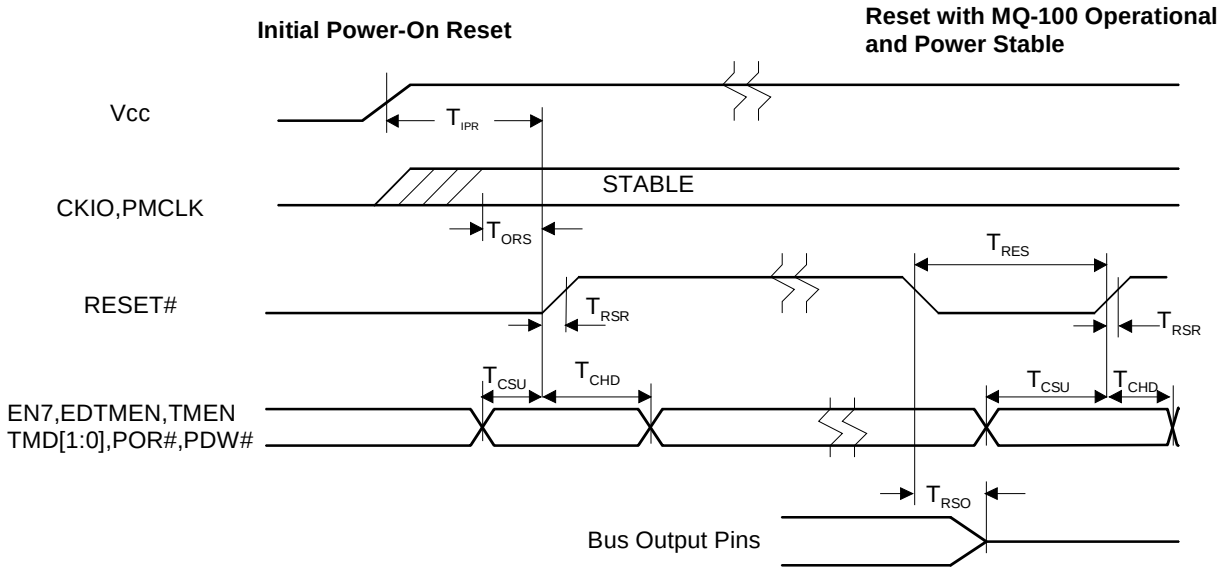


Figure 6C: Reset Timing

Table 6-10 AC Timing Characteristics - Panel Output Timing

Symbol	Parameter	Signaling	Min	Max	Units
T_{SCLK}	SHFCLK cycle time	Measured at $0.4V_{CC}$	15	–	ns
T_{DOVD}	DE and P[35.0] Output Valid Delay		-3	4	ns
T_{COVD}	LP and FLM Output Valid Delay		-3	3	ns
	SHFCLK Duty Cycle		40	60	%

Note: AC Timing is valid when max output loading=25pF.

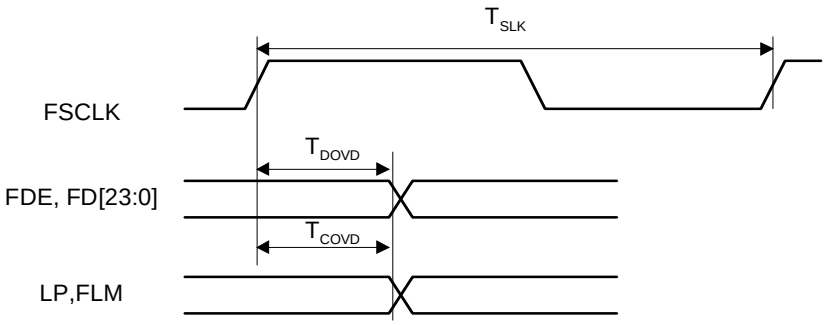


Figure 6D: Panel Output Timing

Table 6-11 AC Timing Characteristics - CPU Read Timing

Symbol	Parameter	Signaling	Min	Max	Units
t_{BSD}	BS delay time		-	10	ns
t_{AD}	Address delay time		-	10	ns
t_{RWD}	RW delay time		-	10	ns
t_{CSD}	CS delay time		-	10	ns
t_{WED1}	WE delay time		-	10	ns
t_{WEDF}	WE delay time at falling edge (to CKIO falling edge)		-	10	ns
t_{RDYS}	RDY setup time		-	-2	ns
t_{RDYH}	RDY hold time		-	1.5	ns
t_{RDS}	RD setup time		-	-2	ns
t_{RDH}	RD hold time		-	1.5	ns

Note:

- AC Timing is valid when max output loading=25pF.
- CPU read timings are based on SH-7750 and SH-7709 operating at 66 MHz.

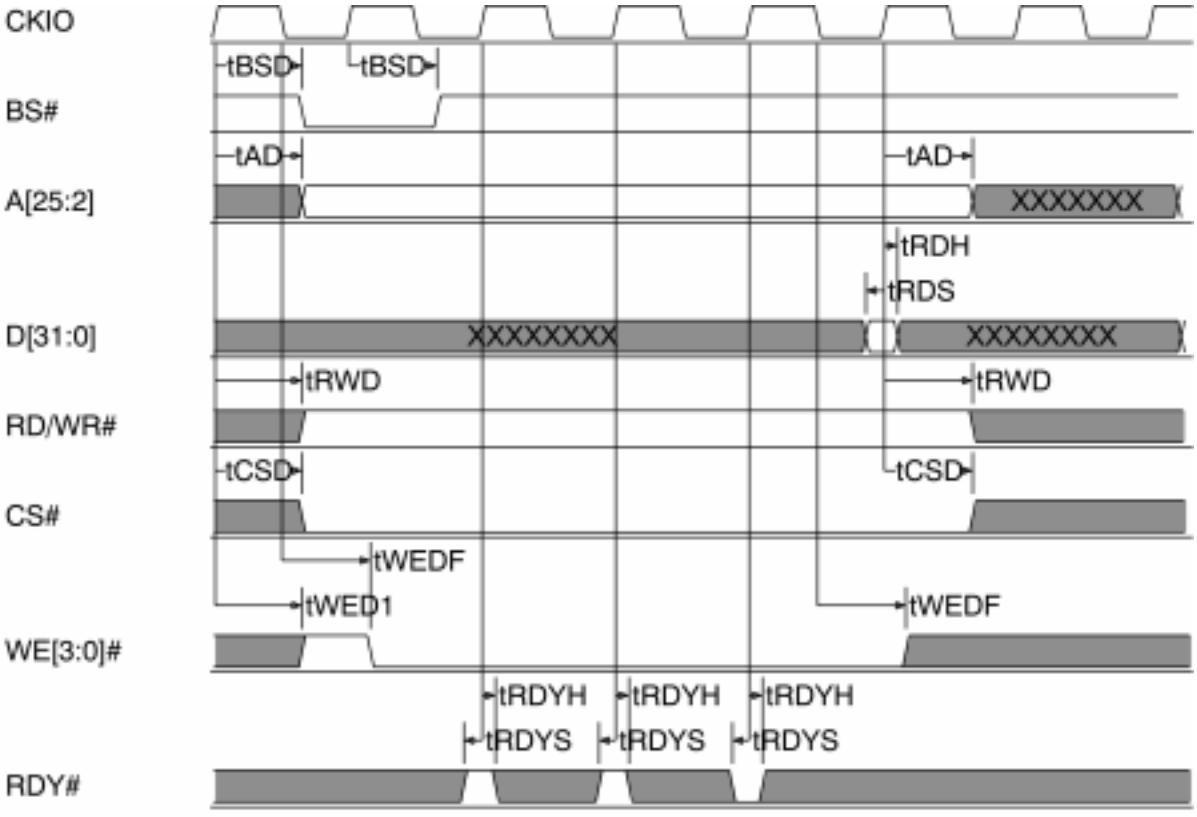


Figure 6E: CPU Read Timing

Table 6-12 AC Timing Characteristics - CPU Write Timing

Symbol	Parameter	Signaling	Min	Max	Units
t_{BSD}	BS delay time		-	10	ns
t_{AD}	Address delay time		-	10	ns
t_{WDD}	Write data delay time		-	10	ns
t_{RWD}	RW delay time		-	10	ns
t_{CSD}	CS delay time		-	10	ns
t_{WED1}	WE delay time		-	-2	ns
t_{WEDF}	WE delay time at falling edge (to CKIO falling edge)		-	1.5	ns
t_{RDYS}	RDY setup time		-	-2	ns
t_{RDYH}	RDY hold time		-	1.5	ns

Note:

- AC Timing is valid when max output loading=25pF.
- CPU read timings are based on SH-7750 and SH-7709 operating at 66 MHz.

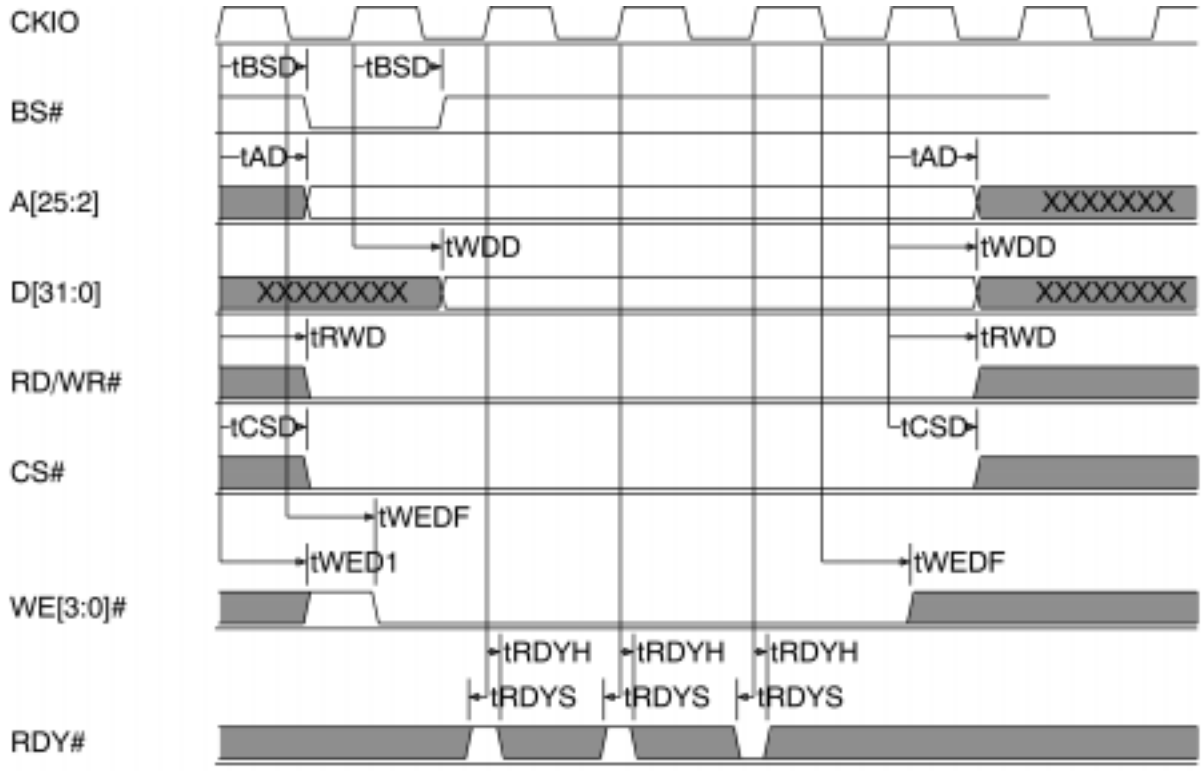


Figure 6F: CPU Write Cycle Timing

HITACHI

MediaQ, Inc.
3303 Octavius Drive
1001-A
Santa Clara, California 95054
Phone: 408-588-0080
FAX: 408-588-0084

Title: MQ-100/HD64464 Data Book
Publication #: MQ-100/HD64464-DB-
Revision No.: 1.08
Date: 7/29/98